

Natural Computing Series

Nataša Jonoska
Erik Winfree *Editors*



Visions of DNA Nanotechnology at 40 for the Next 40

A Tribute to Nadrian C. Seeman

OPEN ACCESS

 Springer

Natural Computing Series

Founding Editor

Grzegorz Rozenberg

Series Editors

Thomas Bäck , Natural Computing Group–LIACS, Leiden University, Leiden,
The Netherlands

Lila Kari, School of Computer Science, University of Waterloo, Waterloo, ON,
Canada

Susan Stepney, Department of Computer Science, University of York, York, UK

Scope

The Natural Computing book series covers theory, experiment, and implementations at the intersection of computation and natural systems. This includes:

- **Computation inspired by Nature:** Paradigms, algorithms, and theories inspired by natural phenomena. Examples include cellular automata, simulated annealing, neural computation, evolutionary computation, swarm intelligence, and membrane computing.
- **Computing using Nature-inspired novel substrates:** Examples include biomolecular (DNA) computing, quantum computing, chemical computing, synthetic biology, soft robotics, and artificial life.
- **Computational analysis of Nature:** Understanding nature through a computational lens. Examples include systems biology, computational neuroscience, quantum information processing.


Nataša Jonoska · Erik Winfree
Editors


Visions of DNA Nanotechnology at 40 for the Next 40

A Tribute to Nadrian C. Seeman

 Springer

Editors

Nataša Jonoska 
Department of Mathematics and Statistics
University of South Florida
Tampa, FL, USA

Erik Winfree 
Computer Science, Bioengineering,
Computation and Neural Systems
California Institute of Technology
Pasadena, CA, USA



ISSN 1619-7127

Natural Computing Series

ISBN 978-981-19-9890-4

ISBN 978-981-19-9891-1 (eBook)

<https://doi.org/10.1007/978-981-19-9891-1>

© The Editor(s) (if applicable) and The Author(s) 2023. This book is an open access publication.

Open Access This book is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this book are included in the book's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the book's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.

The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore



Nadrian C. Seeman (December 16, 1945–November 16, 2021)

In Memoriam

Ned Seeman, whose scientific vision and accomplishments this volume was conceived to honor, passed away while this book was in the late stages of preparation. It is a minor sadness that he will not see the completion of this book; it is a considerable sadness that he will not continue to oversee the development of the field that he created; it is a great sadness that he will continue to guide and inspire his friends and colleagues only in their memories.

As a tribute to Ned, and as a way to remind the upcoming generation of his influence, we (Erik and Nataša) organized a panel discussion at the 28th International Conference on DNA Computing and Molecular Programming held on August 11, 2022. The following consists of combination of notes prepared for the panel, recollections of what was said, and revisions reflecting what the participants wished they had said—an imperfect transcript. It is followed by a series of vignettes and remembrances from a few of Ned’s colleagues and friends. (We wish we could have included more!)

Panel Discussion

Moderator (Erik Winfree): Ned Seeman passed away November 16, 2021. He was a guiding light of our community since the very first DNA computing conference in 1995 and had been building the foundations of DNA nanotechnology for decades before that. It is sad to acknowledge that the younger members of our community will not have the chance to chat with him during coffee breaks, hear his colorful invective first-hand, and be inspired by his scientific creativity and dedication to truth. Today’s panel discussion is meant as a tribute to his life and science—to recognize what we have lost and to celebrate what we have gained.

For many of us here today, Ned needs no introduction. But for those whose intersection with this community is more in the future than in the past, I would like to give a brief biography. Ned was born a few months after the end of World War II. He was an undergraduate at the University of Chicago, where he transitioned

from being a medical student to focusing on biochemistry. As a graduate student at the University of Pittsburgh he settled on crystallography, which he pursued further as a postdoc, first at Columbia, and then at MIT where he worked with Alex Rich. Despite doing spectacular work with Alex on transfer RNA (*Science* 1974) and on sequence-specific recognition of DNA by proteins (*PNAS* 1976), he was not one of the lucky ones when looking for a faculty position. He started as an assistant professor at SUNY Albany in 1977—for what he later called his “Albanian exile”. Another phrase he coined about that time was the inevitable sequence: “No crystals, no crystallography, no crystallographer”. That was basically the way his career was going by 1980 when he was contemplating the prospects of tenure, or lack thereof, having no graduate students and, as he said, “having crystallized nothing of great importance”. The invention of DNA nanotechnology later that year was really a case of snatching victory from the jaws of defeat. Still, his 1982 paper in *Theoretical Biology*—the one that sketched out his ideas for designing multi-branched DNA structures that could self-assemble into rationally designed crystals—was largely ignored for decades, despite his 1983 paper in *Nature* demonstrating experimentally that immobile branched junctions can be designed and synthesized. How’d he get that done, with a basically empty lab? He did whatever he had to. He collaborated with a postdoc at the University of Pennsylvania, who later helped him move to New York University in 1988, where he stayed for the rest of his career.

Why do I emphasize this particular period of what might be considered prehistory, when the really exciting advances in DNA nanotechnology—which Ned participated in equally—came much later as the field started to flourish? After all, when Ned was awarded the Kavli Prize in Nanoscience in 2010, the citation lauded him for “pioneering the use of DNA as a non-biological programmable material for a countless number of devices that self-assemble, walk, compute, and catalyze”. There are several reasons. Perhaps the main one is that this period helped forge Ned’s scientific character, and thus it helps for understanding him and his science. Another is that it reminds us to be thankful to him, for coming through rather than giving up. A third reason is perhaps as a message to the younger people here, that yes, not all careers in science get the lucky breaks, and no, it is definitely not fair, but much of what we are is determined by how we respond to that. Ned modeled one way.

This panel will remind us of some aspects of that way. We’ll organize the discussion into comments from the panelists on 3 broad topics, and then open the floor for participation from the audience. The 3 topics are (1) His science. A sampling of Ned’s papers that touched us somehow. (2) His person. Remembrance of his influence on us through mentoring and inspiration. (3) His legacy. The field of DNA nanotechnology has come so far in the 40 years since Ned envisioned the foundations, so where is it going in the next 40 years?

OK, before we get going, I’d like to introduce the panelists. We were fortunate to get people with a range of perspectives and backgrounds. I am Erik Winfree, from Caltech. I was a graduate student with John Hopfield at Caltech, but collaborated with Ned during that time. I look at the world through the lens of a theoretical computer scientist and natural philosopher. From left to right, the other panelists are: Hao Yan, from Arizona State University. He was a chemistry graduate student

with Ned, then a postdoc with John Reif's group in computer science at Duke. His contributions to the field are too numerous to mention, and among the most beautiful—his group's demonstration of 3D DNA origami with curved shapes, such as a nanoscale vase, comes to mind. Tom Ouldridge, from Imperial College London. He was a physics graduate student with Ard Louis and Jon Doye at Oxford, where he developed the oxDNA model that has proven to be the premier simulator for DNA nanotechnology. His research has expanded dramatically into fundamental questions about biophysics, circuits, and molecular machines. Simon Vecchioni, from NYU. He did his Ph.D. in biomedical engineering with physicist Shalom Wind at Columbia University, where he looked at the conductivity of DNA-based nanowires, and he is currently a postdoc in the Seeman group, where he has been straddling fields. Shelley Wickham, from the University of Sydney. She was a graduate student in physics with Andrew Turberfield at Oxford, where she worked on DNA-based motors and robots, and then a postdoc with William Shih at Harvard, where she developed modular barrel-shared DNA origami. Her current research includes interfacing origami with biological membranes. Nataša Jonoska, the professor of mathematics at the University of South Florida and frequent collaborator with Ned, was supposed to be with us today as well, but unfortunately at the last minute she was not able to attend the conference. She is the main reason this panel happened. In a few places later on, I will read a few comments she sent by email.

Now, let's dive into the first topic: Papers by Ned. Hao will go first.

Hao Yan: My name is Hao Yan, a former Ph.D. student of Ned; I graduated in 2001 from Ned's lab. My favorite paper of his is the 1993 Biochemistry paper titled "DNA double crossover molecules", in which he and his student Tsu Ju Fu reported and characterized a series of DNA molecules containing two crossover sites between helical domains. These include both antiparallel and parallel DNA double-crossover molecules. These DNA constructs laid the foundation for future works of structural DNA nanotechnology, including Seeman and Winfree's 1998 Nature paper on designed two-dimensional DNA arrays using antiparallel DNA double-crossover molecules, and later Rothemund's DNA origami, and many more. In fact, in many of Ned's earlier papers he proposed ideas that later were realized and expanded by colleagues in our field. Examples include the use of meso-junction and anti-junction to create reconfigurable cascading arrays, replicable single-stranded nanostructures, to name a few.

Tom Ouldridge: Early in my Ph.D., Ned Seeman visited Oxford to give a talk. At the time I was deep into developing oxDNA and very focused on using it to simulate self-assembly. He talked about Omabegho's motor: "A bipedal DNA Brownian motor with coordinated legs" (2009). The idea was to create a DNA-only system (no cheating with enzymes, Shelley) that would walk along a track. I remember being inspired by the idea of creating active, fuel-consuming machines like those in biology. The audacity of rationally engineering molecular machines, where chemical-free energy is carefully channeled into mechanical action at a microscopic scale, still blows me away. Indeed, I think an underappreciated aspect of biology is just how good it is at applying chemical-free energy in a standard form to achieve essentially any goal.

I also realized that this was an area that oxDNA could be immediately applied to. Nowadays the model is mostly used for simulating large structures, but at the time the opportunity to say something useful about such small, intricate systems was invaluable. Helpfully for me, Andrew Turberfield's lab, just round the corner, was also very interested in molecular machines, and simulating those machines and the underlying displacement motif generated oxDNA's first useful results.

Simon Vecchioni: I have two favorite papers by Ned. The first is "DNA Nanotechnology at 40" (2020), his perspective in *Nano Letters*. It is a short, non-technical read, and in it, Ned highlights that he is often asked what is next in the field. In published, academic text, he writes: "I DON'T KNOW". I think this captures his acerbic and brilliant wit fairly completely.

More broadly, I am drawn again and again to "Designed Two-Dimensional DNA Holliday Junction Arrays Visualized by Atomic Force Microscopy" (1999), a *JACS* paper authored by Chengde Mao. As a graduate student in a nanotechnology and nanofabrication lab, DNA nanotech was a fascination for all of us, but the methodology hadn't been trained into our bones the way it had been for Ned's group. At the time, this paper felt like a fount of knowledge—it is laid out well, identifies the design rules, and enumerates the composable parts. It's a roadmap for the aspiring DNA nanotechnologist and presents a means of creating and analyzing arbitrary DNA constructs from simple parts. Integer numbers of helical turns, stable immobile Holliday junctions, sequence design, and periodic windows are visualized by atomic force microscopy—it's all there. Engineering new chemistry and swapping components within the parallelogram arrays in this paper felt accessible and doable. For anyone looking to enter the field as I was, this paper presents an excellent launch point.

Shelley Wickham: My favorite paper of Ned's is "A proximity-based programmable DNA nanoscale assembly line" (2010)—a *Nature* paper with first author Hongzhou Gu. This paper came out while I was in my Ph.D., and I was really impressed by the ambition. It had DNA walkers using strand displacement and DNA origami and cargo loading and assembly and AuNPs and it worked. It also had the most lurid color scheme for AFM images I have ever seen. A tripod DNA "walker" rolls along a track, picking up and putting down feet with toehold mediated strand displacement. At each of 3 sites along the track, a depot tile or "cassette" sits pre-loaded with different types of AuNPs. Depending on the trigger DNA strand the cassette swings the cargo over, which then gets added to the motor. Without a trigger, no cargo is loaded at that depot. The paper shows they produce 8 different structures at the end, depending on whether the cassette was switched on or off at each step. The system requires a total of 11 sequential switches.

This paper is still very cool and is something people are still trying to do—bring together DNA circuits with nanostructures to make assembly lines that construct other materials.

Erik Winfree: My first encounter with Ned's work came in early 1995, when I met with Paul Rothemund—then not even a graduate student yet—to discuss Len Adleman's paper on DNA computing. Paul had been working on how to build Turing machines with restriction enzymes, as an undergraduate class project, and he had a

huge stack of photocopied papers for me to look at, mostly by Ned. It's hard to choose a specific paper to talk about for this panel, as I read many of his papers line by line at that time, and they all influenced me deeply, and I've read many since then and can say the same. But the one I've chosen is "De novo design of sequences for nucleic acid structural engineering" (1990). This wasn't his first paper on sequence design, but it was the most detailed one available to study when I was designing my first double-crossover molecules as a graduate student, so it made a real impression on me. To be honest, so did his earlier 1983 paper, "Design of immobile nucleic acid junctions", where he described how he designed the sequences for his first experimental demonstration of an immobile 4-way junction. It's amazing! Ned had practically written a little mini-*proto-NUPACK*! It used base-pairing constraints to specify the target multi-stranded structure; it used test tube concentration calculations to determine the fraction of target assembly as a function of temperature, to maximize stability; it used partition function calculations to maximize the fidelity of specific binding and minimize spurious binding—all reminiscent of what Niles Pierce calls positive and negative design principles. Only, it was written in FORTRAN and ran on a Univac 1100/82, and it didn't really scale up beyond the 64-nt four-armed junction that they designed. So what did the 1990 paper do? Basically, it described what Ned learned for how to scale up as needed for the experimental project they were in the middle of at the time: the DNA wire-frame cube that you all do or should love. Since Ned didn't have the algorithmic chops that Niles later brought to bear on the problem, instead of finding a better algorithm to more efficiently solve the original design criteria, Ned's approach now was to identify a set of simpler criteria that were "good enough"—at least for his purposes. These criteria centered on sequence symmetry minimization—trying to minimize the length of repetitive or complementary subsequences. What was exciting is that these criteria generalized naturally to not just junctions, but essentially arbitrary DNA constructs based on Watson-Crick pairing, and were susceptible to basic optimization algorithms. The field of course has come a long way since Ned's early forays into sequence design, but even today many DNA systems are designed based on the frameworks he established 40 years ago.

There's another take-away that I find to be really interesting. Ned is often called a chemist, for obvious reasons, so it's easy to forget that he was also a coder, a computer programmer. He wrote a lot of programs: for crystallography, for simulating branch migration, for computer graphics, and for sequence design. I think it's no accident that identifying DNA as a uniquely programmable molecule was the insight of a computer programmer. And I also think it's no accident that the magic that powered the birth of this field was unleashed by a person whose understanding and capability spanned from algorithm design to experimental investigation, where all these aspects of reality could get tangled up with each other.

Nataša Jonoska: I heard Ned's talk for the first time at the second DNA computing conference in Princeton in 1996. He showed the infamous image of the cube, from "Synthesis from DNA of a molecule with the connectivity of a cube" (1991). I am a mathematician and did not know too much about biochemistry, and my knowledge of biochemistry was at the level of freshman biology. Constructing a 3D structure

that self-assembles through careful sequence design and predictable topology was completely off the wall for me. It was an inspiration that made me think about 3D structures and to design 3D DNA graphs that we later published together. The subject opened up several new mathematical problems in graph theory (study of new invariants) and topological graph theory (new approaches in graph embedding in surfaces). Those early days of DNA computing were filled with various approaches to solve NP problems (often related to questions about graphs). Our first approach was to envision a variation of DX to solve the Hamiltonian problem through assembling the graph structure itself. When I emailed Ned whether this variation of DX was feasible, he said “make a model”. I brought the model to the next DNA computing meeting and was thrilled to get a nod of approval from him. [Editor’s note: Here Natasha is not talking about a mathematical model or a computer simulation model, but rather something more like a tangle of metal jacks and plastic tubing—a physical thing you can get your hands on and feel.]

Moderator: Great. The second topic concerns Ned as a mentor, visionary, and human being. Again, Hao will go first.

Hao Yan: Ned was a great mentor, a great friend, and a great inspiration. Ned left a legacy that will last: not only did he single-handedly found the field of structural DNA nanotechnology in 1982, a field many of us are enjoying working in, a field that has become highly disciplinary with lots of creative minds working together to give DNA unique capabilities to build, sense, compute, and actuate, but he also founded the International Society for Nanoscale Science, Computation and Engineering (ISNSCE) and served as the founding president of the society. In Ned’s vision, the society will promote the study of the control of the arrangement of the atoms in matter, examine the principles that lead to such control, to develop tools and methods to increase such control, and to investigate the use of these principles for molecular computation and for engineering on the finest possible scales. On a personal note, there have been many things I have learned from Ned. Two important characters that have impacted me are: persistence and supporting students. Ned never gives up, and he once told us “if something doesn’t work, use a hammer, if it still doesn’t work, use a bigger hammer”. Ned has been very supportive of his students and junior colleagues in the field. I was serving on the faculty search committee in my department these days and have had the chance to read some of Ned’s recommendation letters for his students. Ned’s letter can be 5 pages long, and he inserts figures in the letter and really elaborates on what the students’ contribution to the project were. I have never seen letters from other people with figures; Ned really spent time working on his letters to support us. I nowadays do the same to my own students, putting my heart into supporting them and promoting their academic careers. Ned and I have developed from a mentor–student relationship into more of a close personal friendship. I remember, 5 years ago, Ned was trapped in an ICU due to illness in Nanjing while he was traveling in China and India. I emailed him and asked what I can help, he emailed back and said “get me out of the jail”. For Ned it was torture that he lost freedom and could not work on papers and proposals. Ruojie and I flew over to China immediately and tried to get him out, but the doctors wouldn’t release

him. I witnessed how vulnerable a big giant can be and how desperate Ned was when he was isolated from work, like a fish out of water.

Tom Ouldrige: There are two things I'd like to recognize. The first is that, to me at least, Ned's actions in founding the field seem to epitomize an attitude of "if we really believe that DNA has the properties we say it does, we should be able to do these amazing things with it". I think this is a good maxim that can be applied more broadly across scientific endeavors. It drives us to be ambitious, and it also drives us to build things that actually test and extend the limits of our scientific understanding.

Secondly, Ned coined—or at least used for the first time in my hearing—a term that I think describes what I (and many others) do. Since we build functional biomolecular systems from scratch, I've always thought that "synthetic biologists" is a better description of us than of the people who build transcription factor networks in cells. But to be fair, they should have been called genetic engineers, and that term was already taken. But I once heard Ned describe some of his work as "kleptobiology"—klepto being derived from the ancient Greek word for "theft". And that's a brilliant summary of how we steal useful biological molecules and apply them wildly out of context for our own purposes.

Simon Vecchioni: Nanoscience as a larger field is full of industrial partners and big-name physicists and chemists. Fitting into this web of well-established and competitive giants is a rather lonely affair. After joining Ned's lab as a postdoc, I find myself a part of a family I never expected to have. There's a sort of crusty human warmth to the whole field that is surprising and beautiful. You can hear everyone's stories and trials and efforts and aspirations, and it all intersects back to Ned, to his wild but methodical vision.

Ned was a man without pretense: he lived as himself, every day. He made the same walk to the lab, asked the same hard questions, and had a gigantic hidden soft spot for other people, and he loved the art of science—no matter his health or condition. When faced with challenges, he would often tell stories that fell squarely as non-sequitur, but had some roundabout hidden insight that would point you where you were going. He staunchly defended the intellectual property of figures in the field and would tolerate no theft of ideas. To a fault, he made sure that authorship remained with inventors and contributors; this current still runs through our field with its strong spirit of collaboration and uplift, rather than fierce competition and infighting. He was a light to all, and the character and tenor of this community—this ad hoc family—is the evidence.

Shelley Wickham: I first spoke to Ned at a GRC in nanofabrication 2010 in NH, as a Ph.D. student when he came up to my poster on DNA motors—just after the paper I described above came out. When he first came up to my poster, I was a bit intimidated. He took a long pause and my anticipation built up, and then his first words were...“deus ex machina”. It was not the reaction I was expecting! I now know this phrase refers to “a person or thing (as in fiction or drama) that appears or is introduced suddenly and unexpectedly and provides a contrived solution to an apparently insoluble difficulty”. At the time, I had no idea what it meant. Ned did explain it to me though, my work used a DNA motor powered by a nicking enzyme. We ended up having a great chat, and I visited his lab in NYU after the conference.

He was advocating the elegance of DNA only systems, as opposed to the “god-like” power of enzymes. Although through great work like the PEN toolbox and genelets, the field is getting better at harnessing these god-like properties of enzymes. Like others, I found Ned was always very open and engaged with people at all levels, especially students and postdocs. He was honest about academic careers—that it is hard, but it is worth it to be ambitious, and it’s ok to have a messy office. The last time I spoke to Ned was DNA25 (2019) in Seattle. Despite having recently been very ill, he was still ambitious and talking about the next grant, and the next project in a way that I found very inspiring.

Erik Winfree: I first met Ned when I was a graduate student at Caltech, and he’s a big reason why I am still at Caltech. I met him in Princeton, actually, when we both attended the first conference on DNA computing, in 1995. At the time I considered myself a mathematician and computer scientist, and only a few months earlier I had learned about Ned’s work from my friend Paul Rothmund, who was more familiar with chemistry. As part of my talk at the conference, I discussed my embryonic ideas for how Ned’s DNA nanotechnology work could be used as the foundation for a way to compute by molecular self-assembly. I was super surprised to discover that Ned was in the audience and super excited that he invited me, Paul, and a few others to get a beer at the pub afterward. That’s where he told me my ideas were unlikely to work—not with the molecules I was envisioning, at least. He also told me what he thought was the fix: a different flavor of DNA molecule called the double crossover that his group had recently invented and characterized. So he invited me to visit his lab over the summer, where he trained me as an experimental chemist. A few years later, we had a paper in *Nature*.

Ned’s efforts to help young people become successful in science is a common story. His heart and his actions both pushed that way. But more than that, his example. He is someone who illustrates what it means to have a vision, to see things that others don’t see—or don’t see the value of—and to pursue it for decades in near-isolation. Because that’s what it takes. It takes a strength of character and a belief in oneself that very few people have. So I learned from Ned what it means to bring something really new to science: it is to value things that others don’t (yet) value, to care about things that others don’t (yet) care about—that is where one can contribute something truly unique, something better than just being able to run the race faster and be first to a well-recognized goal. Of course, it might also mean that one doesn’t have an easy job getting funding or recognition. I am really glad that, ultimately, Ned has been recognized, not just by his friends and close colleagues, but by the broader scientific community, as the visionary and genius that he truly is. He will be missed and remembered. I think of the “soul” as the collection of recurrent patterns and algorithm snippets that characterize a person’s behavior, so I mean it quite literally when I say that fragments of Ned’s soul will live on in all the science and in every scientist that he touched.

Nataša Jonoska: Those of us who knew him can attest to his unique personality, the hidden kindness, and grand humanism. He offered to host me in his lab for my first sabbatical; so I spent two semesters in his lab. It was a super high learning curve for me as I had not done any lab work prior to that. But that was probably the best

professional (at the time very hard) decision. It turned out to be a great personal decision as well. It was 2001, we started a collaboration that transformed into a deep friendship that lasted 20 years. I was privileged to know him. His perseverance in getting the 3D DNA crystal materialized only in 2009. Lots of other results were happening in the lab during that time. Many great papers got published, but his persistence and ultimate vision was the main driving force; it brought new ideas and pushed students and junior researchers to think in a unique way and to generate lots of productive discussions. We should learn from his example: in science one must have a grand vision and a goal as a motivation. There are often disappointments in science, but one must see those as temporary setbacks. Also, a colorful language can sometimes ease up the frustration.

Moderator: Let's move on to the third and final topic: Where is the field going and how might it develop?

Hao Yan: Future directions of our field that I expect will be important include the following: (1) Molecular machines of the future will entail the combination of rational design and directed evolution to develop molecular machines driven not only by DNA strand displacement, pH, or light but by chemical fuels like those used by nature's molecular motors; (2) universal design platforms that integrate design and modeling tools from DNA, RNA, and protein design to create ribonucleoprotein complexes, an exciting example of which would be the design of an artificial ribosome; (3) structural control at the atomic level, as Ned emphasized when he founded ISNSCE.

Tom Ouldridge: I am a natural pessimist in this sense and usually wrong about what has potential. It is interesting to note that I'm seeing more companies pop up that want to make use of origami or strand displacement technology—in many cases using side products of the scientific research. But scientifically, I think a move toward systems that can operate autonomously over long timescales is an important area to develop, if we want to interface with biology or build synthetic life-like systems.

Simon Vecchioni: To get any part of structural DNA nanotechnology to work at all, Ned painstakingly built the field from its theoretical foundations up. The structure-function relationship of biological systems was a powerful undercurrent of the early works, and the helical twist to emergent topology relationship was rehashed again and again. Each subsequent step forward in the architectural expansion of DNA nanotechnology was accompanied by a well-elucidated structural toolkit. Though Ned staunchly claimed not to know what was coming, he had a strong proclivity toward expanding the chemical design language of DNA. Metal base pairs, guanine tetraplexes, modified backbones—each of these components required a new understanding of structure to enable the growth of new functions.

The goals of DNA nanotech, surprisingly, have more or less been clear since the beginning: (1) engineer soft matter into designed architectures; (2) use these architectures to organize biological and/or nanoscale components for structural and functional studies; and (3) generate emergent (nanoelectronic or otherwise) properties from DNA shapes. To this end, it has become clear that DNA is fantastic as a building material, but lacks some of the physical pizzazz of other nanoscale components. In engineering functional nanotechnologies, the field will need to carefully carve out

new structure-function relationships to equip DNA and DNA nanotechnologists to harness novel optical, electronic, computational, and mechanical properties.

Ned's lab today is deeply committed to expanding DNA nanoscience "Beyond Watson and Crick", and by that we mean adding new base pairs, new backbones, new secondary and tertiary interactions, and overall (carefully) breaking the design rules that have become so fundamental and dear to us. Per Ned's tried and true method, structure comes first. As we step into new chemistry, we imagine and hope for an expanded DNA design language that can encompass the deep and far-seeing vision that Ned had for this field.

Shelley Wickham: This ties back to Ned's ambitious 2010 paper, which influenced me a lot. I see in the future more work on bringing structures and circuits together to use DNA as a dynamic scaffolding tool to make other things. These are still hard goals, and as a field we are still not great at them but there is lot of recent progress that is really exciting. Could we make molecular 3D printers? Crystal lattices to template optical properties? Reconfigurable responsive materials/structures? Have kinetic control of assembly? Those are some future directions I am excited by.

Erik Winfree: In the early days of DNA nanotechnology, Ned often emphasized that the ultimate goal is *control*. The masterful scientist should know EXACTLY what is in the test tube, should be able to specify *exactly*, like Angstrom-level exactly, where each atom is. This was the crystallographer in him speaking. But DNA nanotechnology rarely reaches that pinnacle of control, especially in the hands of neophyte newcomers from fields like mathematics and computer science, such as myself and many others in this room. There are synthesis errors, unintended folds, and stuff sticking everywhere, and even a covalently perfect molecule will warp and wobble in the strangest ways. This can hardly be considered a good thing, but nonetheless I started to get the impression that Ned was coming to the view that being able to do a great many different things, imperfectly, was also of considerable merit, even if perfect control is forsaken. I'd like to use that position as a starting point for where DNA nanotechnology and molecular programming might be going.

If you examine two crystals of, say, diamond, they will be identical to each other down to the Angstrom level, differing only in relatively minor ways such as overall size and shape, or the locations of defects. They're basically superimposable. In contrast, consider two individual bacterial cells, say *E. coli*: no attempt to rotate and translate one to superimpose on the other will match more than a handful of atoms. The level of randomness in biological cells is astounding. Where are the chromosomes? Where are the enzymes? The ion channels? The flagella? And it's all moving about and rearranging constantly. Of course, many other things are nearly precisely constant and reliable—the sequence of the DNA paramount among them. But the point is that it can be hard even to draw the dividing line between what is precisely controlled and what is left to vary in every which way. Yet, that may be what we need to figure out how to do. I believe that it is only a matter of decades before DNA nanotechnology scales up to constructing molecular systems as large and as complex as biological cells, and that the biggest bottleneck to doing so will be knowing how to design such systems to be functional without over-specifying and

trying to control the parts and aspects of the system that do not need to be controlled—those things are better left to be random. It is my hope that design paradigms that reflect this distinction can be developed for DNA nanotechnology, and it is my guess that successful approaches will deeply borrow concepts from neural computation, machine learning, and constraint satisfaction programming: define the goals, not the details.

Moderator: That's it for the pre-arranged portion of this panel. For those of you who knew Ned, I hope it helped you remember him more deeply for a while. For those of you who didn't know him, I hope it inspired you with the desire to take some time to get to know him better. Read his papers. Find out about his life. A good place to start for the latter is the autobiography that's on the Kavli Prize web page or the related memoir that appeared in the American Crystallographic Association in 2014. You can also find online some extensive interviews by other scientists and by historians such as Patrick McCray, or, of course, obituaries such as the one by Phillip Ball in *Nature*.

Vignettes

Jim Canary: Ned's research and DNA nanotech expanded greatly as other structural tools (besides X-ray crystallography) such as AFM became available. The early designs, such as characterization of the DNA cube, required brilliant control experiments carefully designed and executed. Those experiments could not confirm structure but only topology. At the time of his passing, he was eager to learn more about cryo-electron microscopy. Make no mistake, though, that Ned's first love was X-ray crystallography. Ned was the most focused scientist that I have met. One day in his office I noticed that there were more than 15 cheap umbrellas scattered everywhere. When I asked about this, he said that when he leaves and finds out that it is raining, he buys an umbrella from one of the people selling them on the street. The point being that when he was leaving the office, he was not checking or thinking about the weather—or even looking out the window—he was still thinking about his work.

Paul Chaikin: Ned was a larger-than-life character who stood out even among the other characters that haunted the East Village, Chinatown, and Washington Square, places he enjoyed much more in the rough old days before they got gentrified. I first met Ned when I organized the Condensed Matter Physics Gordon Conference in 2001 with no theme but the idea that it would be great gather the most creative people. Albert Libchaber at Rockefeller told me “you have to invite Ned”. I picked Ned up at the train station in New London, he wanted to get a beer, and we spent the evening talking about DNA, knots, cubes, and why the hell, physicists couldn't find a way to “see”/prove his remarkable structures. I was sold and we started collaborating later that year. He's one of the reasons I relocated to NYU. When I got there, we started a group to see how difficult it would be to make our own non-living system that could self-replicate. From then on, the Self-Replication Group, SRG, met every

other week in his office, walking and sitting around seemingly random piles of old papers and books, dodging flimsy models of his structures hanging from the walls and ceiling. Amazingly, when Ned wanted to make a point, after a flurry of language reminiscent of my Brooklyn youth (Ned probably got it from Chicago's streets) he could always find the appropriate paper or toy. It was a great collaboration. Every crazy idea I had, Ned either had done it, knew how to do it, or figured it out by the next meeting. Ned was brilliant. The students and postdocs, who survived (and enjoyed) the spicy (putting it mildly) rants, loved Ned and learned how science was really done. Ned and I went to many meetings together, often by car, and I heard in typical Ned salt about the incompetence of all administrators and some colleagues. I heard many of his talks. He always warned me they were the same talk. I must have seen the slides "DNA: Not merely the secret of life" and Escher's "Depth" a hundred times.

Anne Condon: I remember clearly the first time I heard Ned speak, at an early DNA computing workshop. His contributions to DNA nanotechnology were inspiring, and the technical points of his talk were illuminated with colorful pictures of artifacts and puzzles that he had encountered on his travels. I was mesmerized! His personal story was also inspiring; I'm sure I wasn't the only one who appreciated his down to earth manner and his candor in recounting the rocky and unpredictable turns of his early career. A few years later, when I hosted Ned's visit to the University of British Columbia, Rob Scharein, a mathematical topology and knot visualization expert, attended the talk. I came to appreciate Ned's deep knowledge not only of chemistry but also of the mathematical aspects of topology. Ned stands out for me as someone who cared deeply about community, and he shaped ours for the better.

Lila Kari: Ned's contributions to DNA nanotechnology have been so profound and transformative, and his influence on multiple fields of life sciences and computational sciences so far-reaching, that I think Hollywood will eventually make a movie about him. I thought about that and imagined what such a movie would be like. I thought about Ned, about our many professional and personal interactions, including during his visit to London, Canada, for the "Unconventional Computation and Natural Computation" conference in 2014. (His invited talk was "DNA: Not merely the secret of life"—a wording that only Ned's mind, a mix of far-reaching scientific ideas, artistic bend, and sparkling wit, could conjure.) I thought then, what could the title be of this movie about Ned? How would I characterize Ned, in just a few words? Yes, he was a brilliant scientist. And yes, he was brash, direct, and used a lot of swear words. But, at the same time, there was always a palpable and unmistakable gentleness, fairness, and humility behind this rugged appearance. And then I saw it, in my mind's eye: The Hollywood movie about Ned's life story would be made, a blockbuster, and it would be called "A Scientist and a Gentleman". Because this is what Ned was. A scientist without equal, and a gentleman beyond words. We will miss you, Ned.

Tim Liedl: Needless to say how difficult it is to fit into a few words the immensity of impact Ned had on so many researchers and so many lives. On the one side he created visionary and fundamental science, of his own and of many dear colleagues who are part of his direct and indirect academic family. And at the same time he has

been an attentive and caring friend to us. I feel that his dedication to scientific reason and his eminent humanism might not be the only two factors making DNA nanotech such a great field to contribute to, but they are certainly the most important ones.

Paul Rothmund: In the late nineties when I was a graduate student, Ned graciously offered to let me stay at his apartment while I was visiting his lab. I remember a wall of books and finding Ned's favorite, *Slaughterhouse Five* by Vonnegut, from which he often quoted "so it goes". As a bed he offered up his couch, clearly unused for several years, as it was coated inches deep in the leaves of a nearby, dead, decorative ficus. As I curled up in the leaves I remember feeling excited to be on such a great adventure, getting to spend time with Ned in New York.

Ned seemed to get to roughly steady state, as a productive PI, for at least a couple decades. As a young scientist, trying to find a set of mechanics by which I could function within academics, I always found Ned's methods at least admirable, even when for some reason I found it hard to implement them myself. These mechanics include a lot of things: where and how to find and train students, where and how to apply for funding, with whom and how to collaborate, etc. I'll mention a couple that I remember as remarkable.

For the "where to find students", the answer was Beijing University. Many academics cultivate such a supply of talent; I have a Texan chemist friend who depends on Thai universities and know a German-Californian computer scientist whose large lab is predominantly South Korean. Some future historian will figure out what Ned's lab has meant for the diffusion of ideas back and forth between the USA and China, especially should DNA nanotechnology have large impact on health or electronics. I like to think that Ned's lab has had a positive effect on US-China relations at some scale.

For the "where to find funding", Ned drew from a variety of sources, the National Institutes of Health (NIH), the National Science Foundation, and the Office of Naval Research (ONR). NSF support for DNA nanotech is well documented, and Ned's earliest NSF grant dates to 1997, but Ned was the only person I have known who was able to keep his DNA nanotech program alive with NIH funding on a sustained basis (started in 1982), primarily through grants for studying the intermediates of biological recombination. Despite the evidently thrice yearly "box from hell" of grants to review, which he received from whichever panel he served on, Ned seemed to benefit greatly from the NIH. Early NIH grants are cited in his 1987 paper "The design of a biochip: a self-assembling molecular-scale memory device". Similarly Ned seems to be the one who made the original connection between the field of DNA nanotech and the ONR in 1989. Generations of scientists both within his lab and without (including my own lab) benefited from the enlightened largesse of the ONR, whose money was about as "no-strings-attached" as money can be. If DNA nanotechnology is ever successful at an industrial scale, ONR's early support of Ned's work should be lauded.

For many years Ned relied on Ruojie Sha for training, lab culture, and management. Their symbiosis made Ned's lab and its extraordinary output possible. From a funding point of view, having someone besides the PI to help maintain lab coherence is difficult (although NIH and biology seem more amenable to supporting such

a model). The churn of graduate students and postdocs makes many laboratories experience a sort of boom-and-bust cycle of productivity. The sustained success of laboratories like Seeman lab, which are able to pull off a PI-and-partner model, suggests to me that perhaps there should be more explicit support for those who would try it.

With respect to training graduate students, Ned had a formula for success. It began with a boot camp of sorts (in my memory this may have been anything from two weeks to two months), in which a student new to the lab had to learn all the techniques the lab was practicing, and reproduce its classical results. Early on I think this training involved techniques such as native polyacrylamide gels “formation gels” of multi-stranded complexes like double crossovers. Later I know it included experiments like atomic force microscopy of double-crossover lattices. This rite of passage was further insurance against boom-and-bust cycles in the lab and ensured continuity of lab protocols—had we at Caltech ever been able to implement such program it would have saved years of reinvention as new generations entered the lab.

After the training period, Ned would have a student work on an initial project that was carefully titrated to be incremental-yet-significant—a project that would test the mettle of the student, but would be guaranteed to yield a publication and safeguard the progress of the student toward a Ph.D. Success at this project would land the student a second, high-risk project—the sort of project that could open a new chapter for the lab and launch an independent career. Mediocre performance on the initial project would result in more incremental projects, intended to get the student safely out into the world. Ned articulated this strategy to me more than once as I struggled to manage students. My own interests and way of attempting to do science meant I couldn’t consistently implement Ned’s strategy; to a certain extent I consider Ned’s model a more general model for management and even childraising. Ned’s strategy provides a scaffolding, by which a neophyte can gradually find their way to independence, and I am still trying to consciously apply Ned’s model, or adaptations of it to different aspects of my life.

Ned was also vigilant and diligent with respect to the fortunes of young academics outside of his lab. His DNA nanotechnology track at the FNANO conference in Snowbird Utah was notoriously difficult to get into. Submissions for talks by senior researchers were routinely shot down. However, for any postdoc going out for faculty job talks or assistant professor giving a tenure tour, Ned would make special dispensation to get them a talk in his session.

Personally, I still haven’t found a set of mechanics by which I can sensibly function within academics, or more generally, navigate the daily vicissitudes of life. However, I find that independent of any details, I find Ned’s ethos and approach to science and life has deeply affected me. His being, and now having been, helps prop up a set values that I would like to keep, and I feel very grateful to have known him.

Greg Petsko: I will always remember Ned’s case because it included what is perhaps the most endearing description I’ve ever seen in a support letter. In his statement, a good friend of Ned’s wrote: “Ned dresses and looks like the mountain man Grizzly Adams—or, more precisely, like Grizzly Adams on a bad hair day. His beard gives the impression that a small mammal is eating his face. His language can

be, on occasion, shall we say, colorful (only my drill sergeant could use profanity more imaginatively). He is as honest as the day is long, wears his heart on his sleeve, and cares about what should be cared about. You may ask what any of that has to do with his science; I would answer that it is inseparable from his science. Someone more conventional in their appearance or more concerned about how they appeared to others would probably have followed a trend rather than created one. Someone less well-centered in terms of character might have been tempted to cut corners to achieve rapid success or to over hype their work. Ned would never do either”.

Niles Pierce: Ned was an inspirational scientist, gruff and humorous with a twinkle in his eye, and irreplaceable and warmly remembered.

Grzegorz Rozenberg: On November 16, 2021, I lost a dear friend and the scientific community suffered an irreplaceable loss: Ned Seeman passed away. I met Ned during the early days of DNA-based computation (this is the name that we used then) and, as both of us stated many times, there was an instant special chemistry between us. We certainly enjoyed being together as well as exchanging emails. He was a great scientist and a wonderful human being: curiosity driven, unpretentious, and genuinely natural. We never collaborated on research, but during the DNA11 conference in London, Ontario, in 2005 we had a long discussion on a topic that I suggested for our collaboration, and Ned was really positive about it. We agreed that I would visit him on one of my commutes between Leiden and Boulder so that we could work on this topic, but, regretfully, this never happened because of our very busy lives. We collaborated intensely on creating the International Society for Nanoscale Science, Computation and Engineering (with Ned becoming the founding president, while I served as the founding vice president). He was very enthusiastic about this project, and I witnessed his impressive professionalism and efficiency. We had a lot in common. For example, he enjoyed good jokes, and I always had a good supply of them. Also, he was an art lover. It is well known that he loved Escher, but he also liked Hieronymus Bosch. Since I have studied the paintings of Bosch (for over 50 years by now), I was able to explain many aspects of them. I even advised him which room in the Prado museum in Madrid he should visit (and why) and, indeed, he went there. As a matter of fact, I gave him many images of paintings by Bosch, which he clearly appreciated. We also talked a lot about magic. I am a performing magician, and Ned was very enthusiastic about my card magic. I told him about an important similarity between top magic and top science: both aim to approach the impossible as close as possible. He liked this explanation a lot. In fact, in our correspondence he often referred to my artist name, Bolgani, for example by writing “All the best to you and Bolgani”. When we exchanged new year wishes for 2020, he was hoping to travel (he expected to go to DNA 26 and to China) so we could meet again, and I could present to him my new magic show. Sadly, this did not happen. Let me end by saying that while Ned has vanished, his friendship will stay with me—this is what the magic of a true friendship is about.

Lulu Qian: The first time that I met Ned was at the DNA 10 conference in Milan. I was a first-year graduate student who had just found the field of DNA nanotechnology and molecular programming. When I walked into the lecture hall, my attention was immediately drawn to a man sitting on the floor, immersed in a laptop held on his

knees and a backpack sprawled next to him. I would have mistaken him for an undergrad who couldn't find a seat in a popular class, except that he had distinctive gray hair and a gray beard, and there were plenty of empty seats in the room. It was a memorable scene to me, and it became much more memorable later, when I found out that he was Ned Seeman, the father of the field. Since then, I have associated the image of a scientific giant with the image of a man who sits on the floor and disappears into his own world of thinking in a busy crowd.

Damien Woods: The two memories that come to mind are unpublishable. First one. Ned: "The way to kill a good project is give it to someone who doesn't move it forward". Second one. Young postdoc who (perhaps nervously, or at least quietly) approaches Ned with a criticism couched as a question: "I really like your paper in [major journal] but I have a question about how you analysed the data". Ned, gruffly: "That paper is a piece of shit". I'm paraphrasing the first. Not so much the second.

Andy Ellington: This is almost certainly not appropriate, but: one of the things I most loved about Ned was how profane he was. He was brilliant, he was erudite, he was artistic, and he could curse a blue streak. This was a man who had no fucks left to give about anything other than what was true and good and real and who knew exactly who he was. I miss him greatly.

Phil Lukeman: When Sir Christopher Wren, the great architect, physicist, and polymath, died, his epitaph was displayed in his magnum opus, St. Paul's Cathedral. The epitaph reads, "Si monumentum requiris circumspice" translated roughly as "If you would seek my monument, look around you". In my opinion Ned's legacy is comparable to Wren's—a legacy of de novo design principles for, and the construction of, startlingly elegant nanoscale objects. The objects themselves can't be seen with the naked eye of course; but the impact of his work in the field he founded—in the scientific literature and in the hundreds of scientists he trained and inspired—absolutely can. This volume is a small sampling of that impact. If you would seek his monument, look around you.

Preface

The idea of this book arose in response to a short paper by Ned Seeman, “DNA Nanotechnology at 40”, published in *Nano Letters* in February, 2020. The reference point is, of course, Ned’s 1982 paper on “Nucleic acid junctions and lattices”, or perhaps the moment two years earlier in an Albany bar when he first made the connection between DNA 6-arm junctions and Escher’s woodcut *Depth*. Either way, skipping forward roughly 40 years, Ned concluded his assessment with a humble but optimistic observation:

So where is the field going? The short answer is “I DON’T KNOW”. [. . .] Every day I open a journal and I’m surprised by another unit of progress in the nanoscale control of the structure of matter that DNA nanotechnology offers. I like being surprised that way.

We—and we are bold enough to speak both for ourselves as individuals and for the community as a whole—don’t know where the field is going either. However, we took Ned’s article as a sign that the time was right for the community as a whole to take stock of where we are and where we’re going. Coincidentally, Ned’s 75th birthday was coming up in December 2020, and we thought that starting this project would be a suitable way of honoring him and his contributions to science. Thus this project was born.

Our goal was to have a mixture of technical and non-technical material, reviews, tutorials, perspectives, and open questions—content that may not have a natural home in a journal but that may inspire current researchers to sit back and think about the big picture and that may entice new researchers to enter the field. What are the untold stories, the unspoken concerns, the underlying fundamental issues, the unappreciated opportunities, the unifying grand challenges? What will help us see more clearly, see more creatively, or see farther? What are the things going on right now that may pave the way for the future?

The contributions we received have been grouped into five parts.

Perspectives. The discussion is kicked off by Simon Vecchioni, Roujie Sha, and Yoel Ohayon, three of Ned’s collaborators at New York University. Having between them over four decades of experience in Ned’s laboratory, they illustrate concretely how Ned’s concept of “semantomorphic” science—how the information

in nucleic acid sequences gives rise to semantic meaning that directs formation of morphology—has guided and will continue to guide DNA nanotechnology in the future. This is followed by an essay on non-equilibrium molecular systems by Fritz Simmel, where he assesses both the exciting potential of molecular machines with dynamic behaviors—all the way to robots—and the challenges that have made progress in this direction so much slower than in structural DNA nanotechnology. An important subclass of dynamic behaviors is circuit computations, and developments in this area are reviewed by Fei Wang, Qian Li, and Chunhai Fan. They discuss issues such as scalable architectures, massive parallelism, randomness and search, spatial organization, and repeated or continuous operation. Satoshi Murata gives an overview of research in Japan, which started in the mid-1990s as DNA computing, grew steadily into molecular programming, expanded to molecular robotics, and now pioneers a vision of molecular cybernetics. In perhaps the most personal essay in this volume, Thom LaBean reminisces about his path from the science of random sequence proteins and protein engineering to the world of DNA nanotechnology and characters and concepts he met along the way.

Chemistry and Physics. DNA nanotechnology studies the actions of information in the material world, so understanding and exploiting the properties of the material world is at the heart of everything we do. Greg Tikhomirov asks whether this foundation is restricted to the chemistry and physics of nucleic acids per se, or whether a more general foundation can be found by generalizing to novel engineered information-bearing polymers—and how that might open the door to new applications. Rikke Hansen and Kurt Gothelf consider a very specific application: self-assembly of complex molecular electronic circuits. Here the strategy is to use information in DNA to direct the self-organization of homopolymers and oligomers that have useful electronic properties as single molecules. David Walker, Eric Szmuc, and Andy Ellington also explore using DNA to organize electronic circuits, but with different types of conductors: metallized DNA, carbon nanotubes, and more! In perhaps the most ambitious proposal in this volume, Bernie Yurke gives a theoretical outline for how DNA-organized layouts of optically responsive dyes could serve as the basis for quantum computing circuits.

Structures. As reviewed by Fei Zhang, nucleic acid nanotechnology is rapidly approaching the point where complex finite structures as large as a bacterial cell can be engineered and where periodic designer crystals can be grown large enough and robustly enough to serve the variety of applications envisioned by Ned four decades ago. Beyond applications, the structures that assemble in DNA nanotechnology have fascinating resonances in mathematics; Ellis-Monaghan and Jonoska emphasize connections to topology, graph theory, and algebra. Resonances with arts and crafts can also be quite striking—sometimes not only beautiful, but also scientifically insightful, as is elegantly articulated by Cody Geary for the case of paper-folding origami and cotranscriptional folding of RNA. A paper by Pierre Marcus, Nicolas Schabanel, and Shinnosuke Seki go further, envisioning a theoretical model wherein the kinetics of cotranscriptional RNA folding amounts to a programmable computation. A third theoretical contribution, from Matt Patitz, reviews the impressive range of models and questions that have been explored to understand the programmable

self-assembly of molecular components called tiles. In probably the most abstract chapter of this volume, Jack and Robyn Lutz muse about a form of reasoning—the counterfactual “as if”—that has had proven to be surprisingly powerful for understanding deterministic processes both in tile self-assembly and in distributed computing.

Biochemical Circuits. Beyond using information in DNA to direct the self-assembly of molecules into structures, DNA nanotechnology allows the principled construction of systems of molecular machines that dynamically interact with each other—otherwise known as information-processing circuits. While over the past decades, DNA strand displacement circuit implementations have gradually grown from a few to a few hundred computing gates, Yuan-Jyue Chen and Georg Seelig present a vision—with preliminary experimental results!—for scaling up circuit size by several orders of magnitude using array-based synthesis and high-throughput sequencing. Building on this approach, Luca Cardelli outlines a theoretical application circuit: recording the relative timing of a large number of distinct molecular events. In a software-oriented article, Matt Lakin, Carlo Spaccasassi, and Andrew Phillips give an overview of the past ten years of development of the Visual DSD system for specifying, analyzing, and compiling DNA strand displacement programs (and beyond). Their experience and vision is full of insights for the future of molecular programming.

Spatial Systems. Biochemical circuits have been studied predominantly in the well-mixed limit, where signals are carried by spatially uniform concentrations of distinct molecular species. Guillaume Gines, Anthony Genot, and Yannick Rondelez review experimentally demonstrated biochemical circuits that are spatially non-uniform—whether in reaction–diffusion systems or in droplets or other forms of compartmentalization—and discuss the potential for parallelism in computation and evolution. While Gines et al. touch on the connection between ecological systems and chemical systems—discussing a reaction–diffusion system that is an analog of a predator-prey ecosystem—Ming Yang and John Reif take this metaphor much further in a theoretical exploration of surface-bound single-molecule DNA nanorobots whose interactions are programmed to have collective behaviors similar to social insects. While Yang and Reif employ a stochastic cellular automaton model wherein each cell represents a single molecule, Masami Hagiya and Taiga Hongu take a larger-scale view wherein each cell represents a region of space within a programmable gel, and they propose algorithms for maze solving and for learning Boolean circuits. Returning back from theory-land to the realm of the real, the final chapter of this volume—by Beatrice Ramm, Alena Khmelinskaia, Henri Franquelin, and Petra Schwille—showcases the incredible and beautiful spatiotemporal patterning that develops on reconstituted lipid membranes in the presence of the *E. coli* MinDE system, delicately modulated by DNA origami nanostructures.

Of course, the logistics of putting a volume together, the disruptions of the pandemic, and the fact that so many of us are overcommitted means that many voices and topics that could have been added in this volume are, alas, missing.

Nevertheless, as incomplete as they might be, the diversity and depth of the contributions in this volume attest to the rich accomplishments and continued potential of Ned Seeman's vision for DNA nanotechnology. We can see that the past 40 years brought us from the concept of programmable DNA crystal hosts that organize guest molecules, to the practice of 3D DNA crystals, complex micron-scale structures, reconfigurable nanomechanical devices, programmable computing circuitry, primitive molecular robotic systems, and spatially organized materials—accompanied by increasingly sophisticated software and mathematics. And the next 40 years? We don't know! From the discussions in this volume, we might expect that integrating the many aspects of DNA nanotechnology will result in a dynamic, non-equilibrium, autonomous, cell-scale, molecular robot that can be programmed to sense, compute, and act. Or... more likely, we will be surprised!

Pasadena, CA, USA
Tampa, FL, USA
November 2022

Erik Winfree
Nataša Jonoska

Contents

Perspectives

Beyond Watson-Crick: The Next 40 Years of Semantomorphic Science	3
Simon Vecchioni, Ruojie Sha, and Yoel P. Ohayon	
DNA Nanotechnology Out of Equilibrium	17
Friedrich C. Simmel	
The Evolution of DNA-Based Molecular Computing	31
Fei Wang, Qian Li, and Chunhai Fan	
DNA Nanotechnology Research in Japan	47
Satoshi Murata	
Reminiscences from the Trenches: The Early Years of DNA Nanotech	59
Thomas H. LaBean	

Chemistry and Physics

Beyond DNA: New Digital Polymers	71
Grigory Tikhomirov	
Controlling Single Molecule Conjugated Oligomers and Polymers with DNA	87
Rikke Asbæk Hansen and Kurt Vesterager Gothelf	
Organizing Charge Flow with DNA	107
David J. F. Walker, Eric R. Szmuc, and Andrew D. Ellington	
DNA Assembly of Dye Aggregates—A Possible Path to Quantum Computing	125
Bernard Yurke	

Structures

Building with DNA: From Curiosity-Driven Research to Practice	173
Fei Zhang	

From Molecules to Mathematics	189
Joanna Ellis-Monaghan and Nataša Jonoska	

Origami Life	207
Cody Geary	

Ok: A Kinetic Model for Locally Reconfigurable Molecular Systems . . .	229
Pierre Marcus, Nicolas Schabanel, and Shinnosuke Seki	

Implementing a Theoretician’s Toolkit for Self-Assembly with DNA Components	241
Matthew J. Patitz	

Reasoning As If	271
Jack H. Lutz and Robyn R. Lutz	

Biochemical Circuits

Scaling Up DNA Computing with Array-Based Synthesis and High-Throughput Sequencing	281
Yuan-Jyue Chen and Georg Seelig	

Sequenceable Event Recorders	295
Luca Cardelli	

Computational Design of Nucleic Acid Circuits: Past, Present, and Future	311
Matthew R. Lakin, Carlo Spaccasassi, and Andrew Phillips	

Spatial Systems

Parallel Computations with DNA-Encoded Chemical Reaction Networks	349
Guillaume Gines, Anthony J. Genot, and Yannick Rondelez	

Social DNA Nanorobots	371
Ming Yang and John Reif	

Models of Gellular Automata	397
Masami Hagiya and Taiga Hongu	

Patterning DNA Origami on Membranes Through Protein Self-Organization	411
Beatrice Ramm, Alena Khmelinskaia, Henri G. Franquelin, and Petra Schwille	

Perspectives

Beyond Watson-Crick: The Next 40 Years of Semantomorphic Science



Simon Vecchioni, Ruojie Sha, and Yoel P. Ohayon

Abstract It should come as no surprise that the world of DNA nanotechnology is still learning how to fully master the different steps of the self-assembly process. Semantomorphic science, as the late Ned Seeman would describe DNA nanotechnology, relies on the programmability of nucleic acids (*semanto-*) to encourage short oligomers to put themselves together (*-morphic*) into designed architectures (*science?*). In the same way that Gibson assembly frustrates the molecular biologist, semantomorphic self-assembly has for decades, and continues to, defy the scientist in question. In a brief analogy, Gibson assembly can be thought of as enzymatically directed self-assembly [1] that follows the same general rules as Seeman assembly: (1) guess conditions; (2) set up reaction; (3) pray to entity of choice; (4) check result; and (5) repeat as needed. In other words, when it works, it works well; when it doesn't, troubleshooting the sticky-ended cohesion between too-large or too-small building blocks with imperfect assays can take months. Returning to semantomorphic science, it is still mesmerizing that any of this works at all, and for that, we owe our deepest gratitude to Ned and his generations-spanning vision.

1 A Brief Retrospective

If we look back at the last forty years of work done in Ned's Lab, we can roughly break this period into decades, anchored to the founding ideology. In 1982, Ned proposed the idea of using self-assembling DNA oligomers as crystalline scaffolds for the structure determination of biomolecules [2]. By 1991, with Junghuei Chen's

S. Vecchioni (✉) · R. Sha · Y. P. Ohayon
Structural DNA Nanotechnology Laboratory, Department of Chemistry, New York University,
New York, NY 10003, USA
e-mail: sv1091@nyu.edu

R. Sha
e-mail: rs17@nyu.edu

Y. P. Ohayon
e-mail: yoel.ohayon@nyu.edu

development of the first complex topological object, a DNA cube, it was apparent that Ned's vision *could* be possible [3]. In 1998, Ned worked with Erik Winfree to establish that periodic 2D DNA crystals were attainable through self-assembly [4]. This study and the ensuing cluster of papers on 2D lattices [5–9] may be described as a time when it was clear that Ned's vision *should* be possible. The following decade brought nanomechanical devices (Chengde Mao's B-Z device [10], Bernard Yurke's nano-tweezers [11], and Hao Yan's PX-JX₂ actuator, [12] among others), DNA origami [13] and LEGO-ology [14], and a vast proliferation in the complexity of self-assembled shapes and devices. In 2004, Chengde Mao published the tensegrity triangle motif [15], which Jianping Zheng modified in 2009 to self-assemble into the first 3D macroscopic DNA crystal [16], demonstrating that rigid, simple, and programmable self-assembly could be carried out in 3D: Ned's vision *would* be possible after all.

The start of the 2010s brought about an expansion of complexity: devices involving crystalline logic gates [17], self-replicating DNA machines with Paul Chaikin [18], and natural DNA computing with Natasha Jonoska [19, 20]; and for the first time, these devices were no longer able to be exchanged between laboratories in the same way—the outputs and techniques had now become so complex that composable parts were no longer as simple or modular. Rather than a single motif or reaction, DNA nanodevices had begun to involve a vast number of moving parts, atypical constituents, and specialized conditions. Describing Hongzhou Gu's nanoparticle assembly line [21] for grant applications remains difficult to this day, let alone the modifications necessary to change its function and the critical role that this device plays as a use case for DNA nanotechnology. The current state of semantomorphic science in the Seeman Lab, with the anticipated publication of our first biomolecular structures attained through 3D DNA crystals (manuscript in preparation at the time of writing), is that Ned's vision *is* possible. Not only is it possible, but the feedback gained through modification of simple parts and predetermined solutions to the diffraction phase problem can allow rapid screening of a target library in a way that is impractical using state-of-the-art crystallography. While the advent of cryogenic electron microscopy has already shaken traditional crystallography with promises of imminent obsolescence, there are use cases for Ned's 3D diffraction lenses that we believe are critical for the next 40 years—particularly in the realm of non-canonical, non-Watson-Crick architectures, which are chronically overlooked by dogmatic crystallographic science. Watson-Crick interactions can be, in broad strokes, defined as double helical interactions involving G:C and A:T base pairs. From this point on, we believe that eclipsing Watson-Crick semantics will allow for technologically relevant morphologies, i.e., semantomorphic science as Ned once envisioned it.

2 A Science Allegory

Ned's last publication in 2021 was Brandon Lu's hexagon [22], a structure that can in many ways be thought of as a representation of DNA nanotechnology more broadly. In the outward-facing story of the hexagon, it was discovered that use of non-Watson-Crick sticky-ends in a tensegrity triangle caused non-canonical cohesive interactions involving A:C and G:T. Rather than inhibiting self-assembly, this non-Watson-Crick "mismatch" drives the growth of much larger and, apparently, more thermodynamically-favored crystals with bent Holliday junctions, or branch points. The crystals possess a pseudo-infinite hexagonal channel along the crystallographic $P6_3$ screw axis, presenting a technological opportunity for nanoscopic organization of composable parts in a metachiral system (see Sect. 4 for elaboration).

Preceding this publication came years of head-scratching and some tunnel vision by way of optimism. Ned once said that the use of a microscope—any kind of microscope—was a heuristic trap: the observer would, more often than not, see whatever they were looking for. In the world of self-assembly, this led to almost fifteen years of curating rhombohedral—and only rhombohedral—crystals. Lurking in the corners of thesis figures and oral histories since the first days of Ned's 3D assemblies are needle-like, chunky crystals that appeared at first glance to be failure products. As the designs in our laboratory became more complex, the chemistry further from Watson-Crick, and the geometries more distant from B-form double helices, we began seeing crystals with jagged, improper shapes and space group "failures" in the X-ray diffraction patterns (see Fig. 1). Hao Yan published a similar study in collaboration in 2016, in which a "tensegrity square" became an unwound, flower-like design with $P3_2$ symmetry [23]. In the summer of 2020, the conundrum of the hexagon began a rapid convergence. Updated processing software and the simultaneous result of three separate projects with $P6_3$ symmetry made it clear that our "accidents" and "failures" were telling us something. In abject disregard for Ned's adage "garbage in, garbage out," Brandon Lu and Karol Woloszyn—two extremely talented researchers in the laboratory—on the same day obtained molecular replacement results that showed tensegrity triangles arranged in a chair-hexagon conformation.

It was at that moment starkly clear that self-assembly had produced something different: packing effects had imposed curvature on the double helix to form a corkscrew-like architecture from a linear monomer. Nature finds a way, especially when trying to impose Watson-Crick symmetry on strained oligomers engaging in the dubious process of self-assembly. But it is precisely this departure from Watson-Crick that uncovered a new mode of self-assembly—a change to DNA primary sequence had profound implications on the quaternary structure of those oligomers. To this end, semantomorphic science may find its next relevant morphologies by altering its core semantics and by attending to the unexpected results.

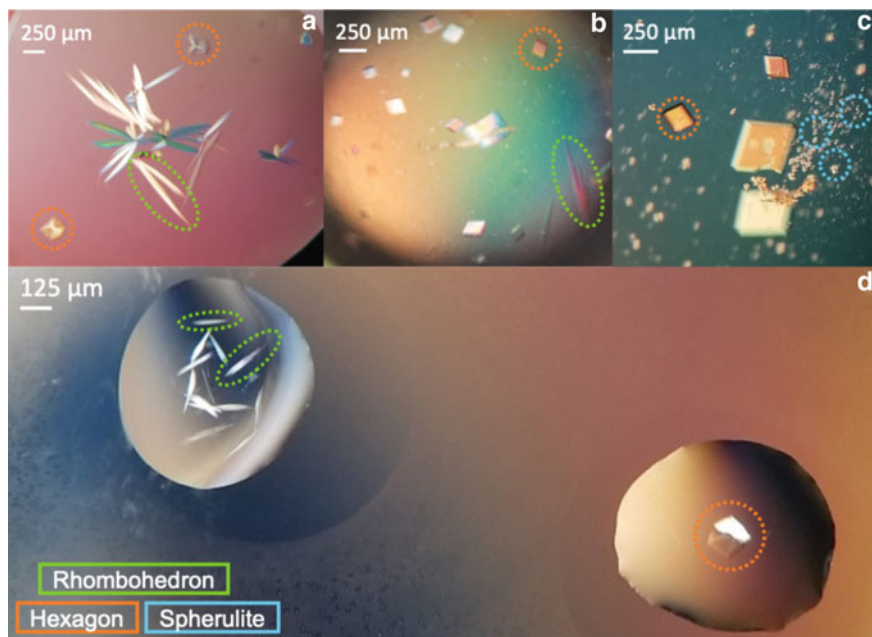


Fig. 1 Self-assembly of tensegrity triangles [24]. **a–c** Various crystallization conditions generate rhombohedra (orange), hexagons (green), and spherulites (blue) in the same drop, despite general optimism for rhombohedral assembly. **d** Two identical hanging drops on the same glass slide, with hexagonal crystals (left) and rhombohedral crystals (right) attained simultaneously. It is clear that macroscopic effects can result from minute differences in the crystal microenvironment

3 A Roadmap

We envision the departure from Watson-Crick architectures through (1) modification of *DNA semantics* by way of new base pairing rules; (2) the augmentation of *DNA syntax* leading to diverse secondary and tertiary structures; and (3) the expansion of the *DNA operating system* through modifications to the sugar-phosphate backbones. We elaborate upon these categories below.

3.1 *DNA Semantics: Schrödinger Crystals Versus Seeman Crystals*

In meetings and conferences, Ned would often repeat that “symmetry is the *opposite* of control!” As it is well known, the very first step toward designing and programming synthetic branched junctions involved the development of SEQUIN, a FORTRAN algorithm explicitly implemented for symmetry minimization [25]. In a computer

science sense, this can be thought as the greatest common substrating optimization algorithm, and in a biochemical sense the free energy minimization of desired secondary structures. Schrödinger posited in his 1951 treatise, “Was ist Leben,” that the fundamental nature of life required an aperiodic crystal as the carrier of genetic information [26]. Only by minimizing repeating segments of information in a geometrically isomorphic crystal would living systems be able to propagate the stored information between generations. Schrödinger’s postulation preceded SEQUIN by four decades, but proposed a similar principle of symmetry minimization contained in topological units unperturbed by the information carried within.

By contrast, Seeman crystals can be thought of as *mostly periodic crystals*, whose sequences repeat (tesselate) in the semantomorphic bundles that we call motifs. These motifs are subject to crystallographic symmetries to generate theoretically infinite nanomaterials. Within this system, local symmetry minimization is critical in imposing long-range order: Schrödinger’s aperiodic crystal must be tightly constrained. The tensegrity triangle, Ned’s crowning joy, contained a periodic unit with only 42 nucleotides (nt). Geometric asymmetry, in the form of 3:3:1 annealing ratios of three interwoven oligomers, counteracts the rotational symmetry of the triangle arms to allow for sequence symmetry down to 42 nt (Fig. 2a vs. Fig. 2c). A similar motif with asymmetric sequences across the triangle and no rotational symmetry between arms contains 126 nt in each periodic unit for the same unit cell, this time with symmetric strand ratios between seven unique oligomers (1:1:1:1:1:1:1) (see Fig. 2d). There is a clear exchange between geometric and sequence symmetries—penalties in sequence minimization require more complex geometry, which in turn makes experiments more difficult and the analysis more complex. It is clear that Schrödinger’s prescription holds for DNA nanotechnology and can be amended for Seeman crystals: *the more aperiodic a crystal’s sequence, the more symmetric the geometry can be*. With this in mind, expanding the genetic code will have a clear effect on the types of structures that nanotechnologists can engineer.

There are many existing strategies for expanding “vanilla DNA”—as Ned would describe Watson-Crick chemistry. The most obvious lies in nucleobase analogs, which typically employ rearranged hydrogen bonding groups: inosine (I) [24], isocytosine (S), isoguanine (B) [27], 2,6-diamino-purine, 2,4 diamino-pyrimidine [28], and “hachimoji” nucleotides dZ and dP [29]. Indeed, very extensive applications of these nucleotides have been carried out by generations of Ned’s students, and it is clear that they satisfy the requirements for Seeman crystals [30]. The full implementation of an expanded lexicon has, to our knowledge, not been carried out in a self-assembly context.

A further development lies in metal-mediated DNA base pairing (mmDNA), which has emerged over the last two decades as a viable means to add symmetry (reducing the aperiodicity of DNA crystals) by allowing non-Watson-Crick pyrimidine:pyrimidine stabilization through the coordination of diverse metal ions (most commonly Ag^+ and Hg^{2+}) [31, 32]. The payoff here lies in reducing the regularity of the overall behavior—metal base pairs are known to be pseudo-covalent, stabilize the helix, and add novel electronic and magnetic properties [33–35], all of which

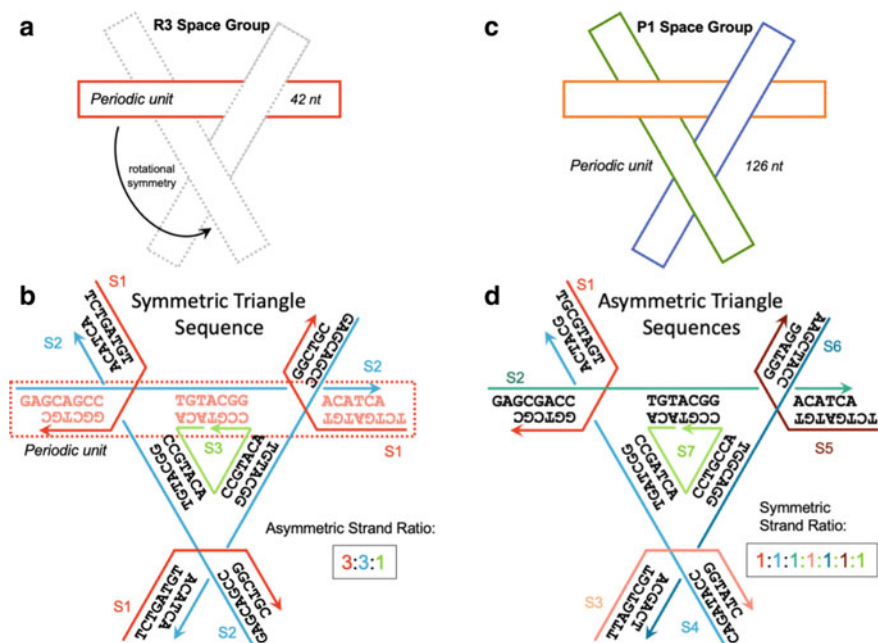


Fig. 2 Tensegrity triangle motifs. **a–b** The symmetric two-turn triangle developed by Jianping Zheng employs a repeating 42 nt unit that attains higher-order complexity by asymmetric strand ratios (strands S1–S3) and sequence weaving. **c–d** The asymmetric version of this motif, developed by Rachel Chernet, only repeats over 126 nt, but involves easier annealing with symmetric and logical strand ratios (strands S1–S7)

represent a general departure from hydrogen bonding as the sole means of implementing Schrödinger and Seeman crystals. By adding bioinorganic diversity to the properties and behaviors of nucleic acids, there lies an opportunity to increase the complexity of DNA structural motifs—more than compensating for the increase in information entropy. A main focus of the work in Ned’s group lies in these types of architectures, and we look forward to sharing Ned’s latest works with the community in the coming months and years.

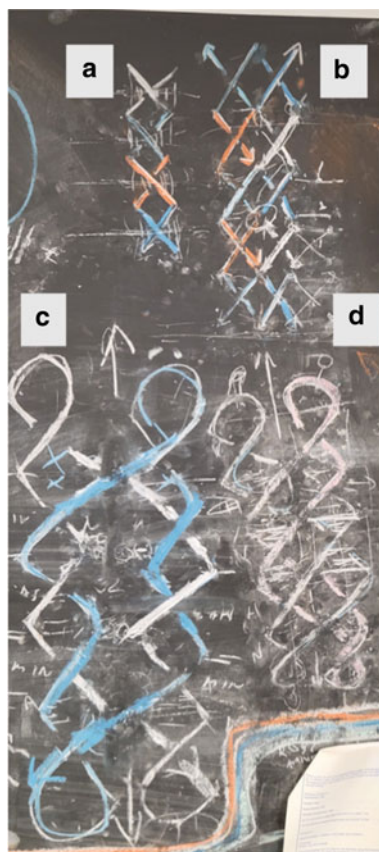
We envision that the expansion of the coding language of DNA will enable more precise, geometrically minimized, and behaviorally complex semantomorphism, and we have only begun to scratch the surface of these techniques.

3.2 DNA Syntax—Information Bundles and Secondary Structures

With the advent of the first functional poly-crossover (PX) motifs by Zhiyong Shen [36], Ned found a way to augment the complexity of DNA branches (Fig. 3b). Beyond

topologically-closed knots and catenates [37], PX motifs represent a wonderful conundrum of the double helix that has driven the creation of new dynamic devices. For years, Ned tried to convince biologists to utilize PX, to search for it in biological systems that Xing Wang demonstrated [38]. The utility of PX DNA was made clear with the PX-JX₂ device, first designed by Hao Yan in 2002, in which geometric control was attainable through the organization of DNA within differentially shaped information packets [12]. Through the addition of small changes attained through strand displacement, it was possible to reorganize the whole semantomorphic structure—this represents a fundamental change in the grouping and behavior of information, and in the same way syntax dictates the order and function of words in a sentence (Fig. 3). To this end, expanding the syntax of DNA nanotechnology will enable more complex materials in the coming decades. A road to syntactic expansion follows Ned's ideas, involving hairpins, triplexes, quadruplexes, and yet-to-be -discovered or developed secondary structures.

Fig. 3 Archival chalk drawings from Ned's office of various DNA secondary structures. **a** Duplex DNA; **b** PX DNA; **c** JX₂ structure; **d** PX DNA with hairpins



The hairpin was one of the earliest tools in the semantomorphic book of style—Hao Yan used this technique to give a 2D crystal the power of nano-actuation: by interfacing with strand displacement, hairpins can act as geometric parentheses, putting topological information on hold until it is needed [39]. Hao's motif changed shape like an omni-directional accordion, performing nanoscale work stored in hairpin parentheticals. Hairpin technology has been extensively explored, but likely has new tricks and interfaces to tell us, especially when implemented with other syntactical elements.

Triplex-forming oligomers (TFOs) exploit the major groove of pyrimidine-enriched DNA sequences to bind a third strand. The overall architecture of the DNA is likely altered to some degree, as Seeman crystals do not form with TFOs during self-assembly—there is, however, no difficulty in decorating these crystals with TFOs after the fact. In this way, the effect of a TFO is not unlike an exclamation mark—placed properly it can manifest a new behavior in an existing sentence. This technology, developed in collaboration with David Rusling, has been interfaced with various attachment chemistries [40] and shows strong promise for adding reporter and sensor behavior to semantomorphic structures. A complete structural description of TFO-bound DNA nanostructures has not been attained, but it is clear that these motifs will enhance and emphasize DNA technologies in the coming years.

It is known that repeats of guanine bases can interrupt the double helix by forming a weaving tetraplex structure (G4), and Ruojie Sha and Nongjian Tao built on this idea to show that G4 molecules are excellent electronic conductors [41, 42]. Unlike Watson-Crick DNA, which could be generously described as a weak resistor (or a fantastic molecular heater), G4 motifs are now known to be wide-bandgap semiconductors. In essence, these structures bind alkali cations between poly-guanine stacking planes and impart electronic functionality to an otherwise inert DNA. This structural organization changes both the orientation and the underlying electron delocalization of the material—in this way, G4 acts as a semicolon. To be useful in a semantomorphic sense, it must come in the middle of a DNA sentence, but it changes the focus and structure of the surrounding information. One does not use a semicolon lightly; and conversely, G4 must be used sparingly for its symmetry penalties and strong departure from Watson-Crick helical parameters. The implementation of G4 within structural DNA motifs has been sparing, but as designs become more complex, we can expect G4 to become increasingly common in the future.

There exist other secondary structures that rely on semantomorphism [43], such as aptamers [44], i-motifs [45], and of course, paranemic cohesion (PX motifs). The future of DNA-based technologies will strongly rely on sequence context and behavior, and the ability to code for novel heterostructures is a powerful and underexplored aspect of semantomorphic science.

3.3 *Nucleic Acid Operating Systems: XNA and Beyond*

If polymerases can be considered the read and write heads of the DNA hard drive, the metaphor can be extended to consider the sugar-phosphate backbone as the operating system in which nucleobase information is stored. Entire classes of enzymes are employed to extract DNA information into the RNA operating system and furthermore translate that information into peptides. Alterations to the nucleic acid operating system generate what are known as xenobiotic nucleic acids (XNAs) [46], and they offer what any orthogonal digital platform can: data encryption, novel behaviors and features, and the need to rewrite basic programs.

Encryption is a result of enzymatic orthogonality—with a different form of aperiodic crystal, information can neither be read nor copied and therefore cannot escape from a synthetic biological system encoded using XNA [47]. This concept has powerful implications for synthetic polymers and synthetic life. The novel features result from the chemical modifications present in XNA via the addition of hydrophobic groups, linkers, and various anchors and substituted sugar moieties [46], leading to different interactions with the surrounding solvent and ultimately presenting the opportunity to program interfaces with chemistry beyond neutral (physiologically adjacent) aqueous systems. Finally, a departure from Watson-Crick geometry changes the shape of the ensuing molecule, including the helical period, the radial dimensions, and the overall pitch of the nucleobases. In order to be successful in achieving nanotechnological programming in an XNA system, the connection between semantics and morphology must be painstakingly re-established for each polymer. The benefits of this expansion have been clear for some time, and work toward structural DNA(+) nanotechnology has been carried out robustly for at least a decade.

Cody Geary and colleagues demonstrated this concept by folding RNA nanostructures while they were transcribed from a DNA template [48]. This represented a fundamental shift toward “living nanotechnology” or “in vivo nucleic acid nanotechnology” by showing that the RNA operating system could be (occasionally) relied upon to generate semantomorphic structures. Ned implemented true XNA backbones within double-crossover motifs, measuring the double helical periodicity of peptide nucleic acids (PNA) [49] and 2'-fluoro-deoxyribonucleic acid (F-DNA) [50]. PNA has subsequently been shown to operate well in organic solvents [51], which has long considered a barrier to self-assembled architectures predicated upon hydrogen bonding.

Ned and Jim Canary at NYU have been collaborating since 1997 to attach organic polymers to the sugars of 2'-O-propargyl-modified ribonucleotides. This approach employs a sterically-unhindered linking site to “plug in” a variety of molecules to the double helix. This was most elegantly demonstrated by Xiao Wang through polyaniline/emeraldine functionalization of a DNA cage that imparted optoelectronic functionality into Seeman crystals [52]. This has been further extended into nylon-DNA [53] and tertiary structure-like crosslinking [54].

Further orthogonal backbones of interest might overcome charge polarity in the phosphates, impart an integer number of nucleotides per helical turn (to avoid the “curse” of 10.5 bp/turn), increase thermostability, frustrate endonucleases, provide a sequence-agnostic attachment strategy for nanomaterials, or even impart electrical or magnetic functionality to the polymer of life [46, 55]. It is clear that backbone modifications are an underexplored, technically difficult, but rewarding addition to nucleic acid nanotechnology. With any operating system change, the new environment must be first explored, structurally characterized, and tested. As with PNA and RNA nanotechnologies, future XNAs can be expected to mature and allow a fully-elucidated and adaptable semantomorphic coding environment.

4 Beyond Watson-Crick: A Call to Action

The benefits of escaping Watson-Crick geometry are hopefully made clear: semantomorphic science stands to gain (1) a much greater *geometric* diversity through a more versatile genetic code; (2) expanded diversity through syntactic groupings of bases employed in new secondary structures (see Fig. 4); (3) orthogonality to enzymatic and aqueous systems through altered backbone operating systems; and (4) an integration of these various approaches to impart novel mechanical, material, electronic, or catalytic behaviors into semantomorphic polymers. As with DNA nanotechnology in its early days, the changes required to carry out this paradigm shift will require an immense amount of experimental exploration and community action; but, unlike the first forty years, the next forty years can rely on the incredible foundation laid out by the pioneering work of Ned Seeman. It is with the deepest gratitude that we remember Ned and his mission and the painstaking groundwork that he and his early team performed to create the field from scratch. With the guidance left in his writings and with his students, colleagues and collaborators across the globe, there exists a full and flourishing roadmap forward. In his work, “DNA Nanotechnology at 40,” Ned concluded “Every day I open a journal and I’m surprised by another unit of progress ... I like being surprised that way.” [56] Let us continue to surprise him.

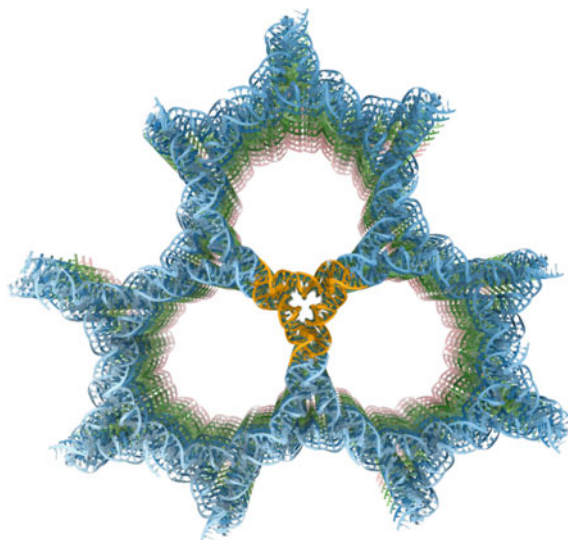


Fig. 4 Self-assembly beyond Watson-Crick: cross-section of 3D hexagonal DNA assembly, PDB 7R96 [22]. The periodic unit forms a tensegrity triangle (orange center, see Fig. 2) which packs into a star-like lattice. The space group $P6_3$ indicates a sixfold symmetry in each channel (see the hexagon-like nature of the triangle arrangement) and that three layers of the material are required in the z-direction for the meta-helix to make a full turn around the cavity (the crystallographic “screw axis”). This discovery was a joyful surprise to Ned

References

1. D.G. Gibson et al., Enzymatic assembly of DNA molecules up to several hundred kilobases. *Nat. Methods* **6**, 343–345 (2009)
2. N.C. Seeman, Nucleic-acid junctions and Lattices. *J. Theor. Biol.* **99**, 237–247 (1982)
3. J. Chen, N.C. Seeman, Synthesis from DNA of a molecule with the connectivity of a cube. *Nature* **350**, 631–633 (1991)
4. E. Winfree, F.R. Liu, L.A. Wenzler, N.C. Seeman, Design and self-assembly of two-dimensional DNA crystals. *Nature* **394**, 539–544 (1998)
5. C.D. Mao, W.Q. Sun, N.C. Seeman, Designed two-dimensional DNA Holliday junction arrays visualized by atomic force microscopy. *J. Am. Chem. Soc.* **121**, 5437–5443 (1999)
6. T.H. LaBean et al., Construction, analysis, ligation, and self-assembly of DNA triple crossover complexes. *J. Am. Chem. Soc.* **122**, 1848–1860 (2000)
7. N.C. Seeman, A.M. Belcher, Emulating biology: building nanostructures from the bottom up. *P. Natl. Acad. Sci.* **99**, 6451–6455 (2002)
8. P. Sa-Ardylen, A.V. Vologodskii, N.C. Seeman, The flexibility of DNA double crossover molecules. *Biophys. J.* **84**, 3829–3837 (2003)
9. F. Liu, R. Sha, N.C. Seeman, Modifying the surface features of two-dimensional DNA crystals. *J. Am. Chem. Soc.* **121**, 917–922 (1999)
10. C. Mao, W. Sun, Z. Shen, N.C. Seeman, A nanomechanical device based on the B-Z transition of DNA. *Nature* **397**, 144–146 (1999)
11. B. Yurke, A.J. Turberfield, A.P. Mills, F.C. Simmel, J.L. Neumann, A DNA-fuelled molecular machine made of DNA. *Nature* **406**, 605–608 (2000)
12. H. Yan, X. Zhang, Z. Shen, N.C. Seeman, A robust DNA mechanical device controlled by hybridization topology. *Nature* **415**, 62–65 (2002)

13. P.W. Rothemund, Folding DNA to create nanoscale shapes and patterns. *Nature* **440**, 297–302 (2006)
14. Y. He et al., Hierarchical self-assembly of DNA into symmetric supramolecular polyhedra. *Nature* **452**, 198–201 (2008)
15. D. Liu, M. Wang, Z. Deng, R. Walulu, C. Mao, Tensegrity: construction of rigid DNA triangles with flexible four-arm DNA junctions. *J. Am. Chem. Soc.* **126**, 2324–2325 (2004)
16. J. Zheng et al., From molecular to macroscopic via the rational design of a self-assembled 3D DNA crystal. *Nature* **461**, 74 (2009)
17. Y. Hao et al., A device that operates within a self-assembled 3D DNA crystal. *Nat. Chem.* **9**, 824–827 (2017)
18. T. Wang et al., Self-replication of information-bearing nanoscale patterns. *Nature* **478**, 225–228 (2011)
19. P. Sa-Ardyen, N. Jonoska, N.C. Seeman, Self-assembly of irregular graphs whose edges are DNA helix axes. *J. Am. Chem. Soc.* **126**, 6648–6657 (2004)
20. G. Wu, N. Jonoska, N.C. Seeman, Construction of a DNA nano-object directly demonstrates computation. *Biosystems* **98**, 80–84 (2009)
21. H. Gu, J. Chao, S.-J. Xiao, N.C. Seeman, A proximity-based programmable DNA nanoscale assembly line. *Nature* **465**, 202–205 (2010)
22. B. Lu et al., 3D hexagonal arrangement of DNA tensegrity triangles. *ACS Nano* **15**, 16788–16793 (2021)
23. C.R. Simmons et al., Construction and structure determination of a three-dimensional DNA crystal. *J. Am. Chem. Soc.* **138**, 10047–10054 (2016)
24. Y.P. Ohayon et al., Designing higher resolution self-assembled 3D DNA crystals via strand terminus modifications. *ACS Nano* **13**, 7957–7965 (2019)
25. N.C. De Seeman, novo design of sequences for nucleic acid structural engineering. *J. Biomol. Struct. Dyn.* **8**, 573–581 (1990)
26. E. Schrödinger, *Was ist Leben?* Vol. 1 (Lehnen München, 1951)
27. O. Bande et al., Isoguanine and 5-methyl-isocytosine bases, in vitro and in vivo. *Chem. Eur. J.* **21**, 5009 (2015)
28. I. Singh et al., Structure and Biophysics for a Six Letter DNA Alphabet that Includes Imidazo [1, 2-a]-1, 3, 5-triazine-2 (8 H)-4 (3 H)-dione (X) and 2, 4-Diaminopyrimidine (K). *ACS Synth. Biol.* **6**, 2118–2129 (2017)
29. S. Hoshika et al., Hachimoji DNA and RNA: A genetic system with eight building blocks. *Science* **363**, 884–887 (2019)
30. C. Hernandez, *Couture Crystal Constructs: Self-Assembly of Three-Dimensional DNA Crystalline Lattices*. (New York University, 2016).
31. A. Ono et al., Specific interactions between silver(I) ions and cytosine-cytosine pairs in DNA duplexes. *Chem. Commun. (Camb.)*, 4825–4827 (2008)
32. Y. Miyake et al., MercuryII-mediated formation of thymine-HgII-thymine base pairs in DNA duplexes. *J. Am. Chem. Soc.* **128**, 2172–2173 (2006)
33. G.H. Clever, S.J. Reitmeier, T. Carell, O. Schiemann, Antiferromagnetic coupling of stacked CuII–salen complexes in DNA. *Angew. Chem. Int. Edit.* **49**, 4927–4929 (2010)
34. E. Toomey et al., Comparison of Canonical versus Silver(I)-Mediated Base-Pairing on Single Molecule Conductance in Polycytosine dsDNA. *J. Phys. Chem. C* **120**, 7804–7809 (2016)
35. H. Torigoe et al., Thermodynamic and structural properties of the specific binding between Ag(+) ion and C: C mismatched base pair in duplex DNA to form C–Ag–C metal-mediated base pair. *Biochimie* **94**, 2431–2440 (2012)
36. Z. Shen, *DNA Poly-Crossover Molecules and Their Applications in Homology Recognition* (New York University, 1999)
37. Y.P. Ohayon et al., Topological linkage of DNA tiles bonded by paranemic cohesion. *ACS Nano* **9**, 10296–10303 (2015)
38. X. Wang, X. Zhang, C. Mao, N.C. Seeman, Double-stranded DNA homology produces a physical signature. *Proc. Natl. Acad. Sci.* **107**, 12547–12552 (2010)

39. L. Feng, S.H. Park, J.H. Reif, H. Yan, A two-state DNA lattice switched by DNA nanoactuator. *Angew. Chem. Int. Edit.* **115**, 4478–4482 (2003)
40. S.D. Osborne et al., Selectivity and affinity of triplex-forming oligonucleotides containing 2'-aminoethoxy-5-(3-aminoprop-1-ynyl) uridine for recognizing AT base pairs in duplex DNA. *Nucleic Acids Res.* **32**, 4439–4447 (2004)
41. R. Sha et al., Charge splitters and charge transport junctions based on guanine quadruplexes. *Nat. Nanotechnol.* **13**, 316–321 (2018)
42. G.I. Livshits et al., Long-range charge transport in single G-quadruplex DNA molecules. *Nat. Nanotechnol.* **9**, 1040–1046 (2014)
43. N.C. Seeman, DNA nicks and nodes and nanotechnology. *Nano Lett.* **1**, 22–26 (2001)
44. C. Lin, E. Katilius, Y. Liu, J. Zhang, H. Yan, Self-assembled signaling aptamer DNA arrays for protein detection. *Angew. Chem. Int. Edit.* **118**, 5422–5427 (2006)
45. W. Wang et al., Reconfigurable two-dimensional DNA lattices: static and dynamic angle control. *Angew. Chem. Int. Edit.* **20**, 25781–25786 (2021)
46. V.B. Pinheiro, P. Holliger, Towards XNA nanotechnology: new materials from synthetic genetic polymers. *Trends Biotechnol.* **32**, 321–328 (2014)
47. M. Vanmeert et al., Rational design of an XNA ligase through docking of unbound nucleic acids to toroidal proteins. *Nucleic Acids. Res.* **47**, 7130–7142 (2019)
48. C. Geary, P.W. Rothmund, E.S. Andersen, RNA nanostructures. A single-stranded architecture for cotranscriptional folding of RNA nanostructures. *Science* **345**, 799–804 (2014)
49. P.S. Lukeman, A.C. Mittal, N.C. Seeman, Two dimensional PNA/DNA arrays: estimating the helicity of unusual nucleic acid polymers. *Chem. Comm.*, 1694–1695 (2004)
50. Uddin. The helicity of a DNA-2'-fluoro DNA hybrid duplex structure. *Int. J. Nanotechnol. Nanomed.* **2**, 1–3 (2017)
51. A. Sen, P.E. Nielsen, On the stability of peptide nucleic acid duplexes in the presence of organic solvents. *Nucleic Acids. Res.* **35**, 3367–3374 (2007)
52. X. Wang et al., An organic semiconductor organized into 3D DNA arrays by “bottom-up” rational design. *Angew. Chem. Int. Edit.* **56**, 6445–6448 (2017)
53. L. Zhu, P.S. Lukeman, J.W. Canary, N.C. Seeman, Nylon/DNA: single-stranded DNA with a covalently stitched nylon lining. *J. Am. Chem. Soc.* **125**, 10178–10179 (2003)
54. M. Ye et al., Site-specific inter-strand cross-links of DNA duplexes. *Chem. Sci.* **4**, 1319–1329 (2013)
55. G.F. Joyce, Toward an alternative biology. *Science* **336**, 307–308 (2012)
56. N.C. Seeman, DNA nanotechnology at 40. *Nano Lett.* **20**, 1477–1478 (2020)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



DNA Nanotechnology Out of Equilibrium



Friedrich C. Simmel

Abstract Dynamic DNA nanotechnology aims at the realization of molecular machines, devices, and dynamic chemical systems using DNA molecules. DNA is used to assemble the components of these systems, define the interactions between the components, and in many cases also as a chemical fuel that drives them using hybridization energy. Except for biosensing, applications of dynamic DNA devices have so far been limited to proof-of-concept demonstrations, partly because the systems are operating rather slowly, and because it is difficult to operate them continuously for extended periods of time. It is argued that one of the major challenges for the future development of dynamic DNA systems is the identification of driving mechanisms that will allow faster and continuous operation far from chemical equilibrium. Such mechanisms will be required to realize active molecular machinery that can perform useful tasks in nanotechnology and molecular robotics.

1 DNA Nanotechnology: A Personal Account

I stumbled into DNA nanotechnology at the turn of the millenium and since then actively participated in about half of its 40 year history which is celebrated here. With a background in solid-state nanophysics—having spent time in clean rooms for the fabrication of semiconductor nanodevices—I was initially fascinated by the promise of “self-assembled electronic devices” that would become possible with the help of DNA’s magic programmable base-pairing rules in the future. In fact, I was completely unaware of DNA nanotechnology until I saw a talk by Uri Sivan from Technion in 1998 who, together with Erez Braun and co-workers, had just realized the first DNA-templated metal nanowires [1]. I actually first considered doing a postdoc with Uri Sivan, but during a visit at Bell Laboratories in 1999, Bernard Yurke introduced me into his work on “DNA tweezers” that he had started with Allen

F. C. Simmel (✉)
Department of Bioscience, Technical University of Munich, TUM School of Natural Sciences,
Garching, Germany
e-mail: simmel@tum.de

© The Author(s) 2023
N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*,
Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_2

P. Mills and Andrew Turberfield (who did a Sabbatical at Bell Labs in this year).¹ The vision of making artificial molecular machines from DNA molecules seemed so unconventional that I felt immediately attracted to it. Fortunately, Bernie Yurke offered me a position, but due to a delay in funding I could only start at Bell Labs in early 2000. Bernie and his co-workers were so nice to let me do a series of control experiments for the DNA tweezers paper [2], which was already under review at that time, and made me a co-author on this extremely important paper. During our work on strand displacement-based DNA machines I finally became aware of Nadrian (Ned) Seeman's pioneering work in DNA nanotechnology—in fact, together with Chengde Mao he had already published a different type of DNA nanodevice in 1999, which was based on the B-Z transition of CpG rich DNA sequences [3].

A major initiating event for me in this period was the seventh International Workshop on DNA computing in Tampa, FL, where for the first time I was exposed to the deep connections between theoretical computer science and self-assembly processes—I still remember a wonderful tutorial introduction by Erik Winfree—and also various ideas on computing in biological systems—topics that have immensely widened my view on the subject and have fascinated me ever since.

2 Designing and Programming with DNA

DNA nanotechnology is an extremely interdisciplinary field of research. Compared to many other fields, it also has a relatively low entry barrier for researchers with diverse scientific backgrounds. There are various reasons for this, but in a deep sense really the unique properties of DNA molecules are responsible for it.

2.1 DNA—A Programmable Molecule

DNA's biological role is intimately connected with its molecular and structural properties, which are heavily utilized in DNA nanotechnology. Of course, there is base-pairing, adenine (A) pairs with thymine (T), guanine (G) with cytosine (C), but in such a manner that A-T and G-C base-pairs have the same size in the context of the double helix. This makes a DNA duplex a structurally very uniform heteropolymer, which is biologically important because this allows DNA and RNA polymerase to run smoothly over it regardless of its sequence. Of course, this also means that information can be encoded in the sequence of base-pairs. As a double-stranded helix, DNA is mechanically relatively rigid. Binding between complementary strands is highly specific and depends on the base sequence. The thermodynamic properties can be determined in an approximately additive manner by just summing up the con-

¹ When I was looking for a postdoc position, I also considered to switch to a more biological topic, but several potential postdoc advisors intimidated me by saying “well, then you need to spend several years to learn more about biology to get even started”. It turns out that DNA nanotechnology has a relatively low entry barrier, which still is an attractive feature for young researchers.

tributions of nearest neighbors [4]—there are no long-range interactions between distinct sequences along a double helix that complicate matters. Put in the nanotechnology context: Double-stranded DNA is a rigid, information-encoding polymer with uniform geometric properties. The interactions between DNA single strands can be *programmed* by choosing the appropriate sequences.

This is almost everything you need to know to get started with DNA nanotechnology—which is quite astonishing, because really *a lot* is known about DNA—you can ignore the details in the beginning, but they become important later. Because of its central role to life, DNA has attracted scientists from very different disciplines and next to biologists, of course, chemists who developed schemes to synthesize and chemically modify DNA at will and who also came up with synthetic DNA and RNA analogues. Physicists like to study the structure of DNA, its mechanics and dynamics as a polymer, its thermodynamics, and its charge interactions [5, 6]. A whole sub-discipline of computer science—bioinformatics—is devoted to the study and comparison of DNA sequences. DNA nanotechnology greatly benefits from all these achievements—automated synthesis, structural and thermodynamic information, and computational tools are all available. This makes DNA nanotechnology to be the most advanced molecular technology so far, and because it is sequence based, it is also programmable in a relatively straightforward manner.

2.2 *Learning by Building*

Researchers in synthetic biology often talk about the design-build-test-learn (DBTL) cycle and about *understanding by building*. In DNA nanotechnology, these principles are in fact realized! The availability of computer tools combined with automated synthesis and a wide range of established characterization tools enable even inexperienced researchers to quickly create novel DNA nanostructures, study their behavior, and use them in applications. When mistakes are made, one can relatively easily go through another DBTL cycle to improve the results. Importantly, this makes it possible to find a good balance between rational design (taking into account the wealth of nucleic acids knowledge) and a learning-by-doing approach.

Among the many fascinating achievements of DNA nanotechnology [7], DNA origami is certainly the best known outside of the community [8, 9]. DNA utilizes a long single-stranded DNA molecule termed the “scaffold” and a large number of shorter staple strands that sequence specifically bind to designed positions along the scaffold. This creates links between these positions, which fold the scaffold into a three-dimensional molecular shape. The same scaffold strand can be “programmed” to adopt completely different shapes simply by the choice of the staple sequences. Using a DNA origami design program allows researchers to design such origami structures on the computer and choose staple sequences for synthesis that will later physically assemble the desired shapes in the test tube [10]. An interesting development sets in here: Designing DNA nanostructures using computer programs and the corresponding DNA nanotech methodology represents a new skill set that draws

from, but is distinct from that of traditional biophysics or biochemistry. As stated already before, you do not need to know every chemical detail about DNA to build such structures. At the same time, origami researchers develop a “feeling” for what can be built, and what not.

Think of modern computer programmers—it is not required to have detailed knowledge about computer hardware (or electronics and solid-state physics, for that matter) to be able to write code in a high-level programming language and solve complex tasks with it. While it may be a little sad that programmers do not know how certain things work in detail—it is also a big achievement that they do *not need* to know! Modularization and abstraction help to make much bigger leaps much faster. And, more or less, this is what happens in DNA nanotechnology—this is the power behind the idea of *molecular programming* [11, 12].

2.3 Challenges and Limitations

Already now the ability to create DNA-based nanostructures of almost arbitrary shape combined with the availability of a wide range of chemical modifications for these structures has found use in nanomedicine, biosensing, nanoscale science, and biophysics. In such applications, DNA nanostructures are typically used to spatially organize molecules and nanoparticles with nanometer precision.

More or less obvious questions (and challenges) for the future of static DNA nanostructures regard the scale of the structures: Can we make bigger and bigger functional structures, what quantities and at what cost [13]? Can we achieve greater precision in the arrangement of molecules and increase the chemical diversity and functionality of DNA nanostructures? Using modified DNA and nucleic acid analogues, e.g., it would be great if one could rationally design DNA nanostructure-based catalysts with similar catalytic versatility and power as enzymes. These challenges are rather chemical in nature and will hopefully be addressed by the gifted chemists in the field (if we come back to the comparison with computer programming: chemists provide the hardware for DNA nanotechnology and thus define and extend the capabilities of the hardware-agnostic molecular programmers).

In my opinion, a greater challenge lies in the realization of *dynamic* functions and the control and utilization of non-equilibrium processes. Over the past two decades, DNA nanotechnology has been concerned with the realization of molecular switches (such as the tweezers), machine-like devices (molecular walkers [14, 15], rotors [16], and the like), DNA-based chemical reaction networks (CRNs), and DNA computers [17, 18]. There are different motivations for creating such systems: On a fundamental level, one would simply like to learn how to build artificial molecular machines or generally realize dynamic molecular functions. Then, such systems should be of great use in nanotechnology—examples for applications of dynamic DNA processes already exist: enzyme-free DNA-based sensors (hybridization chain reaction [19], catalytic hairpin assembly [20]) and DNA-based super-resolution microscopy (DNA-PAINT) [21].

In several aspects, however, dynamic DNA systems are still very limited. Next to their low chemical functionality (as for static structures, see above), their main restrictions are the difficulty of operating them continuously and autonomously and their relatively low speed. If one considers DNA nanotechnology as an approach to emulate and thus understand biological self-organization, these will be major hurdles.

3 From Self-Assembly to Non-equilibrium Dynamics and Self-Organization

Realizing dynamic, biology-inspired self-organizing systems, and building molecular machinery in particular, involves the challenge of controlling the flow of chemical energy through a non-equilibrium chemical system to generate interesting dynamics and structure (cf. Fig. 1).

3.1 Molecular Machines

Molecular machines and devices made from DNA have usually been based on DNA conformational changes (formation of duplexes, hairpins, triplexes, i-motifs, etc.) that take place in the presence of a DNA input or in response to a change in environmental

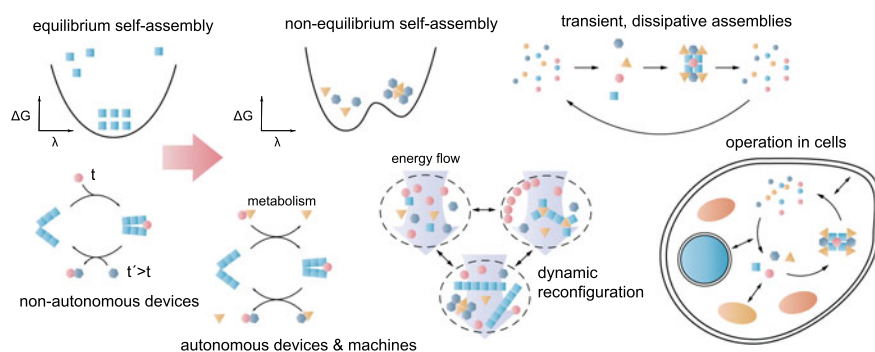


Fig. 1 DNA nanotechnology has very successfully generated nanostructures, which self-assemble as the thermodynamically most stable structure of a mixture of components. Various molecular machines and devices have been realized by periodically driving switchable molecular structures out of equilibrium in a non-autonomous manner. In the future, DNA nanotechnology is anticipated to increasingly operate further away from equilibrium, which will be of interest in various contexts: accessing far-from-equilibrium self-assembled and dissipative structures; driving molecular machines continuously and autonomously; realizing active matter and cell-like soft robotic systems that can respond to their environment and quickly switch between different states of organization. A specific challenge is the production and operation of nucleic acid devices inside living cells. In the figure, λ denotes some parameter that characterizes the deviation of the system from equilibrium

conditions (salt, pH, light). Due to the nature of these stimuli, most such systems are operated in a clocked manner—they are kicked out of equilibrium by a change in conditions (addition of fuel), followed by relaxation to a new equilibrium state.

Continuous and autonomous operation out of equilibrium is challenging. An ingenious idea to achieve this was developed by Turberfield and Yurke and further developed by many others [22, 23]. DNA fuel can be forced into an unreactive secondary structure (like a hairpin loop), which can be activated via strand invasion by a DNA trigger strand. The activated sequence domains can then invade another inert fuel hairpin (or secondary structure). This principle can be used to store chemical (hybridization) energy in hairpin loops, which can be released in a dynamical DNA system in catalytic cycles or cascades. In essence, this is an example of kinetic pathway engineering: The inert fuels are present in a metastable state, which can only relax through reactions that drive a process of interest [23]. This principle has been used to drive autonomous molecular walkers and also lies at the heart of a variety of nucleic acid amplification schemes [24].

Even though these achievements are extremely impressive, the field of DNA machines has not yet “taken off,” and the capabilities of DNA nanomachines are still far from those of biological machinery. DNA-based molecular machines are slow, they have not been used to exert appreciable forces to carry out tasks or move objects over larger distances. Why is that?

Using DNA as a fuel is one of the major conceptual developments in DNA nanotechnology—DNA is an *information-bearing fuel* and can simultaneously act as a molecular address code that switches only specific molecular processes. As can be seen from the applications that have emerged, this is particularly useful for biosensing applications and for the realization of chemical reaction networks. DNA fuels have not been very successfully used, however, to generate fast movements and quick conformational changes as found in biological molecular machines.

Biology does not use an information-bearing fuel molecule—it uses a couple of small molecule fuels (ATP, GTP) for almost *everything*. These molecules are present at high concentrations (in the mM range), are constantly (re)generated by cellular metabolism, and shared by a large diversity of different processes that run in parallel.

Using small molecule fuels at high concentrations results in much faster kinetics. Millimolar concentrations are much higher than the typical concentrations used for DNA fuels, with an accordingly higher on-rate for the fuel.² Furthermore, binding of a small molecule to its binding site is faster than binding between large molecules with multiple interactions or complex interaction surfaces—DNA hybridization requires nucleation of a few base-pairs “at the right position,” for which the participating DNA molecules must be in the appropriate orientation to each other. Of course, also the off-rate is affected: While a small molecule with a comparatively small ΔG for binding will quickly dissociate, we have to use branch migration or other means to wrest off a DNA fuel molecule from a DNA machine to start another machine cycle.

² Experiments with optimized, “leakless” strand displacement systems have demonstrated operation at concentrations of up to 10 μM [25].

Other aspects that play a role in biological machines and may guide the further improvement of artificial machines relate to the coupling of fuel consumption to conformational changes. In biology, sometimes only a small change—binding of ATP into a tight pocket or changing a single charge by phosphorylation—is transduced mechanically to a large change in geometry. DNA nanotechnology has not achieved this level of molecular control yet. It remains to be seen whether DNA—even in chemically modified form—can be used for such mechanisms at all.

If we want to stick with chemical driving mechanisms rather than physical manipulation of DNA nanodevices with magnetic [26] or electrical fields [27], or light [28–30], we will also need to figure out a mechanism to provide high-energy fuels for longer periods of time. If we do not want to use manual or microfluidic addition of fuels and removal of waste [31], we will need to embed the DNA devices within a reaction cycle that generates the fuels—in other words, some kind of metabolism is required. This probably cannot be achieved with DNA alone.

Potential tasks for building future molecular machines based on DNA might thus be the following: (i) find the “ATP equivalent” for DNA nanotechnology. It would be great to have a universal fuel that is used by all kinds of DNA machinery (which, of course, would mean that we have to give up sequence-addressability through the fuel); (ii) find a way to couple fuel consumption to a *quick* conformational change; (iii) realize some kind of metabolism that generates fuel at high concentrations or find some other clever way to replenish it.

What will be the role of *DNA programmability* in this context?

3.2 *Non-equilibrium Chemical Dynamics and Self-Assembly*

Similar considerations as for molecular machines apply to dynamic and growing biological structures and materials in general. Structures in biology are rarely built to last. There is a constant turnover of molecules within supramolecular assemblies like the cell membrane or the cytoskeleton, which also gives biological systems the ability to respond to their environment, to grow, change shape, move, etc.

For instance, treadmilling of cytoskeletal filaments [32] involves addition of new monomers at one end, consuming ATP or GTP, and dissociation at the other end, which results in an apparent movement of the filaments. Several groups have succeeded in generating polymers from DNA tiles or helix bundles [33–35], but the polymerization process was usually based on DNA hybridization and was not coupled to consumption of a high-energy fuel. In biology, the growth of filaments can actually exert forces on membranes (in addition to the Brownian ratchet force generated by growth of the filaments, molecular motors are involved [36]). It is currently unclear, how the slow growth of DNA filaments could be applied for something similar. At this point, DNA nanotechnology can provide the building blocks, but DNA-fuelled processes do not yet compete with the ATP/GTP-driven processes involved in the formation of non-equilibrium structures.

There are opportunities, obviously, in fusing ATP consuming proteins (or other, potentially synthetic catalytic units) to DNA structures and thus generate molecular systems with active dynamics. Here the DNA components will provide the spatial organization, while the dynamics are driven by some other chemical process, which can be supplied with fuel more easily. In principle, one could also use cellular metabolism itself to drive DNA or RNA-based systems and thus solve (or circumvent) the problem of energy supply and creation of non-equilibrium conditions. Several examples of toehold-mediated strand invasion processes have been demonstrated *in vivo*, where all of the components were generated by transcription reactions. This so far mainly refers to switchable regulatory RNA molecules (toehold switches [37] or conditional guide RNAs [38, 39]), but recent developments (still *in vitro*) also suggest a route to realize more complex RNA strand displacement circuitry from transcription products [40, 41].

Somewhat intermediate in this context is the utilization of DNA or RNA polymerases, ligases, and nucleases that can be adopted to power dynamic nucleic acid-based systems *in vitro* [42–44]. These enzymes accomplish the production and degradation of RNA or DNA fuels consuming nucleotide triphosphates and can therefore keep a system out of equilibrium as long as the NTPs are supplied and waste products are removed. Enzyme-driven systems retain much of the sequence programmability of pure DNA systems and have been used to create bistable systems and oscillators [42, 43, 45, 46], pattern forming systems [47, 48], transient dynamics [44, 49, 50], or to control assembly/disassembly reactions [35]. In contrast to dynamic DNA systems based on hybridization interactions alone [51], however, enzyme-driven systems have to cope with the biochemical idiosyncrasies of the enzymes used and are intrinsically less programmable.

In spite of the limitations mentioned, it is still conceivable to extend dynamic DNA nanotechnology to certain far-from-equilibrium processes and assemblies based exclusively on DNA and thus to create assemblies that are not the thermodynamically most stable ones. For instance, control of kinetics has previously been shown to be important in the context of origami folding and allowed to direct the folding process toward one of several alternative possible structures [52, 53]. Kinetics of strand displacement reactions can further be tuned via the length of the toehold [54], or by using tricks such as remote toeholds [55], which allow the engineering of kinetic pathways. “Handhold-mediated strand displacement” has just recently been demonstrated to enable the formation of templated far-from-equilibrium DNA assemblies [56].

3.3 Robots

Of course, speed is not everything. Many applications will not require fast movement or response. For instance, if we just want to arrange molecules into specific geometries, static DNA nanotechnology is sufficient. Also growth and pattern formation processes do not strictly have to be fast—think of emulating plant growth or programming the development of an organism. Where speed probably matters, is in

robotics [27, 57–63]. At least in my interpretation, robotic systems should be able to interact with their environment—sense, make decisions, respond, and act on their surroundings. In order to be able to do that sensing, computation, and action have to take place on a relevant timescale. In this context, it may be useful to adopt an engineering approach toward molecular robotic systems. We need to identify tasks for the robots and define desired performance characteristics and benchmarks. Very likely, for many applications it will not be possible to reach the aims set for the robots using DNA alone—any physical or chemical trick to speed up the systems will here be welcome.

Another challenge for the realization of DNA-based molecular robotics is the integration of different robotic components into a consistent system. DNA nanodevices have already been shown to be capable of sensing, computation, and movement, but these have been combined into a molecular robotic system only in a few cases. Rather than programming DNA robots at the sequence level, programming at a modular level could be interesting. Taking inspiration from macroscopic modular robots, one could strive for generating optimized DNA robotic components with standardized interfaces that allow the reuse of known functional DNA modules.

Already now, many researchers like to apply the same type of DNA origami structures (think of Paul Rothemund's rectangle [8]) to many different tasks—simply because they are useful and have been proven to work—generating ever more differently shaped structures is not necessary for many applications. Further, using localization and spatial organization of DNA components not only allows speeding up slow hybridization reactions by generating high local concentrations [62, 64–67]—it also allows reuse of the same components by spatial separation and thus avoids cross-talk. Thus, strategies for defined modular interfaces and spatial arrangements of known components might become a new branch of molecular programming.

4 What Lies Ahead?

DNA nanotechnology 40 years after its inception by Ned Seeman is still developing rapidly. Hundreds of laboratories worldwide harness the power of DNA self-assembly to arrange molecules and particles into specific geometries to address scientific questions in a wide range of research areas. Due to all the favorable properties of DNA already mentioned above, DNA nanotechnology is here to stay—it has (or will) simply become a standard approach for molecular engineering, similar as lithography or the production of nanoparticles in nanoscience. Further progress in the field will require improving the chemical versatility of DNA and DNA analogues without sacrificing the unique capability for sequence-programmable self-assembly.

The greatest challenges for the field lie in the realization of dynamic functions. Can one generate DNA-based molecular machines that are really useful? Can one generate complex chemical dynamics based only on DNA? Can one emulate biological behaviors with programmed DNA nanosystems? Will it be possible to speed up the systems to make them useful for applications? If not, what are the general principles, the insights we can derive from DNA-based model systems?

It is likely that we will need to use some kind of dissipative process—chemical catalytic cycles, enzymes, cellular metabolism—or physical driving to generate interesting dynamic behaviors. The challenge, then, is to use the power of DNA nanotechnology to arrange the respective active components in space and to direct the non-equilibrium energy flow to generate and control self-organization processes. A major challenge will be to develop schemes to abstract these behaviors and ultimately make them as programmable as static DNA structures.

Let me end with a personal note. I am glad and grateful that I can be part of this great interdisciplinary adventure called *DNA nanotechnology*—DNA nanotech is a wonderful scientific community, with so many highly inspiring personalities from all fields of research, starting, of course, with Ned Seeman who in his talks always emphasized the importance of knowing everything about chemistry, physics, biology, math, ..., and arts. Even though mentioned several times in my text, one also should *not* overemphasize usefulness and applicability. Using DNA as a generic, programmable molecular substrate to explore interesting concepts and ideas “in the test tube” is a worth on its own. It allows you to stand back and approach problems in self-assembly and chemical dynamics at a more abstract level—and thus gain insights into their general governing principles.

References

1. E. Braun, Y. Eichen, U. Sivan, G. Ben-Yoseph, DNA-templated assembly and electrode attachment of a conducting silver wire. *Nature* **391**(6669), 775–778 (1998)
2. B. Yurke, A.J. Turberfield, A.P. Mills, F.C. Simmel, J.L. Neumann, A DNA-fuelled molecular machine made of DNA. *Nature* **406**(6796), 605–608 (2000)
3. C. Mao, W. Sun, Z. Shen, N.C. Seeman, A nanomechanical device based on the B-Z transition of DNA. *Nature* **397**(6715), 144–146 (1999)
4. J. SantaLucia, A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics. *Proc. Natl. Acad. Sci.* **95**(4), 1460–1465 (1998)
5. J. SantaLucia, D. Hicks, The thermodynamics of DNA structural motifs. *Ann. Rev. Biophys. Biomol. Struct.* **33**, 415–440 (2004)
6. A. Vologodskii, *Biophysics of DNA* (Cambridge University Press, Cambridge, 2015)
7. N.C. Seeman, H.F. Sleiman, DNA nanotechnology. *Nat. Rev. Mater.* **3**(1), 17068 (2017)
8. P.W.K. Rothemund, Folding DNA to create nanoscale shapes and patterns. *Nature* **440**(7082), 297–302 (2006)
9. S.M. Douglas, H. Dietz, T. Liedl, B. Högberg, F. Graf, W.M. Shih, Self-assembly of DNA into nanoscale three-dimensional shapes. *Nature* **459**(7245), 414–418 (2009)
10. S. Dey, C. Fan, K.V. Gothelf, J. Li, C. Lin, L. Liu, N. Liu, M.A.D. Nijenhuis, B. Sacca, F.C. Simmel, H. Yan, P. Zhan, DNA origami. *Nat. Rev. Methods Primers* **1**(1), 13 (2021)
11. Molecular Programming Project. <http://molecular-programming.org/>
12. E. Winfree, in *Advances in Molecular Programming and Computing: Toward Chemistry as a New Information Technology, Report from a workshop in Copenhagen, 2–4 May 2013* (2013). https://www.dna.caltech.edu/Papers/Denmark_Workshop_2013.pdf
13. F. Praetorius, B. Kick, K.L. Behler, M.N. Honemann, D. Weuster-Botz, H. Dietz, Biotechnological mass production of DNA origami. *Nature* **552**(7683), 84–87 (2017)
14. T. Omabegho, R. Sha, N.C. Seeman, A bipedal DNA Brownian motor with coordinated legs. *Science* **324**(5923), 67–71 (2009)

15. S.J. Green, J. Bath, A.J. Turberfield, Coordinated chemomechanical cycles: a mechanism for autonomous molecular motion. *Phys. Rev. Lett.* **101**(23), 238101 (2008)
16. P. Ketterer, E.M. Willner, H. Dietz, Nanoscale rotary apparatus formed from tight-fitting 3D DNA components. *Sci. Adv.* **2**(2), e1501209–e1501209 (2016)
17. L. Qian, E. Winfree, Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**(6034), 1196–1201 (2011)
18. D. Woods, D. Doty, C. Myhrvold, J. Hui, F. Zhou, P. Yin, E. Winfree, Diverse and robust molecular algorithms using reprogrammable DNA self-assembly. *Nature* **567**(7748), 366–372 (2019)
19. R.M. Dirks, N.A. Pierce, Triggered amplification by hybridization chain reaction. *Proc. Natl. Acad. Sci.* **101**(43), 15275–15278 (2004)
20. B. Li, A.D. Ellington, X. Chen, Rational, modular adaptation of enzyme-free DNA circuits to multiple detection methods. *Nucleic Acids Res.* **39**(16), e110–e110 (2011)
21. R. Jungmann, C. Steinhauer, M. Scheible, A. Kuzyk, P. Tinnefeld, F.C. Simmel, Single-molecule kinetics and super-resolution microscopy by fluorescence imaging of transient binding on DNA origami. *Nano Lett.* **10**(11), 4756–4761 (2010)
22. A.J. Turberfield, J.C. Mitchell, B. Yurke, A.P. Mills, M.I. Blakey, F.C. Simmel, DNA fuel for free-running nanomachines. *Phys. Rev. Lett.* **90**(11), 118102 (2003)
23. P. Yin, H.M.T. Choi, C.R. Calvert, N.A. Pierce, Programming biomolecular self-assembly pathways. *Nature* **451**(7176), 318–322 (2008)
24. F.C. Simmel, B. Yurke, H.R. Singh, Principles and applications of nucleic acid strand displacement reactions. *Chem. Rev.* **119**(10), 6326–6369 (2019)
25. B. Wang, C. Thachuk, A.D. Ellington, E. Winfree, D. Soloveichik, Effective design principles for leakless strand displacement systems. *Proc. Natl. Acad. Sci.* **115**(52), 201806859 (2018)
26. S. Lauback, K.R. Mattioli, A.E. Marras, M. Armstrong, T.P. Rudibaugh, R. Sooryakumar, C.E. Castro, Real-time magnetic actuation of DNA nanodevices via modular integration with stiff micro-levers. *Nat. Commun.* **9**(1), 1446 (2018)
27. E. Kopperger, J. List, S. Madhira, F. Rothfischer, D.C. Lamb, F.C. Simmel, A self-assembled nanoscale robotic arm controlled by electric fields. *Science* **359**(6373), 296–301 (2018)
28. X. Liang, H. Nishioka, N. Takenaka, H. Asanuma, A DNA nanomachine powered by light irradiation. *ChemBioChem* **9**(5), 702–705 (2008)
29. M. You, Y. Chen, X. Zhang, H. Liu, R. Wang, K. Wang, K.R. Williams, W. Tan, An autonomous and controllable light-driven DNA walking device. *Angew. Chem.-Int. Ed.* **51**(10), 2457–2460 (2012)
30. F. Lohmann, D. Ackermann, M. Famulok, Reversible light switch for macrocycle mobility in a DNA rotaxane. *J. Am. Chem. Soc.* **134**(29), 11884–11887 (2012)
31. T.E. Tomov, R. Tsukanov, Y. Glick, Y. Berger, M. Liber, D. Avrahami, D. Gerber, E. Nir, DNA bipedal motor achieves a large number of steps due to operation using microfluidics-based interface. *ACS Nano* **11**(4), 4002–4008 (2017)
32. V.I. Rodionov, G.G. Borisy, Microtubule treadmilling in vivo. *Science* **275**(5297), 215–218 (1997)
33. P.W.K. Rothmund, A. Ekani-Nkodo, N. Papadakis, A. Kumar, D.K. Fygenson, E. Winfree, Design and characterization of programmable DNA nanotubes. *J. Am. Chem. Soc.* **126**(50), 16344–16352 (2004)
34. P. Yin, R.F. Hariadi, S. Sahu, H.M.T. Choi, S.H. Park, T.H. LaBean, J.H. Reif, Programming DNA tube circumferences. *Science* **321**(5890), 824–826 (2008)
35. L.N. Green, H.K.K. Subramanian, V. Mardanlou, J. Kim, R.F. Hariadi, E. Franco, Autonomous dynamic control of DNA nanostructure self-assembly. *Nat. Chem.* **11**(6), 510–520 (2019)
36. R.B. Dickinson, L. Caro, D.L. Purich, Force generation by cytoskeletal filament end-tracking proteins. *Biophys. J.* **87**(4), 2838–2854 (2004)
37. A. Green, P. Silver, J. Collins, P. Yin, Toehold switches: de-novo-designed regulators of gene expression. *Cell* **159**(4), 925–939 (2014)
38. L.M. Hochrein, H. Li, N.A. Pierce, High-performance allosteric conditional guide RNAs for mammalian cell-selective regulation of CRISPR/Cas. *ACS Synth. Biol.* **10**(5), 964–971 (2021)

39. L. Oesinghaus, F.C. Simmel, Controlling gene expression in mammalian cells using multiplexed conditional guide RNAs for Cas12a. *Angew. Chem. Int. Ed.* **60**(44), 23894–23902 (2021)
40. W. Bae, G.-B.V. Stan, T.E. Ouldrige, In situ generation of RNA complexes for synthetic molecular strand-displacement circuits in autonomous systems. *Nano Lett.* **21**(1), 265–271 (2021)
41. S.W. Schaffter, E.A. Strychalski, Cotranscriptionally encoded RNA strand displacement circuits. *Sci. Adv.* **8**, eabl4354 (2022)
42. J. Kim, K.S. White, E. Winfree, Construction of an in vitro bistable circuit from synthetic transcriptional switches. *Mol. Syst. Biol.* **2**(1), 68 (2006)
43. K. Montagne, R. Plasson, Y. Sakai, T. Fujii, Y. Rondelez, Programming an in vitro DNA oscillator using a molecular networking strategy. *Mol. Syst. Biol.* **7**(1), 466 (2011)
44. J. Deng, A. Walther, ATP-powered molecular recognition to engineer transient multivalency and self-sorting 4D hierarchical systems. *Nat. Commun.* **11**(1), 3658 (2020)
45. E. Franco, E. Friedrichs, J. Kim, R. Jungmann, R. Murray, E. Winfree, F.C. Simmel, Timing molecular motion and production with a synthetic transcriptional clock. *Proc. Natl. Acad. Sci.* **108**(40), E784–E793 (2011)
46. S.W. Schaffter, R. Schulman, Building in vitro transcriptional regulatory networks by successively integrating multiple functional circuit modules. *Nat. Chem.* **11**(9), 829–838 (2019)
47. A. Padirac, T. Fujii, A. Estevez-Torres, Y. Rondelez, Spatial waves in synthetic biochemical networks. *J. Am. Chem. Soc.* **135**(39), 14586–14592 (2013)
48. A.S. Zadorin, Y. Rondelez, G. Gines, V. Dilhas, G. Urtel, A. Zambrano, J.C. Galas, A. Estevez-Torres, Synthesis and materialization of a reaction-diffusion French flag pattern. *Nat. Chem.* **9**(10), 990–996 (2017)
49. E. Del Grosso, A. Amodio, G. Ragazzon, L.J. Prins, F. Ricci, Dissipative synthetic DNA-based receptors for the transient loading and release of molecular cargo. *Angew. Chem. Int. Ed.* **57**(33), 10489–10493 (2018)
50. S. Wang, L. Yue, V. Wulf, S. Lilienthal, I. Willner, Dissipative constitutional dynamic networks for tunable transient responses and catalytic functions. *J. Am. Chem. Soc.* **142**(41), 17480–17488 (2020)
51. N. Srinivas, J. Parkin, G. Seelig, E. Winfree, D. Soloveichik, Enzyme-free nucleic acid dynamical systems. *Science* **358**(6369), eaal2052 (2017)
52. K.E. Dunn, F. Dannenberg, T.E. Ouldrige, M. Kwiatkowska, A.J. Turberfield, J. Bath, Guiding the folding pathway of DNA origami. *Nature* **525**(7567), 82–86 (2015)
53. A.E. Marras, L. Zhou, V. Kolliopoulos, H.-J. Su, C.E. Castro, Directing folding pathways for multi-component DNA origami nanostructures with complex topology. *New J. Phys.* **18**(5), 055005 (2016)
54. N. Srinivas, T.E. Ouldrige, P. Sulc, J.M. Schaeffer, B. Yurke, A.A. Louis, J.P.K. Doye, E. Winfree, On the biophysics and kinetics of toehold-mediated DNA strand displacement. *Nucleic Acids Res.* **41**(22), 10641–10658 (2013)
55. A.J. Genot, D.Y. Zhang, J. Bath, A.J. Turberfield, Remote toehold: a mechanism for flexible control of DNA hybridization kinetics. *J. Am. Chem. Soc.* **133**(7), 2177–2182 (2011)
56. J. Cabello-Garcia, W. Bae, G.-B.V. Stan, T.E. Ouldrige, Handhold-mediated strand displacement: a nucleic acid based mechanism for generating far-from-equilibrium assemblies through templated reactions. *ACS Nano* **15**(2), 3272–3283 (2021)
57. K. Lund, A.J. Manzo, N. Dabby, N. Michelotti, A. Johnson-Buck, J. Nangreave, S. Taylor, R. Pei, M.N. Stojanovic, N.G. Walter, E. Winfree, H. Yan, Molecular robots guided by prescriptive landscapes. *Nature* **465**(7295), 206–210 (2010)
58. H. Gu, J. Chao, S.-J. Xiao, N.C. Seeman, A proximity-based programmable DNA nanoscale assembly line. *Nature* **465**(7295), 202–205 (2010)
59. S.M. Douglas, I. Bachelet, G.M. Church, A logic-gated nanorobot for targeted transport of molecular payloads. *Science* **335**(6070), 831–834 (2012)
60. R.A. Muscat, J. Bath, A.J. Turberfield, A programmable molecular robot. *Nano Lett.* **11**(3), 982–987 (2011)

61. S.F.J. Wickham, J. Bath, Y. Katsuda, M. Endo, K. Hidaka, H. Sugiyama, A.J. Turberfield, A DNA-based molecular motor that can navigate a network of tracks. *Nat. Nanotechnol.* **7**(3), 169–173 (2012)
62. A.J. Thubagere, W. Li, R.F. Johnson, Z. Chen, S. Doroudi, Y.L. Lee, G. Izatt, S. Wittman, N. Srinivas, D. Woods, E. Winfree, A cargo-sorting DNA robot. *Science* **357**(6356), eaan6558 (2017)
63. S. Li, Q. Jiang, S. Liu, Y. Zhang, Y. Tian, C. Song, J. Wang, Y. Zou, G.J. Anderson, J.Y. Han, Y. Chang, A DNA nanorobot functions as a cancer therapeutic in response to a molecular trigger in vivo. *Nat. Biotechnol.* **36**(3), 258–264 (2018)
64. M. Teichmann, E. Kopperger, F.C. Simmel, Robustness of localized DNA strand displacement cascades. *ACS Nano* **8**(8), 8487–8496 (2014)
65. G. Chatterjee, N. Dalchau, R.A. Muscat, A. Phillips, G. Seelig, A spatially localized architecture for fast and modular DNA computing. *Nat. Nanotechnol.* **12**(9), 920–927 (2017)
66. J. Li, A. Johnson-Buck, Y.R. Yang, W.M. Shih, H. Yan, N.G. Walter, Exploring the speed limit of toehold exchange with a cartwheeling DNA acrobat. *Nat. Nanotechnol.* **13**(8), 723–729 (2018)
67. J. Chao, J. Wang, F. Wang, X. Ouyang, E. Kopperger, H. Liu, Q. Li, J. Shi, L. Wang, J. Hu, L. Wang, W. Huang, F.C. Simmel, C. Fan, Solving mazes with single-molecule DNA navigators. *Nat. Mater.* **18**(3), 273–279 (2019)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



The Evolution of DNA-Based Molecular Computing



Fei Wang, Qian Li, and Chunhai Fan

Abstract The first demonstration of DNA computing was realized by Adleman in 1994, aiming to solve hard combinatorial problems with DNA molecules. This pioneering work initiated the evolution of the field of DNA computing during the last three decades. Up to date, the implemented functions of DNA computing have been expanded to logic operations, neural network computations, time-domain oscillator circuits, distributed computing, etc. Herein, the history of DNA computing is briefly reviewed, followed by discussions on opportunities and challenges of DNA-based molecular computing, especially from the perspective of algorithm design. Future directions and design strategies for next-generation DNA computing is also discussed.

1 A Brief History of DNA Computing

Nature-evolved DNA molecules are the primary information-carrying medium of life [1]. The computing power of DNA relies primarily on its structural potential. In 1953, Watson and Crick first proposed the double-helix structure of DNA, which marks a key step to uncover the secret of life [2]. In 1982, Seeman for the first time proposed a rational design of Holliday junction-like branched DNA structure [3], pioneering the endeavor to construct human-defined structures using DNA beyond the secret of life. This work and the subsequent progress in DNA nanotechnology provide insights and toolbox for the design of dynamic structures to implement computing algorithms.

In 1994, Adleman proposed a DNA-based algorithm to solve a Hamiltonian path problem [4]. This work for the first time demonstrated the feasibility of carrying out computations using synthetic DNA molecules, thereby signaling the start of DNA computing. The parallelism far beyond that of conventional silicon-based computers attracted wide interest. Following this work, efforts were made to explore the parallel

F. Wang · Q. Li · C. Fan (✉)

School of Chemistry and Chemical Engineering, Frontiers Science Center for Transformative Molecules, Institute of Translational Medicine, Shanghai Jiao Tong University, Shanghai 200240, China

e-mail: fanchunhai@sjtu.edu.cn

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*, Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_3

31

computation ability of DNA molecules to solve various mathematically complex problems, including Boolean satisfiability problem (SAT) [5–9], maximal clique problem [10], traveling salesman problem [11], etc. During this period, it was realized that the available molecular parallelism was not enough to combat the slow clock speed of biochemical operations and the redundancy required to combat the high intrinsic error rate of the operations. Evolutionary computation models were proposed to overcome the limitations, although not experimentally demonstrated [10, 12–15].

Subsequently, a variety of new molecular mechanisms for implementing computational algorithms were proposed, which greatly enriched the toolbox of DNA computing [16–24]. In 2000, as a milestone in DNA computing, an enzyme-free DNA “tweezer” was proposed by Yurke et al., which could switch between ON and OFF states through strand displacement reactions [25]. Based on accurate base pairing principle, with tunable reaction kinetics [26–31] and spontaneous execution, strand displacement reactions immediately enable various modular design of DNA computing architectures [16, 18, 32–34]. In 2004, Dirks and Pierce realized triggered amplification by hybridization chain reaction (HCR) [17], which has been broadly applied in computing [35], biosensing [36], and self-assembly [37, 38]. In 2006, Seelig et al. proposed a toehold-mediated strand displacement scheme to construct enzyme-free logic circuits [18]. In 2007, Zhang et al. developed a signal amplification reaction network that uses DNA strand as catalyst [39].

With the advances of DNA computing toolbox, various computational functions (e.g., automaton [40, 41], logic computing [42, 43], neural network computing [44, 45], cargo-sorting [46], and maze solving [35]) have been realized. Figure 1 presents a timeline of representative advances in DNA computing, classified mainly according to the realized functions and design principles. In 2003, Benenson et al. reported a molecular automaton that uses DNA both as data and fuel [40]. In 2010, Pei et al. first developed a programmable computing device to play a game [41]. In 2011, Qian and Winfree proposed a simple yet modular computing unit “seesaw” motif, with which large-scale digital computation [42] and neural network computation [44] were implemented experimentally with improved performance. In 2012, Padiac et al. developed switchable memories using DNA and DNA processing enzymes [19]. In addition to state jumps, they also implemented time-domain programming. Recently, temporal dynamics programming was further developed to implement predator-prey reaction network [47], adaptive immune response simulator [48], and enzyme-free oscillators [49]. Logic computing has also been developed with the introduction of DNA origami-based logic gates [50], spatially localized logic gates [33], integrated gene logic chip [51], single-stranded gates [52], and DNA switching circuits [53]. Meanwhile, task-oriented DNA molecular algorithms have been demonstrated in recent years, such as edge detection [54], cargo-sorting [46], maze solving [35], and pattern recognition [45].

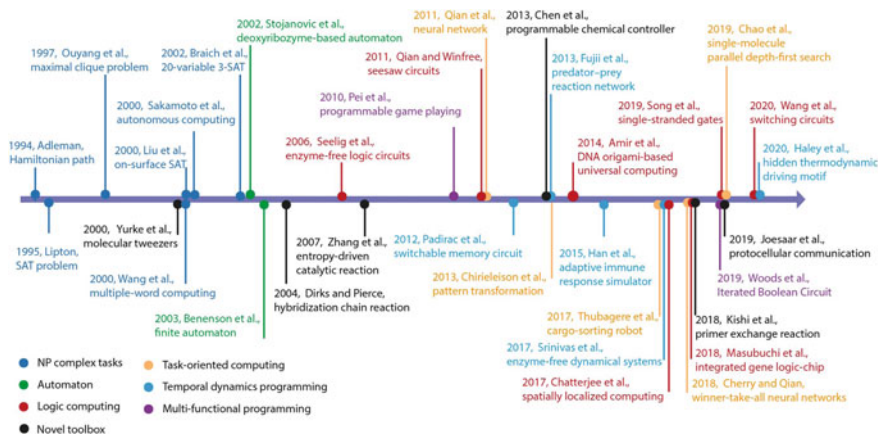


Fig. 1 Timeline of representative advances in the field of DNA computing [4–10, 16–19, 21, 25, 30, 32, 33, 35, 39–41, 44, 47–56]

2 Opportunities and Challenges

Highly ordered arrangement of physically addressable computing units at the nanoscale facilitates high-performance computing of silicon-based conventional computers. To complete a calculation task, the corresponding electronic transmission paths are activated by addressing to map the algorithm to the hardware. In contrast, DNA computing functions are mainly implemented with interactions between DNA and DNA [18], other biological molecules (e.g., RNAs [57], proteins [19, 51], and small molecules [58]), or environmental conditions (e.g., light [59], pH [60], and ions [61]). These molecule computing units are addressed chemically rather than physically, since they are mixed in solution with indistinguishable locations. To carry out a required computing task, a DNA-based chemical reaction network is programmed to run according to specified rules (algorithms) by designing and controlling the interactions between DNA and above elements. The differences in underlying implementations between DNA and solid-state computing devices (e.g., electronic and optical computers) result in unique advantages and challenges for further development of DNA computing.

2.1 Bridge Between Matter and Information

Computing relies on information processing, transfer, and storage. For conventional storage media, information is stored in the specified (magnetic, optical, electrical) states of matters by spatial manipulation of these storage media. In contrast, the aperiodic nucleobases make a DNA strand itself a piece of information. This not only leads to ultrahigh information density in DNA, but also bridges matter and information.

With aperiodic nucleobases and regularly sized hydrogen bonds, DNA double helix is an aperiodic crystal, which supports reliable information storage and transfer [62]. The information stored in DNA is transferred into RNA and proteins through genetic transcription and translation. The stereoregular duplex structure allows DNA to be accessed and processed by sequence-independent enzymes for self-replication and degradation. Thus, DNA links a wide matter world at the molecular level with the sequence information it carries.

2.2 *Massive Parallelism*

DNA computing was initially proposed to solve mathematically complex problems, such as Hamilton path problem [4], SAT problem [5], and maximum clique problem [10], taking advantage of specific and highly parallel binding between DNA molecules. Due to the stochasticity of molecular interactions, every individual molecule of the same population randomly follows certain permitted reaction paths. According to the law of large numbers, all possibilities are covered as long as the number of molecules is sufficient. Compared with algorithms that search for every possible combination one by one until the answer is found, DNA-based molecular algorithms can greatly reduce the time complexity.

It should be noted that the parallelism underlying molecular interactions relies on the availability of participating molecules. For example, a 500-node traveling salesman problem has more than 10^{1000} potential solutions, which is beyond the number of available molecules, as 1 L of 1 M solution could only provide 6×10^{23} manipulative molecules. Besides, as proposed by Back et al. [12], a huge number of DNA molecules can participate a calculation in parallel to generate a random population of candidate solutions, followed by a filtering step to remove all DNA molecules not representing a solution to the problem. Such “filtering approach” becomes infeasible as the problem size grows, since it becomes difficult to select a small number of answer products from a large number of non-answer products. Theoretically, this limitation of parallelism in problem size could be overcome by evolutionary algorithms [12, 13, 63].

2.3 *Scalability*

Conventional computing devices are based on the integration and spatial arrangement of same building blocks. For these systems, scaling is realized by integrating more building blocks. In contrast, DNA computing relies on specific interactions of orthogonal sequences. The 4-base coding nature and base pairing rule of DNA sequences support a rich orthogonal molecule library for large-scale computing. However, with the increasing number of required molecule types, the Hamming distances between DNA strands become smaller, leading to transient or stable unwanted binding of

DNA strands [15]. In addition, molecules participating in a basic function need to search for each other through diffusion. Increasing molecule types will increase the chance for a molecule to bind to an unwanted molecule and thus decrease the probability to be found by its target molecule, which would result in the decrease of computing speed together with increases of leakage. Therefore, scalability of DNA computing is limited to a certain finite size that cannot be extended by simply adding more computing units.

3 Directions for Future Development and Potential Approaches

3.1 *Scaling-Up*

The number of computing elements in a computing system determines its executable program complexity. Similar to other computational machines, the improvement of circuit scale is an important direction of evolution. The first modern electronic computer ENIAC contained 18,000 electron tubes [64]. Nowadays, tens of billions of transistors are integrated into an everyday mobile phone chip. For DNA computing, the effective scope is still solving problems that contain a small number of nodes or variables, using less than a few hundred participating DNA strands. Therefore, circuit scaling-up is an important requisite for functional evolution of DNA computing, which raises challenges including: (1) the specificity and efficiency of molecular reactions deteriorated with the increase of participating molecules; (2) the lack of a scalable computing architecture to realize the automatic mapping of complex algorithms to hardware implementations. For scaling up DNA computing, several strategies have been proposed and further explorations could be worthwhile.

3.1.1 Spatial Separation

A whole reaction can be split into different compartments by spatial separation. As a result, the molecules from different compartments are restricted from meeting each other; therefore, the reactions can be carried out efficiently in each compartment. Both semiconductor circuits and cells use the spatial separation strategy to control the material and information transmission pathways to complete complex computing tasks. In DNA computing, spatial separation has also been explored [65–68]. In 2011, Chandran et al. proposed a theoretical framework to implement parallel and scalable computation, using localized strand displacement reactions on the surface of DNA nanostructure [65]. In 2016, Genot et al. realized simultaneous observation of 10^4 reactions, by encapsulating a computational reaction system with various input conditions into droplets [67]. The molecules in each droplet reacted independently,

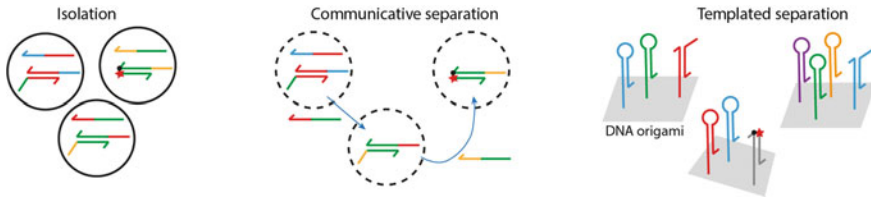


Fig. 2 Three types of spatial separation that may inspire design of DNA computing systems

since the oil film blocked molecular communication between droplets (Fig. 2a), which enabled parallel and high-resolution mapping of circuits.

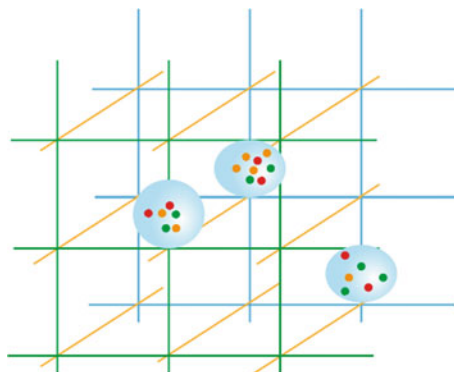
Communicative separation allows different spatial positions to store different chemical information with input/output communication. In 2019, Joesaar et al. encapsulated basic logic blocks into proteinosomes as protocells to mimic the function of natural cells [55]. As the output of one reaction, the released DNA strands in one protocell could pass through protocell membranes, diffuse in solution, and then enter another protocell, triggering the reactions in the downstream as input strands. This approach was supported by protocell–protocell communications (Fig. 2b), which holds potential for high-performance DNA computing with distributed systems. However, the computation demonstrated so far is limited to several protocells with single logic gates inside. Circuit size in each protocell and communication efficiency between protocells remain to be explored.

The addressability of DNA origami at nanometer precision facilitated the confinement of molecule reactions on DNA origami surface, which enables templated separation of reactions on different origamis (Fig. 2c). In 2017, Chatterjee et al. proposed spatialized information propagation by using DNA origami as the canvas to design circuits [33]. The circuit on DNA origami receives input signals and fuel strands from the solution and releases output strands for readout, allowing signal communication between origamis. However, circuits across multiple origamis via inter-origami communication have not been realized. Due to the lack of threshold and amplifier functions in such spatial-separated systems, this strategy still faces challenges when increasing scale of the circuit. DNA self-assembly may provide alternative solutions to construct localized response elements with noise suppression and signal amplification functions.

3.1.2 Combination of Order and Disorder

Collision events of DNA molecules in solution are disordered, which brings a high degree of parallelism to DNA computation together with inaccuracy at some extent. The parallelism means subdividing a deterministic space into a probability space for one calculation. Taking the search algorithm as an example, in sequential computing, each time a possible path is explored, and a specific output is generated. For parallel DNA molecular reactions, all possible paths are explored simultaneously. As the

Fig. 3 Schematic illustration of high-level ordered and low-level disordered architecture



number of possible reaction paths increases, detectors with higher sensitivity and accuracy are required to obtain the solution for a problem. In addition, as the number of molecules increases in the system, the probability for a single molecule of being at the state of non-specific binding is higher, which will consequently reduce the speed and probability along the correct reaction path and increase the probability of signal leakage.

Living systems provide unique examples for coordinating reactions involving a large amount and variety of molecules. In cells, spatial compartments are utilized to confine reactions into small reaction containers. With a certain degree of fluidity, the skeletal structures of cells provide a heterogeneous environment for disordered reactions. Cells aggregate to form ordered tissues, organs, and finally the organisms. With the combination of disorder and order, organisms have evolved advanced computing capabilities (e.g., learning, thinking, and decision-making). High-precision manufacturing technologies, including DNA nanotechnology, hold potential to build highly ordered containers for DNA molecular reactions. The ordered organization of computing modules, together with the disordered molecular reactions, will provide high parallelism and overall coordination to the computing system (Fig. 3). It is possible to develop more complex artificial DNA-based computing systems *in vitro* with improved synthetic intelligence.

3.1.3 Reversible and Directional Reaction

Currently, the generate-and-test approach is the most widely utilized one to experimentally demonstrate DNA computing process. When the scale of potential solutions exceeds the amounts of available molecules, a problem becomes theoretically infeasible. In fact, the faithful readout of the result is also limited by the proportion of correct calculation result. A solution to a problem could be viewed as a correct assembly of DNA molecules, and a high yield of the correct DNA assembly will be of benefit to the filtration of correct answer. If the yield is too low, a solvable problem may be misinterpreted as no solution. Condon and coworkers proposed a strategy

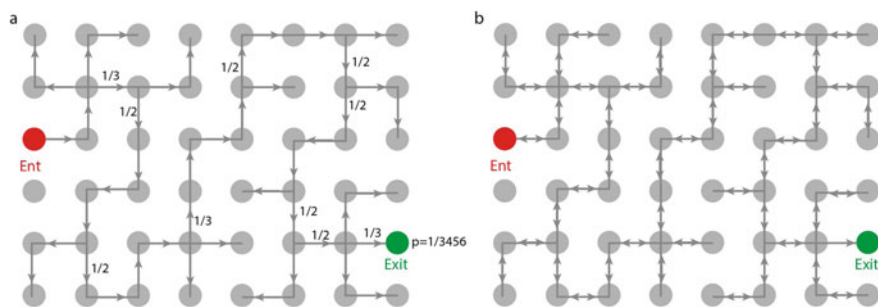


Fig. 4 Schematic illustrations of irreversible search algorithm (a) and reversible search algorithm (b) for a maze

using reversible strand displacement reactions for DNA computations that are space and energy efficient [69]. Thubagere et al. proposed a random walk-based algorithm for cargo-sorting [46], in which a single-stranded DNA robot picks up cargo irreversibly through toehold-mediated strand displacement reactions. Carrying the cargo, the robot performs a reversible random walk among adjacent tracks via toehold exchange until reaching the goal track for cargo drop-off.

The reversible motion strategy mentioned above could be extended to solve complex optimization problems. Taking a maze-solving problem for example, assuming the probability of stepping in each allowed direction is equal, the probability for a navigator to reach the exit of the maze shown in Fig. 4a is $1/3456$. As each individual navigator randomly follows a correct or wrong path for the maze solution, it only requires 16 steps to reach the exit (Fig. 5a). However, the probability suggests that only ~ 0.03 nM correct assembly would be obtained with 100 nM navigators, making it hard to detect the correct solution. Using a reversible search algorithm, the navigator could return to a node that has been visited, while its last step to exit is irreversible (Fig. 4b). Through this approach, there is no dead end except the exit; thus, every navigator is capable of reaching the exit. In a simulation, more than 50% navigators reached the exit within 1000 steps (Fig. 5b). With possible repeated visit of an intermediate node, it takes remarkably more time to reach the exit for the reversible navigator than the irreversible one. However, the arrived percentage of reversible navigators exceeds that of irreversible navigators in 100 steps (Fig. 5c). Sacrificing time for higher success probability may provide an approach. For a reversible system, time sacrifice will probably lead to higher success probability with higher yields of correct DNA assemblies, which may provide solutions to complex tasks beyond the practical computing power.

3.1.4 More Efficient Molecular Searching Modes

In homogeneous solutions, molecules recognize each other mainly through diffusion. This is why DNA computing systems constructed from diffusive building blocks

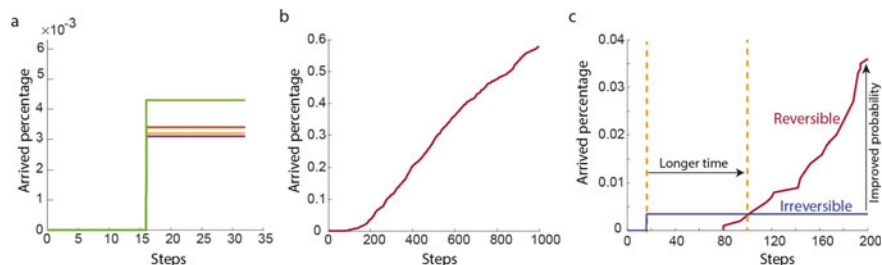


Fig. 5 Simulated overall success rates of maze solving. **a** All navigators that could reach the exit undergoes 16-step propagation in the irreversible algorithm. **b** Arrived percentage increases with step numbers in the reversible algorithm. **c** Comparison of arrived percentage in the two propagation modes within 200 steps

usually face a limitation in reaction rates toward the correct pathway. Commonly, tens of hours are needed to finish calculation when hundreds of DNA strands are involved [45]. Fast computing under low DNA concentrations could be realized by introducing new molecular searching modes. As a successful demonstration, calculations were completed in minutes under nanomolarity concentrations [33], by using DNA origami to confine the diffusion of each computing element into nanoscale regions.

Inspiration may also be obtained from the natural systems. In the cellular environment, the searching process of a protein toward its target DNA segments generally involves complex motions, including sliding, hopping, and intersegmental transfer as well as diffusion [70] (Fig. 6). When the DNA strand is stretched, protein prefers 1-dimensional lateral search. When the DNA strand is coiled, which is the native state, proteins can transfer between spatially adjacent segments. Combining these searching modes, even at very low concentrations, proteins can realize fast target search throughout the whole cell. Learning from these natural molecular searching modes, high-performance DNA computing may be developed based on novel molecular interaction mechanisms. The accurate spatial addressing property of DNA nanotechnology and tunable mechanical rigidity of DNA nanostructures may offer novel scaffolds to construct molecular machines with new molecule motion modes. With these molecular machines, more fast and efficient molecular recognition may be possible, which would realize the increase of the executable program complexity of DNA computing systems.

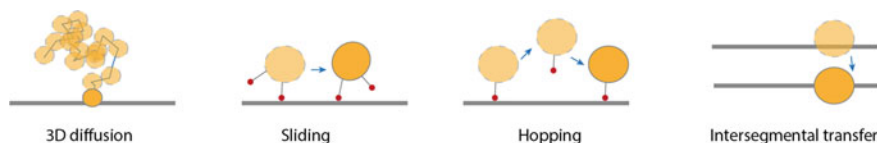


Fig. 6 Possible DNA searching modes of proteins in cells that may inspire more efficient molecular interactions for DNA computing in solution

3.2 *Updating and Reusing*

As a natural computer, life accumulates memories from past experiences, renews, and upgrades itself. Electrical computing chips also update their states either triggered by periodic clock or aperiodic pulse signals. Most proposed DNA computing devices are disposable and cannot be reused, because computing elements are permanently destroyed after calculation. Therefore, DNA computing is less developed in this respect, and developing renewable and reusable circuits will greatly expand the application scope of DNA computing.

3.2.1 **Hardware Resetting**

In DNA computing, the reset of the hardware state is based on chemical reactions, which is an important challenge that limits the sustainable use of computing devices. Spontaneous chemical reactions follow the tendency of energy change. By adding new strands to trigger reverse strand displacement [71], or by using the action of nicking enzyme to change the energy state of the system [72], the reaction can be reversed, and the input signal can be degraded, thereby realizing the reset of the computing device. Although resettable circuit implementations have been validated, the recovered concentration of input strands for next computing cycle reduced rapidly, making the circuit incapable of recycling. To perform sequential operations like electronic computers, further exploration on the design of a resettable DNA computing system is needed. Meanwhile, in combination with the parallelism of molecular reactions, highly parallel computing within one clock cycle may be developed. In this direction, to improve the reset efficiency, it is necessary to simplify the molecular structure design with further understanding of the underlying mechanisms.

3.2.2 **Iteration and Update of Molecular Reaction Networks**

Neuromorphic computing empowers artificial devices to learn from new inputs and realize self-update. Recently, DNA circuits-based neural network computing has been demonstrated [44, 45, 57]. However, the weight values of these neural networks were trained *in silico*. This one-time-use feature makes DNA circuits unable to renew themselves and thus incapable of learning.

Evolutionary DNA algorithms have been proposed to overcome parallelism limitations by dividing the selection of the final answer from a single huge pool into recursive selections from various small pools [12, 13]. For example, in Systematic Evolution of Ligands by EXponential Enrichment (SELEX), a destructive process is performed to remove intermediate results that do not fit the constraints of a problem. These iterative selections allow evolution of computing results to approach the solution, which is different from the Adleman-style computing with a single selection at the final step [73–76]. The evolutionary strategy also provides possibilities to

achieve molecular model training in machine learning. Rondelez's team has developed a series of evolutionary DNA reaction networks using a toolbox of DNA processing enzymes, i.e., polymerase, nickase, and exonuclease [19, 47, 77–79]. With a rich library of DNA processing enzymes to generate, transfer, and degrade DNA signals, it is promising to experimentally implement more complicated evolutions with DNA-based reaction networks to mimic biological systems in vitro.

4 Summary

DNA computing has evolved, slowly but steadily during the last 30 years. Despite the remarkable progress, challenges remain in many facets, such as function diversity, feasible circuit size, and computing efficiency. In particular, DNA computing relies on molecular diffusion and recognition of DNA molecules, which is fundamentally different with conventional and other type of computing systems that use a universal signal (e.g., electron or photon). We envision that next-generation DNA computing with molecular intelligence may evolve with inspiration from both natural living systems and electronic computers.

Acknowledgements We thank Erik Winfree, Darko Stefanovic, and other two anonymous reviewers for many helpful suggestions during the revision of the manuscript.

References

1. T. van der Valk, P. Pecnerova, D. Diez-Del-Molino, A. Bergstrom, J. Oppenheimer, S. Hartmann, G. Xenikoudakis, J.A. Thomas, M. Dehasque, E. Saglican, F.R. Fidan, I. Barnes, S. Liu, M. Somel, P.D. Heintzman, P. Nikolskiy, B. Shapiro, P. Skoglund, M. Hofreiter, A.M. Lister, A. Gotherstrom, L. Dalen, Million-year-old DNA sheds light on the genomic history of mammoths. *Nature* **591**(7849), 265–269 (2021)
2. J.D. Watson, F.H.C. Crick, Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid. *Nature* **171**(4356), 737–738 (1953)
3. N.C. Seeman, Nucleic acid junctions and lattices. *J. Theor. Biol.* **99**(2), 237–247 (1982)
4. L.M. Adleman, Molecular computation of solutions to combinatorial problems. *Science* **266**(5187), 1021–1024 (1994)
5. R.J. Lipton, DNA solution of hard computational problems. *Science* **268**(5210), 542–545 (1995)
6. Q.H. Liu, L.M. Wang, A.G. Frutos, A.E. Condon, R.M. Corn, L.M. Smith, DNA computing on surfaces. *Nature* **403**(6766), 175–179 (2000)
7. K. Sakamoto, H. Gouzu, K. Komiya, D. Kiga, S. Yokoyama, T. Yokomori, M. Hagiya, Molecular computation by DNA hairpin formation. *Science* **288**(5469), 1223–1226 (2000)
8. L.M. Wang, Q.H. Liu, R.M. Corn, A.E. Condon, L.M. Smith, Multiple word DNA computing on surfaces. *J. Am. Chem. Soc.* **122**(31), 7435–7440 (2000)
9. R.S. Braich, N. Chelyapov, C. Johnson, P.W.K. Rothmund, L. Adleman, Solution of a 20-variable 3-SAT problem on a DNA computer. *Science* **296**(5567), 499–502 (2002)
10. Q. Ouyang, P.D. Kaplan, S. Liu, A. Libchaber, DNA solution of the maximal clique problem. *Science* **278**(5337), 446–449 (1997)

11. J.Y. Lee, S.Y. Shin, T.H. Park, B.T. Zhang, Solving traveling salesman problems with DNA molecules encoding numerical values. *Biosystems* **78**(1–3), 39–47 (2004)
12. T. Back, J.N. Kok, G. Rozenberg, Evolutionary computation as a paradigm for DNA-based computing. *Evol. Comput.*, 15–40 (2002)
13. W.P. Stemmer, The evolution of molecular computation. *Science* **270**(5241), 1510 (1995)
14. A.D. Ellington, M.P. Robertson, K.D. James, J.C. Cox, Strategies for DNA computing. *DNA Based Comp.* **3**, 173–184 (1997)
15. K.D. James, A.R. Boles, D. Henckel, A.D. Ellington, The fidelity of template-directed oligonucleotide ligation and its relevance to DNA computation. *Nucleic Acids Res.* **26**(22), 5203–5211 (1998)
16. M.N. Stojanovic, T.E. Mitchell, D. Stefanovic, Deoxyribozyme-based logic gates. *J. Am. Chem. Soc.* **124**(14), 3555–3561 (2002)
17. R.M. Dirks, N.A. Pierce, Triggered amplification by hybridization chain reaction. *Proc. Natl. Acad. Sci. U.S.A.* **101**(43), 15275 (2004)
18. G. Seelig, D. Soloveichik, D.Y. Zhang, E. Winfree, Enzyme-free nucleic acid logic circuits. *Science* **314**(5805), 1585–1588 (2006)
19. A. Padirac, T. Fujii, Y. Rondelez, Bottom-up construction of in vitro switchable memories. *Proc. Natl. Acad. Sci. U.S.A.* **109**(47), E3212–E3220 (2012)
20. J.Y. Kishi, T.E. Schaus, N. Gopalkrishnan, F. Xuan, P. Yin, Programmable autonomous synthesis of single-stranded DNA. *Nat. Chem.* **10**(2), 155–164 (2018)
21. Y.J. Chen, N. Dalchau, N. Srinivas, A. Phillips, L. Cardelli, D. Soloveichik, G. Seelig, Programmable chemical controllers made from DNA. *Nat. Nanotechnol.* **8**(10), 755–762 (2013)
22. C.D. Mao, T.H. LaBean, J.H. Reif, N.C. Seeman, Logical computation using algorithmic self-assembly of DNA triple-crossover molecules. *Nature* **407**(6803), 493–496 (2000)
23. P. Yin, H.M. Choi, C.R. Calvert, N.A. Pierce, Programming biomolecular self-assembly pathways. *Nature* **451**(7176), 318–322 (2008)
24. X. Chen, A.D. Ellington, Shaping up nucleic acid computation. *Curr. Opin. Biotechnol.* **21**(4), 392–400 (2010)
25. B. Yurke, A.J. Turberfield, A.P. Mills, F.C. Simmel, J.L. Neumann, A DNA-fuelled molecular machine made of DNA. *Nature* **406**(6796), 605–608 (2000)
26. B. Yurke, A.P. Mills, Using DNA to Power Nanostructures. *Genet. Prog. Evolvable Mach.* **4**(2), 111–122 (2003)
27. N. Srinivas, T.E. Ouldrige, P. Sulc, J.M. Schaeffer, B. Yurke, A.A. Louis, J.P.K. Doye, E. Winfree, On the biophysics and kinetics of toehold-mediated DNA strand displacement. *Nucleic Acids Res.* **41**(22), 10641–10658 (2013)
28. D.Y. Zhang, E. Winfree, Control of DNA strand displacement kinetics using toehold exchange. *J. Am. Chem. Soc.* **131**(47), 17303–17314 (2009)
29. R.R.F. Machinek, T.E. Ouldrige, N.E.C. Haley, J. Bath, A.J. Turberfield, Programmable energy landscapes for kinetic control of DNA strand displacement. *Nat. Commun.* **5**(1), 1–9 (2014)
30. N.E.C. Haley, T.E. Ouldrige, I. Mullor Ruiz, A. Geraldini, A.A. Louis, J. Bath, A.J. Turberfield, Design of hidden thermodynamic driving for non-equilibrium systems via mismatch elimination during DNA strand displacement. *Nat. Commun.* **11**(1), 2562 (2020)
31. J. Li, A. Johnson-Buck, Y.R. Yang, W.M. Shih, H. Yan, N.G. Walter, Exploring the speed limit of toehold exchange with a cartwheeling DNA acrobat. *Nat. Biotechnol.* **13**(8), 723–729 (2018)
32. L. Qian, E. Winfree, A simple DNA gate motif for synthesizing large-scale circuits. *J. R. Soc. Interface* **8**(62), 1281–1297 (2011)
33. G. Chatterjee, N. Dalchau, R.A. Muscat, A. Phillips, G. Seelig, A spatially localized architecture for fast and modular DNA computing. *Nat. Nanotechnol.* **12**(9), 920–927 (2017)
34. H. Lv, Q. Li, J. Shi, C. Fan, F. Wang, Biocomputing based on DNA strand displacement reactions. *ChemPhysChem.* **22**(12), 1151–1166 (2021)
35. J. Chao, J. Wang, F. Wang, X. Ouyang, E. Kopperger, H. Liu, Q. Li, J. Shi, L. Wang, J. Hu, L. Wang, W. Huang, F.C. Simmel, C. Fan, Solving mazes with single-molecule DNA navigators. *Nat. Mater.* **18**(3), 273–279 (2019)

36. S. Bi, S.Z. Yue, S.S. Zhang, Hybridization chain reaction: a versatile molecular tool for biosensing, bioimaging, and biomedicine. *Chem. Soc. Rev.* **46**(14), 4281–4298 (2017)
37. W. Meng, R.A. Muscat, M.L. McKee, P.J. Milnes, A.H. El-Sagheer, J. Bath, B.G. Davis, T. Brown, R.K. O'Reilly, A.J. Turberfield, An autonomous molecular assembler for programmable chemical synthesis. *Nat. Chem.* **8**(6), 542–548 (2016)
38. C.A. Figg, P.H. Winegar, O.G. Hayes, C.A. Mirkin, Controlling the DNA Hybridization Chain Reaction. *J. Am. Chem. Soc.* **142**(19), 8596–8601 (2020)
39. D.Y. Zhang, A.J. Turberfield, B. Yurke, E. Winfree, Engineering entropy-driven reactions and networks catalyzed by DNA. *Science* **318**(5853), 1121–1125 (2007)
40. Y. Benenson, R. Adar, T. Paz-Elizur, Z. Livneh, E. Shapiro, DNA molecule provides a computing machine with both data and fuel. *Proc. Natl. Acad. Sci. U.S.A.* **100**(5), 2191–2196 (2003)
41. R. Pei, E. Matamoros, M. Liu, D. Stefanovic, M.N. Stojanovic, Training a molecular automaton to play a game. *Nat Nanotechnol* **5**(11), 773–777 (2010)
42. L. Qian, E. Winfree, Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**(6034), 1196–1201 (2011)
43. H. Su, J. Xu, Q. Wang, F. Wang, X. Zhou, High-efficiency and integrable DNA arithmetic and logic system based on strand displacement synthesis. *Nat. Commun.* **10**(1), 5390 (2019)
44. L. Qian, E. Winfree, J. Bruck, Neural network computation with DNA strand displacement cascades. *Nature* **475**(7356), 368–372 (2011)
45. K.M. Cherry, L.L. Qian, Scaling up molecular pattern recognition with DNA-based winner-take-all neural networks. *Nature* **559**(7714), 370–376 (2018)
46. A.J. Thubagere, W. Li, R.F. Johnson, Z. Chen, S. Doroudi, Y.L. Lee, G. Izatt, S. Wittman, N. Srinivas, D. Woods, E. Winfree, L. Qian, A cargo-sorting DNA robot. *Science* **357**(6356), eaan6558 (2017)
47. T. Fujii, Y. Rondelez, Predator-prey molecular ecosystems. *ACS Nano* **7**(1), 27–34 (2013)
48. D. Han, C. Wu, M. You, T. Zhang, S. Wan, T. Chen, L. Qiu, Z. Zheng, H. Liang, W. Tan, A cascade reaction network mimicking the basic functional steps of adaptive immune response. *Nat. Chem.* **7**(10), 835–841 (2015)
49. N. Srinivas, J. Parkin, G. Seelig, E. Winfree, D. Soloveichik, Enzyme-free nucleic acid dynamical systems. *Science* **358**(6369), eaal2052 (2017)
50. Y. Amir, E. Ben-Ishay, D. Levner, S. Ittah, A. Abu-Horowitz, I. Bachelet, Universal computing by DNA origami robots in a living animal. *Nat. Nanotechnol.* **9**(5), 353–357 (2014)
51. T. Masubuchi, M. Endo, R. Iizuka, A. Iguchi, D.H. Yoon, T. Sekiguchi, H. Qi, R. Inuma, Y. Miyazono, S. Shoji, T. Funatsu, H. Sugiyama, Y. Harada, T. Ueda, H. Tadakuma, Construction of integrated gene logic-chip. *Nat. Nanotechnol.* **13**(10), 933 (2018)
52. T. Song, A. Eshra, S. Shah, H. Bui, D. Fu, M. Yang, R. Mokhtar, J. Reif, Fast and compact DNA logic circuits based on single-stranded gates using strand-displacing polymerase. *Nat. Nanotechnol.* **14**(11), 1075–1081 (2019)
53. F. Wang, H. Lv, Q. Li, J. Li, X. Zhang, J. Shi, L. Wang, C. Fan, Implementing digital computing with DNA-based switching circuits. *Nat. Commun.* **11**(1), 121 (2020)
54. S.M. Chirieleison, P.B. Allen, Z.B. Simpson, A.D. Ellington, X. Chen, Pattern transformation with DNA circuits. *Nat. Chem.* **5**(12), 1000–1005 (2013)
55. A. Joesaar, S. Yang, B. Bogels, A. van der Linden, P. Pieters, B. Kumar, N. Dalchau, A. Phillips, S. Mann, T.F.A. de Greef, DNA-based communication in populations of synthetic protocells. *Nat. Nanotechnol.* **14**(4), 369–378 (2019)
56. D. Woods, D. Doty, C. Myhrvold, J. Hui, F. Zhou, P. Yin, E. Winfree, Diverse and robust molecular algorithms using reprogrammable DNA self-assembly. *Nature* **567**(7748), 366–372 (2019)
57. C. Zhang, Y. Zhao, X. Xu, R. Xu, H. Li, X. Teng, Y. Du, Y. Miao, H.C. Lin, D. Han, Cancer diagnosis with DNA molecular computation. *Nat. Nanotechnol.* **15**(8), 709–715 (2020)
58. J. Chen, S. Zhou, J. Wen, Concatenated logic circuits based on a three-way DNA junction: a keypad-lock security system with visible readout and an automatic reset function. *Angew. Chem. Int. Ed.* **54**(2), 446–450 (2015)

59. X. Xiong, M. Xiao, W. Lai, L. Li, C. Fan, H. Pei, Optochemical control of DNA-switching circuits for logic and probabilistic computation. *Angew. Chem. Int. Ed.* **60**(7), 3397–3401 (2020)
60. Y.J. Guo, D.B. Yao, B. Zheng, X.B. Sun, X. Zhou, B. Wei, S.Y. Xiao, M. He, C.X. Li, H.J. Liang, pH-Controlled detachable DNA circuitry and its application in resettable self-assembly of spherical nucleic acids. *ACS Nano* **14**(7), 8317–8327 (2020)
61. R. Orbach, F. Wang, O. Lioubashevski, R.D. Levine, F. Remacle, I. Willner, A full-adder based on reconfigurable DNA-hairpin inputs and DNAzyme computing modules. *Chem. Sci.* **5**(9), 3381–3387 (2014)
62. E. Schrodinger, *What is Life? The Physical Aspect of the Living Cell* (University Press, Cambridge, 1951)
63. D.J.F. Jeng, I. Kim, J. Watada, DNA-based evolutionary algorithm for cable trench problem. *Knowl. Based Intell. Inform. Eng. Syst. Pt 3, Proc.* **4253**, 922–929 (2006)
64. A.W. Burks, From ENIAC to the stored-program computer: two revolutions in computers, in *A History of Computing in the Twentieth Century* (Elsevier, 1980), pp. 311–344
65. H. Chandran, N. Gopalkrishnan, A. Phillips, J. Reif, Localized hybridization circuits. *DNA Comput. Mole. Program.* **6937**, 64–83 (2011)
66. L. Qian, E. Winfree, in *Parallel and Scalable Computation and Spatial Dynamics with DNA-based Chemical Reaction Networks on a Surface (DNA Computing and Molecular Programming. 20th International Conference, DNA 20. Proceedings: LNCS 8727)*, (2014), pp. 114–131
67. A.J. Genot, A. Baccouche, R. Sieskind, N. Aubert-Kato, N. Bredeche, J.F. Bartolo, V. Taly, T. Fujii, Y. Rondelez, High-resolution mapping of bifurcations in nonlinear biochemical circuits. *Nat. Chem.* **8**(8), 760–767 (2016)
68. M. Weitz, J. Kim, K. Kapsner, E. Winfree, E. Franco, F.C. Simmel, Diversity in the dynamical behaviour of a compartmentalized programmable biochemical oscillator. *Nat. Chem.* **6**(4), 295–302 (2014)
69. C. Thachuk, A. Condon, in *Space and Energy Efficient Computation with DNA Strand Displacement Systems, DNA Computing and Molecular Programming* (Berlin, Heidelberg, 2012), pp. 135–149.
70. M. Klein, T.J. Cui, I. MacRae, C. Joo, M. Depken, Skipping and sliding to optimize target search on protein-bound DNA and RNA. *BioRxiv* (2020). <https://doi.org/10.1101/2020.06.04.133629>
71. S. Garg, S. Shah, H. Bui, T. Song, R. Mokhtar, J. Reif, Renewable time-responsive DNA circuits. *Small* **14**(33), 1801470 (2018)
72. C. Zhang, Z. Wang, Y. Liu, J. Yang, X. Zhang, Y. Li, L. Pan, Y. Ke, H. Yan, Nicking-assisted reactant recycle to implement entropy-driven DNA circuit. *J. Am. Chem. Soc.* **141**(43), 17189–17197 (2019)
73. D. Faulhammer, A.R. Cukras, R.J. Lipton, L.F. Landweber, Molecular computation: RNA solutions to chess problems. *Proc. Natl. Acad. Sci. U.S.A.* **97**(4), 1385–1389 (2000)
74. A. Drame-Maigne, A.S. Zadorin, I. Golovkova, Y. Rondelez, Quantifying the performance of micro-compartmentalized directed evolution protocols. *Life* **10**(2), (2020)
75. R. Deaton, R.C. Murphy, J. Rose, M. Garzon, D.R. Franceschetti, S. Stevens, in *A DNA based implementation of an evolutionary search for good encodings for DNA computation. Proceedings of 1997 IEEE International Conference on Evolutionary Computation (ICEC'97) (IEEE, 1997)*, pp. 267–271
76. J.H. Chen, D.H. Wood, Computation with biomolecules. *Proc. Natl. Acad. Sci. U.S.A.* **97**(4), 1328–1330 (2000)
77. A. Dramé-Maigné, R. Espada, G. McCallum, R. Sieskind, Y. Rondelez, Directed evolution of enzymes based on in vitro programmable self-replication. *BioRxiv* (2021). <https://doi.org/10.1101/2021.04.22.440993>

78. H.Q. Dinh, N. Aubert, N. Noman, T. Fujii, Y. Rondelez, H. Iba, An effective method for evolving reaction networks in synthetic biochemical systems. *IEEE Trans. Evol. Comput.* **19**(3), 374–386 (2015)
79. N. Aubert, C. Mosca, T. Fujii, M. Hagiya, Y. Rondelez, Computer-assisted design for scaling up systems based on DNA reaction networks. *J. R. Soc. Interface* **11**(93), 20131167 (2014)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



DNA Nanotechnology Research in Japan



Satoshi Murata

Abstract In this essay, the evolution of DNA nanotechnology research in Japan to date will be reviewed. The expansion of the research community in Japan and the trends in regard to the selection of project themes will be elucidated, along with the identification of the researchers who participated in these projects. Some aspects of the research history of the author, who entered from the field of robotics, are introduced, as this information may be of interest to young students and researchers.

1 Introduction

In 1982, when Professor Seeman began his research on DNA nanostructures, there were no studies of this kind in Japan. In 1994, Adleman's work on DNA computers was published, and two years later, Masami Hagiya of the University of Tokyo initiated a research project on molecular computing [1]. Using this as a starting point, the history of DNA nanotechnology in Japan has evolved over approximately 25 years. Hagiya would then lead research in Japan for the next 20 years. Figure 1 shows the genealogy of related projects in Japan.

Early projects explored the possibility of massively parallel computation with molecules, and in 2001, a project on molecular programming [2] began, which included researchers from disciplines such as mechanical, materials, and medical engineering. These scientists introduced the idea of extending the function of the system beyond "computation," to sensing of the environment and acting accordingly. After several years of preparation, the project of "molecular robotics" was launched in 2012 [3].

There are several reasons why the concept of "molecular robots" originated in Japan. One is that Japan is a country where robotics research is very active, and there are a large number of researchers, in both the theoretical and applied fields. There is also a latent familiarity with the word "robot" through various depictions in cartoons and animations. On the other hand, research in chemistry, especially

S. Murata (✉)

Department of Robotics, Graduate School of Engineering, Tohoku University, Sendai, Japan
e-mail: satoshi.murata.a4@tohoku.ac.jp

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*,
Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_4

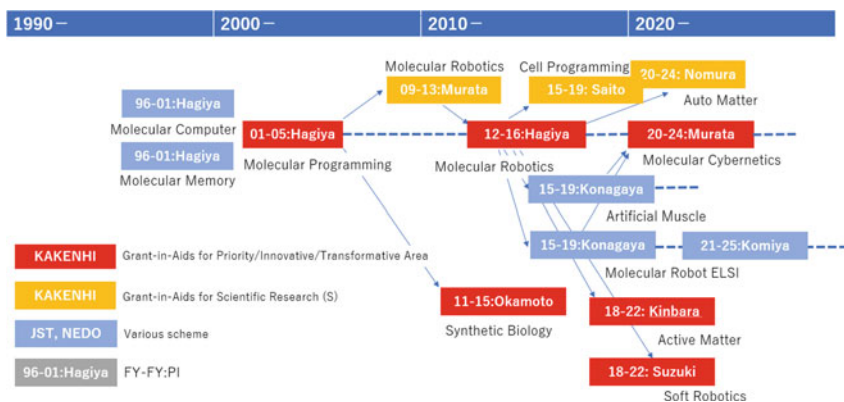


Fig. 1 DNA nanotechnology-related projects in Japan

organic chemistry and polymer science, is also active in Japan, as is evidenced by the fact that the country has produced a number of Nobel laureates in chemistry. Many researchers in these fields are working on specific applications, while others are attempting to create new molecules out of scientific curiosity, with the expectation that the application of their molecules will be beneficial in the future. The author believes that these types of researchers may have been attracted by the concept of “building robots with molecules.”

During the five years of the project, several prototypes have been developed. The amoeba-type molecular robot contained various molecular devices that were encapsulated in artificial cells. In the gellular automaton, programming of a molecular computing system on a gel medium was investigated. In parallel with the progress of the project, a community of researchers in molecular robotics was organized as an arm of the Society of Instrument and Control Engineers (SICE), and research workshops and national conferences were held regularly. In 2020, a project on “molecular cybernetics” was launched, with the author as its representative [4]. The goal of this project was to create a system with a certain information-processing capabilities by combining artificial cells developed through molecular robotics.

2 How the Author Got Involved in DNA Nanotechnology

The author specializes in robotics, specifically autonomous distributed robotic systems (DARS) [5]. Research on these systems focuses on flexibility and adaptability, which is not possible with conventional robots, through the coordination of a large number of autonomous robotic agents. The author is particularly interested in “homogeneous” autonomous distributed robot systems, that is, systems consisting of a large number of robot modules of exactly the same type. Each module is a small robot with a microcomputer, sensors, and actuators. (This type of robot is therefore

called a modular robot.) Similar to building a house with Lego blocks, the modules not only connect with each other to form different shapes, but also move and recombine themselves to change the shape of the whole structure without external aid. My group proposed the concept of such modular robots and developed several prototypes to demonstrate this concept [6]. For example, we have created a “self-repairing” robot that can automatically replace a failed module with a spare one, and a modular robot that can autonomously transform itself into different forms, such as that of a dog or a snake, and move according to these forms.

Several other groups have also developed homogeneous distributed robots, but the number of modules was limited to between 10 and 100. Therefore, these prototypes had reduced functionality. There are various reasons why the number of modules cannot be increased (i.e., difficulty in creating a scalable system). For instance, each module must contain such things as a microcomputer, sensors, motors, batteries, and communication devices in a compact space, which is difficult to design in itself; it is challenging to guarantee mechanical, electrical, and information reliability, and, when possible, it comes with a large cost. Modular robots are expected to be used for exploration and rescue in unknown environments because of their ability to adapt to the conditions, but full-scale applications would require thousands or tens of thousands of modules, which is hindered with current technology. The Kilobot developed by Nagpal et al. is a system with 1000 modules and is probably the largest modular robot built to date [7]; however, each module is only able to move on a plane, and they cannot be connected to each other.

In 2000, when I moved from the National Mechanical Engineering Laboratory (MEL) to the Tokyo Institute of Technology, I was unsure if I would continue my research on modular robots. I had been simulating the formation of networks of various shapes using a model in which mass points were connected by virtual springs [8]. I learned that it is possible to make a jigsaw puzzle of molecules using DNA, that is, they could self-assemble [9]. I had always been interested in the term “self-assembly” and even called my modular robot a “self-assembling machine” [10]. Therefore, I began studying DNA tiles with my students and developed a method to improve the reliability of algorithmic self-assembly, which had been proposed by Winfree and others at that time [11]. Subsequently, I learned directly from Professor Winfree and gradually entered this field.

Many mechanical engineers have changed their research subjects significantly in the course of their research life. My personal impression is that approximately 50% alter their research interests at some point. Professor Klavins at the University of Washington is another scientist who moved from the field of modular robots to that of bio/nanotechnology. He has been working on modular robots that operate stochastically [12] (his system of randomly moving pizza piece-like modules on an air-hockey table to make perfectly round pizzas is fascinating to watch). He is now a synthetic biologist.

Young people who are about to start their research should be prepared for the possibility that they may completely change their research field in the future. This does not mean starting the research from scratch, as the methodologies acquired up

to that point are often applicable for different research areas. Changing their focus expands their futures by broadening the range of methodologies available to them.

3 The Evolution of Projects in Japan

Let us return to the research projects in Japan. First, I would like to briefly explain the scientific research system in Japan. Most basic research in the sciences, engineering, and humanities is funded by public grants from the government. The main subsidy is a Grant-in-Aid for Scientific Research (KAKENHI) provided by the Ministry of Education, Culture, Sports, Science, and Technology (MEXT) [13]. In addition, the Japan Science and Technology Agency (JST), affiliated with MEXT, and the New Energy and Industrial Technology Development Organization (NEDO), affiliated with the Ministry of Economy, Trade and Industry (METI), also provide large research funds, but these funds are only for application-oriented projects.

In the field of DNA nanotechnology, KAKENHI is the main source of funding. There are various levels of KAKENHI, ranging from a few million yen per year for a small number of researchers to several hundred million yen per year for a few dozen researchers. All of these are competitive research funds, and each obtained with different levels of difficulty. As expected, projects with larger budgets have more difficulty obtaining these grants, and often the largest schemes are awarded after years of repeated applications. In particular, the KAKENHI scheme, which is awarded to groups of several dozen researchers (the scheme name has changed from “Priority Area Research” to “Innovative Area Research” to “Transformative Area Research,” but the content is similar), aims to establish a new academic community in the field. By following this transition, we can observe the trend of research in Japan.

3.1 *The 1980s and 1990s*

In the 1980s, DNA research in molecular biology was active in Japan, but no projects in relation to DNA nanotechnology had yet been initiated. DNA nanotechnology can be divided into two categories: structural DNA nanotechnology founded by Seeman, and DNA computing founded by Adleman, and considering these, DNA computing research actually began earlier in Japan. That is, two years after the first paper on DNA computing by Adleman [14], a research project on molecular computers (1996–2002) [1] was conducted under the leadership of Masami Hagiya at the University of Tokyo. Professor Adleman was well known in the field of computer science, and DNA computing was expected to be a massively parallel computing architecture that could outperform supercomputers. At that time, Takashi Yokomori (Waseda University) and others in the Japanese computer science community had obtained information about DNA computing through researchers in formal language theory.

In other words, expectations of molecular computation were raised at a relatively early stage in Japan. Senior researchers Yuichiro Anzai, in computer science (Keio University) and Shigeyuki Yokoyama, in structural molecular biology (University of Tokyo), who were in a position to select new projects under the “Research for the Future Program”, chose Professor Hagiya, who had been working on a follow-up to Adleman’s experiment, as the leader of the project. The influence of both the bottom-up interest of young researchers and the top-down direction of senior researchers at the beginning of this field in Japan is evident.

The molecular computer project [1] started by Hagiya et al. was joined by Takashi Yokomori, Akira Suyama (University of Tokyo) in biophysics, and later by Masayuki Yamamura (Tokyo Institute of Technology) in computer science. The SAT engine using DNA hairpin formation and Whiplash PCR are among the results of the project. The SAT engine solves the satisfiability problem known for hard combinatorial problems by using the hairpin formation of DNA strands [15, 16]. Whiplash PCR combines the opening/closing of DNA hairpins with polymerase elongation to achieve state representation and state transitions in a single hairpin molecule; the term “whiplash” came into use subsequently [17, 18]. In the same year, 1996, the project of “Molecular Memory” was conducted under Hagiya [19]. In this project, Azuma Ouchi (Hokkaido University), Jun Tanida (Osaka University), and others in computer science participated, and various types of molecular memories were developed including “nested primer molecular memory” [20], which amplified only DNA with a specific sequence by four-level PCR, and a molecular memory that read information spatially by transduction between two secondary structures using the infrared laser excitation of hairpin DNA fixed on the substrate plane. Conformational addressing of multiple hairpin DNAs [21] was also developed in this project.

3.2 *The 2000s*

In 2002–2007, the Grant-in-Aid for Scientific Research on Priority Areas “Molecular Programming” was conducted, again with Hagiya as the representative [2]. This was one of the largest KAKENHI research projects, involving approximately 50 researchers and lasting five years. In contrast to the previous research on “molecular computation,” which aimed to synthesize a desired function or structure by utilizing the computational potential of biomolecules, “molecular programming” considered the process of designing biomolecules and their chemical reactions as “programming” and aimed to develop a systematic programming methodology for molecular computation. For this project, in addition to the members of the previous projects, Daisuke Kiga (Waseda University) in synthetic biology, Kenzo Fujimoto (Japan Advanced Institute of Science and Technology) in nucleic acid chemistry, and the present author (Tohoku University) in robotics were involved. The first half of the project focused on the refinement of DNA sequence design technology, and the second half on its application in nanotechnology and synthetic biology.

For this project, Suyama et al. developed a molecular computational reaction system (R-TRACS) using DNA and RNA enzymes [22], and Kiga et al. created a bacterial computer using *E. coli* [23]. In addition, the author worked on the DNA tile error rate evaluation [24] with Winfree. At the same time, the author attempted to grow DNA tiles in microfluidic devices with Teruo Fujii (now President of the University of Tokyo). In Fujii's laboratory, Yannick Rondelez began studying the reaction system using DNA enzymes in 2009, and this developed into the PEN DNA toolbox [25].

Subsequently, frequent meetings have been held, mainly by those involved in the molecular programming project, and a community of researchers using DNA nanotechnology as a tool has been formed. In June 2002, the DNA8 workshop was held in Sapporo (Hokkaido University), hosted by the molecular programming project (Professor Seeman presented a keynote lecture).

3.3 *The 2010s*

After the end of the molecular programming project, proposals were made every year to obtain large project research funds, but none were accepted. In 2010, the author's proposal, "Development of molecular robotics by DNA nanoengineering," was selected as a relatively large project under the "Grant-in-Aid for Scientific Research (S)" scheme. Akinori Kuzuya (Kansai University), who studied DNA origami in Seeman's laboratory, and Shin-ichiro Nomura (Tohoku University) who was involved in artificial cell engineering participated in this project. This project introduced the term "molecular robotics," which uses a DNA nanotechnology to design component molecules and assemble them to create a functional molecular system that can respond autonomously to changes in the environment. The "Molecular Robotics Research Group" was established in the Society of Instrument and Control Engineers (SICE) during this time. The author served as the representative of this group. This research group continues, and the majority of the researchers involved in DNA nanotechnology and DNA computing in Japan have participated.

In 2011, Grant-in-Aid for Innovative Field "Synthetic Biology" was implemented (2011–2016) with Masahiro Okamoto (Kyushu University) as the representative. Yamamura, Kiga, and Suyama, who participated in the molecular programming, moved to this group.

In 2012, the project "Molecular Robotics—Creation of molecular robots with sensation and Intelligence" was adopted as Innovative Area, led by Hagiya [3]. The core members of this project were a combination of computer science researchers (Hagiya, Satoshi Kobayashi (University of Electro-Communications), Akihiko Konagaya (Tokyo Institute of Technology), and experimental and implementation researchers (the author and Hirohide Saito (Kyoto University), who was in the field of RNA nanotechnology).

Generally, a "robot" can be defined as an "autonomous system" that acquires information from the external environment using sensors, processes this information,

and acts on the environment accordingly. A body (structure) is also required to distinguish the system from the environment and to integrate these components. The goal of the project was to develop technologies for the design, fabrication, integration, and control of molecular robots.

An evolutionary scenario for molecular robots was proposed at the inception of the molecular robotics project (Fig. 2) [26]. This scenario predicts the gradual evolution of molecular robots over four stages: Generation 0: A single molecule acts as a robot. Its behavior is random due to thermal fluctuations. This would be similar to the molecular spider of Stojanovic et al. [27]; Generation 1: Artificial cell membranes encapsulating various molecular devices. When several kinds of molecules are enclosed in a container, the concept of “concentration” arises, and its behavior can be predicted to some extent by chemical reaction kinetics. This is called an amoeba-type molecular robot; Generation 2: Various molecular devices are dispersed in a gel medium, whereby each molecule has a spatial distribution, resulting in a system with information on both “concentration” and “position.” This is called a slime-type molecular robot; Generation 3: The molecular robot of Generation 1 is multicellularized. Multicellular robots can be realized with more complex and diverse functions; and Generation 4: Hybridization, in which the molecular robot is expected to be combined with conventional nanotechnology such as photolithography.

In the molecular robotics project, the goal was to achieve the first and second generations of this scenario. For Generation 1, the development of an amoeba-like molecular robot, we succeeded in constructing a system combining a light sensor, a DNA amplification circuit, a DNA molecular clutch, and a microtubule-kinesin molecular motor in a single artificial cell (liposome). We demonstrated that the liposome deforms like an amoeba and turns on and off using light stimuli [28]. For Generation 2, the slime-type molecular robot, a combination of BZ gel actuator (a self-oscillating gel actuator using the Belousov-Zhabotinsky reaction) and DNA computing was initially planned, but the operating condition of the BZ gel actuator required strong acidity, which is incompatible with the operating conditions for DNA,

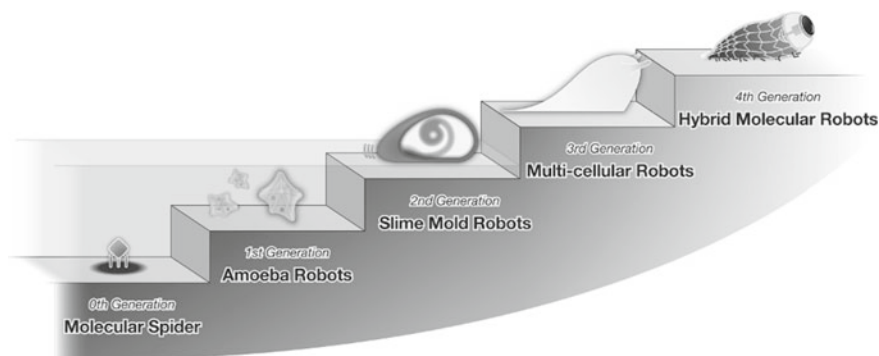


Fig. 2 Evolutionary scenario of molecular robotics

and the project was abandoned. Instead, the molecular implementation of a cellular automaton system in gel space, called a gellular automaton, was explored [29, 30].

In addition to amoeba-type molecular robots, various technologies have been developed. For example, RNA nanostructure devices have been developed as therapeutics. They function under the strong noise of various biomolecules and successfully control the fate of cancer cells [31]. Another example is a technique to speed up various DNA computational reactions by combining synthetic DNA and artificial nucleic acids [32]. Furthermore, microtubule assemblies controlled by light-responsive DNA [33] and gel actuators driven by DNA-controlled gel-sol phase transition [34] have been developed as actuators for molecular robots.

The results of the “molecular robotics” project were summarized in a textbook, “Introduction to Molecular Robotics” [35], which was published in Japanese. The English version of the textbook is currently being edited. In 2014, DNA20 was held in Kyoto (Kyoto University), hosted by the molecular robotics project. The conference included a keynote lecture by Professor Seeman on “Molecular machines made from DNA.”

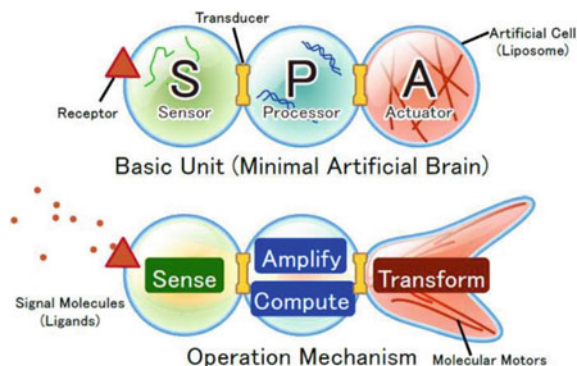
In the year following the completion of the molecular robotics project, two new KAKENHI projects, as Innovative Areas, were launched, namely “Molecular Engine” (2018–2023), led by Kazu Kinbara and “Creation of Soft Robotics” (2018–2023), led by Koichi Suzumori, both of whom were from the Tokyo Institute of Technology. These were the research areas of active matter and soft robotics, respectively. Some of the members of Molecular Robotics have participated in these projects, and collaboration between these areas is progressing.

3.4 *Current Research*

Four years after the molecular robotics project, a project entitled “Molecular Cybernetics—Construction of a minimal artificial brain by the power of chemistry” was adopted as a KAKENHI Transformative Area (A) [4]. Transformative Areas (A) are similar to Innovative Areas. The Molecular Cybernetics is led by the author with 18 core researchers including Taro Toyota (University of Tokyo) and Shin-ichiro Nomura, Akinori Kuzuya, and Takashi Nakakuki (Kyushu Institute of Technology) in control engineering.

The amoeba-type molecular robot can have a variety of molecular devices implemented in a single artificial cell (liposome). However, it is difficult to derive appropriate solution conditions under which all the different molecular species, such as light-responsive artificial nucleic acids, DNA devices, and molecular motors, can function. To solve this problem, we use multiple liposomes that contain solutions suitable for each of these molecular devices and bind them together to build a system. The liposomes are connected by a “transducer” for communication, which transmits molecular information without mixing the solutions. Our goal was to realize a simple learning function on a three-liposome system called a minimal artificial brain (Fig. 3).

Fig. 3 Minimal brain concept in molecular cybernetics project



In molecular cybernetics, there are several new approaches that have yet to be attempted. One of these is a variety of free services for members of the group. Kazunori Matsuura of Tottori University (peptide engineering) runs a peptide synthesis center that provides various peptide chains upon request from the members of the project. Keiji Murayama of Nagoya University (nucleic acid chemistry) runs the nucleic acid synthesis center, which provides nucleic acid sequences containing special artificial bases. In addition, Kuzuya of Kansai University will provide a range of single-molecule observation services using various microscopes and atomic force microscopes. Another new activity of the project is journalist-in-residence. Miki-hito Tanaka of Waseda University (social analysis) hosts external non-specialists, such as newspaper journalists, science writers, and science fiction authors. They will help society to accept the advanced concept of molecular cybernetics by providing an objective view of the ongoing project and disseminating information as links between researchers and the general public. In 2023, DNA29 will be held in Sendai (Tohoku University), hosted by the molecular cybernetics project.

4 Summary

This essay describes the history of the development of the research community in Japan, focusing on the evolution of research projects conducted to date. In addition, an example of how researchers can become involved in an emerging field such as DNA nanotechnology is given through the author's personal research history. Currently, the COVID-19 pandemic continues to be unpredictable (January 2022). In Japan, the spread of the sixth wave is now feared, and almost all communication between researchers is restricted to online contact only. Once the pandemic ends, we will again be able to freely hold conferences and more readily collaborate to bring this research into the future.

Professor Seeman passed away during the editing of this manuscript. He first revealed the possibility of molecular robotics based on DNA nanotechnology, and

many researchers in Japan have been directly or indirectly influenced by him. I would like to express my gratitude for his guidance and provide my sincere condolences on his loss.

Acknowledgements The author would like to express profound gratitude to the reviewers for their helpful comments and to Professor Masami Hagiya of the University of Tokyo and Professor Emeritus Akihiko Konagaya of the Tokyo Institute of Technology for providing with various information about the history of research in Japan. This work was supported by a Grant-in-Aid for Transformative Research Areas (A) Molecular Cybernetics, 20H05968.

References

1. M. Hagiya (PI), in *Theory and Construction of Molecular Computers*. JSPS-RFTF. 96I00101; 1996–2001
2. M. Hagiya (PI), in *Molecular Programming, Grant-in-Aid for Scientific Research on Priority Areas*. 2001–2005; 766
3. M. Hagiya (PI), in *Molecular Robotics, Grant-in-Aid for Scientific Research on Innovative Areas*. 2012–2017; 24104001
4. S. Murata (PI), in *Molecular Cybernetics, Grant-in-Aid for Transformative Research Areas*. 2020–2025; 20A403
5. International symposium on distributed autonomous robotics systems (DARS). RAS, IEEE. 2020; 15th DARS in Kyoto (2020)
6. S. Murata, H. Kurokawa, Self-organizing robots. *Star* **77** (2012)
7. M. Rubenstein, A. Cornejo, R. Nagpal, Programmable self-assembly in a thousand-robot swarm. *Science* **345**(6198), 795–799 (2014)
8. K. Fujibayashi, S. Murata, K. Sugawara, M. Yamamura, Self-organizing formation algorithm for active elements. *Forma* **18**(2), 83–95 (2003)
9. E. Winfree, F. Liu, L. Wenzler, N.C. Seeman, Design and self-assembly of two-dimensional DNA crystals. *Nature* **394**, 539–544 (1998)
10. S. Murata, H. Kurokawa, S. Kokaji, Self-assembling machine. *Proc. IEEE ICRA* (1994)
11. K. Fujibayashi, D. Zhang, E. Winfree, S. Murata, Error suppression mechanisms for DNA tile self-assembly and their simulation. *Nat. Comput.* **8**(3), 589–612 (2009)
12. E. Klavins, R. Ghrist, D. Lipsky, A grammatical approach to self-organizing robotic systems. *IEEE Trans. Autom. Control.* **51**(5), 949–962 (2006)
13. Grants-in-Aid for Scientific Research (KAKENHI), JSPS. <https://www.jspis.go.jp/english/e-grants/index.html>
14. L. Adleman, Molecular computation of solutions to combinatorial problems. *Science* **266**, 1021–1024 (1994)
15. M. Hagiya, M. Arita, D. Kiga, K. Sakamoto, S. Yokoyama, Towards parallel evaluation and learning of Boolean μ -formulas with molecules. DIMACS Workshop, DNA Based Comp., 57–72 (1997)
16. K. Sakamoto, H. Gouzu, K. Komiya, D. Kiga, S. Yokoyama, T. Yokomori, M. Hagiya, Molecular computation by DNA hairpin formation. *Science* **288**(5469), 1223–1226 (2000)
17. K. Komiya, K. Sakamoto, A. Kameda, M. Yamamoto, A. Ohuchi, D. Kiga, S. Yokoyama, M. Hagiya, DNA polymerase programmed with a hairpin DNA incorporates a multiple-instruction architecture into molecular computing. *Biosystems* **83**, 18–25 (2006)
18. K. Komiya, M. Yamamura, J.A. Rose, Experimental validation and optimization of signal dependent operation in whiplash PCR. *Nat. Comput.* **9**, 207–218 (2010)
19. M. Hagiya (PI), in *Polymorphic Molecular Interaction for Constructing Huge Memory*, CREST, JST, 2001–2007

20. M. Yamamoto, S. Kashiwamura, A. Ohuchi, M. Furukawa, Large-scale DNA memory based on the nested PCR. *Nat. Comput.* **7**, 335–346 (2008)
21. M. Takinoue, A. Suyama, Hairpin-DNA memory using molecular addressing. *Small* **2**, 1244–1247 (2006)
22. M. Takinoue, D. Kiga, K. Shohda, A. Suyama, Experiments and simulation models of a basic computation element of an autonomous molecular computing system. *Phys. Rev. E.* **78**, 041921 (2008)
23. R. Sekine, M. Yamamura, S. Ayukawa, K. Ishimatu, S. Akama, M. Takinoue, M. Hagiya, D. Kiga, Tunable synthetic phenotypic diversification on Waddington's landscape through autonomous signaling. *Proc. Nat. Acad. Sci. USA* **108**(44), 17969–17973 (2011)
24. K. Fujibayashi, R. Hariadi, S.H. Park, E. Winfree, S. Murata, Toward reliable algorithmic self-assembly of DNA tiles: a fixed-width cellular automaton pattern. *Nano Lett.* **8**(7), 1791–1797 (2008)
25. A. Padirac, T. Fujii, Y. Rondelez, Nucleic acids for the rational design of reaction circuits. *Curr. Opin. Biotechnol.* **24**, 575–580 (2013)
26. S. Murata, A. Konagaya, S. Kobayashi, H. Saito, M. Hagiya, Molecular robotics: a new paradigm for artifacts. *N. Gener. Comput.* **31**(1), 27–45 (2013)
27. K. Lund et al., Molecular robots guided by prescriptive landscapes. *Nature* **465**, 206–210 (2010)
28. Y. Sato, Y. Hiratsuka, I. Kawamata, S. Murata, S.M. Nomura, Micrometer-sized molecular robot changes its shape in response to signal molecules. *Sci. Robot.* **2**, 4 (2017). <https://doi.org/10.1126/scirobotics.aal3735>
29. M. Hagiya, S. Wang, I. Kawamata, S. Murata, T. Isokawa, F. Pepper, K. Imai, in *On DNA-Based Gellular Automata*. Proceedings of the 13th UCNC (Springer, 2014), pp. 177–189
30. T. Hosoya, I. Kawamata, S.M. Nomura, S. Murata, Pattern formation on discrete gel matrix based on DNA computing. *N. Gener. Comput.* **37**(1), 97–111 (2019)
31. E. Osada, Y. Suzuki, K. Hidaka, H. Ohno, H. Sugiyama, M. Endo, H. Saito, Engineering RNA-protein complexes with different shapes for imaging and therapeutic applications. *ACS Nano* **8**(8), 8130–8140 (2014)
32. S. Nakamura, H. Hashimoto, S. Kobayashi, K. Fujimoto, Photochemical acceleration of DNA strand displacement by using ultrafast DNA photo-crosslinking. *ChemBioChem* (2017). <https://doi.org/10.1002/cbic.201700430>
33. J.J. Keya, R. Suzuki, A.M.R. Kabir, D. Inoue, H. Asanuma, K. Sada, H. Hess, A. Kuzuya, A. Kakugo, DNA-assisted swarm control in a biomolecular motor system. *Nat. Commun.* **9**, 453 (2018)
34. D. Kandatsu, K. Cervantes-Salguero, I. Kawamata, S. Hamada, S.M. Nomura, K. Fujimoto, S. Murata, Reversible gel–sol transition of a photo-responsive DNA gel. *ChemBioChem.*, 1118–1121 (2016)
35. Bunshi robotics Gairon (in Japanese), ed. by S. Murata. ebook3_MolRobo_color.pdf (cbi-society.org)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Reminiscences from the Trenches: The Early Years of DNA Nanotech



Thomas H. LaBean

Abstract Research toward the use of nucleic acids as a medium in which to encode non-biological information or as a structural material with which to build novel constructs has now been going on for 40 years. I have been participating in this field for approximately 24 years. I will use my space within this dedicated volume to relate some of my personal experiences and observations throughout my own DNA nanotech journey.

1 Discovering DNA Computing

DNA-based nanotechnology, for me, began in 1994 when Len Adleman's paper "Molecular Computation of Solutions to Combinatorial Problems" came out [1]. I was at a meeting out west, and people were talking about this new *Science* paper at lunches and dinners. It was the buzz subject of social gatherings for the week. Adleman is the "A" in the RSA encryption standard, and here, he was dabbling in actual biochemistry in order to prototype the encoding and processing of non-biological information in DNA molecules for what seemed to be the first time ever. Adleman used a "generate and sort" strategy to solve an NP-complete problem (i.e., the Hamiltonian Path Problem) using a library of DNA molecules as the scratchpad upon which a library of possible solutions was written. Contrary to many popular press reports at the time, Adleman did not solve the Traveling Salesman Problem; he answered the question "is there a Hamiltonian path" through this specific 7-node, directed graph, not "what is the shortest Hamiltonian Path?" His system was designed such that biochemical laboratory steps could be used to sort through the molecules and reject those with incorrect answers recorded upon them (i.e., invalid or non-Hamiltonian paths through the graph). Through Adleman's paper, molecular computing or DNA-based computing was introduced to me and to a broad scientific audience. Of course, by then, Ned Seeman had already conceived of and been toiling

T. H. LaBean (✉)

Materials Science and Engineering Department, North Carolina State University, Raleigh, NC, USA

e-mail: thlabean@ncsu.edu

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*, Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_5

59

within the field of DNA nanotechnology for over a decade; it was just that he toiled mostly alone and outside the eye of a large audience, and certainly outside of my notice, up until then. DNA nanotech entered into my attention and seemingly into the wider public attention through the door of computer science (CS), through molecular computing.

I didn't join the DNA nanotech community until 1998. At the time, I was a postdoc in Jane and Dave Richardson's lab working on *de novo* protein engineering. We were testing our knowledge of the rules of protein folding by trying to design well-folding proteins from scratch. We were deeply involved with the Diversity Biotechnology Consortium (DBC) including Jane and Dave, plus Stu Kauffman (my Ph.D. advisor and pioneer of "self-organization in complex systems"), Mario Geysen (the father of "mimetopes"), Frances Arnold (world-famous protein engineer/evolver who would later be awarded a Nobel Prize), Andy Ellington (who actually coined the term "aptamer"), Pim Stemmer (who invented "sexual PCR"), and many other prominent folks with whom I had the good fortune to interact. The DBC was busy furthering the quite new field of combinatorial chemistry within the biological realm by designing clever selection techniques and applying them to libraries of random-sequence biopolymers (i.e., polypeptides and polynucleotides) in order to winnow the vast populations of possible polymers within sequence space and find/evolve functional, individual sequences for specific purposes. This focus was a direct, logical follow-on from my Ph.D. project in which I randomly searched amino acid sequence space for polypeptide strands that were able to fold into compact, globular 3D structures as do many evolved, biological proteins [2–5].

2 Connections to Broader Scientific Themes

The more general context of my work at the time was that combinatorial chemistry was sweeping the pharmaceutical industry and revolutionizing drug discovery. Then, 1990 was a very big year, in which new tools emerged that allowed combinatorics and *in vitro* evolution to successfully break through into biochemistry and structural biology. The various biopolymer sequence spaces could now be explored and mapped. Specifically, Ellington and Szostak [6] and Tuerk and Gold [7] described the formation of randomized nucleic acid libraries and *in vitro* selection of RNA molecules with specific sequences that provided self-folding and affinity binding to a variety of chemical targets, while Scott and Smith [8] did the same thing for polypeptides via the implementation of phage-display libraries. In the late 80s, we were pursuing similar goals in Stu Kauffman's lab, but our early genetic strategies turned out to be too recombinogenic, and thus stable subclones could not be effectively isolated and propagated.

On a personal front, I was divorced in 1996 and had a clause added to the separation agreement that my sons remain in Durham for the indefinite future, so that my ex-wife could not try to move my sons away from me. While this worked fine for my family life, it put a severe crimp in my professional plans, since I was unable to

perform a national search and pursue a tenure-track, Assistant Professor position with any geographic freedom. Moving into DNA nanotech enabled instead a slow but steady climb up through the Research Professor track at Duke. I was fortunate to finally land a tenure-track faculty job as Associate Professor at North Carolina State University in 2011, be awarded tenure in 2012 and be promoted to full professor in 2017. Professionally (as well as scientifically), DNA nanotech has been a great field to work in, and it has allowed me to follow an unconventional career path in which I have never been an Assistant Professor.

Back in 1998, the invitation to abandon *de novo* and library-based protein design and start designing novel nucleic acid structures came when John Reif walked across Research Drive in the middle of the Duke campus looking for some biochemist to teach him about DNA and molecular biology. Another postdoc in Jane's and Dave's group said there was a crazy, theoretical computer scientist (redundant (?)) looking around for someone to talk to. I taught John a lot of very basic stuff like what 'ligase' was, but he was a quick learner and was offering a decent salary to me if I were to dump Biochemistry and join the forces of Computer Science. I ended up making a deal with the Richardsons, and we maintained our wet lab operations in the Duke Biochemistry Department for a number of years until our DNA team grew too large and started crowding Jane and Dave in their own space.

Shifting gears from protein engineering to nucleic acid engineering was not a difficult move. In fact, a researcher must be comfortable with the manipulation of DNA, through synthetic chemistry, molecular biology, and microbiology procedures in order to hope to do experimental protein design. Moving to DNA structural engineering actually increased the speed of work by eliminating many gene expression and protein purification headaches and other tedious steps in the macromolecular design cycle.

3 Ned Seeman: Founder of the Field

The official start of DNA nanotech (as is reflected in the title of this volume) was actually in 1982. Ned Seeman's paper in the *Journal of Theoretical Biology* [9] challenged people to think about DNA as a structural material instead of as a genetic material. The concept was to assemble periodic matter from DNA for various uses such as guest–host systems for docking protein molecules to reliably form 3D crystals for use in X-ray diffraction studies to solve the atomic-scale structures of the guest proteins. The way Ned frequently explained it in later years was that he was trying to succeed as a crystallographer but was failing to produce high quality protein crystals; therefore, no crystals meant no crystallographer, so he had to try something novel. Probably most people, like me, did not know anything about the 1982 *JTB* paper when it first came out. I was an undergraduate at the time the paper was published, and it would not come to my attention until about sixteen years later. My first introduction to Ned's work was learning about his closed, geometric-like, or topological objects

such as the DNA cube and truncated octahedron, the family of double-crossover tiles, and finally the 2D DNA crystals imaged on mica by atomic force microscopy [10].

In 1998, John Reif organized a team that brought major DARPA funding to the nascent field of DNA nanotech. I signed on at that point. At my first DNA Computing conference, DNA4 in Philadelphia, I got to visit my old stomping grounds at UPENN and to meet many of the characters who were busy developing the discipline of DNA nanotech. Seeman's lab at NYU was, of course, a hot spot that was producing the first rounds of young experimental pioneers in the field including Junghuei Chen, Chengde Mao, and Hao Yan. In those early days, people typically were coming into DNA nanotech through either chemistry/biochemistry/molecular biology or through CS and were joining the community to learn more about the opposite wing of the field. People who came in through the CS door included Grzegorz Rozenberg, Len Adleman, Anne Condon, Richard Lipton, Natasha Jonaska, and Lila Kari. They worked on widely varying topics including word/code design, encoding strategies, simulation or theoretical models of DNA computing strategies for addressing demanding CS problems (NP-hard or NP-complete problems), or simpler problems like molecular implementations of Boolean logic, databasing, cryptography, etc. It was also exciting to meet people like physicists Andrew Turberfield and Bernie Yurke who were fluent in both experimental and theoretical languages. These thinkers expanded the use of DNA in several directions including not only as a structural material but also to function as fuel for these new molecular machines [11].

At the DNA4 meeting in 1998, I first met Erik Winfree, mature for his age and obviously brilliant, he was a graduate student who understood all the scientific background implicitly, acted socially with the grace and ease of a senior academic, and was responsible for establishing the idea of algorithmic assembly at the center of the burgeoning field of DNA nanotech. Erik would soon continue "a family tradition" and receive a MacArthur Prize as did his father, Art Winfree (sidenote: Art was a friend of my Ph.D. advisor Stu Kauffman, so I knew his name and his work long before meeting his son, Erik. Also, I have had the privilege of working directly under two MacArthur fellows, Kauffman, as well as Jane Richardson). The application of algorithmic assembly to DNA computing was a revolutionary concept. Prior to that, the "generate and sort" strategy predominated, in which all or many possible solutions were created within a molecular library, and then, the set was sorted biochemically by discarding incorrect or suboptimal solutions to the problem, similar to what Adleman did in 1994. On the other hand, algorithmic assembly was a molecular implementation of Wang tiling, a theoretical computing model in which colored tile edges specify allowed assembly rules and a small tile set is capable of generating very large, complex, programmed patterns. Algorithmic assembly allowed implementations in which only correct answers would be assembled in the first place. This opened the field to a whole world of new possibilities and pushed creative thinking in diverse directions.

Directly following DNA4, John Reif and I rented a car and drove Ned back to New York where we visited his lab for a couple of days. I enjoyed the distinct privilege of going up to Ned's apartment in NYU faculty housing and seeing exactly what you would expect from a dedicated bachelor/scientist in Manhattan: a room lined with

overflowing bookshelves and a single reading chair in the middle of the room. I was amused to note that the well-worn reading chair had leaked a small pile of crumbled polyurethane foam that rested at the angle of repose right where it had fallen out of the back of the aging chair. It just looked like he didn't waste time with unimportant distractions like decorating or cleaning. My other fun memory of Ned came a couple years later when he was talking at a meeting somewhere in Europe (I think), and he whipped his belt off from around his waist to use it as a visual aid for demonstrating DNA supercoiling. The funny thing was his joke announcement, something like: "Sorry to disappoint you, folks, but that's as far as it goes." There have been so many personal and professional memories surrounding lots of characters in the field of DNA nanotech, and I have very much enjoyed being part of this community.

At DNA4, I was just joining the DNA nanotech club, observing and learning, but by DNA5 in 1999, I was deeply embedded in the field and co-authored three papers at that conference. One of my contributions included the first use of the term "scaffold strand" to indicate a long strand around which other oligos would assemble and generate a structure larger than a standard "tile" [12]. I took Seeman's concept of a "reporter strand," a strand that is ligated together only within the context of the desired, assembled structure, and inverted it by preassembling the long "scaffold" so that it could act as a nucleation element for a larger superstructure. The concept of scaffold strands was further developed in [13] where a "fully addressable" 2D structure was illustrated and proposed. We also proposed another structural variant of a scaffolded assembly that same year [14]. Years later, Paul Rothemund mentioned to me that these early uses of scaffold strands and schematic proposals for scaffolded structures were used against him as pre-existing technology when the US patent office first turned down his patent application describing DNA origami, a big mistake, in my opinion, on the part of the patent examiner.

4 Personal Milestones

Among the milestones of DNA nanotech that I am proudest to have been a part of, I would include our 2000 demonstration of cumulative XOR computation using assembly of TX tiles [15]. This was the first published experimental demonstration of a molecular computation by algorithmic self-assembly using DNA tiles. Winfree and others would create more impressive control and computational scale soon enough [16], but it was fun to be on the cutting edge however briefly. This community has been highly collaborative and cooperative even while also being competitive.

While we were still using labs in the Biochemistry Department, I managed to lure "Ned's best student," Hao Yan to join us at Duke in 2001, and thus ensued a number of highly productive years, often centering around the "golden hands" of my first graduate student, Sung Ha Park. At that time, Sung Ha could get any experiment to work; he calmly solved many complex, sticky experimental problems that other people banged their heads against unsuccessfully. Hao was known around the group as "the finisher" because he could see, plan, and execute the final experiments

necessary to finish a story and complete what was needed for the next publication. Hao then moved on in 2004 to set up his own research group at Arizona State, and somehow that felt like the “early years” of the field were starting to come to a close.

Another major event of 2004 was the publication of William Shih’s DNA octahedron [17]. This was the first time I had ever heard William’s name, but not surprisingly, thereafter, I would never forget it. A few extremely impressive things about the groundbreaking accomplishments of that report include: self-folding (via thermal annealing) of a long molecule with the assistance of a few “helper strands,” production of the nanostructure by DNA polymerase (rather than solid-phase chemical synthesis of short oligonucleotides), and surprisingly detailed cryo-EM structure evaluation. These attributes went against the grain of many long-held tenets and traditions of the field and presaged several revolutions to come.

In 2006, I was again proud to be at the cutting edge of the field by helping to create what was at the time, the “largest” (since we were trying to construct nanostructures of increasing size), fully addressable 2D array; these were 80×80 nm grids with 16 pixels that were “turned off and on” by programmable binding of avidin protein molecules [18]. We compared a couple of hierarchical assembly strategies and employed the tile-lattice method. Our accomplishment and “world record” held for only a couple of months and was soon shattered by the advent of DNA origami.

Paul Rothemund’s description of DNA origami in 2006 changed all the rules of DNA nanotechnology [19]. First of all, a single-author, cover-of-Nature paper was relatively anomalous but not too surprising once you got to know Paul and his body of work which was and is uniformly groundbreaking, intellectually deep, and widely divergent in topics. The design constraints that the field had been laboring under including: (1) Religious re-use of a small number of specific nucleotide sequences at Holliday-junction-like strand exchange points for crossovers (previously everyone only used variants of the J1 junction sequence first worked out by Seeman). Paul showed that essentially any sequence would work. (2) Exact strand stoichiometry matching during assembly reactions was no longer an issue; excess staple strands effectively folded “all” of the available scaffold. (3) Impure oligonucleotides (i.e., standard desalt from IDT instead of laborious, in-house, PAGE purification of each individual oligo) did not affect assembly yield. These changes may not seem like much from today’s perspective, but at the time, they arrived as a major, earth-shattering revolution. It also felt like almost everybody shifted gears and got into the origami game, including quite a few people who came into DNA nanotech on the wave of relative ease with which origami structures could be adapted to different uses, modified, and even designed anew from scratch. However, completely redesigning an origami cost several thousand dollars back then, so the basic cost of a new architectural design increased significantly versus the old repetitive, tile-and-lattice strategy. There also came an immediate echo from China that heralded the presence of another brilliant, self-motivated student when Lulu Qian rapidly and independently designed and assembled an origami in the shape of China [20]. Extension of DNA origami into all types of massive and fantastic 3D shapes has come from William Shih’s

group, especially during the Douglas, Dietz, Liedl, Högberg years, and continued subsequently in their individual labs and those of many others.

5 The End of the Early years

In attempting to restrict this missive to a reasonable length and also to the “early years” of DNA nanotech as stated in the title, I will quickly point to a few more important developments and then conclude. Another key signpost for the field came when single-strand tile or the SST strategy was implemented and first published in 2008 by Peng Yin [21]. Peng had been gone from the Duke group for a while by then, but this work resulted from his time in Durham. We had been working on a project that Sung Ha Park and I referred to as “tile-less lattice” where individual oligos would assemble upon the growing lattice without first forming tile-like units. The most successful design using this concept was finalized by Peng while he was at Caltech and became the programmed tube circumference strategy which subsequently became the 2D [22] and finally 3D SST DNA bricks or DNA Lego architectures which are now a successful subfield of DNA nanotech. Major efforts are now under way to build ever larger self-assembled DNA materials including the incredible 2D multi-origami, hierarchical structures of Lulu Qian’s group and wireframe tessellated 3D structures from several groups. Progress has also been breathtaking on functional DNA hydrogels for soft robotics by Rebecca Schulman and others. Single-stranded RNA origami has been a thing since 2014 [23], and I have recently had the good luck to co-founded a startup called Helixomer, to commercialize an RNA origami-based anticoagulant (with reversal agent) for human health care. There are a number of startup companies now bringing DNA nanotechnology in various forms to the marketplace.

Following its founding by Ned Seeman in 1982, the field of DNA nanotechnology has grown up through the efforts of a large and expanding community of researchers. Ned also founded the International Society for Nanoscale Science, Computation, and Engineering (ISNSCE) (in which I am currently serving as President). The society has sponsored two conferences per year for a long time: DNA-X focuses on computational aspects of DNA nanotech while FNANO draws in a larger community of people studying self-assembly in nucleic acids but also in other molecular systems. I believe that there have already been 26 annual DNA-X meetings (of which, I have been to 11) and 18 FNANO meetings (of which I have been to 17). FNANO has always been held at the Snowbird resort in Utah, partially due to John Reif’s lifelong love of downhill skiing. DNA-X has been held alternately in North America, Europe, North America, Asia (repeat) for quite some time (until the pandemic forced it online in 2020). Traveling to the DNA-X conferences, as well as to work in the labs of collaborators in Denmark and China, has provided me with fantastic opportunities to explore the world. Memories of hanging out with the Italians in Japan, for example, still bring a smile to my face. These travels have led to lifelong friendships.

During this journey through the world with DNA nanotech, I have managed to amuse myself at various times by embedding personal symbols and souvenirs or unnecessary cultural references within the scientific literature. Some examples include a photograph of William S. Burroughs that I put into a cryptography paper, the phrase “sub-Lilliputian holiday” I tucked into a book review, and a molecularized likeness of the Venus de Milo placed in a news-and-views piece. I think people have to play small games like that just to keep things interesting and as minor acts of mischief even if the only one in on the joke is yourself. If someone else had written this type of piece, it would almost certainly highlight a different subset of people and events, but this is the way I remember the unfolding of DNA nanotech. I apologize if anyone is offended or hurt by anything I have written or failed to write in this short memoir-style paper. My intention is simply to record some of my memories for posterity, and I am not trying to cause trouble, attack/insult anyone, or grind any particular axes. When I received the invitation to contribute to this volume, the editors described a colorful array of possible essay types that they were imagining and hoping to include. This sort of personal history and reminiscence was the option that caught my interest. I am hoping to be able to read similar musings from other longtime members of the DNA nanotechnology community because, of course, my idiosyncratic points of view about events and developments through time are necessarily limited. Still, I hope this slim contribution adds to the overall historic document that the editors have bought into the world.

This volume is now imbued with added poignancy since Ned Seeman passed away during the month before the due date of this manuscript. His creativity, irreverence, and intelligence will be sorely missed in our community.

References

1. L.M. Adleman, Molecular computation of solutions to combinatorial problems. *Science* **266**, 1021–1024 (1994). <https://doi.org/10.1126/science.7973651>
2. T.H. LaBean, in *Examination of Folding in Random Sequence Proteins*. Ph.D. Dissertation from Department of Biochemistry and Biophysics at the University of Pennsylvania (1993)
3. T.H. LaBean, S.A. Kauffman, Design of synthetic gene libraries encoding random sequence proteins with desired ensemble characteristics. *Protein Sci.* **2**, 1249–1254 (1993)
4. T.H. LaBean, S.A. Kauffman, T.R. Butt, Libraries of random-sequence polypeptides produced with high yield as carboxy-terminal fusions with ubiquitin. *Mol. Diversity* **1**, 29–38 (1995)
5. T.H. LaBean, T.R. Butt, S.A. Kauffman, E.A. Schultes, Protein folding absent selection. *Genes* **2**, 608–626 (2011). <https://doi.org/10.3990/genes2030608>
6. A. Ellington, J. Szostak, In vitro selection of RNA molecules that bind specific ligands. *Nature* **346**, 818–822 (1990). <https://doi.org/10.1038/346818a0>
7. C. Tuerk, L. Gold, Systematic evolution of ligands by exponential enrichment: RNA ligands to bacteriophage T4 DNA polymerase. *Science* **249**, 505–510 (1990). <https://doi.org/10.1126/science.2200121>
8. J.K. Scott, G.P. Smith, Searching for peptide ligands with an epitope library. *Science* **249**, 386–390 (1990). <https://doi.org/10.1126/science.1696028>
9. N.C. Seeman, Nucleic acid junctions and lattices. *J. Theor. Biol.* **99**, 237–247 (1982). [https://doi.org/10.1016/0022-5193\(82\)90002-9](https://doi.org/10.1016/0022-5193(82)90002-9)

10. E. Winfree, F. Liu, L. Wenzler, N.C. Seeman, Design and self-assembly of two-dimensional DNA crystals. *Nature* **394**, 539–544 (1998). <https://doi.org/10.1038/28998>
11. B. Yurke, A. Turberfield, A. Mills, F.C. Simmel, J.L. Neumann, A DNA-fuelled molecular machine made of DNA. *Nature* **406**, 605–608 (2000). <https://doi.org/10.1038/35020524>
12. T.H. LaBean, E. Winfree, J.H. Reif, Experimental progress in computation by self-assembly of DNA tilings. DIMACS Ser. Discr. Mathe. Theor. Comp. Sci. **54**, 123–140. ed. by E. Winfree, D.K. Gifford, Proceedings of the 5th DIMACS Workshop on DNA Based Computers (MIT, Cambridge, 1999). ISBN 0-8218-2053-2
13. T.H. LaBean, in *Introduction to Self-assembling DNA Nanostructures for Computation and Nanofabrication*, eds. by J.T.L. Wang, C.H. Wu, P.P. Wang, Computational Biology and Genome Informatics, (World Scientific Publishing, Singapore, 2003), pp 35–58. ISBN 981-238-257-7
14. H. Yan, T.H. LaBean, L. Feng, J.H. Reif, Directed nucleation assembly of DNA tile complexes for barcode-patterned lattices. *Proc. Nat. Acad. Sci. USA* **100**, 8103–8108 (2003)
15. C. Mao, T.H. LaBean, J.H. Reif, N.C. Seeman, Logical computation using algorithmic self-assembly of DNA triple-crossover molecules. *Nature* **407**, 493–496 (2000). <https://doi.org/10.1038/35035038>
16. P.W.K. Rothmund, N. Papadakis, E. Winfree, Algorithmic self-assembly of DNA Sierpinski triangles. *PLoS Biol.* **2**(12), e424 (2004). <https://doi.org/10.1371/journal.pbio.0020424>
17. W.M. Shih, J.D. Quispe, G.F. Joyce, A 1.7-kilobase single-stranded DNA that folds into a nanoscale octahedron. *Nature* **427**, 618–621 (2004). <https://doi.org/10.1038/nature02307>
18. S.-H. Park, C. Pistol, S.-J. Ahn, J.H. Reif, A. Lebeck, C. Dwyer, T.H. LaBean, Finite-size, fully-addressable DNA tile lattices formed by hierarchical assembly procedures. *Angew. Chem. Int. Ed.* **45**, 735–739 (2006). <https://doi.org/10.1002/anie.200503797>
19. P.W.K. Rothmund, Folding DNA to create nanoscale shapes and patterns. *Nature* **440**, 297–302 (2006). <https://doi.org/10.1038/nature04586>
20. L. Qian, Y. Wang, Z. Zhang, J. Zhao, D. Pan, Y. Zhang, Q. Liu, C. Fan, J. Hu, L. He, Analogic China map constructed by DNA. *Chin. Sci. Bull.* **51**, 2973–2976 (2006). <https://doi.org/10.1007/s11434-006-2223-9>
21. P. Yin, R. Hariadi, S. Sahu, H.M.T. Choi, S.-H. Park, T.H. LaBean, J.H. Reif, Programming molecular tube circumferences. *Science* **321**, 824–826 (2008). <https://doi.org/10.1126/science.1157312>
22. B. Wei, M. Dai, P. Yin, Complex shapes self-assembled from single-stranded DNA tiles. *Nature* **485**, 623–626 (2012). <https://doi.org/10.1038/nature11075>
23. C. Geary, P.W.K. Rothmund, E.S. Andersen, A single-stranded architecture for cotranscriptional folding of RNA nanostructures. *Science* **345**, 799–804 (2014). <https://doi.org/10.1126/science.1253920>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Chemistry and Physics

Beyond DNA: New Digital Polymers



Grigory Tikhomirov

Abstract From a programming perspective, DNA is stunningly simple: a string of bits coding two types of interactions. The specific *chemical* form of DNA given to us by evolution imposes significant constraints on what is possible with DNA nanotechnology. In this paper, I propose three designs for new digital DNA-like polymers that retain the essential information-bearing properties of DNA while enabling functions not achievable with DNA such as greater stability, programmability, and precision.

A scientist working in the field of DNA nanotechnology can be very successful without knowing how to draw the chemical structure of DNA. Indeed, from a programming perspective, DNA is stunningly simple: a string of bits coding two types of interactions, i.e., a *digital polymer* (Fig. 1, left). But from a chemistry perspective, DNA is exceedingly complex: heterocycles, sugars, and phosphates are organized via a diverse arsenal of chemical bonds—covalent, hydrogen, ionic, and π – π stacking—to give just the right structure enabling robust recognition by many complex biomolecules for reading, copying, evolution, and a multitude of other biological functions (Fig. 1, right).

This specific form of DNA given to us by evolution imposes significant constraints on what is possible with DNA nanotechnology. For example, G-quadruplex formation limits sequences available for strand displacement, and cleavage by nucleases limits *in vivo* utility of DNA-only structures. Nevertheless, DNA remains the most powerful molecule for molecular programming due to the simplicity of its programming rules and its practical advantages including: (i) fast and cheap custom synthesis (e.g., by integrated DNA technologies and Twist Bio), (ii) mature design and modeling tools (e.g., scadnano [1], Peppercorn [2], MagicDNA [3], NUPACK [4], and OxDNA [5]), (iii) ability to amplify copy number (e.g., by PCR), and (iv) fast and cheap sequencing (e.g., by Pacific Bio and Oxford Nanopore). In this paper, I will describe three new digital polymers that can enable functions not achievable with DNA while retaining the essential information-bearing properties and some practical advantages of DNA.

G. Tikhomirov (✉)

Electrical Engineering and Computer Sciences, University of California, Berkeley, USA
e-mail: gt3@berkeley.edu

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*, Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_6

71

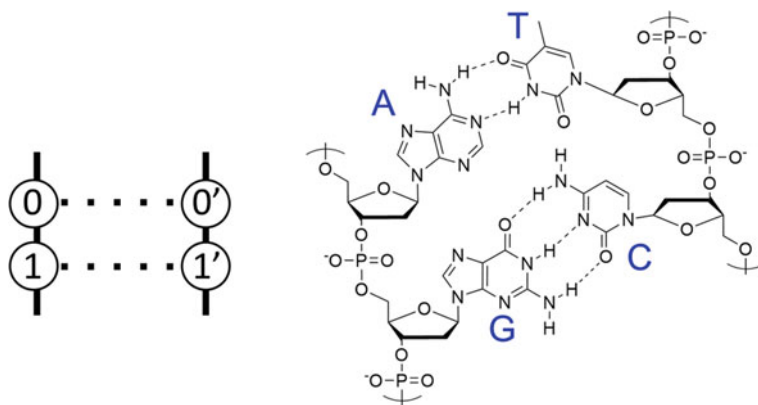


Fig. 1 DNA from the perspective of a molecular programmer (left) and a chemist (right)

An astute reader may be wondering, “this is a very ambitious proposal. People have worked on DNA alternatives for many years. What about all the existing polymers such as PNA, GNA, and other XNAs?” Yes, many people have worked on DNA alternatives for decades exploring questions ranging from pure curiosity about molecular origins of life [6] to very practical goals of improving microarray sensors [7]. Typically, these alternatives have been focused on creating molecules possessing a specific attribute of DNA, most notably (a) transcription and translation by enzymes [8], (b) ability to recognize DNA sequences while being more stable [9], and (c) information storage [10]. These alternatives include GNA (a DNA analog in which sugar is replaced with a more stable glycerol [11]), phosphorothioate DNA (a DNA analog in which oxygen in phosphate is replaced with sulfur [12]), PNA (a DNA analog in which sugar is replaced with a neutral peptide-like backbone [13]), and L-DNA (a DNA analog with opposite chirality to DNA [14]). It may seem surprising, but no one, to my knowledge, has tried to redesign DNA to enlarge its functionality for the purposes of DNA computing or DNA nanotechnology. As the new digital polymer designs below will demonstrate, without the constraint to bind natural DNA or be processed by natural enzymes, we have much more freedom in molecular design.

Before describing the new polymer designs, it is instructive to first consider what desirable functions of a digital polymer DNA currently lacks, and how these might be achievable by existing approaches without building an entirely new polymer. Significant limitations of DNA, summarized in Table 1, include low stability, limited chemical functionality with 4-word alphabet, and lack of bio-orthogonality. Generally, to overcome these limitations, chemical modifications or non-DNA coating are used. However, these modifications are frequently hard to implement for non-chemists and often disrupt the desired structure of DNA.

While combining all the above functions in one polymer is neither feasible nor necessary, I propose three new polymers below that draw from this pool of functions to create DNA analogs that better serve specific tasks. For each polymer, I will specifically discuss redesign of the recognition elements and backbone, the new

Table 1 Desired properties/abilities and approach to achieve them with DNA-based structures

Property	New abilities (examples)	How to do it with DNA (examples)	Limitations of doing it with DNA (examples)
Improved stability in vivo	Wider biomedical applications	Ligation, increasing cross-overs, chemical modification, coatings [15]	Coatings reduce surface access, chemical modifications can be costly, and disrupt dsDNA structure
Improved thermal stability	Wider applications in hot environments	T-T welding [16], ligation [17]	Can distort the structure
Solubility in non-polar solvents, such as hexane	Compatibility with industrial processes such as spin casting	Attaching alkyl chains [18]	Limited to making a small number of modifications
More chemical variety	More compact and stronger aptamers	Biological and chemical synthesis [19]	Modifications can be costly and disrupt dsDNA structure
More than four letters (ACGT)	Increased information density	Biological and chemical synthesis [8]	Modifications can be costly
Higher density breadboard	Better control over electronic and biological coupling	Use square lattice origami (~2.5 nm spacing) [20]	Need extremely closely-packed DNA
Response to stimuli (pH, light, RedOx)	External control of binding	Hoogsteen pairing designs, chemical modifications [21]	Modifications can be costly and disrupt dsDNA structure
Wider range of hybridization energies	Stronger than G-C pairs can reduce leak by minimizing fraying/breathing	Currently cannot be done	N/A
Controlled helical pitch	Low pitch → less knots	Currently cannot be done	N/A
Fluorescent base options	Biophysics studies, in vivo tracking	Chemical synthesis [22]	Modifications can be costly and disrupt dsDNA structure
Bio-orthogonality	Minimize interference with physiology	L-DNA [23]	Modifications can be costly and disrupt dsDNA structure
Conductivity or semi-conductivity	Nanoelectronics	Regular DNA is a poor conductor [24], but it can be coated with metals [25] or semiconducting polymers [26]	Poor performance and uniformity of coated nanostructures

(continued)

Table 1 (continued)

Property	New abilities (examples)	How to do it with DNA (examples)	Limitations of doing it with DNA (examples)
Flexibility control	Nanomechanical devices	Adding defects or flexible linkers in multi-strand structures [27]	Limited control for ssDNA

functions this enables, and synthesis. The first polymer, NP1, marginally redesigns DNA bases while introducing a simpler and more modular α -peptide backbone. This should expand its chemical vocabulary and programmability, among other advantages. A second, more ambitious polymer design, NP2, proposes a recognition code based entirely on natural peptides. This should also increase chemical functionality while enabling scalable production *in vivo*. A third polymer, NP3, aims to make a covalent recognition code for applications requiring very stable architectures. This should yield polymers with ultimate stability, e.g., for making portable devices. I hope that the ideas explored in these three designs can lead to an extended toolkit for molecular programming that will enable nanotechnologies currently impractical with DNA.

1 New Polymer 1 (NP1)

The design of NP1 builds on the successes of DNA, proteins, and peptide nucleic acid (PNA) while adding new functions unattainable by any single class of these molecules such as higher stability and larger design space. First, I propose to “fix” DNA bases for a slimmer and stronger code. Then, I explain why an α -peptide backbone is better than any other backbone for a digital polymer, especially when it comes to scalable production of the new polymer. Finally, I provide an overview of related existing approaches and discuss ways to overcome potential pitfalls.

Recognition elements. The recognition between two strands of NP1 is based on unmodified DNA bases (T and C) and modified DNA bases (A and G). First, to enable denser molecular pegboards, I propose to make the double helix slimmer by replacing purine bases with pyrimidine, effectively chopping off the 5-membered ring (Fig. 2, blue fragments; compare to **DNA**). One concern is that this “slimming surgery” may disrupt the balance between hydrophobic and hydrophilic interactions that has been finely tuned during molecular evolution, thus destabilizing the helix. Another concern is that this design goes against the size complementarity hypothesis, which states that large purines must pair with small pyrimidines. In fact, however, “skinny” DNA where purines are replaced with pyrimidines preserves its structure and even gains stability [28]. Another advantage of removing the 5-membered heterocycle is that this eliminates guanine’s ability to form quadruplexes, a big nuisance for some

DNA applications such as information retrieval in G-rich DNA sequences for data storage.

To improve the stability of A-T pairs, I propose to add an amino group to the modified adenine (Fig. 2, red). This converts an asymmetric double hydrogen bond to a symmetric triple hydrogen bond, which has been shown to increase duplex stability while precluding stray recognition by some biomolecules normally binding to the minor groove [8]. To preserve hydrogen-bonding motifs, I propose to move the nitrogens in the new pyrimidines away from the ring location linked to the backbone. The reason I retain the nitrogens is that their electron withdrawing effect is important to maintain the same energy of lowest unoccupied molecular orbital (LUMO) localized on the lone pair of the hydrogen-bonding amino group in position 2 of the new

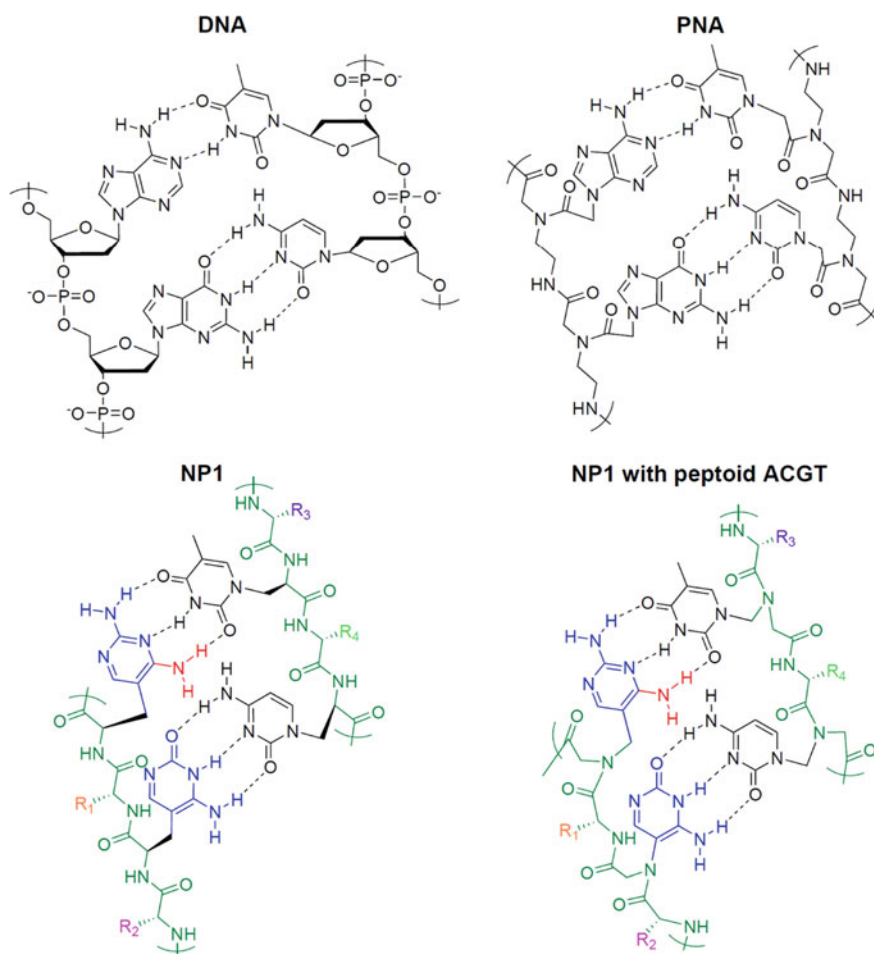


Fig. 2 Structures of DNA, PNA, new polymer 1 (NP1), and NP1 with peptoid ACGT analogs

pyrimidine analog of adenine. One can also achieve a similar electron withdrawing effect by introducing a nitro group in position 6 of the ring [28]. Another reason to move nitrogen to the new position 3 is that it is easier to synthesize such a base compared to one with carbon.

Backbone. Unlike PNA, NP1 organizes hydrogen-bonding elements on a simpler yet more versatile natural α -amino acid backbone. This can potentially allow adaptation of not only existing chemical but also biological peptide syntheses to produce NP1. Adding a canonical amino acid after every recognition bearing non-canonical amino acid (Fig. 2, colorful Rs) also allows a great expansion of the chemical functionality available to NP1 compared to DNA or PNA. A larger chemical space afforded by more diverse and numerous amino acid residues compared to nucleotides is likely one of the main reasons why Nature has eventually chosen peptides and not polynucleotides to build most molecular machines.

To keep the initial designs as close to DNA as possible, I propose to place hydrogen-bonding bases on every second amino acid, which should give a base spacing very close to the ones in DNA and PNA (PNA forms a double helix with 18 bases per period versus 10.5 for DNA [29].) Unlike PNA, NP1 is designed to be chiral by adding chiral (L-) ACGT amino acids and natural L-amino acids. As a reminder, PNA lacks chirality due to the rapid inversion of the tertiary prochiral nitrogens. This inversion leads to racemic mixtures of left- and right-handed helices, as well as promiscuity in C \rightarrow N polarity of the peptide backbone, an equivalent of 5' \rightarrow 3' polarity of the sugar backbone that enforces the antiparallel requirement for DNA helices. While it is possible to add chirality to the PNA backbone by synthesizing γ -modified PNA, NP1 is more modular compared to such a γ -modified PNA, because NP1 requires only 24 monomers (four ACGT analogs and 20 α -amino acids) to enable more design freedom than possible with 80 γ -modified PNA monomers (4 \times 20). Additionally, while PNA's neutrality makes it challenging to deliver in vivo, NP1's hydrophilicity may be tuned by incorporating a variety of hydrophilic and hydrophobic amino acid residues enabling applications in media with a range of lipophilicities from water to hexane. However, care should be taken to avoid combinations which are unsuitable, including large regions of highly polar side chains or unbroken polar-non-polar alternating regions to avoid amphiphilic or sheet-forming behavior.

Looking at DNA through the eyes of a chemist also presents a tantalizing opportunity to explore the design space of two-stranded helical molecules. For example, in NP1, nucleobase analogs can be integrated more sparsely (with more natural amino acids inserted between the ACGT analogs compared to just a single amino acid in Fig. 2), allowing natural biologically-active peptide fragments to be organized with a precision and complexity infeasible using current protein engineering approaches [30]. Also, by varying the number and the type (α , β , γ , δ) of amino acids, it should be possible to control the helical pitch, i.e., the angle between two adjacent stacks, which is 34.3° for DNA and 20° for PNA. In an extreme case, it may be possible to reduce the rotation per base pair to 0° by directly linking two nucleobase analogs (skipping a canonical amino acid). This is because the distance between two adjacent nucleobase analogs attached to a fully stretched α -amino acid peptide backbone is

3.5 Å, which means that the scaffold almost does not need to twist for the bases to reach the typical stacking distance of 3.4 Å.

Synthesis. To realize its practical potential, a new digital polymer should be amenable to automated synthesis, allowing efficient production of custom sequences as can be done now for DNA, peptides, and oligosaccharides [31]. Initial efforts may most profitably be dedicated to the synthesis of the four amino acids nucleobase analogs that are compatible with the standard automated peptide or peptoid syntheses [32]. Next, the effect of sequence and other design elements on the structure of the double-stranded helix and simple multi-strand constructs using nuclear magnetic resonance (NMR), X-ray crystallography, and atomic force microscopy (AFM) should be investigated.

Once optimal structures are elucidated, efforts should be focused on developing larger-scale sustainable synthesis. One approach can be to adapt natural cellular protein synthesis machinery to produce large amounts of NP1; this will leverage the last two decades of rapid progress in techniques for co-translational incorporation of unnatural amino acids into proteins produced in cells via genetic code expansion [33]. Finally, directed evolution may be used to modify components of the translation machinery such as synthetase/tRNA pairs to selectively incorporate the four new amino acids with A, C, G, T analog residues first in cell-free media and then in cells [34].

Existing approaches to design a simple non-covalent recognition code. Much work has been done to create digital polymers with recognition between strands based on hydrogen bonds and other non-covalent interactions [35–38]. However, a DNA-like polymer with robust recognition of DNA has not yet been reported, partially because creating such a digital polymer for the purposes of DNA nanotechnology was not the main reason for these efforts.

Potential pitfalls and ways to overcome them. Experts on protein structure may have a valid concern that instead of the desired double helix, NP1 may form α -helix, β -sheet, or other motifs common in proteins due to the presence of carbonyls and NH groups. I expect that the three locally organized hydrogen bonds in the base pairs will be more favorable than the two relatively remote hydrogen bonds of α -helices or β -sheets. Also, proline, an amino acid that lacks NH, can be sparsely added to break up α -helices or β -sheets. Yet another way to minimize spurious hydrogen bonding is to use peptoid versions of ACGT where the base is attached to the nitrogen (Fig. 2). As a note, PNA has the capacity to form spurious $\text{NH}\cdots\text{OC}$ hydrogen bonds (twice as few as α -peptides but the same number as the peptoid version of NP1) but still prefers to form a double helix. However, I propose to keep regular α -amino acid monomers for non-ACGT monomers instead of fully switching to peptoids to retain the desired chirality and higher chance for in vivo synthesis. In planning in vivo applications, one should keep in mind that while gaining stability to nucleases NP1 will become susceptible to proteases. In addition, one should consider that NP1 may bind DNA especially in a single stranded form due to the similarity of its recognition elements to DNA bases.

2 New Polymer 2 (NP2)

Design. NP1 requires developing synthesis and incorporation of synthetic DNA base analogs. Is it possible to avoid this and design a new DNA-like polymer entirely from natural amino acids? The design of NP2 also relies on the natural α -amino acid peptide backbone. However, instead of familiar ACGT analogs, this design seeks to construct recognition elements from natural amino acid side chains (Fig. 3). Hydrophobic interactions such as V–L or F–W or single hydrogen bonds in pairs such as S–T or C–Y are too weak and not specific to provide a reliable code. One excellent candidate is R–D pair that is maintained by two hydrogen bonds and electrostatic attraction. This interaction is likely even stronger than the G–C hydrogen bond. In principle, it is possible to construct a digital polymer using just a single *heterophilic interaction* (i.e., binding partners are different) such as G–C or R–D, but this would substantially increase possible spurious interactions and decrease information density for the same length compared to a polymer with two heterophilic interactions such as DNA. Therefore, it is desirable to introduce another specific interaction that is as orthogonal to R–D as possible. Since the only other available heterophilic interaction—K–E—also involves a carboxylic acid (E, on a slightly longer leash than D) and will likely bind to R, I suggest using a *homophilic interaction* (i.e., binding partners are the same) such as N–N or Q–Q or Q–N. This interaction is also maintained by two hydrogen bonds, but the partners are not charged as in the R–D pair. The lack of charges should bring HOMO and LUMO of binding partners closer, making this interaction more orthogonal to R–D. Among the three possible interactions (N–N or Q–Q or Q–N), I show N–N in Fig. 3, because according to preliminary molecular dynamics calculations performed in my group, N–N gives more stable double-stranded complexes compared to Q–Q, Q–N, and N–Q. R, D, and G essentially comprise a three-letter code which will likely have more spurious interactions than a four-letter code (A, T, G, C). My group is currently studying a series of complexes such as shown in Fig. 3 (R = G and L) by NMR, AFM, and X-ray diffraction. P is inserted to disrupt potential α -helices or β -sheets as discussed for NP1.

Existing approaches to design a simple recognition code with peptides. The idea that it is possible to construct a recognition code similar in simplicity to A–T and G–C of DNA with natural amino acids may seem preposterous to some. “If there had been such a simple peptide motif, Nature (not the journal) would surely have found it by now. If not Nature, then David Baker”—an active reader may think. And the reader would be partially right: Some recognition elements have indeed been identified both by stochastic natural evolution and more rational protein design. However, these motifs are still not as simple, programmable, and practical as A–T and G–C of DNA, as I explain next.

Some obvious hydrogen-bonding recognition analogs of A–T and G–C mentioned above are R–D, R–E, K–D, K–E, N–N, Q–Q, and Q–N with their two hydrogen bonds. Protein database (PDB) search indeed reveals quite frequent intra- and interpeptide contacts with shorter than Van-der-Waals radii distances for these pairs indicating

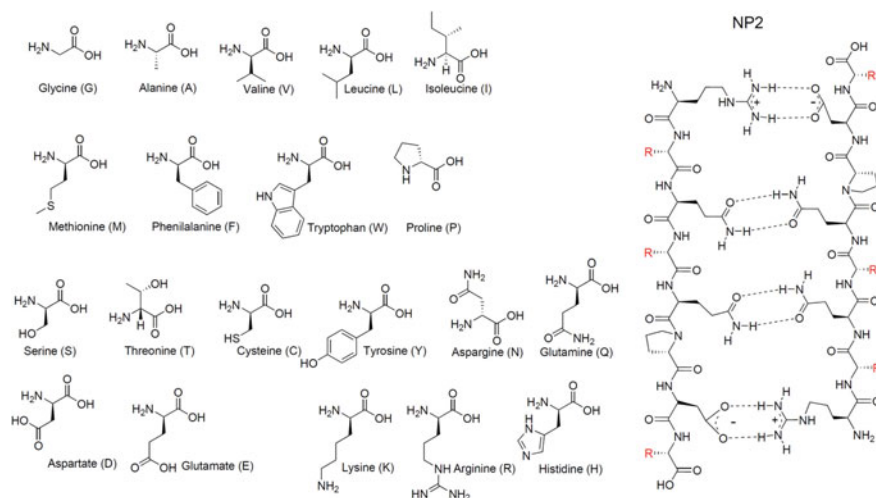


Fig. 3 Structures of 20 essential amino acids (left) and an example structure of New Polymer 2 duplex (right) made with a self-complementary strand. Red Rs can potentially be various side chains from the amino acids on the left

bonding. So, Nature does use them, but it does so very rarely compared to NH-CO hydrogen bonds and hydrophobic residue interactions that dominate protein folding landscapes. Learning from the wealth of protein structures in PDB, protein designers have made huge progress in predicting unknown structures (e.g., with Rosetta [39] and AlphaFold [40]) and designing new ones [41]. However, even the simplest recognition codes elucidated so far are still not as compact and modular as A–T and G–C of DNA, limiting their practical use [42].

The great wealth of natural PDB structures combined with machine learning algorithms is very powerful, but unlikely to uncover a conceptually new code. The current protein world is biased to represent structures achievable by evolution due to its evolutionary pathway to complexity. *To overcome this bias and build a new, potentially better protein world we need to develop design principles that are not based on reinforcing the current set of “rules.”* If we are successful in elucidating these new more rational design principles, we will potentially be able to build a new protein world and even new forms of life that can live longer and healthier than us. After all, we humans possess the power of systematic rational design as well as tools of directed and accelerated evolution [43–46]. But let us start by designing a new digital polymer.

Potential pitfalls and ways to overcome them. Even though the recognition elements in NP2 are placed with regular spacing (every 6 sigma bonds) on the peptide scaffold, the resulting minimum energy conformation of a double-stranded complex may not be as periodic as DNA double helix where a regular 3.4 Å spacing is ensured by π – π stacking. This potential lack of uniform periodicity, also known as Schrodinger’s aperiodic crystal requirement for information-bearing polymers

(Schrodinger postulated that this requirement is needed for recognition by enzymes), is not necessarily a problem for many applications such as strand displacement. Strand displacement has already been demonstrated with α -helix based peptides and would benefit significantly from the acceleration that is expected for a more flexible single peptide chain compared to a rigid α -helix structure [47].

To address the recurrent concern about α -helix or β -sheet formation, I expect the hydrogen bonds of the code (R–D and G–G) to be more favorable than NH–OC hydrogen bonds as in NP1. As with NP1, this is expected because more compact interactions are entropically more favorable than the spread-out NH–OC interactions.

When planning synthesis, one should keep in mind that even at high coupling efficiency, solid-state reaction yields drop off rapidly, limiting the length of peptide or peptoid strands at sub-50-mer lengths (99% efficiency per-step gives less than 80% yield after 25 steps [48]). The high coupling efficiencies may be lower due to different monomers. However, the hybridization energy per base will likely be higher compared to DNA due to the absence of negative charges on single strands. This may enable shorter strands of NP2 be sufficient for building nanostructures and behaviors analogous to ones in structural and dynamic DNA nanotechnology.

Future outlook. A digital polymer made of only natural amino acids could be synthesized *in vivo*, enabling a multitude of biomedical applications for molecular programming—imagine being able to shape proteins into structures similar to DNA or RNA origami via co-translational folding, or being able to attach DNA-like strands in precise locations on protein surfaces to enable complex protein-based chemical reaction networks (CRNs) *in vivo*. Furthermore, biotechnological production of proteins is generally more scalable than that of nucleic acids. Perhaps one day we will grow designer nanoscale machines built with principles of DNA programming with the ease and cost of a cheese factory.

3 New Polymer 3 (NP3)

Design. DNA, NP1, and NP2 use hydrogen bonds in recognition elements. This puts upper bounds on the thermal stability of architectures designed with these polymers. NP3 aims to make a covalent recognition code for applications requiring very stable architectures. Is it possible to use stronger covalent bonds instead of weaker non-covalent ones to encode recognition between two strands of a digital polymer? Nature evolved to rely on weaker non-covalent interactions in many biological digital polymers, presumably to enable many dynamic behaviors at physiological temperatures. For example, the base pairing energy in DNA (hydrogen bonding + π – π stacking) is typically in 0.4–20.0 kJ/mol range; and more than 99.999% of proteins have free energy of folding below 33.5 kJ/mol [41]. Covalent bonds are an order of magnitude stronger, for example: N–H (386 kJ/mol), C–N (305 kJ/mol), C = C (602 kJ/mol), and C = N (615 kJ/mol). The main concern one may imagine with using covalent bonds for constructing a recognition code is that it would not allow error correction, i.e., a single incorrect bond will be so strong that it would prevent the system from

reaching the desired global thermodynamic minimum. The strongest non-covalent DNA bonds are labile above 90 °C allowing for the global minima to be reached by slow annealing. Thermal annealing of covalently bound structures will likely destroy the constituent molecules. However, some covalent bonds can be made labile under mild conditions at room temperature, for example, in the presence of catalysts that lower their transformation activation energy. These bonds are known as “dynamic covalent.” I propose to use dynamic covalent bonds to construct NP3 (Fig. 4).

Like NP2, NP3 uses only natural amino acids and relies on a 3-letter code constructed from heterophilic (S-D) and homophilic (C-C) interactions. Coincidentally, both ester and disulfide bonds can be rendered labile under the same conditions—low pH (ester) and reducing environment (S-S bond)—which is not the case for many other pairs of dynamic covalent bonds that may require incompatible conditions to become labile simultaneously [49].

Existing approaches to design a simple recognition code with dynamic covalent bonds. Dynamic covalent bonds have been explored extensively for the purposes of drug discovery, material design, and other goals [50]. Substantial efforts have been directed toward creating two-letter [51, 52] and even four-letter [53] heterophilic codes. The main challenge has been overcoming kinetically-trapped species with some recent successes for two-letter systems [54].

Potential pitfalls and ways to overcome them. In case the example shown in Fig. 4 does not form, there are many degrees of design freedom to explore. For example, artificial amino acids such as catechol and boronic acid can be incorporated [52]. Artificial amino acids would also allow positioning groups that form dynamic

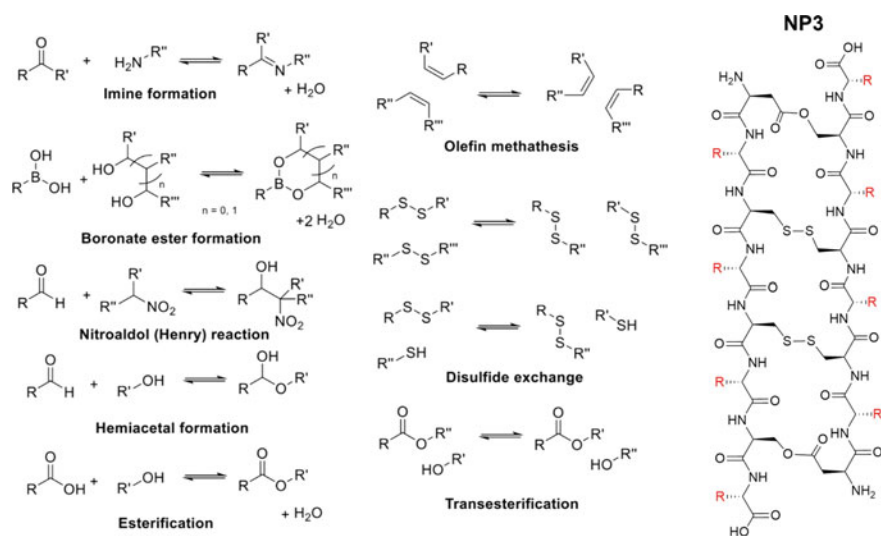


Fig. 4 Examples of dynamic covalent bonds (left and middle) and an example structure of New Polymer 3 duplex (right) made with ester and disulfide bonds from a self-complementary strand. Red Rs can potentially be any side chain

covalent bonds on aromatic residues such as benzene, which can enable stacking bonds for a more regular periodicity of duplex via π - π stacking like in DNA. When planning synthesis, consideration should be made for how non-canonical amino acids for the recognition elements of NP2 might be efficiently protected and deprotected in conventional solid-state peptide and peptoid syntheses. If protecting these groups is restrictive, post-polymerization reactions (i.e., thiolene click chemistry) to append desired recognition moieties can be explored. If one plans to use NP3 for in vivo applications, they should carefully plan the fate of the assemblies given potential swings in pH (e.g., acidic lysosomes), redox potential, etc.

4 Example Applications

The new digital polymers described above will enable many new capabilities unattainable with DNA. Some of the capabilities were summarized in Table 1. Below are a few more examples.

Activatable strand displacement. Overcoming leak reactions in strand displacement circuits is a significant challenge for dynamic DNA nanotechnology. One way to reduce or eliminate leak is to introduce a single covalent base pair such as cysteine-cysteine (C-C) from NP3 to a branch migration domain of a double-stranded complex based on non-covalent bonds (NP1 or NP2). The displacement will proceed only if the one special base is activated, e.g., with a reducing agent. The activation can be designed to be controlled by light [55] and other stimuli, expanding the complexity and versatility of dynamic behaviors that can be implemented with molecular programming.

Bio-orthogonality for in vivo molecular programming. As mentioned above, most nucleic acid analogs such as PNA were designed to bind nucleic acids. This limits their applications in vivo due to their interference with endogenous chemistry. NP2 and NP3 can be used to build a system for molecular programming in vivo that is orthogonal to existing nucleic acids.

Enzymeless translation and self-replication. The covalent code of NP3 makes binding by a single monomer stable. Simply by mixing a single template strand with free monomers, a new complementary strand can be templated. A ligation mechanism can be introduced (e.g., via terminal alkene methathesis) that would zip the new templated strand. After this, the complex can be “melted” via labilization of the covalent bonds, and the process repeated. Depending on whether the new monomers are the same as or different from monomers in the templating strand, this process can be conceptually viewed as replication or translation. Neither process would require enzymes, unlike their biological counterparts. Enzyme-free information transfer in digital polymers has been an focus of experimental efforts for many [56]. Enzyme-free protein replication (protein PCR) has been but the wild dream of few.

Molecular breadboard with 100 × pattern density of DNA origami. Using a version of NP1, it is also possible to design a fully addressable breadboard with

average functionalization spacing of 0.6 nm (compare to current ~ 6 nm boards). I leave out the details of this design to stimulate the imagination of the reader.

5 Conclusions

Digital technologies are becoming essential to our existence. The digital paradigm enables construction of extremely complex hardware and software from simple building blocks. The original digital code can be found in life. The digital nature of naturally-evolved polymers such as DNA and proteins is essential to the existence and proliferation of life, by allowing robust information flow through precise encoding of genetic/structural information in sequences of base pairs/amino acids. While molecular programming has so far been dominated by work with DNA, chemists have been developing artificial polymers for many decades, though only a few of these constitute digital polymers, in which the sequence is precisely controlled down to single monomers. The few synthetic digital polymers that do exist have been shown to be superior to their less precise non-digital counterparts for applications in materials [57], molecular electronics [58], tracking [10], and molecular programming, because they allow precise control through their sequence of melting point, bandgap energy, compact mass-spec signature, and sequence-dependent recognition, respectively. Yet, despite this progress, these non-biological digital polymers are still largely underutilized.

DNA will likely remain the molecule of choice for molecular programming for the foreseeable future due to its current practical advantages. But I hope that the ideas outlined here will encourage exploration of new digital polymers with expanded capabilities—both experimentally by chemists and theoretically by molecular programmers. Nature initially used RNA and DNA as the substrate for life during the “RNA world” before evolving construction based on proteins, a paradigm shift that enabled much more diverse forms and functions. Similarly, the approaches outlined here could lead to more diverse nanotechnologies built with molecular programming.

Acknowledgements I am grateful to Prof. Helen Tran, her lab, and other anonymous reviewers for their helpful critiques on the manuscript.

References

1. D. Doty, B.L. Lee, T.S. Stérin, in *A Browser-Based, Scriptable Tool for Designing DNA Nanostructures*. eds. C. Geary, M.J. Patitz, 26th International Conference on DNA Computing and Molecular Programming (DNA 26) [Internet]. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum für Informatik; 2020 [cited 2021 Nov 22]. pp. 9:1–9:17. (Leibniz International Proceedings in Informatics (LIPIcs); vol. 174). Available from: <https://drops.dagstuhl.de/opus/volltexte/2020/12962>

2. S. Badelt, C. Grun, K.V. Sarma, B. Wolfe, S.W. Shin, E. Winfree, A domain-level DNA strand displacement reaction enumerator allowing arbitrary non-pseudoknotted secondary structures. *J. R. Soc. Interface* **17**(167), 20190866 (2020)
3. C.-M. Huang, A. Kucinic, J.A. Johnson, H.-J. Su, C.E. Castro, Integrated computer-aided engineering and design for DNA assemblies. *Nat. Mater.* **20**(9), 1264–1271 (2021)
4. J.N. Zadeh, C.D. Steenberg, J.S. Bois, B.R. Wolfe, M.B. Pierce, A.R. Khan, R.M. Dirks, N.A. Pierce, NUPACK: analysis and design of nucleic acid systems. *J. Comput. Chem.* **32**(1), 170–173 (2011)
5. E. Poppleton, R. Romero, A. Mallya, L. Rovigatti, P. Šulc, OxDNA.org: a public webserver for coarse-grained simulations of DNA and RNA nanostructures. *Nucleic Acids Res.* **49**(W1), W491–8 (2021)
6. S.A. Benner, D. Hutter, Phosphates, DNA, and the search for nonterrean life: a second generation model for genetic molecules. *Bioorg. Chem.* **30**(1), 62–80 (2002)
7. N.C. Hauser, R. Martinez, A. Jacob, S. Rupp, J.D. Hoheisel, S. Matysiak, Utilising the left-helical conformation of L-DNA for analysing different marker types on a single universal microarray platform. *Nucleic Acids Res.* **34**(18), 5101–5111 (2006)
8. S. Hoshika, N.A. Leal, M.-J. Kim, M.-S. Kim, N.B. Karalkar, H.-J. Kim, A.M. Bates, N.E. Watkins, H.A. SantaLucia, A.J. Meyer, S. DasGupta, J.A. Piccirilli, A.D. Ellington, J. SantaLucia, M.M. Georgiadis, S.A. Benner, Hachimoji DNA and RNA: a genetic system with eight building blocks. *Science* **363**(6429), 884–887 (2019)
9. K. Duffy, S. Arangundy-Franklin, P. Holliger, Modified nucleic acids: replication, evolution, and next-generation therapeutics. *BMC Biol.* **18**(1), 112 (2020)
10. L. Charles, J.-F. Lutz, Design of abiological digital poly(phosphodiester)s. *Acc. Chem. Res.* **54**(7), 1791–1800 (2021)
11. M.K. Schlegel, L.-O. Essen, E. Meggers, Duplex structure of a minimal nucleic acid. *J. Am. Chem. Soc.* **130**(26), 8158–8159 (2008)
12. W.J. Taylor, J. Ott, F. Eckstein, The rapid generation of oligonucleotide-directed mutations at high frequency using phosphorothioate-modified DNA. *Nucleic Acids Res.* **13**(24), 8765–8785 (1985)
13. M. Egholm, O. Buchardt, L. Christensen, C. Behrens, S.M. Freier, D.A. Driver, R.H. Berg, S.K. Kim, B. Norden, P.E. Nielsen, PNA hybridizes to complementary oligonucleotides obeying the Watson-Crick hydrogen-bonding rules. *Nature* **365**(6446), 566–568 (1993)
14. W. Zhong, J.T. Sczepanski, Direct comparison of d-DNA and l-DNA Strand-displacement reactions in living mammalian cells. *ACS Synth. Biol.* **10**(1), 209–212 (2021)
15. A.R. Chandrasekaran, Nuclease resistance of DNA nanostructures. *Nat. Rev. Chem.* **5**(4), 225–239 (2021)
16. T. Gerling, M. Kube, B. Kick, H. Dietz, Sequence-programmable covalent bonding of designed DNA assemblies. *Sci. Adv.* **4**(8), eaau1157 (2018)
17. P. O’Neill, P.W.K. Rothmund, A. Kumar, D.K. Fygenson, Sturdier DNA nanotubes via ligation. *Nano Lett.* **6**(7), 1379–1383 (2006)
18. S. Arangundy-Franklin, A.I. Taylor, B.T. Porebski, V. Genna, S. Peak-Chew, A. Vaisman, R. Woodgate, M. Orozco, P. Holliger, A synthetic genetic polymer with an uncharged backbone chemistry based on alkyl phosphonate nucleic acids. *Nat Chem.* **11**(6), 533–542 (2019)
19. M. Kimoto, R. Yamashige, K. Matsunaga, S. Yokoyama, I. Hirao, Generation of high-affinity DNA aptamers using an expanded genetic alphabet. *Nat. Biotechnol.* **31**(5), 453–457 (2013)
20. Y. Ke, S.M. Douglas, M. Liu, J. Sharma, A. Cheng, A. Leung, Y. Liu, W.M. Shih, H. Yan, Multilayer DNA origami packed on a square lattice. *J. Am. Chem. Soc.* **131**(43), 15903–15908 (2009)
21. H. Asanuma, X. Liang, H. Nishioka, D. Matsunaga, M. Liu, M. Komiyama, Synthesis of azobenzene-tethered DNA for reversible photo-regulation of DNA functions: hybridization and transcription. *Nat. Protoc.* **2**(1), 203–212 (2007)
22. W. Xu, K.M. Chan, E.T. Kool, Fluorescent nucleobases as tools for studying DNA and RNA. *Nat. Chem.* **9**(11), 1043–1055 (2017)

23. G. Ke, C. Wang, Y. Ge, N. Zheng, Z. Zhu, C.J. Yang, l-DNA molecular beacon: a safe, stable, and accurate intracellular nano-thermometer for temperature sensing in living cells. *J. Am. Chem. Soc.* **134**(46), 18908–18911 (2012)
24. X. Guo, A.A. Gorodetsky, J. Hone, J.K. Barton, C. Nuckolls, Conductivity of a single DNA duplex bridging a carbon nanotube gap. *Nat. Nanotech.* **3**(3), 163–167 (2008)
25. E. Braun, Y. Eichen, U. Sivan, G. Ben-Yoseph, DNA-templated assembly and electrode attachment of a conducting silver wire. *Nature* **391**(6669), 775–778 (1998)
26. J.B. Knudsen, L. Liu, A.L. Bank Kodal, M. Madsen, Q. Li, J. Song, J.B. Woehrstein, S.F.J. Wickham, M.T. Strauss, F. Schueder, J. Vinther, A. Krissanaprasit, D. Gudnason, A.A.A. Smith, R. Ogaki, A.N. Zelikin, F. Besenbacher, V. Birkedal, P. Yin, W.M. Shih, R. Jungmann, M. Dong, K.V. Gothelf, Routing of individual polymers in designed patterns. *Nat. Nanotech.* **10**(10), 892–898 (2015)
27. C. Lee, K.S. Kim, Y.-J. Kim, J.Y. Lee, D.-N. Kim, Tailoring the mechanical stiffness of DNA nanostructures using engineered defects. *ACS Nano* **13**(7), 8329–8336 (2019)
28. S. Hoshika, I. Singh, C. Switzer, R.W. Molt, N.A. Leal, M.-J. Kim, M.-S. Kim, H.-J. Kim, M.M. Georgiadis, S.A. Benner, “Skinny” and “Fat” DNA: two new double helices. *J. Am. Chem. Soc.* **140**(37), 11655–11660 (2018)
29. H. Rasmussen, J.S. Kastrop, J.N. Nielsen, J.M. Nielsen, P.E. Nielsen, Crystal structure of a peptide nucleic acid (PNA) duplex at 1.7 Å resolution. *Nat. Struct. Mol. Biol.* **4**(2), 98–101 (1997)
30. G.C. Pugh, J.R. Burns, S. Howorka, Comparing proteins and nucleic acids for next-generation biomolecular engineering. *Nat. Rev. Chem.* **2**(7), 113–130 (2018)
31. A.A. Joseph, A. Pardo-Vargas, P.H. Seeberger, Total synthesis of polysaccharides by automated glycan assembly. *J. Am. Chem. Soc.* **142**(19), 8561–8564 (2020)
32. R.N. Zuckermann, J.M. Kerr, S.B.H. Kent, W.H. Moos, Efficient method for the preparation of peptoids [oligo(N-substituted glycines)] by submonomer solid-phase synthesis. *J. Am. Chem. Soc.* **114**(26), 10646–10647 (1992)
33. K. Lang, J.W. Chin, Cellular incorporation of unnatural amino acids and bioorthogonal labeling of proteins. *Chem. Rev.* **114**(9), 4764–4806 (2014)
34. T. Kawakami, H. Murakami, H. Suga, Ribosomal synthesis of polypeptoids and peptoid–peptide hybrids. *J. Am. Chem. Soc.* **130**(50), 16861–16863 (2008)
35. A.E. Stross, G. Iadevaia, D. Núñez-Villanueva, C.A. Hunter, Sequence-selective formation of synthetic H-bonded duplexes. *J. Am. Chem. Soc.* **139**(36), 12655–12663 (2017)
36. P. Troselj, P. Bolgar, P. Ballester, C.A. Hunter, High-fidelity sequence-selective duplex formation by recognition-encoded Melamine Oligomers. *J. Am. Chem. Soc.* **143**(23), 8669–8678 (2021)
37. J.-F. Lutz, 100th anniversary of macromolecular science viewpoint: toward artificial life-supporting macromolecules. *ACS Macro Lett.* **9**(2), 185–189 (2020)
38. Y. Zhang, R. Cao, J. Shen, C.S.F. Detchou, Y. Zhong, H. Wang, S. Zou, Q. Huang, C. Lian, Q. Wang, J. Zhu, B. Gong, Hydrogen-bonded duplexes with lengthened linkers. *Org. Lett.* **20**(6), 1555–1558 (2018)
39. C.A. Rohl, C.E.M. Strauss, K.M.S. Misura, D. Baker, in *Protein Structure Prediction Using Rosetta*. Methods in Enzymology [Internet] (Academic Press, 2004), pp. 66–93 [cited 2021 Jun 24]. (Numerical Computer Methods, Part D; vol. 383). Available from: <https://www.sciencedirect.com/science/article/pii/S0076687904830040>
40. R.F. Service, ‘The game has changed.’ AI triumphs at solving protein structures [Internet]. Science | AAAS (2020) [cited 2021 Jun 24]. Available from: <https://www.sciencemag.org/news/2020/11/game-has-changed-ai-triumphs-solving-protein-structures>
41. P.-S. Huang, S.E. Boyken, D. Baker, The coming of age of de novo protein design. *Nature* **537**(7620), 320–327 (2016)
42. S.E. Boyken, Z. Chen, B. Groves, R.A. Langan, G. Oberdorfer, A. Ford, J.M. Gilmore, C. Xu, F. DiMaio, J.H. Pereira, B. Sankaran, G. Seelig, P.H. Zwart, D. Baker, De novo design of protein homo-oligomers with modular hydrogen-bond network–mediated specificity. *Science* **352**(6286), 680–687 (2016)

43. F.H. Arnold, Design by directed evolution. *Acc. Chem. Res.* **31**(3), 125–131 (1998)
44. A. Ravikumar, G.A. Arzumanyan, M.K.A. Obadi, A.A. Javanpour, C.C. Liu, Scalable, continuous evolution of genes at mutation rates above genomic error thresholds. *Cell* **175**(7), 1946–1957.e13 (2018)
45. B.G. Wong, C.P. Mancuso, S. Kiriakov, C.J. Bashor, A.S. Khalil, Precise, automated control of conditions for high-throughput growth of yeast and bacteria with eVOLVER. *Nat. Biotechnol.* **36**(7), 614–623 (2018)
46. G. Tikhomirov, P. Petersen, L. Qian, Fractal assembly of micrometre-scale DNA origami arrays with arbitrary patterns. *Nature* **552**(7683), 67–71 (2017)
47. M.C. Groth, W.M. Rink, N.F. Meyer, F. Thomas, Kinetic studies on strand displacement in de novo designed parallel heterodimeric coiled coils. *Chem. Sci.* **9**(18), 4308–4316 (2018)
48. H. Tran, S.L. Gael, M.D. Connolly, R.N. Zuckermann, Solid-phase submonomer synthesis of peptoid polymers and their self-assembly into highly-ordered nanosheets. *J. Vis. Exp.* **57**, 3373 (2011)
49. H.M. Seifert, K. Ramirez Trejo, E.V. Anslyn, Four simultaneously dynamic covalent reactions. Experimental proof of orthogonality. *J. Am. Chem. Soc.* **138**(34), 10916–24
50. P.T. Corbett, J. Leclaire, L. Vial, K.R. West, J.-L. Wietor, J.K.M. Sanders, S. Otto, Dynamic combinatorial chemistry. *Chem. Rev.* **106**(9), 3652–3711 (2006)
51. T. Wei, J.H. Jung, T.F. Scott, Dynamic covalent assembly of peptoid-based ladder oligomers by vernier templating. *J. Am. Chem. Soc.* **137**(51), 16196–16202 (2015)
52. M. Hebel, A. Riegger, M.M. Zegota, G. Kizilsavas, J. Gačanin, M. Pieszka, T. Lückcrath, J.A.S. Coelho, M. Wagner, P.M.P. Gois, D.Y.W. Ng, T. Weil, Sequence programming with dynamic boronic acid/catechol binary codes. *J. Am. Chem. Soc.* **141**(36), 14026–14031 (2019)
53. S.C. Leguizamón, M.F. Dunn, T.F. Scott, Sequence-directed dynamic covalent assembly of base-4-encoded oligomers. *Chem. Commun.* **56**(56), 7817–7820 (2020)
54. S.C. Leguizamón, T.F. Scott, Sequence-selective dynamic covalent assembly of information-bearing oligomers. *Nat. Commun.* **11**(1), 784 (2020)
55. Y. Kamiya, H. Asanuma, Light-driven DNA nanomachine with a photoresponsive molecular engine. *Acc. Chem. Res.* **47**(6), 1663–1672 (2014)
56. J. Niu, R. Hili, D.R. Liu, Enzyme-free translation of DNA into sequence-defined synthetic polymers structurally unrelated to nucleic acids. *Nat. Chem.* **5**(4), 282–292 (2013)
57. W. Meng, R.A. Muscat, M.L. McKee, P.J. Milnes, A.H. El-Sagheer, J. Bath, B.G. Davis, T. Brown, R.K. O'Reilly, A.J. Turberfield, An autonomous molecular assembler for programmable chemical synthesis. *Nat. Chem.* **8**(6), 542–548 (2016)
58. D.J. Rizzo, G. Veber, T. Cao, C. Bronner, T. Chen, F. Zhao, H. Rodriguez, S.G. Louie, M.F. Crommie, F.R. Fischer, Topological band engineering of graphene nanoribbons. *Nature* **560**(7717), 204–208 (2018)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Controlling Single Molecule Conjugated Oligomers and Polymers with DNA



Rikke Asbæk Hansen and Kurt Vesterager Gothelf

Abstract The unique specificity of DNA interactions and our ability to synthesize artificial functionalized DNA sequences makes it the ideal material for controlling self-assembly and chemical reactions of components attached to DNA sequences. Inspired by the field of molecular electronics and the lack of methods to assemble molecular components, we have explored the organization of conjugated molecular components using DNA-based self-assembly. In this chapter, we provide an overview of our efforts first to assemble and chemically couple conjugated molecules directed by DNA, and more recently to assemble conjugated polymers in DNA nanostructures. At the end of the chapter, we provide a short overview of work by other groups in the field.

Molecular electronics is an interdisciplinary subject that aims to assemble electronic components in a bottom-up approach using conducting molecules as building blocks. It represents an alternative to the lithography-based top-down preparation of silicon-based electronic circuits. A large variety of organic molecules with potentially useful electronic properties are available [1, 2]; however, one of the major obstacles for taking advantages of the unique properties of these molecules is the challenge associated with connecting the molecular components in a controllable manner.

Conjugated oligomers and polymers have been used extensively in bulk organic electronics such as light-emitting diodes, field-effect transistors, and polymeric solar cells [3, 4]. The study of the single molecule properties of conjugated polymers is more limited; however, studies have been performed using, e.g., scanning tunnel microscopy [5] and single molecule spectroscopy [6, 7]. The major problem in the production of single molecular-based electronics is the inability to arrange the components to form circuits.

In a seminal theoretical paper from 1987, Robinson and Seeman proposed a design for an electronic molecular memory chip formed by DNA guided self-assembly

R. A. Hansen · K. V. Gothelf (✉)
iNANO and Department of Chemistry, Aarhus University, Aarhus, Denmark
e-mail: kvg@chem.au.dk

of conjugated polymer as wires and metal complexes as memory components [8]. Throughout the past 18 years a major focus of our laboratory has been to implement, at least in part, Robinson and Seeman's vision to use DNA-directed self-assembly to generate and/or position molecular conjugated wires [9]. DNA nanotechnology provides a useful tool for controlling and connecting individual conjugated oligomer and polymer molecules at the nanoscale [10]. In order to exploit DNA nanotechnology for controlling organic molecules, the molecules have to be functionalized with single stranded DNA (ssDNA) sequences, which in turn direct the assembly of the molecules. In this paper, we will provide an overview of our work in this area, and we have summarized the contributions from others to the area at the end of the paper.

1 Modular Self-Assembly of Molecular Components

Our entry into this area was inspired by the intense interest in molecular electronics and the emergence of DNA-templated synthesis in beginning of the new millennium [11, 12]. We envisioned that DNA-templated synthesis may serve as a tool to both assemble and couple individual components to a fully conjugated oligomer as shown in Fig. 1.

Bunz and coworkers had shown that it was possible to assemble molecular rod oligomers using rod monomers displaying DNA at each terminus; however, the resulting oligomers were unconjugated as their rods were separated by DNA helices [13]. In 2001, Czapinski and Sheppard reported on the DNA-directed coupling of salicylaldehydes into metal-salen complexes [14]. The metal-salen coupling was ideal for assembly of conjugated oligomers since the linkages are linear between the headgroups to form a conjugated oligomer. Furthermore, the electronic properties of the complexes should be interchangeable by changing the metal ion.

In our first approach to solve this problem, reported in 2004, we described the development of a DNA-directed bottom-up method for programmed assembly that utilized covalent couplings between multiple organic modules [15, 16]. The basic

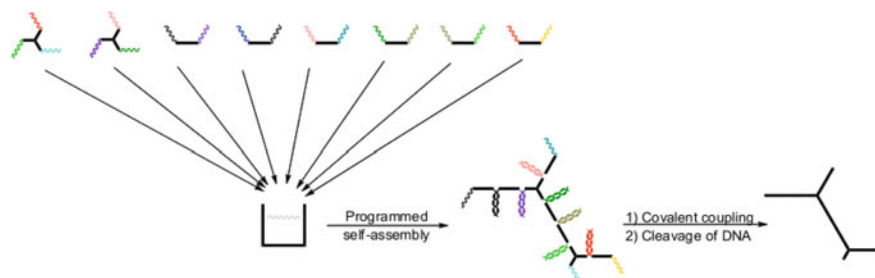


Fig. 1 Concept of DNA-directed assembly and coupling of organic rod monomers into conjugated organic oligomers (linear and tripodal black rods: molecular monomers; colored wavy lines: DNA)

building blocks are rigid conjugated modules with an oligo(phenylene ethylene) backbone and have salicylaldehyde-derived termini. Additionally, each terminus contains a hydroxyl linker that is functionalized with either a 4,4'-dimethoxytrityl (DMT) protecting group or a phosphoramidite moiety. This allowed for incorporation of the module into the 3'-end of a DNA strand followed by removal of the DMT protecting group and synthesis of the second DNA strand. The modules were synthesized both as a linear oligonucleotide-functionalized module (LOM) and a tripoidal oligonucleotide-functionalized module (TOM) (Fig. 2a). By using complementary strands on the modules, it was possible to direct the assembly without the need for additional DNA templates. The modules were covalently coupled with metal-salen formation between the salicylaldehydes at the termini of the modules by reaction with ethylenediamine (EDA) and a manganese salt. The salicylaldehyde groups of two modules are brought in close proximity when their complementary DNA strands are annealed together allowing a pseudo-intramolecular reaction to occur (Fig. 2b).

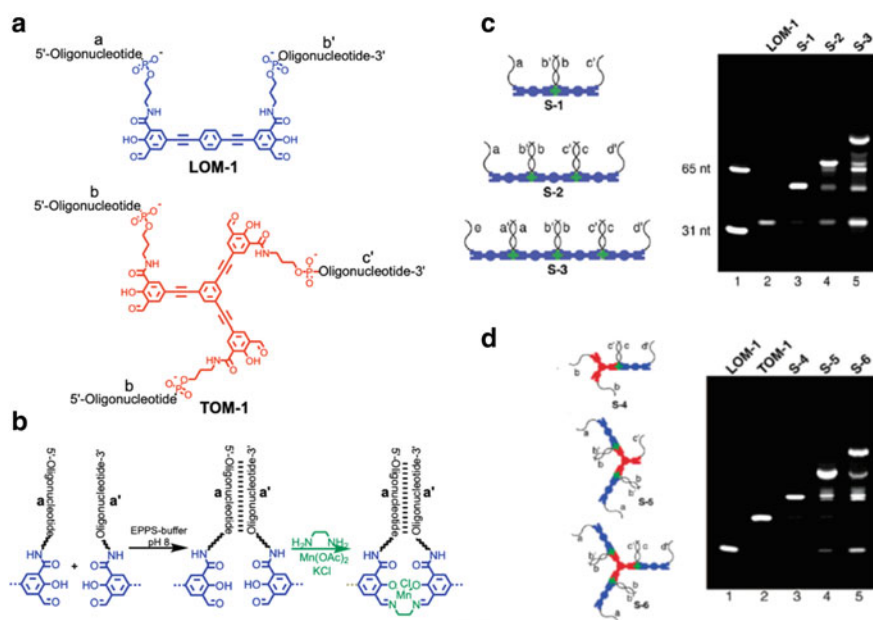


Fig. 2 **a** Chemical structure of LOM and TOM. LOM-1 contains two 15-mer sequences: a and b'. TOM-1 contains three 15-mer sequences: one c' and two b. **b** schematic illustration of the DNA-templated coupling. First step is an annealing of the two complementary strands, bringing the two reactive groups into close proximity. The next step is the formation of a covalent Mn-salen link between the two terminal salicylaldehydes by reaction with EDA and $Mn(OAc)_2$. **c** Illustration of the linear oligomer and gel electrophoresis in 8 M urea shows that Mn-salen products are covalently linked. **d** Analogous illustration of the tripoidal constructs. Adapted with permission from [15]. Copyright (2004) American Chemical Society

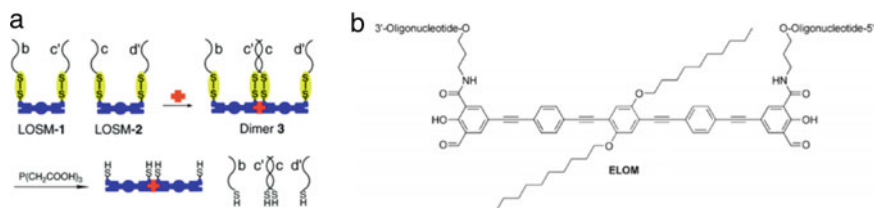


Fig. 3 **a** Illustration of DNA-directed coupling of LOSM by metal-salen formation, followed by cleavage of DNA strands by reaction with TCEP. Adapted from [18] with permission from the Royal Society of Chemistry. **b** Chemical structure of the elongated linear oligonucleotide-functionalized module (ELOM)

The DNA sequences on the modules can be altered, which makes it possible to combine the LOMs and TOMs to form a variety of linear and branched predefined structures. Self-assembly and coupling of up to four differently encoded linear modules was successfully obtained (Fig. 2c). More complex structures were built by combining LOMs and TOMs (Fig. 2d). Up to three LOMs was successfully coupled to a single TOM. This method provides a unique degree of control of the architecture of the molecular wire; however, a major challenge of the approach was that there seemed to be a limit of assembling four modules. Despite many attempts, we never managed to make penta- or higher-order structures in reasonable yields.

The research in this direction was continued to improve the system. In 2005, Nielsen et al. [17] investigated the DNA-directed double reductive amination of salicylaldehydes in the presence of EDA to form tetrahydrosalen. This amine-linked structure was found to be much more stable toward acid, heat, methylamine, and ethylenediamine tetraacetic acid (EDTA).

Additionally, the selective cleavage of DNA strands from the conjugated backbone was enabled by installing disulfide bridges between the DNA sequence and the LOM, called LOMS (Fig. 3a). The disulfide modified strands were successfully cleaved from the backbone by treatment with tris(2-carboxyethyl)phosphine (TCEP) [17, 18].

The attempts to improve the system did not make it possible to assemble more than four modules with a satisfactory yield. We speculated that this may be caused by steric and charge repulsion between the large oligonucleotide duplexes, as the diameter of a duplex is around 2 nm which is the same as the length of a LOM. Therefore, it is expected that the duplexes will induce steric strain as the structures grow. One possible solution to this problem is to extend the length of the linear module. In 2006, Blakskjær et al. [19] reported the synthesis of an elongated linear oligonucleotide-functionalized module (ELOM) (Fig. 3b). The DNA-directed coupling between an ELOM and LOM was successful; however, attempts to couple two or three ELOMs were unsuccessful. It is proposed that this is due to the amphiphilic nature of the ELOMs.

In 2008, a new strategy for DNA-programmed coupling of molecular rods was reported by Andersen et al. [20]. The aim of the new method was to make a simpler

setup that only requires one conjugate to form dimers or higher-order structures. Two 10-mer oligonucleotide-functionalized rod-type modules were synthesized and arranged on a DNA template strand. This design places the rod modules approximately on the same side of the double helix, one helical turn apart. Inspired by the previous studies, the modules were functionalized with salicylaldehydes in each terminus. The rod was also functionalized with an activated ester in the middle of the structure, in order to couple to an amino-modified oligonucleotide. The templated coupling was tested by mixing a 20-mer template strand with two equivalents of reactive strand. The strands were annealed followed by reaction with ethylene diamine and a metal salt to form the pseudo-intramolecular coupling between the two reactive strands (Fig. 4a). Denaturation of the double helix allows for purification of the single stranded coupled product. It was possible to isolate the desired Mn-salen dimer in 10% yield, whereas the Ni-salen dimer was isolated in 25% yield. Additionally, it was also possible to form heterodimers by using two different DNA sequences. The preparation of a trimer was also attempted; however, it was not possible to identify the product by mass spectrometry. The starting materials were consumed in all reactions, which suggests that the low yield could be due to side reactions or loss of material due to aggregation. A more hydrophilic molecule could potentially solve the aggregation problems and allow for the formation of higher-order structures. An advantage of this method compared to the previous published method using LOM and TOM modules is that this method yields a coupled module with extending single strands. These strands could be hybridized to other DNA nanostructures, allowing a precise positioning of the nanowire. While the synthesis was simpler than LOM/TOM approach, it was limited by low yields and inability to form structures more complex than a dimer.

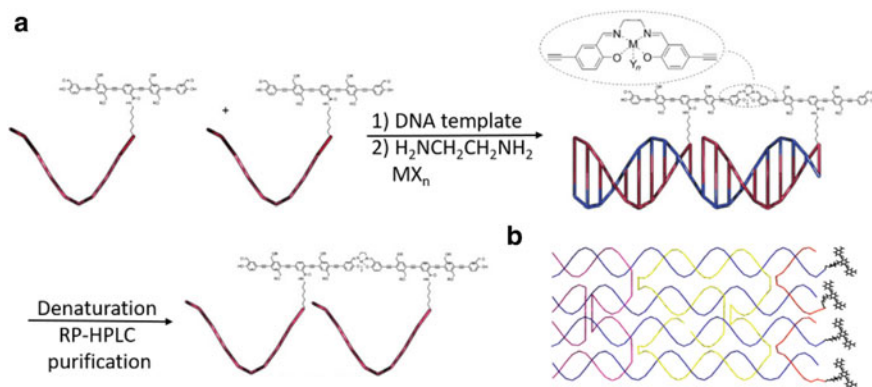


Fig. 4 a DNA-templated dimerization of two molecular rods by metal-salen formation. Two reactive strands are annealed with a template strand to bring the molecular rods in close proximity. The rods are coupled by reaction with EDA and a metal salt. By denaturation, the template strand is removed, and the coupled product is purified by RP-HPLC. Adopted from [20] with permission from John Wiley and Sons. Copyright 2008 Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim. b Illustration of 4-helix bundle with four modules

In the continuous search for DNA structures that can serve as templates for DNA-directed couplings of conjugated organic molecules, interhelical couplings in a DNA 4-helix bundle (4-HB) have been investigated [21]. A conjugated linear molecule containing terminal salicylaldehyde groups and a central activated ester was synthesized. The module was coupled to amino-modified DNA strands and incorporated into specific locations in a well-defined 4-HB, to allow for selective interhelical couplings (Fig. 4b). The coupling of the modules was tested with both a metal-salen and dihydrazone formation; the coupled products were then analyzed by denaturing PAGE analysis. The couplings were found to be very distance dependent, and no non-templated couplings were observed. With the metal-salen coupling, it was possible to obtain dimers in moderate yields; however, higher-order structures were not obtained. For the dihydrazone couplings, trimer formation was achieved.

The coupling of salicylaldehydes to form metal-salen linkages has been successful in other contexts. In a very recent study, the single molecule electronic properties of Mn(III), Co(III), and Fe(III)-salen complexes in a break junction were studied. The study showed that the metal-salen bridge is a relatively poor conductor but that the conductivity is dependent on the nature of the metal [22]. Unfortunately, the metal-salen formation is reversible, and the complex is labile in aqueous media. Therefore, alternative coupling strategies have been investigated in order to form irreversible, hydrolysis-resistant, and conjugated linkages between molecular modules. A method for DNA-directed formation of 1,3-diyne linkages between conjugated molecular building blocks has been reported by Ravnsbæk et al. [23]. The 1,3-diynes can be obtained by a Glaser-Eglinton reaction between terminal alkynes. An oligo(phenylene ethylene) molecular rod containing two terminal acetylene groups was synthesized to enable the formation of a conjugated linear oligomer. Additionally, each monomer was functionalized with DMT and phosphoramidite functionalities for incorporation into a DNA strand by automated oligonucleotide synthesis (Fig. 5a + b). A series of four 30-mer oligonucleotides were prepared by automated oligonucleotide synthesis, during which the phosphoramidite rod monomer was incorporated in the middle of the strand. This resulted in four oligonucleotide-functionalized diacetylene modules (ODM) consisting of organic module with two 15-mer sequences in each terminal region (Fig. 5c). The DNA-directed Glaser-Eglinton reactions were performed between the different ODM strands. Denaturing PAGE analysis of the crude products shows the formation of dimer and trimer in a high yield (Fig. 5d, lanes 2–4), while the tetramer shows a lower yield (Fig. 5d, lane 5). The lower yield is believed to be caused by the increasing electrostatic repulsion and steric hindrance between the increasing number of DNA strands. The electrostatic repulsion could be removed by changing the DNA strands to PNA strands; however, this would also result in a much less soluble molecule. The formed oligomers have a size of around 4–8 nm and could have an application as conducting nanowire. The oligomer contains single stranded DNA in each terminus, which could be used as handles for a specific positioning of the nanowire on a DNA origami structure.

Despite the advances in DNA-directed synthesis, it was not possible to obtain structures longer than tetramers. Significantly longer sequence specific oligomers

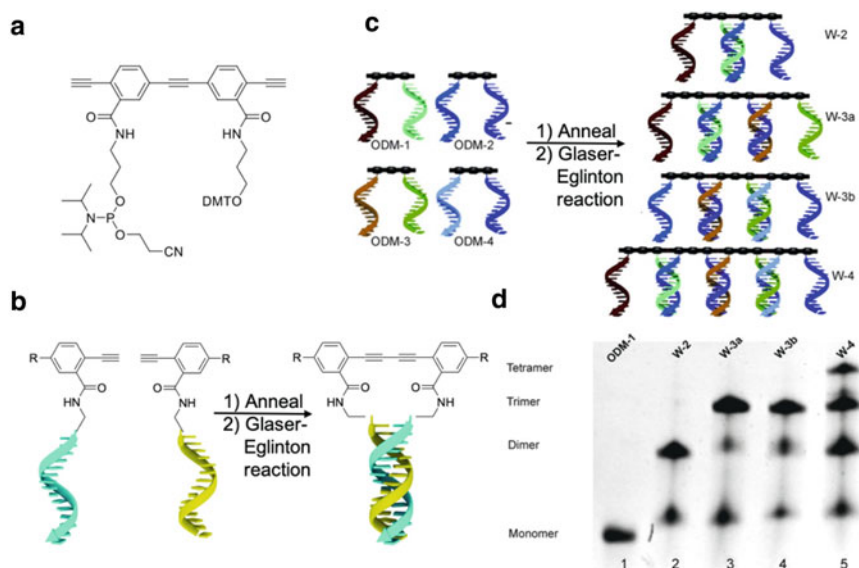


Fig. 5 **a** Chemical structure of the oligo(phenylene ethylene) phosphoramidite monomer. **b** illustration of annealing and subsequent Glaser-Eglinton coupling between two modified DNA strands. **c** oligomerization of ODM monomers by multiple Glaser-Eglinton reactions. **d** denaturing PAGE analysis of DNA-directed Glaser-Eglinton oligomerization of ODM sequences. Reproduced from [23] with permission from John Wiley and Sons. Copyright 2011 Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim

have been prepared by DNA-directed synthesis by others; however, these oligomers were not conjugated, and only one DNA strand is linked to the products [24–26].

2 Conjugated Polymers on DNA Origami

During the course of our work on the modular DNA-directed assembly, the DNA origami method was published by Rothemund in 2006 [27]. This method enabled the self-assembly of, from a DNA point of view, very large structures with unique addressability, such as the rectangular mono-layer origami with dimensions of $100 \times 70 \text{ nm}^2$. Such a structure appeared to be an almost perfect breadboard for assembly and coupling of the molecular modules described above. However, upon closer consideration, it would require extremely efficient immobilization and chemical cross-linking to form a continuous wire of 100 nm across the origami structures. The modules are only 2 nm long, and it would require immobilization of approximately 50 modules and 49 coupling reactions that should all be successful to create a continuous wire. Especially in light of the poor yields obtained when attempting

to template the assembly in a 4-helix sheet as described above [23], we chose to abandon the modular assembly strategy for creating larger wires on DNA origami.

Instead, we turned our attention to conjugated polymers that offers a continuous single molecular conjugated system that may be more than 100 nm long [28]. The drawback is the lack of full control over the length of the polymer; however as long as it is possible to remove shorter polymers, this may be acceptable. In order to control the assembly of such a polymer on DNA origami, it was a requirement to conjugate DNA sequences along the polymer to create a graft-type polymer. Based on our previous experience with hydrophobic wires and DNA, we assumed that a high density of DNA along the polymer (one DNA strand per monomer) would be required to avoid aggregation and precipitation. The DNA strands could be attached to the polymer by three different methods: (1) premade DNA strands are coupled to the monomers which are then polymerized, (2) premade DNA strands with a chemical handle are coupled to a complementary functional group on a premade polymer, and (3) the polymer is immobilized on a solid support, and the DNA strands are synthesized on the polymer by automated synthesis. We decided to pursue the latter approach since we believed this would enable the synthesis of long polymers with a high density of DNA.

As described by Knudsen et al. [29], poly(phenylene vinylene) containing a triethylene glycol (TEG) linker appending from each repeat unit was synthesized (Fig. 6). The TEG linkers were terminated with a *tert*-butyldiphenylsilyl (TBDPS) protecting group, and after deprotection, the terminal alcohol served as starting point for the DNA synthesis. Before DNA synthesis, the polymers were characterized by scanning tunneling microscopy (STM) (Fig. 7a) [30]. In addition to providing information about the molecular structure and composition of the polymer, this also showed that a fraction of the polymers was very long (>200 nm).

By partially deprotecting the side chains and exposing approximately 20–40% of the hydroxyl groups, it was possible to immobilize the polymer onto a solid support through phosphoramidite chemistry. This was followed by deprotection of the remaining side chains and synthesis of ssDNA directly onto the side chains of the polymer through automated solid phase oligonucleotide synthesis. Upon cleavage, deprotection, and purification, a water-soluble material was obtained (poly(APPV-DNA)).

By this method, we obtained very long DNA grafted polymers of around 200 nm that disperse well on a mica surface as shown in the AFM image in Fig. 7b. It was later discovered that the polymers had a tendency to aggregate in the presence of divalent cations, which significantly lowered the intensity of the emission [31]. The polymers were purified by size exclusion chromatography, and the fractions containing the longer polymers were selected for further experiments.

In order to control the immobilization of the polymer along a specific track on DNA origami, we used Rothemund's rectangular DNA origami structure containing a line of single stranded DNA appending for every 5 nm on the origami surface. The density of DNA on the polymer is much higher as approximately 2/3 of each repeat unit of the polymer has a 9-mer sequence. The thermal stability of the attachment

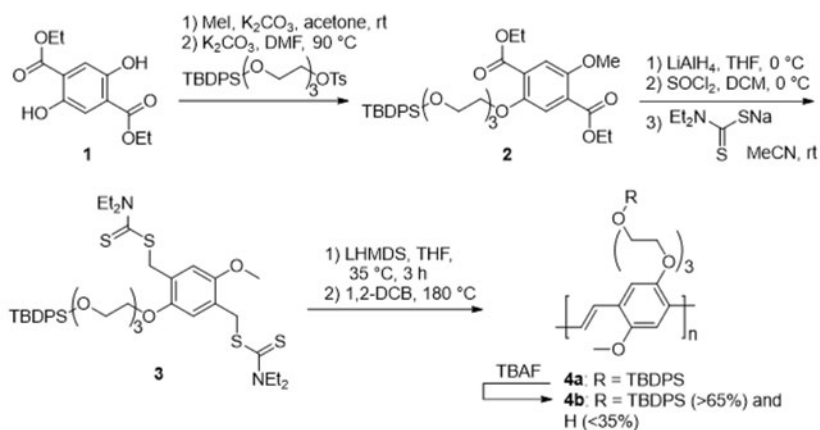
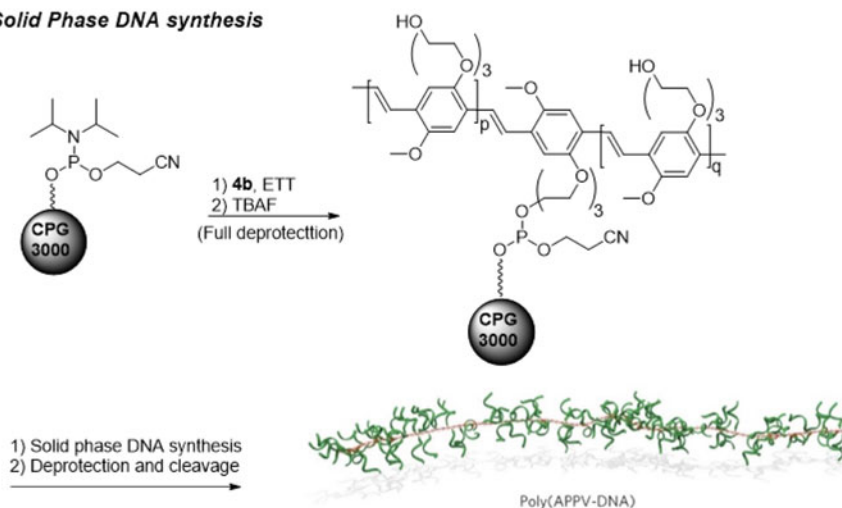
Polymer synthesis**Solid Phase DNA synthesis**

Fig. 6 Synthesis of poly(APPV-DNA). The PPV polymer (**4a**) is synthesized by a dithiocarbamate route. Partial TBDPS deprotection to give **4b** allows immobilization on a phosphoramidite-functionalized CPG support. After removal of the remaining protecting groups, ssDNA sequences are synthesized on the polymer by automated DNA synthesis. As the last step, poly(APPV-DNA) is deprotected and cleaved from the solid support. Reused with permission from MacMillan Publishers Ltd: [Nature Nanotechnology] [29] copyright (2015)

was greater than the 9-mer's melting temperature because of the polyvalent binding of these oligos to the origami.

As shown in Fig. 8a, the polymer was efficiently immobilized along designed linear, bent, and U-shaped tracks on the DNA origami [29]. The polymer was also routed on a 3D barrel shaped DNA origami structure designed by Wickham et al.

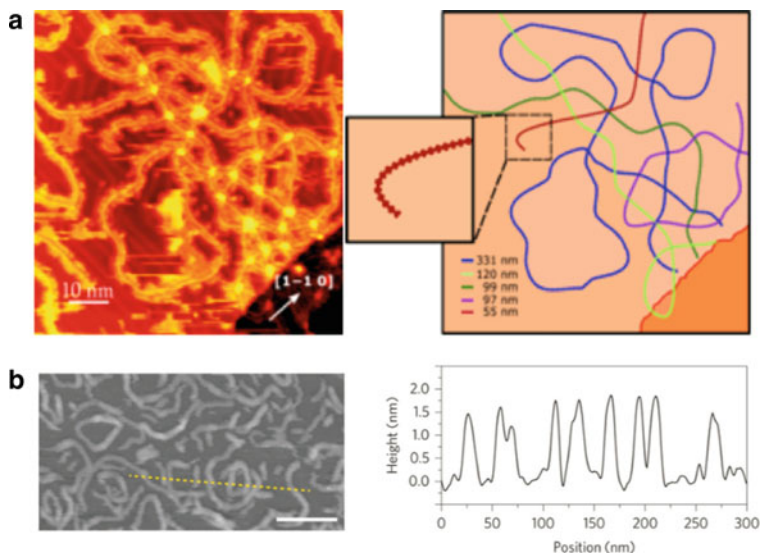


Fig. 7 **a** left: STM image of PPV-TBDPS (4a) on Au(111), right: schematic map showing the different polymer strands and their respective lengths. Polymer strands are only measured as far as the edges of the STM image field. Reproduced from [30] with permission from The Royal Society of Chemistry. **b** left: AFM topography image showing poly(APPV-DNA) dispersed onto a mica surface, right: height measurements of poly(APPV-DNA) on the mica surface. Scale bar = 100 nm. Reused with permission from MacMillan Publishers Ltd: [Nature Nanotechnology] [29] copyright (2015)

[32]. The 3D shape of the polymer was characterized by PAINT super-resolution imaging as shown in Fig. 8b.

In further studies, a method for controlling the dynamics of single polymers on DNA origami was developed by Krissanaprasit et al. [33]. It was possible to switch the conformation of single polymer molecules by toehold-mediated strand displacement reactions. The polymer could be directed to one of two tracks on a DNA origami by addition of linker strands. The linker strands could be removed by toehold-mediated strand displacement with remover stands allowing switching to the other track (Fig. 9).

Immobilizing conjugated polymers onto a DNA origami platform allows the discovery of new properties of these polymers. One of the goals motivating the development of DNA grafted conjugated polymers is to enable characterization of intra- and intermolecular energy transfer for single polymer molecules. For this purpose, a fluorene-based DNA grafted polymer (poly(F-DNA)) was synthesized using the same approach as for the poly(APPV-DNA) [34]. By positioning both the poly(F-DNA) and poly(APPV-DNA) on the same origami structure, the energy transfer between poly(F-DNA) and poly(APPV-DNA) could be investigated (Fig. 10a). When polymers were immobilized on opposite sites of the DNA origami, no energy transfer was detected. This is believed to be due to lacking physical contact between the polymers.

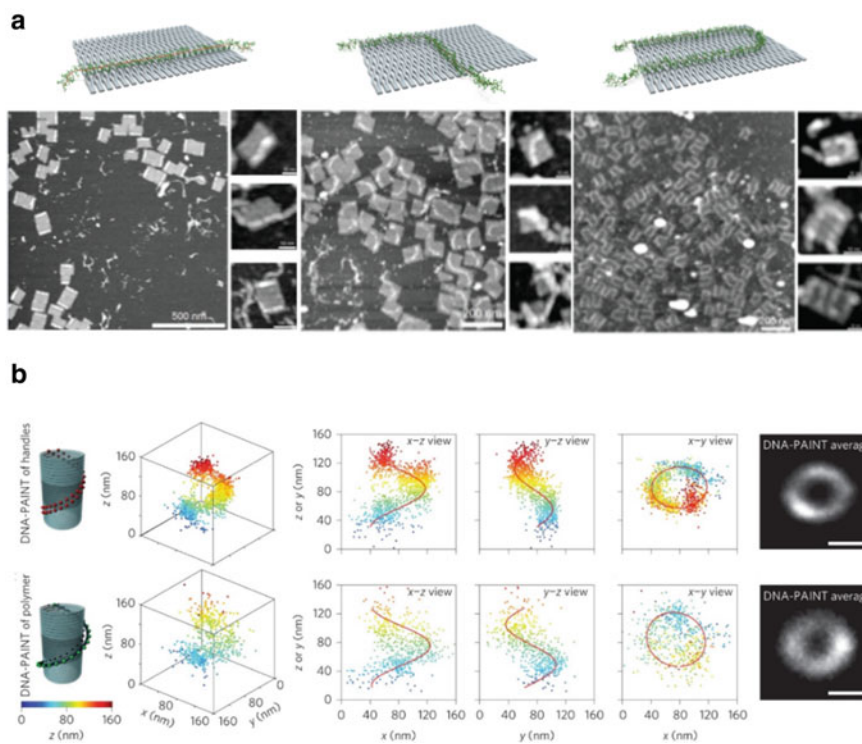


Fig. 8 **a** Top: Illustrations of the of poly(APPV-DNA) immobilized in designed patterns on flat rectangular origami. Bottom: AFM topography images of these structures. **b** left: 3D DNA PAINT of guide staple strands on the DNA structure, right: 3D DNA PAINT of the polymer attached to the DNA structure. Scale bar = 50 nm. Reused with permission from MacMillan Publishers Ltd: [Nature Nanotechnology] [29] copyright (2015)

Therefore, the energy transfer was tested in solution by using complementary DNA strands on the polymers. Energy transfer from poly(F-DNA) to poly(APPV-DNA) was observed with a relative efficiency of 37%. It is assumed that the complementary DNA grafted polymers form a multi-polymer particle, which means that the energy transfer most likely is not taking place between individual polymers (Fig. 10b).

In order to investigate the intermolecular energy transfer between individual polymers, immobilization onto the same side of an origami structure is necessary to both obtain physical contact between the polymers and control over the stoichiometry. Ideally, this setup can be improved in future to arrange multiple different conjugated polymers on one origami platform and transfer energy to a final acceptor. This will allow harvesting light energy at a broad range of wavelength to exploit the white light of the sun.

The nanoscale transport of light in photosynthesis is a fundamental process where light energy is converted to chemical energy and is a prototypical “green” source of energy. Thus, the ability to harvest light at the nanoscale has been investigated

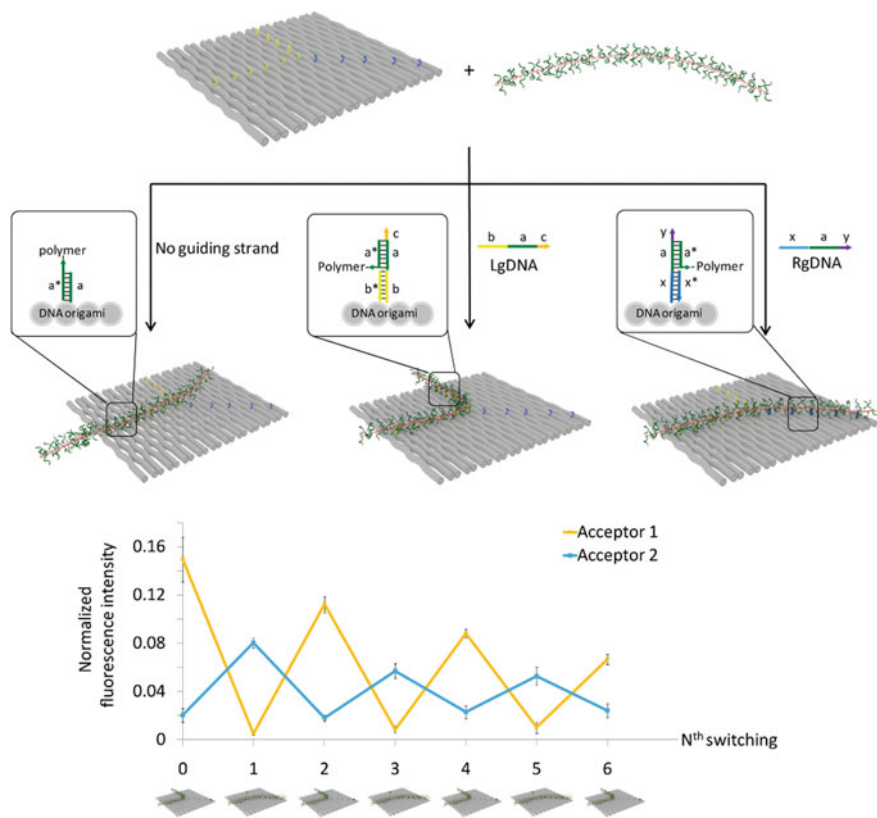


Fig. 9 **Top** Illustration of the conformational switching of single polymer molecules on DNA origami by toehold-mediated strand displacement reactions. **Bottom** FRET measurements of the switching showing six successful events. Reprinted with permission from [33]. Copyright (2016) American Chemical Society

intensely [35, 36]; however, most research has focused on energy transfer cascades using small molecule dyes. In a recent report by Madsen et al. [37], a single molecule polymer poly(APPV-DNA) was investigated for its properties as a photonic wire. The DNA grafted polymer was immobilized onto a single DNA origami by hybridization to a track of single stranded staple strands extending from the origami structure. On the same, origami structure donor and acceptor fluorophores were placed at specific positions along the polymer, allowing for energy transfer from the donor fluorophores to the polymer, through the polymer, and from the polymer to an acceptor fluorophore (Fig. 11a). The energy transfer was studied by both ensemble fluorescence spectroscopy and single molecule spectroscopy. The distance dependence of energy transfer through poly(APPV-DNA) was tested by investigating six different donor-acceptor distances on a DNA origami platform. Distances ranging from 10.5 nm to 24 nm were investigated, and it was found that the energy transfer efficiency

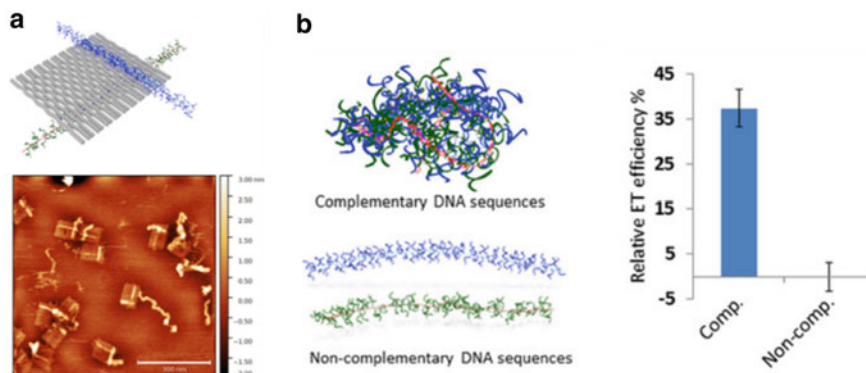


Fig. 10 **a** Illustration and AFM topography image of poly(F-DNA) and poly(APPV-DNA) immobilized in an orthogonal alignment on opposite surfaces of the same flat rectangular origami structure. Scale bar = 300 nm. **b** Energy transfer between poly(F-DNA) and poly(APPV-DNA). Left: Illustration showing that sequence complementarity results in multiple-strand complexes between the DNA grafted polymers, whereas non-complementary strands result in separate DNA grafted polymers. Right: Quantification of energy transfer from poly(F-DNA) to poly(APPV-DNA). Reproduced from [34] with permission from John Wiley and Sons. Copyright 2017 Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim

decreased with increased distance (Fig. 11b). Interestingly, energy transfer was still observed at 24 nm distance between donor and acceptor. As the energy transfer efficiency did not approach zero at the longest distance, it is expected that energy transfer at longer distances should be feasible. This efficient intramolecular energy transfer makes poly(APPV-DNA) an efficient antenna molecule for use in light harvesting nanodevices.

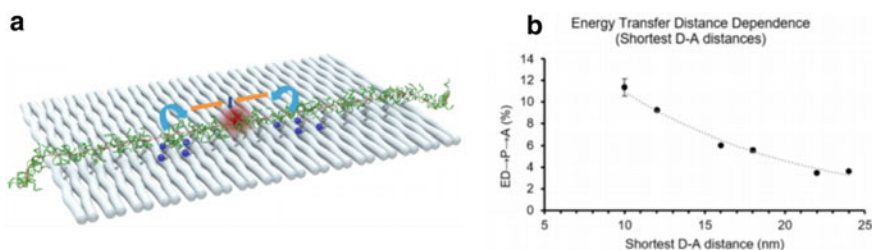


Fig. 11 **a** Schematic illustration of poly(APPV-DNA) hybridized to a DNA origami platform containing donor and acceptor fluorophores. Upon excitation of donor fluorophores energy is transferred to the polymer and finally to acceptor fluorophore in the middle of the origami. **b** Data showing the distance dependence energy transfer through poly(APPV-DNA). Reprinted with permission from [37]. Copyright (2021) Americal Chemical Society

3 Work from Other Groups

Other groups have also contributed to the work within this field. Sleiman's group has shown that nucleobase templated polymerization can be used to control the length and polydispersity of a conjugated polymer [38]. A templated polymer with controlled molecular weight and narrow polydispersity was synthesized by a living polymerization method. A thymine displaying template polymer was synthesized by ring-opening metathesis polymerization. Adenine-modified conjugated monomers were aligned on the template polymer by hydrogen-bonding interactions. Subsequent Sonogashira polymerization leads to the synthesis of a conjugated polymer (Fig. 12). The daughter polymer was found to have a narrow molecular weight distribution and a chain length of around 25 monomeric units, close to the length of the template polymer. This is in contrast to non-templated polymerization, or polymerization with an incorrect template, which gave rise to short polymers with high polydispersities. This method is a very useful tool in the synthesis of conjugated polymers of a defined length, and it could potentially enable the synthesis of sequence specific polymers by exploiting all four nucleobases. However, the method will be limited by the number of different monomers that can be incorporated. Additionally, this method does not allow for incorporation of a conjugated polymer into other DNA nanostructures due to the lack of ssDNA sequences on the polymer.

More recently, other groups have studied conjugated polymers and oligomers using DNA origami platforms. Mertig's group has synthesized end-functionalized polythiophenes [39]. A single DNA strand was attached to the end of each individual polymer, and these end-functionalized polymers were then immobilized onto a DNA origami in different patterns. They were able to demonstrate that the optical properties of densely immobilized conjugated polymers can be fine-tuned by controlling the π - π stacking interactions between the polymers. Addition of surfactant molecules

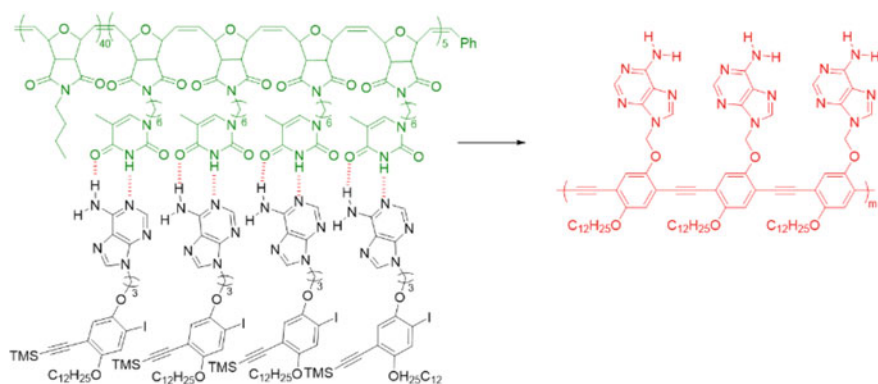


Fig. 12 Templated Sonogashira polymerization of adenine-containing monomers by hydrogen bonding to a thymine containing template polymer. Adapted with permission from [38]. Copyright (2009) American Chemical Society

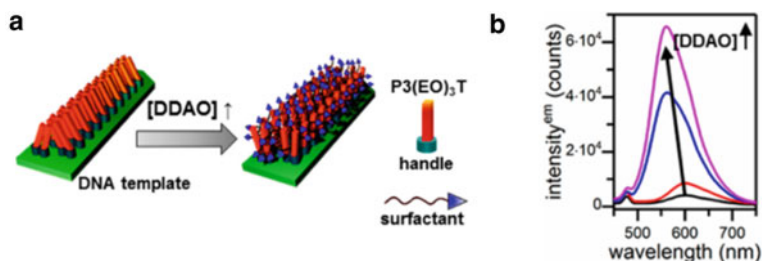


Fig. 13 **a** Scheme of surfactant-induced breakup of stacked polythiophene backbones on a DNA origami platform. DDAO = *N,N*-dimethyldodecylamine *N*-oxide. **b** fluorescence emission spectra of polythiophene at different DDAO concentrations (0.0, 0.003, 0.03, 0.3 wt%). Fluorescence increases with increasing DDAO concentration. Reprinted with permission from [39]. Copyright (2017) American Chemical Society

was able to break up the stacked polythiophene backbones and thus enhanced the fluorescent emission of the polymers (Fig. 13a, b).

The Seeman and Canary groups have presented the synthesis of aniline octamers, which were functionalized with ssDNA sequences in each end [40]. The DNA oligoaniline conjugates were successfully incorporated into 3D DNA crystals. It was possible to switch between different oxidation states of the oligoaniline by chemical treatment, and the oxidation state could be determined by the visual appearance of the DNA crystal (Fig. 14a). This reversible switching opens up the opportunity of controlling the conductivity of DNA-based systems. However, the electronic properties of the DNA structure were not been tested experimentally.

More recently, the same groups reported the preparation of a DNA origami-based molecular electro-optical modulator [41]. Two different types of conjugated oligomers were synthesized and functionalized with DNA sequences at each end. The DNA functionalized oligomers were incorporated into a flat DNA origami structure with a central cavity and formed an “X”-shape. They showed that it was possible to reversibly alter the fluorescence signal output by redox reactions (Fig. 14b).

4 Conclusion

The combined work in this field has developed unique tools to handle conjugated oligomers and polymers at the single molecule level within the field of DNA nanotechnology. The original vision was to develop methods to realize the assembly and coupling of wires and components for molecular electronics, but we are still far away from this, since we have not yet been able to characterize the conductivity of the oligomers or polymers. First of all, it is extremely difficult to make two contacts to a single molecule organic oligomer/polymer to measure conductivity. This kind of measurement has been realized by STM in ultra-high vacuum [6], and it has also been shown for carbon nanotubes on DNA origami [42]. However, in spite of

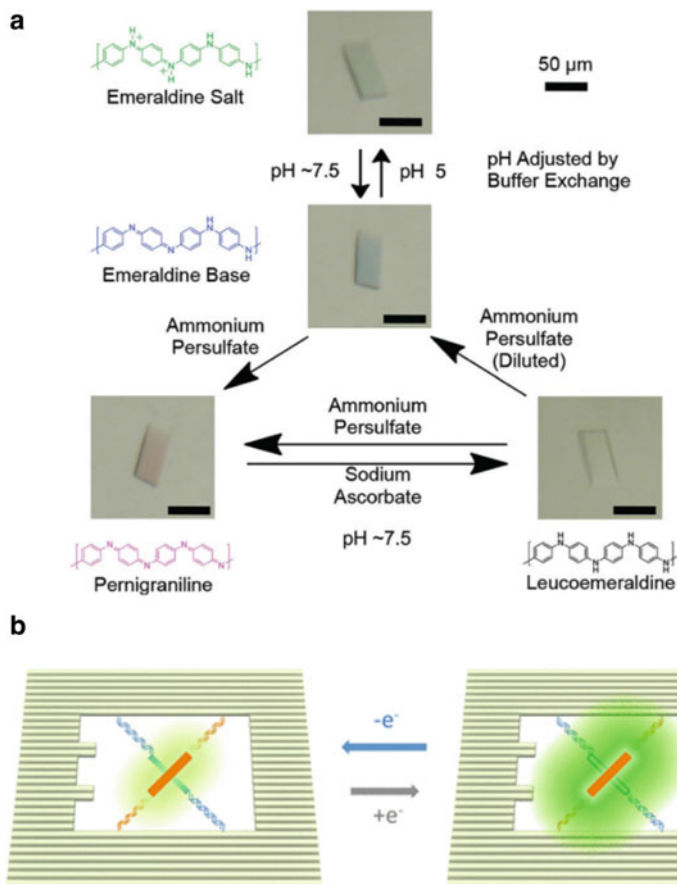


Fig. 14 **a** Image of DNA crystals with incorporated oligoaniline. Changing the charge and oxidation state of oligoaniline leads to color changes directly correlated to the oligoaniline form present. Reproduced from [40] with permission from John Wiley and Sons. Copyright 2017 Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim. **b** Illustration of DNA origami-based molecular electro-optical modulator. The DNA grafted oligomers hybridize to extended stable strands and form an “X” shape. Redox reactions makes it possible to tune the fluorescence intensity of the modulator. Reprinted with permission from [41]. Copyright (2018) American Chemical Society

several attempts, we have not been able to reliably measure the conductivity through any of the oligomers/polymers described above. Secondly, the polymers are semiconductors, and therefore, it is doubtful that the single molecule polymers would show efficient conductivity in the absence of dopants. On the other hand, we believe that the structural control of the conjugated polymers has great potential for making single molecule optical circuits. As we have recently shown, it is possible to transfer excitation from one dye to another through the polymers [37]. For future studies, we believe that the key utility will be to build systems for light harvesting. With the

spatial and chemical control that this approach offers, it may become possible to build systems that mimic photosynthesis.

References

1. S. Kubatkin, A. Danilov, M. Hjort, J. Cornil, J.L. Brédas, N. Stuhr-Hansen, P. Hedegård, T. Bjørnholm, Single-electron transistor of a single organic molecule with access to several redox states. *Nature* **425**, 698–701 (2003)
2. D.W. Steuerman, H.-R. Tseng, A.J. Peters, A.H. Flood, J.O. Jeppesen, K.A. Nielsen, J.F. Stoddart, J.R. Heath, Molecular-mechanical switch-based solid-state electrochromic devices. *Angew. Chemie* **116**, 6648–6653 (2004)
3. A. Facchetti, π -conjugated polymers for organic electronics and photovoltaic cell applications. *Chem. Mater.* **23**, 733–758 (2011)
4. J.H. Burroughes, D.D.C. Bradley, A.R. Brown, R.N. Marks, K. Mackay, R.H. Friend, P.L. Burns, A.B. Holmes, Light-emitting diodes based on conjugated polymers. *Nature* **347**, 539–541 (1990)
5. L. Lafferentz, F. Ample, H. Yu, S. Hecht, C. Joachim, L. Grill, Conductance of a single conjugated polymer as a continuous function of its length. *Science* **323**, 1193–1197 (2009)
6. P.F. Barbara, A.J. Gesquiere, S.J. Park, J.L. Young, Single-molecule spectroscopy of conjugated polymers. *Acc. Chem. Res.* **38**, 602–610 (2005)
7. D.A. Vanden Bout, W.T. Yip, D. Hu, D.K. Fu, T.M. Swager, P.F. Barbara, Discrete intensity jumps and intramolecular electronic energy transfer in the spectroscopy of single conjugated polymer molecules. *Science* **277**, 1074–1077 (1997)
8. B.H. Robinson, N.C. Seeman, *Protein Eng.* **1**, 295–300 (1987)
9. K.V. Gothelf, R.S. Brown, A modular approach to DNA-programmed self-assembly of macromolecular nanostructures. *Chem. A Eur. J.* **11**, 1062–1069 (2005)
10. M. Madsen, K.V. Gothelf, *Chemistries for DNA nanotechnology*. *Chem. Rev.* **119**, 6384–6458 (2019)
11. C. Joachim, J.K. Gimzewski, A. Aviram, Electronics using hybrid-molecular and mono-molecular devices. *Nature* **408**, 541–548 (2000)
12. J. Minshull, W.P. Stemmer, D.S. Wilson, J.W. Szostak, Y. Xu, N.B. Karalkar, E.T. Kool, P. Luo, J.C. Leitzel, Z.-Y.J. Zhan, D.G. Lynn, M.K. Herrlein, J.S. Nelson, R.L. Letsinger, R.K. Bruick, P.E. Dawson, S.B.H. Kent, N. Usman, G.F. Joyce, Y. Gat, L.E. Orgel, A. Luther, R. Brandsch, The generality of DNA-templated synthesis as a basis for evolving non-natural small molecules. *Curr. Opin. Chem. Biol.* **4**, 6961–6963 (2000)
13. S.M. Waybright, C.P. Singleton, K. Wachter, C.J. Murphy, U.H.F. Bunz, Oligonucleotide-directed assembly of materials: defined oligomers. *J. Am. Chem. Soc.* **123**, 1828–1833 (2001)
14. J.L. Czapinski, T.L. Sheppard, Nucleic acid template-directed assembly of metallosalen-DNA conjugates. *J. Am. Chem. Soc.* **123**, 8618–8619 (2001)
15. K.V. Gothelf, A. Thomsen, M. Nielsen, E. Cló, R.S. Brown, Modular DNA-programmed assembly of linear and branched conjugated nanostructures. *J. Am. Chem. Soc.* **126**, 1044–1046 (2004)
16. M. Nielsen, A.H. Thomsen, E. Cló, F. Kirpekar, K. Gothelf, V: Synthesis of linear and Tripoidal Oligo(phenylene ethynylene)-based building blocks for application in modular DNA-programmed assembly. *J. Org. Chem.* **69**, 2240–2250 (2004)
17. M. Nielsen, V. Dauksaite, J. Kjems, K. Gothelf, V: DNA-directed coupling of organic modules by multiple parallel reductive animations and subsequent cleavage of selected DNA sequences. *Bioconjug. Chem.* **16**, 981–985 (2005)
18. R.S. Brown, M. Nielsen, K.V. Gothelf, Self-assembly of aluminium-salen coupled nanostructures from encoded modules with cleavable disulfide DNA-linkers. *Chem. Commun.* **4**, 1464–1465 (2004)

19. P. Blakskjær, K.V. Gothelf, Synthesis of an elongated linear oligo(phenylene ethynylene)-based building block for application in DNA-programmed assembly. *Org. Biomol. Chem.* **4**, 3442–3447 (2006)
20. C.S. Andersen, H. Yan, K.V. Gothelf, Bridging one helical turn in double-stranded DNA by templated dimerization of molecular rods. *Angew. Chemie Int. Ed.* **47**, 5569–5572 (2008)
21. C.S. Andersen, M.M. Knudsen, R. Chhabra, Y. Liu, H. Yan, K.V. Gothelf, Distance dependent interhelical couplings of organic rods incorporated in DNA 4-helix bundles. *Bioconjug. Chem.* **20**, 1538–1546 (2009)
22. F. Kilibarda, A. Strobel, T. Sendler, M. Wieser, M. Mortensen, J. Brender, T. Manfred, H. Jochen, K. Elke, S. Sibylle, V.G. Artur, F. Kilibarda, A. Strobel, T. Sendler, M. Wieser, P.M. Helm, J. Kerbusch, A. Erbe, Single-molecule doping : conductance changed by transition metal centers in salen molecules. *Adv. Electron. Mater.* (2021)
23. J.B. Ravnsbaek, M.F. Jacobsen, C.B. Rosen, N.V. Voigt, K.V. Gothelf, DNA-programmed Glaser-Eglinton reactions for the synthesis of conjugated molecular wires. *Angew. Chemie Int. Ed.* **50**, 10851–10854 (2011)
24. P.J. Milnes, M.L. McKee, J. Bath, L. Song, E. Stulz, A.J. Turberfield, R.K. O'Reilly, Sequence-specific synthesis of macromolecules using DNA-templated chemistry. *Chem. Commun.* **48**, 5614–5616 (2012)
25. M.L. McKee, P.J. Milnes, J. Bath, E. Stulz, A.J. Turberfield, R.K. O'Reilly, Multistep DNA-templated reactions for the synthesis of functional sequence controlled oligomers. *Angew. Chemie - Int. Ed.* **49**, 7948–7951 (2010)
26. M.L. McKee, P.J. Milnes, J. Bath, E. Stulz, R.K. O'Reilly, A.J. Turberfield, Programmable one-pot multistep organic synthesis using DNA junctions. *J. Am. Chem. Soc.* **134**, 1446–1449 (2012)
27. P.W.K. Rothmund, Folding DNA to create nanoscale shapes and patterns. *Nature* **440**, 297–302 (2006)
28. P.O. Morin, T. Bura, M. Leclerc, Realizing the full potential of conjugated polymers: innovation in polymer synthesis. *Mater. Horiz.* **3**, 11–20 (2016)
29. J.B. Knudsen, L. Liu, A.L.B. Kodal, M. Madsen, Q. Li, J. Song, J.B. Woehrstein, S.F.J. Wickham, M.T. Strauss, F. Schueder, J. Vinther, A. Krissanaprasit, D. Gudnason, A.A.A. Smith, R. Ogaki, A.N. Zelikin, F. Besenbacher, V. Birkedal, P. Yin, W.M. Shih, R. Jungmann, M. Dong, K.V. Gothelf, Routing of individual polymers in designed patterns. *Nat. Nanotechnol.* **10**, 892–898 (2015)
30. S.J. Jethwa, M. Madsen, J.B. Knudsen, L. Lammich, K.V. Gothelf, T.R. Linderoth, Revealing the structural detail of individual polymers using a combination of electrospray. *Chem. Commun.* **53**, 1168–1171 (2017)
31. D. Gudnason, M. Madsen, A. Krissanaprasit, K.V. Gothelf, V. Birkedal, Controlled aggregation of DNA functionalized. *Chem. Commun.* **1**, 1–4 (2018)
32. S.F.J. Wickham, A. Auer, J. Min, N. Ponnuswamy, J.B. Woehrstein, F. Schueder, M.T. Strauss, J. Schnitzbauer, B. Nathwani, Z. Zhao, S.D. Perrault, J. Hahn, S. Lee, M.M. Bastings, S.W. Helmig, A.L. Kodal, P. Yin, R. Jungmann, W.M. Shih, Complex multicomponent patterns rendered on a 3D DNA-barrel pegboard. *Nat. Commun.* **11**, 1–10 (2020)
33. A. Krissanaprasit, M. Madsen, J.B. Knudsen, D. Gudnason, W. Surareungchai, V. Birkedal, K.V. Gothelf, Programmed switching of single polymer conformation on DNA Origami. *ACS Nano* **10**, 2243–2250 (2016)
34. M. Madsen, R.S. Christensen, A. Krissanaprasit, M.R. Bakke, C.F. Riber, K.S. Nielsen, A.N. Zelikin, K.V. Gothelf, Preparation, single-molecule manipulation, and energy transfer investigation of a Polyfluorene-graft-DNA polymer. *Chem. A Eur. J.* **23**, 10511–10515 (2017)
35. S. Kundu, A. Patra, Nanoscale strategies for light harvesting. *Chem. Rev.* **117**, 712–757 (2017)
36. M. Kownacki, S.M. Langenegger, S.X. Liu, R. Häner, Integrating DNA photonic wires into light-harvesting supramolecular polymers. *Angew. Chemie Int. Ed.* **58**, 751–755 (2019)
37. M. Madsen, M.R. Bakke, D.A. Gudnason, A.F. Sandahl, R.A. Hansen, J.B. Knudsen, A.L.B. Kodal, V. Birkedal, K.V. Gothelf, A single molecule Polyphenylene-Vinylene photonic wire. *ACS Nano* **15**, 9404–9411 (2021)

38. P.K. Lo, H.F. Sleiman, Nucleobase-templated polymerization: copying the chain length and polydispersity of living polymers into conjugated polymers. *J. Am. Chem. Soc.* **131**, 4182–4183 (2009)
39. J. Zessin, F. Fischer, A. Heerwig, A. Kick, S. Boye, M. Stamm, A. Kiriy, M. Mertig, Tunable fluorescence of a semiconducting polythiophene positioned on DNA Origami. *Nano Lett.* **17**, 5163–5170 (2017)
40. X. Wang, R. Sha, M. Kristiansen, C. Hernandez, Y. Hao, C. Mao, J.W. Canary, N.C. Seeman, An organic semiconductor organized into 3D DNA arrays by “bottom-up” rational design. *Angew. Chemie Int. Ed.* **56**, 6445–6448 (2017)
41. X. Wang, C. Li, D. Niu, R. Sha, N.C. Seeman, J.W. Canary, Construction of a DNA Origami based molecular electro-optical modulator. *Nano Lett.* **18**, 2112–2115 (2018)
42. H.T. Maune, S.P. Han, R.D. Barish, M. Bockrath, W.A.G. Iii, P.W.K. Rothmund, E. Winfree, Self-assembly of carbon nanotubes into two-dimensional geometries using DNA origami templates. *Nat. Nanotechnol.* **5**, 61–66 (2010)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Organizing Charge Flow with DNA



David J. F. Walker, Eric R. Szmuc, and Andrew D. Ellington

Abstract The seminal recognition by Ned Seeman that DNA could be programmed via base-pairing to form higher order structures is well known. What may have been partially forgotten is one of Dr. Seeman's strong motivations for forming precise and programmable nanostructures was to create nanoelectronic devices. This motivation is particularly apt given that modern electronic devices require precision positioning of conductive elements to modulate and control electronic properties, and that such positioning is inherently limited by the scaling of photoresist technologies: DNA may literally be one of the few ways to make devices smaller (Liddle and Gallatin in *Nanoscale* 3:2679–2688 [1]). As with many other insights regarding DNA at the nanoscale, Ned Seeman recognized the possibilities of DNA-templated electronic devices as early as 1987 (Robinson and Seeman in *Protein Eng.* 1:295–300 [2]). As of 2002, Braun's group attempted to develop methods for lithography that involved metalating DNA (Keren et al. in *Science* 297:72–75 [3]). However, this instance involved linear, double-stranded DNA, in which portions were separated using RecA, and thus, the overall complexity of the lithography was limited. Since then, the extraordinary control afforded by DNA nanotechnology has provided equally interesting opportunities for creating complex electronic circuitry, either via turning DNA into an electronic device itself (Gates et al. in *Crit. Rev. Anal. Chem.* 44:354–370 [4]), or by having DNA organize other materials (Hu and Niemeyer in *Adv. Mat.* 31(26), [5]) that can be electronic devices (Dai et al. in *Nano Lett.* 20:5604–5615 [6]).

1 Origami's Rise

While Ned Seeman can be almost uniquely attributed with the idea that DNA could be designed to assemble into higher order supramolecular structures [7], the means by which these structures would be created has varied over the years [8, 9]. Early efforts to generate non-extensible structures (such as the Olympian Borromean rings [10])

D. J. F. Walker · E. R. Szmuc · A. D. Ellington (✉)
Department of Molecular Biosciences, University of Texas, Austin, TX, USA
e-mail: ellingtonlab@gmail.com

© The Author(s) 2023
N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*,
Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_8

were followed by the creation of rigid, modular DNA elements based on more rigid crossover structures resembling Holliday junctions (the so-called double crossover or DX tile; Fig. 1a [11]) that could be assembled as building blocks into larger architectures [12] (Fig. 1b). Attempts to precisely control architecture from the atomic level upwards were further encouraged by Rothemund's discovery that short oligonucleotide "staples" could be used to fold longer DNAs into defined "origami" shapes [13] (Fig. 1c). This realization has since exploded into a huge range of architectures, both two- and three-dimensional [14], marvelous artistic endeavors [15], and a range of potential application areas for origami [16]. Further extending the departure from atomic level control, there is even tantalizing evidence that extensible three-dimensional crystal structures can arise, albeit somewhat fortuitously, allowing enhanced engineering of DNA lattice dimensions [17]. These progressive advances in atomic scale control over complex structure now provide intriguing opportunities for configuring electronics.

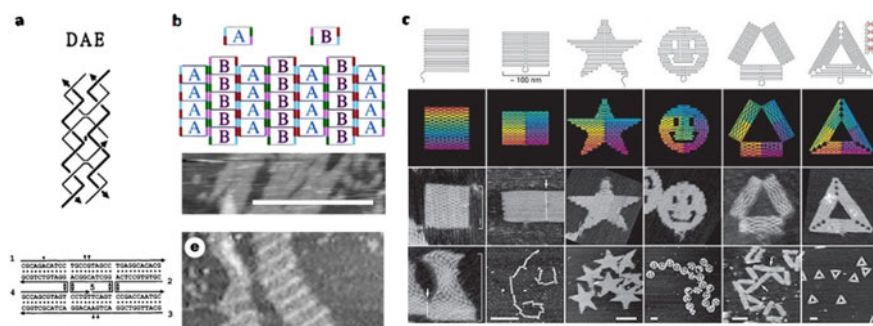


Fig. 1 Progression of DNA origami shapes through technology. **a** Seeman's double-crossover [18] junction, one of five original designs. Self-assembly occurs via a mechanism similar to Holliday junction formation, but these synthetic molecules constitute a new class of DNA structures. Adapted with permission from [11]. Copyright [11] American Chemical Society. **b** Seeman's group self-assembling DNA origami composed of repeating two DNA units, A and B, which form into origami strips with a 33 nm periodicity, as measured by AFM. Scale bar is 300 nm. Adapted by permission [12]. Copyright 1998, Springer **c** since then, DNA origami has become more complex and controlled. Rothemund's utilized computer-aided design and DNA staples to fabricate advanced shapes, including (from left to right); squares, rectangles, stars, smiley faces, triangles with rectangular domains, and trapezoidal domains with bridges between them. Color indicates the base-pair index along the folding path; red is the 1st base, purple the 7,000th. Top row of the AFM images is $165 \mu\text{m} \times 165 \mu\text{m}$, and lower row scale bars are 100 nm, with the exception of the rectangle, which is $1 \mu\text{m}$. Adapted by permission [13], Copyright 2006, Springer

2 Making DNA Nanostructures Conductive Through Metallization

In order to make DNA nanostructures into electronic devices, one approach is to determine how to make DNA conductive. While there is some evidence that DNA itself can be a conductive material [19, 20], its conductivity is relatively low [21]. To solve this problem, nucleic acids can interact with metals (or be modified to interact with metals), and if completely metalated can readily pass current. To this end, Yan et al. constructed nanogrid and nanoribbon lattices from four arm junctions and used a two-step procedure involving aldehyde derivatization of the DNA, silver seeding, and then complete silver metalation to create conductive nanowires [22] (Fig. 2a). The metallized nanowires showed a linear-ohmic current–voltage (I - V) profile using a two-terminal setup. The bulk resistivities of the nanoribbon structures were $2.4 \times 10^{-6} \Omega\cdot\text{m}$, roughly 100-fold more than polycrystalline silver ($1.6 \times 10^{-8} \Omega\cdot\text{m}$). The higher resistance could be due to the difficulty in calculating the true unit cross-section of the silver, granularity in the silver structure, or low densities of the metals following deposition. Nonetheless, this was one of the first demonstrations that organized DNA assemblies might eventually prove useful for the creation of programmable electronic devices. Following up this feat, the LaBean lab then created and metalated nanotubes (Fig. 2b) [23] based on triple-crossover tiles (TX, previously developed in collaboration with the Seeman lab [24]). These structures were found to have around a tenfold higher resistance compared to the nanoribbon strategy, however, still showed the ohmic behavior of silver expected of a conductive metal structure.

In an exciting recent result, Shani et al. became the first to report superconducting nanowires based on metalated DNA [25]. Niobium nitride, a material known to become a superconductor below 16 K, was deposited onto the surface of DNA using magnetron sputtering (Fig. 2c). The coated DNA exhibited a superconducting transition at approximately 5 K, ascribed to thermally activated phase slips, with further improvements in conductivity with the onset of quantum phase slips at 3.7 K. The DNA-templated superconductor exhibited maximal conductivity at the lowest measured temperature of 2.2 K and also showed a large negative magnetoresistance, further indicative of superconducting properties.

Conductivity alone is only one aspect of an electronic device, and thus, there is a continuing need to direct particular electron flows. The irregular nature of DNA origami provides opportunities for creating virtually any pattern, and the Harb and Wooley labs in turn metalated [26] and demonstrated the conductivity [27] of origami, including plating branched structures (Fig. 2d). Of course, mere patterning is not necessarily sufficient to create functional electronic junctions. Strategies for diversifying electronic interfaces include hybridization of DNA-conjugated gold nanoparticles to allow site-specific seeding [28] and organic masking to deposit different metals [29]. Another approach has been to sacrifice continuity of charge transport in favor of creating specific metalated structures side-by-side on “nanoscale printed circuit boards” [30]. The characterization of conductance on C-shaped origami nanowires

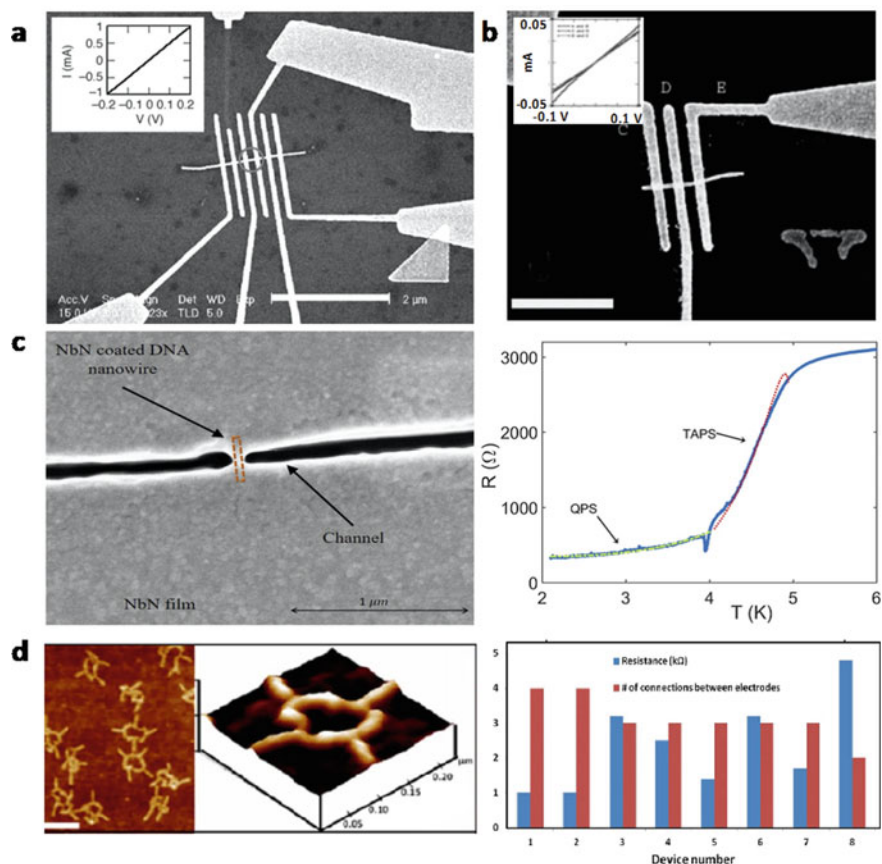


Fig. 2 Making DNA electrically conductive via metallization. **a** the first metallized DNA origami structure. LaBean's group was the first to achieve this milestone. DNA ribbons composed by 4×4 DNA tiles were silver seeded, and the conductivity was measured using current–voltage spectroscopy (a inset) scale bar $2 \mu\text{m}$. Reprinted/adapted by permission from [22]. Reprinted with permission from AAAS. **b** this was followed up by the same group using silver metallated DNA nanotubes instead of their previous DNA ribbons. The conductivity (b inset) was calculated to be tenfold lower than the ribbons. Adapted from [23]. Copyright 2004, National Academy of Sciences. **c** the first superconducting metallated DNA nanowire was published by Shani et al. The left image shows the HR-SEM of the niobium nitride coated DNA nanowire suspended on a black channel. On the right, it is the resistance measurements as a function of the temperature, with special reference to the thermally activated phase slips (TAPS) and quantum phase slips (QPS), associated with superconductivity. Adapted from [25] with permission, CC BY 4.0. **d** more complex DNA origami shapes can also be metallated and still show electrical conductivity. The left image shows the AFM images of the DNA origami CC structures with the corresponding histogram of gold-metallated resistance values for different devices. The number of metallated CC connections between each electrode was counted (red) and the total resistance calculated (blue) [32]. Adapted with permission from [27]. Copyright 2013, American Chemical Society

has indicated a variety of modes of charge transport, including hopping, thermionic, and tunneling mechanisms [31].

Despite the wealth of shapes and patterns afforded by DNA, a variety of methods for casting and plating, numerous possible charge transport mechanisms, and nanoscale electronic devices have for the most part not yet emerged from attempts to generating conductive DNA. This is in part because of the difficulties in actually forming the equivalent of atomic scale bandgap p-n junctions that are key to all electronics (and hauntingly were also key to Seeman's original dreams for self-assembled memories devices [2]).

While these difficulties may yet be overcome via even more refined and specific methods for metal placement and coating on DNA, an alternative approach has arisen in which the key advantage of nanoscale placement, rather than nanoscale patterning, is emphasized.

3 Decorating Origami

Moore's law is of course an observation, rather than a requirement, and suggests that the number of transistors within an integrated circuit will double approximately every two years. As we reach the limitations of lithography methods to achieve ever-increasing circuit densities for advanced computation, alternative fabrication techniques are being explored. In particular, DNA nanotechnology has the theoretical ability to pattern semiconducting elements with sub-nm resolution (since the width of ssDNA is ca. 0.9 nm) and could potentially be used in moderate throughput for the bottom-up, self-assembled fabrication of semiconducting chips.

3.1 DNA Scaffolding for Conductive Metals

While it has proven possible to make origami itself conductive, origami can also be used as an armature or scaffold for identifying, sorting, and positioning conductive elements. This has been achieved with a variety of DNA-material hybrids, including gold nanoparticles [33, 34], nanorods [35, 36], and quantum dots [37]. These hybrids can be further organized into higher order conductive structures. For example, Chad Mirkin's group created spherical nucleic acids (SNA) [33], and three-dimensional nucleic acid nanostructures composed of a nanoparticle core (commonly gold) densely functionalized with DNA. By functionalizing SNAs with DNA sequences that could recognize planar nanoparticle clusters, Oleg Gangs' group created electronically conductive stacked supramolecular assemblies [38] (Fig. 3a). These assemblies were capable of forming filament-like, pillar structures, comprised of alternating SNAs and planar nanoparticles. By controlling the number, position, size, and composition of the nanoparticles, different three-dimensional pillar architectures could be produced with a variety of physical and electrical transport properties.

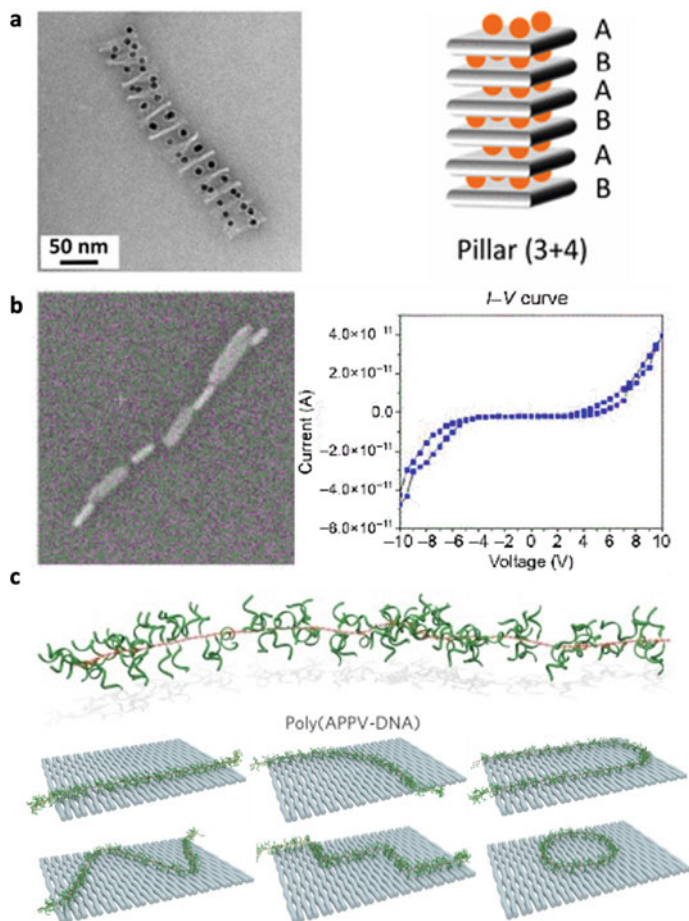


Fig. 3 Using DNA origami to position different materials into high-ordered architectures. **a** Tian et al. used SNA's to control the assembly of planar nanoparticle clusters into electrically conductive pillars. The left image shows the SEM of these structures composed of two different clusters. Cluster A has 3 SNAs, and cluster B has 4 SNAs. The right image shows the assemble scheme for the two alternating clusters (A&B). Adapted with permission from [38]. Copyright 2017, American Chemical Society. **b** the first publication, by Aryal et al., showing electrically connected metal–semiconductor junctions. The left SEM image shows the smaller gold nanorods seeded onto the DNA origami deposited onto silicon oxide wafers. After which, the thinner CTAB-coated tellurium nanorods were deposited within the gaps. The nanorods were electroless plated with gold to electrically connect the structures. Current–voltage spectroscopy (right) exhibited a diode response consistent with a Schottky junction. Adapted by permission [39], Copyright 2020, Springer. **c** This is the first publication showing the placement of polymers into two- and three-dimensional architectures with DNA origami (Knudsen et al. [40]). The top image shows the APPV polymer conjugated to the ssDNA staples (extending from the phenylene groups). Using DNA origami placed on a substrate, the DNA staples route the polymer into specific designed architectures. The six smaller illustrations show the origami design and the polymer structures obtained. The original paper also contains AFM topography data showing correct routing. Adapted by permission [40], Copyright 2015, Springer

DNA origami has also been used as a scaffold to assemble single metal wires composed of contiguous electrically connected metal–semiconductor junctions. In 2020, Aryal et al. [39] deposited DNA origami onto a silicon wafer then positioned three DNA-coated gold nanorods (100 nm × 20 nm) along the DNA with increasing gaps between the nanorods (Fig. 3b). Next, tellurium nanorods were positioned into the gaps between the gold nanorods via electrostatic interactions. Finally, the gaps between the nanorods were filled via electroless gold plating, creating a nanoscale wire composed of alternating Au-Te-Au junctions. When subjected to current–voltage (I-V) electrical characterization, the response showed a non-linear-ohmic nature consistent with Schottky junction (aka Schottky barrier diode) properties. This was the first demonstration of how DNA origami can be used to assemble nanoscale metal–semiconductor junctions.

3.2 DNA Scaffolds for Conductive Polymers

An alternative to introducing metals into DNA origami materials is the incorporation of conductive polymers into the origami. This was originally accomplished by the Gothelf and Dong labs; a conjugated paraphenylene vinylene (APPV) brush polymer that contained a nine nucleotide single-stranded DNA staple was synthesized [40]. To program the placement of the APPV onto a silicon oxide substrate and resultant architectures, DNA origami that could hybridize to the oligonucleotide attached to the polymer was constructed. Using this method, the authors were able to direct the placement and orientation of the APPV-DNA hybrid to create linear structures, 90° curves, U-shape, staircases, and circular designs (Fig. 3c). Ultimately, three-dimensional cylindrical DNA origami structures were constructed, and APPV was routed around. While in this implementation, no electrical conductivity measurements were performed, but the surface potential of the APPV-DNA polymer chain was measured to be -130 mV, implying a charge transfer higher than the underlying silicon oxide. These results provide an excellent proof-of-principle for extension to more conductive polymers and demonstrate the ability to create very compact three-dimensional structures capable of organizing conductive polymers into molecular-scale electronics, with the polymer providing a link to softer, flexible, and perhaps more biocompatible materials for biomedical engineering applications.

3.3 DNA Scaffolds for Carbon Nanotubes

The DNA-based placement of conductive and now semi-conductive metals is powerful, but electronic applications remain elusive because of a need for scaling, an application that may yet arise via the placement of carbon nanotubes (CNTs); though in particular, controlling the placement and positioning of CNTs in three-dimensions is incredibly complicated. Structural precision can potentially be achieved without

DNA using thin-film approaches, but these are prone to assembly defects such as crossing, bundling, and non-conforming pitch [41], parameters essential for creating a precisely ordered three-dimensional CNT architecture for applications in integrated circuits. Utilization of DNA nanostructures as templates will allow greater precision in the nanofabrication process. If architectural problems are overcome, CNT-based computers can potentially outperform the best silicon devices [42], while being at least an order of magnitude more efficient [43]. This efficiency is incredibly important in regard to Dennard Scaling, which states that as a transistor reduces in size, the power density stays constant. This reduction in power usage allows manufacturers to increase the operating frequency of microprocessors and boost performance. Over the past 5 years, there has not been a significant increase in operating frequency, with IC manufacturers resorting to multi-core threading to overcome performance restrictions. Using new highly efficient materials such as CNTs would enable manufacturers to go beyond silicon-based ICs and increase operating frequency with less impact on power requirements.

In 2003, Zheng et al. [44] reported that ssDNA can strongly interact with CNTs to form a stable complex; this advance most importantly allowed CNTs to be solubilized in aqueous solutions. The DNA-CNT hybrid architecture (wrapping) was found to be dependent on a short (10–45 nucleotide) GT-rich sequence which could form a two-dimensional sheet structure. Furthermore, it has been discovered that the chirality of the DNA around the CNT is dependent on the handedness of the helicity of the DNA [45]. Remarkably, GT-rich elements can selectively bind all 12 of the major chiral semiconducting CNT species [46]. The wrapping of the negatively charged DNA around the CNTs was also found to promote the selective separation of chirally pure CNTs via a positively charged anion exchange resin [46], potentially solving one of the major application challenges for CNTs, since chiral structures cannot be readily synthesized but have superior electrical conductivity and semiconducting properties [47].

These results set the stage for using DNA origami to fabricate highly specific and orientated nanostructures with CNTs. An inherent problem with using CNTs is the difficulty of organizing them into specific orientations. This complication was nicely remedied by Erik Winfree's group, who used DNA origami with "hook-binding domains" to perpendicularly place two CNTs into a specific two-dimensional orientation [48] (Fig. 4a). The alignment of the CNTs into cross-junctions positioned them with 6 nm resolution and led to a stable field-effect transistor (FET)-like behavior, two firsts. This was quickly followed by the Goddard group using small-structured DNA linkers to establish highly dense parallel CNT arrays that could be self-assembled on the surface of mica (Fig. 4b). By tuning the length of the DNA linker, the CNT pitch could be tuned accordingly, with distances ranging from <3 nm to >20 nm [49]. Using a different DNA origami architecture that resulted in parallel placements, the Norton group utilized larger "blocks" of DNA origami that acted as linkers to orient two CNTs onto a one-dimensional origami construct, separating CNTs of ~100 nm length by more than 500 nm (Fig. 4c) [50]. Alternative methods of assembling CNTs on DNA origami templates using streptavidin–biotin interactions have also proven successful in generating defined cross-junctions [51].

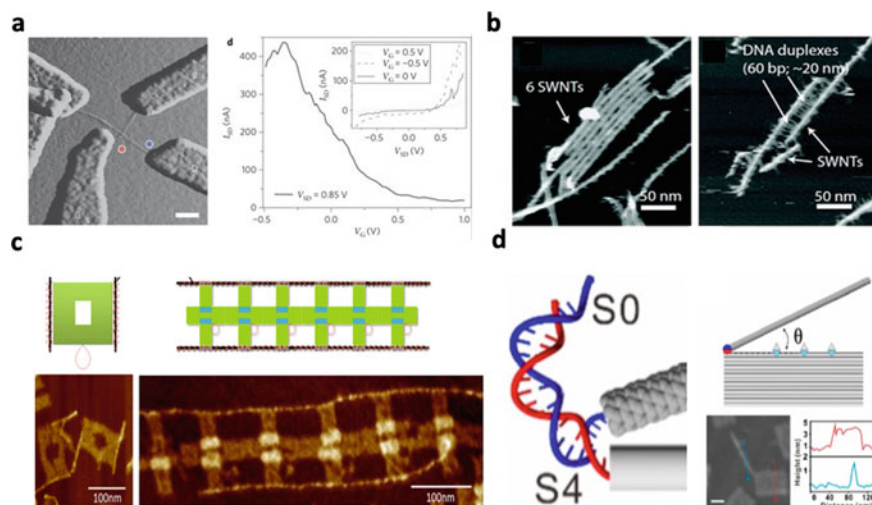


Fig. 4 Using DNA origami to precisely position CNTs into two-dimensional structures. **a** the seminal study by the Winfree group showing the first time DNA origami was used to position CNTs. By using different DNA linkers built upon a 7000 bp scaffold, the authors were able to position two CNTs into a perpendicular cross-junction (left). This architecture was capable of displaying a FET behavior (right). Adapted by permission [48], Copyright 2009, Springer. **b** following up this work, Han et al. used DNA linkers to control the pitch between the CNTs by using different sized linkers. On the left, it is six CNTs positioned with 20 bp linkers creating an array pitch of 8.5 nm, whereas on the right, the linkers are 60 bp and create a pitch of 22 nm. Adapted with permission from [49]. Copyright 2012, American Chemical Society. **c** using a different design architecture, Mangalum et al. also achieved parallelly positioned CNTs using individual “block” DNA origami structures. These blocks contained ssDNA loops which enabled the blocks to be linked together to create a scaffold to position CNTs ~100 nm apart, over a distance of 500 nm, bringing nanometer architectures close to micrometer size domains. Adapted with permission from [50]. Copyright 2013, American Chemical Society. **d** rather than relying on DNA wrapping, Pei et al. created an amide bond between the ssDNA and the ends of the CNTs (left). By using a one-point linkage, they can utilize double-stranded DNA hybridization to create origami rafts (right) which can pivot round the linkage point to positionally control the CNTs. Adapted with permission from [53]. Copyright 2019, American Chemical Society

Even more complex “Y-shape” nanostructures have been created that involve a three-way DNA junction with protruding single-stranded sequences that anneal to CNTs. These nanostructures orient the three CNTs at an approximate 120° angle to each other. Further, higher-ordered networks of each Y-shaped triplet can be assembled to create a DNA-CNT “mesh” [52]. Recently, Seeman’s group [53] has demonstrated that rather than absorbing DNA to the sides of SWCNTs via van der Waals forces, DNA can be specifically localized to the ends of SWCNT structures via conjugation chemistry between a pendant amino-group on the DNA molecule and carboxylates presented on the termini of the SWCNT. By orienting DNA to the terminals of the nanotubes, the conductive lattice structure on the sides is not impeded,

reducing interference with the electrical properties of the SWCNT. The ssDNA conjugated to the reactive end of a SWCNT can thus be used to orient multiple nanotubes via hybridization, creating precisely positioned SWCNTs within a two-dimensional DNA origami “raft” [53] (Fig. 4d). In another approach to organizing supramolecular structures, a conductive SNA nanoparticle “linker” was used for the parallel positioning of DNA-conjugated carbon nanotubes, resulting in a five-fold binding improvement over just relying on the stickiness of the nanotubes themselves [54].

3.4 Highly Ordered, Three-Dimensional DNA-CNT Arrays

Two publications, released on the same day, go a long way toward the implementation of direct biotemplated CNT nanofabrication using DNA origami “bricks.” Sun et al., [55], developed a highly scalable supramolecular assembly method, termed Spatially Hindered Integration of Nanotube Electronics (SHINE). Densely parallel, aligned arrays of CNTs were fabricated within an array of channels, achieving precise intertube spaces of 10.4 nm (Fig. 5a). Using the SHINE assembly method, Zhao et al. [56] then created a multichannel p-channel metal-oxide semiconductor field-effect transistor (FET) by fixing the DNA-templated CNTs onto a polymer-templated silicon wafer. The CNTs were fixed into position with metal bars, and then, electrodes and gate dielectrics were deposited onto the device (Fig. 5b). The relatively poor electronic performance often observed with biotemplating was remedied by removing contaminating DNA and metal ions with ultra-pure water, low-concentration H_2O_2 , and thermal annealing after the CNT fixing stage. This rinsing-after-fixing approach improved the key transport performance metrics by a factor of 10 compared to the previous biotemplated FETs. This approach highlights the advantage of using DNA origami for the organization of truly three-dimensional architectures.

4 The Future of DNA-Organized Electronics

The future of DNA-organized electronics will likely resemble its past and present but will build on substantive technical advances that fall outside the field. The key enabler, the ability to organize and ultimately design structure at the atomic scale, will be augmented by new opportunities for using novel mechanisms to mediate electron flow. In particular, it should be possible to either make DNA itself more conductive by changing its fundamental chemistry, or to better adapt biocompatible electronic materials to nanostructured DNA scaffolds.

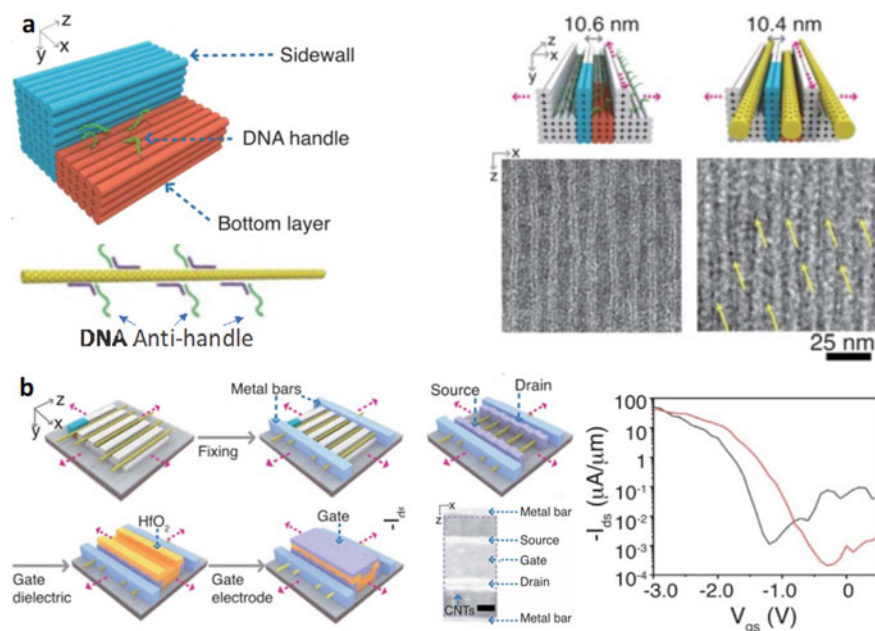


Fig. 5 Creating highly ordered CNTs in three-dimensional architectures. **a** the revolutionary spatially hindered integration of nanotube electronics (SHINE) method developed by Sun et al. DNA origami nano-bricks were assembled into three-dimensional side walls and bottom layers creating trenches in which DNA handles sit (left). The CNTs wrapped with DNA anti-handle linkers when added (right), seat themselves within the confined trenches, creating a highly parallel ordered array with a precise pitch of 10.4 nm. This pitch can be controlled by using different DNA origami sidewall thicknesses. TEM imaging shows the specific topography of the array, with the CNTs positions indicated with the yellow arrows. From [55]. Adapted with permission from AAAS **b** using the SHINE method, Zhao et al. constructed a high-performance FET. In short, the authors constructed a 10.4 nm pitch CNT array, added metal bars, source, drain, and gate dielectrics followed by rinsing to remove the DNA and any contaminating metal ions before finally adding the gate. The current–voltage curves (right) highlight the difference between not removing the DNA and thermal annealing (gray line) and after thermal annealing (red line), improving the FET performance by an order of magnitude. From [56]. Adapted with permission from AAAS

4.1 Making DNA More Electronic

While there are now extremely interesting opportunities for creating Schottky junctions between DNA-patterned and non-DNA-patterned materials, as described above, the opportunities for atomic scale control over the geometry and charge flow, as envisioned by Seeman [2] remains largely unrealized to-date. The key to further enablement may come through the control of chemistry that is afforded by DNA nanotechnology. It is relatively simple to append electronic elements (the most obvious being ferrocene) to nucleobases, allowing more precise control over where electron transfer

occurs. Further, the introduction of unnatural nucleobases offers further opportunities for atomic level placement, including non-standard base-pairs that specifically chelate metals at the Watson–Crick interface [57].

4.2 Scaffolding Biocompatible Electronic Materials

While the electronic properties of native DNA can be improved, it is still somewhat surprising that it is a conductive material at all, given the lack of obvious redox moieties available [58]. Recently, there has been a paradigm shift (entirely in the Kuhnian sense) in our understanding of how biological materials can potentially transfer electrons. The intermediacy of redox active compounds remains a key feature of bio-enabled electrochemistry and has been successfully utilized in constructing photonic DNA-organized excitonic circuits [59–62]. However, the opportunities for long distance electron transfer in the absence of such intermediacy are becoming more and more apparent. This has long been the case for tunneling between organic cofactors in cytochrome couples (for example) but has only recently been implicated for peptides and proteins that lack cofactors.

A particularly relevant example of this shift in understanding has been garnered from the study of a synthetic 29-mer peptide that forms an antiparallel coiled-coil hexamer (ACC-Hex) and has been shown to have a remarkably high electrical conductivity [63]. The mechanism of electron transport is currently under debate, since features of both ohmic electronic transport and metallic-like temperature dependence are observed, despite the fact that it contains neither extended conjugation, π -stacking, nor redox centers. A similar story seems to be unfolding in the study of electrically conductive protein nanowires (e-PNs). *Geobacter sulfurreducens*, a common anaerobic species of bacteria found in anoxic subsurface sediments, is capable of respiring via the reduction of metals, such as Fe(III). These bacteria are able to accomplish this feat by producing long thin filaments that protrude from the cell and electronically link the cell to the metal [64]. Similar to the ACC-Hex fibers, purified e-PNs display ohmic electronic transport and metallic-like temperature dependence. They also contain a high density of aromatic amino acids positioned conservatively throughout one of their constituent proteins [65]. Removal of the aromatic amino acids drastically reduces the electrical conductivity of the nanowires [64].

The potentially critical role of aromatic amino acids in electronic conduction presents a conundrum about what novel electron transport mechanisms may be available to biological polymers. Irrespective of the underlying physics, there has been a remarkable plethora of practical devices produced from the e-pili [66–68], including generating energy from environmental ambient humidity [69]. Currently, these methods involve using thin-film approaches to create devices, and as we have seen in other aspects of microelectronics, this can ultimately limit the overall device architecture and functionality. As with other electronic components, such as CNTs, the use of DNA origami now offers a truly unique opportunity to contort and position

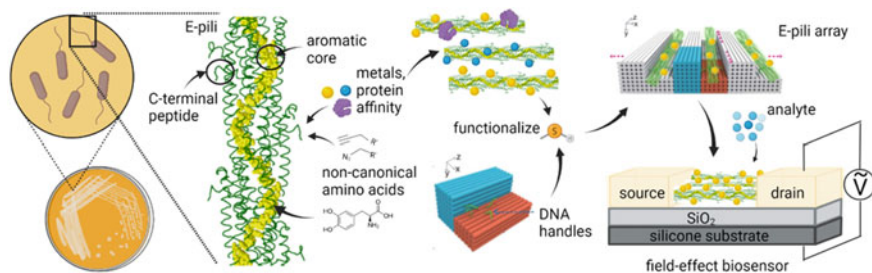


Fig. 6 e-PNs engineering to create hybrid protein: DNA electronic devices. One type of e-PNs are formed from the monomeric type-IV pilin protein of *G. sulfurreducens* which can be modified by engineering the carboxy terminal domains exposed to the environment on the surface of the nanowire. At this termini, the interactions of fused specialty peptides which can bind a multitude of analytes, including metal ions, nanoparticles proteins, or other molecules, can create interactions that will modulate electrical properties. Additionally, the inclusion of non-canonical amino acids can further increase the range of chemistries available for binding and electron transfer. Bound molecules and engineered peptides can further connect with three-dimensional origami scaffolds (the Sun et al. SHINE design is shown) to create bio-produced field-effect transistors which can be used to sense specific analytes for biosensing applications

these conductive wires into specific orientations (Fig. 6). However, the potential for developing sophisticated devices goes well beyond CNTs, since a variety of properties in the pili can be tuned via genetics: They can be functionalized with peptide domains to bind specifically to sites on DNA nanostructures, going beyond the terminal attachment of carbon nanotubes; the number and type of aromatic residues available for conductance can be modulated at will; the length and diameter of the pili can be controlled biologically. The use of DNA origami to organize protein-based nanowires into electrode arrays would represent an entirely new direction in bioelectronics, one which we believe could finally realize the potential of Dennard's and Moore's laws at the nanometer-scale.

References

1. J.A. Liddle, G.M. Gallatin, Lithography, metrology and nanomanufacturing. *Nanoscale* **3**(7), 2679–2688 (2011)
2. B.H. Robinson, N.C. Seeman, The design of a Biochip—a self-assembling molecular-scale memory device. *Protein Eng.* **1**(4), 295–300 (1987)
3. K. Keren, M. Krueger, R. Gilad, G. Ben-Yoseph, U. Sivan, E. Braun, Sequence-specific molecular lithography on single DNA molecules. *Sci.* **297**(5578), 72–75 (2002)
4. E.P. Gates, A.M. Dearden, A.T. Woolley, DNA-templated lithography and nanofabrication for the fabrication of nanoscale electronic circuitry. *Crit. Rev. Anal. Chem.* **44**(4), 354–370 (2014)
5. Y. Hu, C.M. Niemeyer, From DNA nanotechnology to material systems engineering. *Adv. Mat.* **31**(26) (2019)
6. X.P. Dai, Q. Li, A. Aldalbahi, L.H. Wang, C.H. Fan, X.G. Liu, DNA-Based Fabrication for Nanoelectronics. *Nano Lett.* **20**(8), 5604–5615 (2020)
7. N.C. Seeman, Nucleic-acid junctions and lattices. *J. Theor. Biol.* **99**(2), 237–247 (1982)

8. C.H. Liu, A.D. Ellington, DNA nanotechnology: from biology and beyond. *Nucleic Acid Nanotech.* **29**, 135–169 (2014)
9. N.C. Seeman, An overview of structural DNA Nanotechnology. *Mol. Biotechnol.* **37**(3), 246–257 (2007)
10. C.D. Mao, W.Q. Sun, N.C. Seeman, Assembly of Borromean rings from DNA. *Nature* **386**(6621), 137–138 (1997)
11. T.J. Fu, N.C. Seeman, DNA double-crossover molecules. *Biochem.* **32**(13), 3211–3220 (1993)
12. E. Winfree, F.R. Liu, L.A. Wenzler, N.C. Seeman, Design and self-assembly of two-dimensional DNA crystals. *Nature* **394**(6693), 539–544 (1998)
13. P.W.K. Rothemund, Folding DNA to create nanoscale shapes and patterns. *Nature* **440**(7082), 297–302 (2006)
14. X.H. Yan, S.J. Huang, Y. Wang, Y.Y. Tang, Y. Tian, Bottom-up self-assembly based on DNA nanotechnology. *Nanomat.* **10**(10) (2020)
15. A. Gopinath, E. Miyazono, A. Faraon, P.W.K. Rothemund, Engineering and mapping nanocavity emission via precision placement of DNA origami. *Nature* **535**(7612), 401–405 (2016)
16. A.A. Arora, C. de Silva, Beyond the smiley face: applications of structural DNA nanotechnology. *Nano Rev. Exp.* **9** (2018)
17. P.J. Paukstelis, J. Nowakowski, J.J. Birktoft, N.C. Seeman, Crystal structure of a continuous three-dimensional DNA lattice. *Chem. Biol.* **11**(8), 1119–1126 (2004)
18. C.-J. Wu, B.-H. Chu, M.-T. Weng, H.-L. Lee, Enhancement of bandwidth in a Chirped quarter-wave dielectric mirror. *J. Electromag. Waves Appl.* **23**(4), 437–447 (2012)
19. K. Wang, DNA-based single-molecule electronics: from concept to function. *J. Funct. Biomat.* **9**(1), 8 (2018)
20. G.I. Livshits, A. Stern, D. Rotem, N. Borovok, G. Eidelshstein, A. Migliore, E. Penzo, S.J. Wind, R. Di Felice, S.S. Skourtis, J.C. Cuevas, L. Gurevich, A.B. Kotlyar, D. Porath, Long-range charge transport in single G-quadruplex DNA molecules. *Nat. Nanotechnol.* **9**(12), 1040–1046 (2014)
21. C. Dekker, M.A. Ratner, Electronic properties of DNA. *Phys. World* **14**(8), 29–33 (2001)
22. H. Yan, S.H. Park, G. Finkelstein, J.H. Reif, T.H. LaBean, DNA-templated self-assembly of protein arrays and highly conductive nanowires. *Science* **301**(5641), 1882–1884 (2003)
23. D. Liu, S.H. Park, J.H. Reif, T.H. LaBean, DNA nanotubes self-assembled from triple-crossover tiles as templates for conductive nanowires. *Proc. Natl. Acad. Sci. U.S.A.* **101**(3), 717–722 (2004)
24. T.H. LaBean, H. Yan, J. Kopatsch, F.R. Liu, E. Winfree, J.H. Reif, N.C. Seeman, Construction, analysis, ligation, and self-assembly of DNA triple crossover complexes. *J. Am. Chem. Soc.* **122**(9), 1848–1860 (2000)
25. L. Shani, P. Tinnefeld, Y. Fleger, A. Sharoni, B. Y. Shapiro, A. Shaulov, O. Gang, Y. Yeshurun, DNA origami based superconducting nanowires. *AIP Adv.* **11**, 015130 (2021)
26. J.F. Liu, Y.L. Geng, E. Pound, S. Gyawali, J.R. Ashton, J. Hickey, A.T. Woolley, J.N. Harb, Metallization of branched DNA origami for nanoelectronic circuit fabrication. *ACS Nano* **5**(3), 2240–2247 (2011)
27. Y.L. Geng, A.C. Pearson, E.P. Gates, B. Uprety, R.C. Davis, J.N. Harb, A.T. Woolley, Electrically conductive gold- and copper-metallized DNA origami nanostructures. *Langmuir* **29**(10), 3482–3490 (2013)
28. A.C. Pearson, J. Liu, E. Pound, B. Uprety, A.T. Woolley, R.C. Davis, J.N. Harb, DNA Origami Metallized site specifically to form electrically conductive nanowires. *J. Phys. Chem. B* **116**(35), 10551–10560 (2012)
29. B. Uprety, E.P. Gates, Y. Geng, A.T. Woolley, J.N. Harb, Site-specific metallization of multiple metals on a single DNA origami template. *Langmuir* **30**(4), 1134–1141 (2014)
30. S.S. Jia, J.B. Wang, M. Xie, J.X. Sun, H.J. Liu, Y.N. Zhang, J. Chao, J. Li, L.H. Wang, J.P. Lin, K.V. Gothelf, C.H. Fan, Programming DNA origami patterning with non-canonical DNA-based metallization reactions. *Nat. Comm.* **10**(5597) (2019)

31. T. Bayrak, A. Martinez-Reyes, D.D.R. Arce, J. Kelling, E.C. Samano, A. Erbe, Fabrication and temperature-dependent electrical characterization of a C-shape nanowire patterned by a DNA origami. *Scientific Reports* **11**(1) (2021)
32. T. Cai, K. Han, P. Yang, Z. Zhu, M. Jiang, Y. Huang, C. Xie, Reconstruction of dynamic and reversible color change using reflection protein. *Sci. Rep.* **9**(1), 1–11 (2019)
33. C.A. Mirkin, R.C. Mucic, J.J. Storhoff, R.L. Letsinger, DNA-induced assembly of gold nanoparticles: a method for rationally organizing colloidal particles into ordered macroscopic materials. *Nature* **382**, 607–609 (1996)
34. B.Q. Ding, Z.T. Deng, H. Yan, S. Cabrini, R.N. Zuckermann, J. Bokor, Gold nanoparticle self-similar chain structure organized by DNA origami. *J. Am. Chem. Soc.* **132**(10), 3248–3249 (2010)
35. A. Kuzyk, R. Schreiber, H. Zhang, A.O. Govorov, T. Liedl, N. Liu, Reconfigurable 3D plasmonic metamolecules. *Nat. Mater.* **13**(9), 862–866 (2014)
36. P. Zhan, P.K. Dutta, P. Wang, G. Song, M. Dai, S.X. Zhao, Z.G. Wang, P. Yin, W. Zhang, B. Ding, Y. Ke, Reconfigurable three-dimensional gold nanorod plasmonic nanostructures organized on DNA origami tripod. *ACS Nano* **11**(2), 1172–1179 (2017)
37. H. Bui, C. Onodera, C. Kidwell, Y. Tan, E. Graugnard, W. Kuang, J. Lee, W.B. Knowlton, B. Yurke, W.L. Hughes, Programmable periodicity of quantum dot arrays with DNA origami nanotubes. *Nano. Lett.* **10**(9), 3367–3372 (2010)
38. C. Tian, M.A.L. Cordeiro, J. Lhermitte, H.L. Xin, L. Shani, M. Liu, C. Ma, Y. Yeshurun, D. DiMarzio, O. Gang, Supra-nanoparticle functional assemblies through programmable stacking. *ACS Nano*. **11**(7), 7036–7048 (2017)
39. B.R. Aryal, D.R. Ranasinghe, T.R. Westover, D.G. Calvopiña, R.C. Davis, J.N. Harb, A.T. Woolley, DNA origami mediated electrically connected metal—semiconductor junctions. *Nano Res.* **13**(5), 1419–1426 (2020)
40. J.B. Knudsen, L. Liu, A.L.B. Kodal, M. Madsen, Q. Li, J. Song, J.B. Woehrstein, S.F.J. Wickham, M.T. Strauss, F. Schueder, J. Vinther, A. Krissanaprasit, D. Gudnason, A.A.A. Smith, R. Ogaki, A.N. Zelikin, F. Besenbacher, V. Birkedal, P. Yin, W.M. Shih, R. Jungmann, M. Dong, K.V. Gothelf, Routing of individual polymers in designed patterns. *Nat. Nanotechnol.* **10**(10), 892–898 (2015)
41. Y. Zhou, R. Azumi, Carbon nanotube based transparent conductive films: progress, challenges, and perspectives. *Sci. Techn. Adv. Mat.* **17**(1) (2016)
42. A.D. Franklin, M. Luisier, S.J. Han, G. Tulevski, C.M. Breslin, L. Gignac, M.S. Lundstrom, W. Haensch, Sub-10 nm carbon nanotube transistor. *Nano Lett.* **12**(2), 758–762 (2012)
43. M.M. Shulaker, G. Hills, N. Patil, H. Wei, H.Y. Chen, H.S. Wong, S. Mitra, Carbon nanotube computer. *Nature* **501**(7468), 526–530 (2013)
44. M. Zheng, A. Jagota, E.D. Semke, B.A. Diner, R.S. Mclean, S.R. Lustig, R.E. Richardson, N.G. Tassi, DNA-assisted dispersion and separation of carbon nanotubes. *Nat. Mater.* **2**(5), 338–342 (2003)
45. G.H. Zerze, F.H. Stillinger, P.G. Debenedetti, The handedness of DNA assembly around carbon nanotubes is determined by the chirality of DNA. *J. Phys. Chem. B* **125**(26), 5362–5369 (2020)
46. X. Tu, S. Manohar, A. Jagota, M. Zheng, DNA sequence motifs for structure-specific recognition and separation of carbon nanotubes. *Nature* **460**(7252), 250–253 (2021)
47. M.W. Feng Yang, D. Zhang, J. Yang, M. Zheng, Y. Li, Chirality pure carbon nanotubes: growth, sorting, and characterization. *Chem. Rev.* **120**(5), 2693–275 (2020)
48. H.T. Maune, S.P. Han, R.D. Barish, M. Bockrath, W.A. Goddard III., P.W.K. Rothmund, E. Winfree, Self-assembly of carbon nanotubes into two-dimensional geometries using DNA origami templates. *Nat. Nanotechnol.* **5**(1), 61–66 (2009)
49. S. Han, M.H.T. Maune, R.D. Barish, M. Bockrath, W.A. Goddard III, DNA-linker-induced surface assembly of ultra dense parallel single walled carbon nanotube arrays. *Nano Lett.* **12**(3) (2012)
50. A. Mangalum, M. Rahman, M. L. Norton, Site-specific immobilization of single-walled carbon nanotubes onto single and one-dimensional DNA origami. *J. Am. Chem. Soc.* **135**(7) (2013)

51. A.P. Eskelinen, A. Kuzyk, T.K. Kaltiainenaho, M.Y. Timmermans, A.G. Nasibulin, E.I. Kaipinen, P. Törmä, Assembly of single-walled carbon nanotubes on DNA-origami templates through streptavidin-biotin interaction. *Small* **7**(6) (2011)
52. B. Oruc, S. Celik, S. H. Soytaş, H. Unal, DNA directed self-assembly of single walled carbon nanotubes into three-way junction nanostructures. *ACS Omega* **3**(4) (2018)
53. H. Pei, R. Sha, X. Wang, M. Zheng, C. Fan, J.W. Canary, N.C. Seeman, Organizing end-site-specific SWCNTs in specific loci using DNA. *J. Am. Chem. Soc.* **141**(30), 11923–11928 (2019)
54. Y. Zhang, X. Mao, F. Li, M. Li, X. Jing, Z. Ge, L. Wang, K. Liu, H. Zhang, C. Fan, X. Zuo, Nanoparticle-assisted alignment of carbon nanotubes on DNA origami. *Angew Chem. Int. Ed. Engl.* **59**(12), 4892–4896 (2020)
55. W. Sun, J. Shen, Z. Zhao, N. Arellano, C. Rettner, J. Tang, T. Cao, Z. Zhou, T. Ta, J.K. Streit, J.A. Fagan, T. Schaus, M. Zheng, S.J. Han, W.M. Shih, H.T. Maune, P. Yin, Precise pitch-scaling of carbon nanotube arrays within three-dimensional DNA nanotrenches. *Sci.* **368**(6493), 874–877 (2020)
56. M. Zhao, Y. Chen, K. Wang, Z. Zhang, J.K. Streit, J.A. Fagan, J. Tang, M. Zheng, C. Yang, Z. Zhu, W. Sun, DNA-directed nanofabrication of high-performance carbon nanotube field-effect transistors. *Sci.* **368**(6493), 878–881 (2020)
57. Y. Takezawa, J. Muller, M. Shionoya, Artificial DNA base pairing mediated by diverse metal ions. *Chem. Lett.* **46**(5), 622–633 (2017)
58. D.N. Beratan, Why are DNA and protein electron transfer so different? *Annu. Rev. Phys. Chem.* **70**(1), 71–97 (2019)
59. É. Boulais, N.P.D. Sawaya, R. Veneziano, A. Andreoni, J.L. Banal, T. Kondo, S. Mandal, S. Lin, G.S. Schlau-Cohen, N.W. Woodbury, H. Yan, A. Aspuru-Guzik, M. Bathe, Programmed coherent coupling in a synthetic DNA-based excitonic circuit. *Nature Mat.* **17**(2) (2018)
60. E. Graugnard, D.L. Kellis, H. Bui, S. Barnes, W. Kuang, J. Lee, W.L. Hughes, W.B. Knowlton, B. Yurke, DNA-controlled excitonic switches. *Nano Lett.* **12**(4) (2012)
61. C.B.L. Cannon, D.L. Kellis, L.K. Patten, P.H. Davis, J. Lee, E. Graugnard, B. Yurke, W.B. Knowlton, Coherent exciton delocalization in a two-state DNA-templated dye aggregate system. *J. Phys. Chem. A* **121**(37) (2017)
62. P.D. Cunningham, S.A. Díaz, B. Yurke, I.L. Medintz, J.S. Melinger, Delocalized two-exciton states in DNA scaffolded cyanine dimers. *J. Phys. Chem. B* **124**(37) (2020)
63. N. Ing, R.K. Spencer, S.H. Luong, H.D. Nguyen, A.I. Hochbaum, Electronic conductivity in biomimetic α -helical peptide nanofibers and gels. *ACS Nano.* **12**(3), 2652–2661 (2018)
64. D.R. Lovley, D.J.F. Walker, Geobacter protein nanowires. *Front. Microbiol.* **10**, 2078 (2019)
65. D.J.F. Walker, R.Y. Adhikari, D.E. Holmes, J.E. Ward, T.L. Woodard, K.P. Nevin, D.R. Lovley, Electrically conductive pili from pilin genes of phylogenetically diverse microorganisms. *ISME J.* **12**(1), 48–58 (2017)
66. T. Fu, X. Liu, H. Gao, J.E. Ward, X. Liu, B. Yin, Z. Wang, Y. Zhuo, D.J.F. Walker, J.J. Yang, J. Chen, D.R. Lovley, J. Yao, Bioinspired bio-voltage memristors. *Nat. Commun.* **11**(1), 1–10 (2020)
67. A.F. Smith, X. Liu, T.L. Woodard, T. Fu, T. Emrick, J.M. Jiménez, D.R. Lovley, J. Yao, Bioelectronic protein nanowire sensors for ammonia detection. *Nano Res.* **13**(5), 1479–1484 (2020)
68. X. Liu, T. Fu, J. Ward, H. Gao, B. Yin, T. Woodard, D.R. Lovley, J. Yao, Multifunctional Protein Nanowire Humidity Sensors for Green Wearable Electronics. *Adv. Elect. Mat.* **6**(9), 2000721 (2020)
69. X. Liu, H. Gao, J.E. Ward, X. Liu, B. Yin, T. Fu, J. Chen, D.R. Lovley, J. Yao, Power generation from ambient humidity using protein nanowires. *Nat.* **578**(7796), 550–554 (2020)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



DNA Assembly of Dye Aggregates—A Possible Path to Quantum Computing



Bernard Yurke

Abstract DNA-based self-assembly enables the programmable arrangement of matter on a molecular scale. It holds promise as a means with which to fabricate high technology products. DNA-based self-assembly has been used to arrange chromophores (dye molecules) covalently linked to DNA to form Förster resonant energy transfer and exciton-based devices. Here we explore the possibility of making coherent exciton information processing devices, including quantum computers. The focus will be on describing the chromophore arrangements needed to implement a complete set of gates that would enable universal quantum computation.

1 Introduction

Already in the earliest days of DNA nanotechnology, Richardson and Seeman imagined using DNA nanotechnology to fabricate electronic information storage and processing devices, such as three-dimensional memories in which the wires and switching elements were of molecular scale [1]. My own entry into this field, in the late 1990s, at Bell Laboratories, was motivated by this vision. The tools of DNA nanotechnology have undergone considerable development since those early days [2, 3], but its use in the construction of electronic or photonic information processing systems is still in its infancy. One of the more developed areas is the use of DNA nanotechnology to arrange metallic nanoparticles, quantum dots and dyes in desired configurations [4–9], (here referred to as aggregates) aimed at making photonic devices. A recent development is the use of DNA self-assembly to construct dye aggregates in which the distance between the dyes is sufficiently close that excited-state energy can be transferred between neighboring dyes in a manner that maintains quantum coherence [10–17]. The packet of energy that is transferred between dyes is referred to as an exciton and that exhibits quantum mechanical particle-like properties. The possibility of using such aggregates as quantum gates

B. Yurke (✉)

Micron School of Materials Science & Engineering and Department of Electrical and Computer Engineering, Boise State University, Boise, ID 83725, USA
e-mail: bernardyurke@boisestate.edu

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*, Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_9

125

and quantum computers was raised by Castellanos et al. [18, 19]. They exhibited dye aggregate configurations that implemented the function of a set of three types of gates that would enable universal quantum computation. As the authors point out, however, their embodiment of these gates does not allow one to exploit the full power of quantum computation. To accomplish that, the authors note, use must be made of the interaction between two excitons. Here a set of gates is provided for which the interaction between two excitons is employed in the implementation of the controlled not (CNOT) gate. These gates do form a complete set enabling universal quantum computation. The quantum computation scheme employed is that of a many particle quantum walk, similar to the scheme proposed by Childs et al. [20]; however, here a dual-rail architecture is employed that greatly simplifies gate design.

Random perturbations of excitons by molecular vibrations tend to wash out the delicate quantum interference effects on which quantum computing relies. Other would-be quantum computing technologies also must contend with random processes by which quantum interference effects are washed out. This problem, however, is sufficiently severe for exciton-based quantum gates that it poses a significant challenge to exciton-based quantum computation's capacity to win the race against all the other quantum computation technologies being developed [21–23]. Whether dyes can be synthesized for which the interaction between excitons and molecular vibrations is sufficiently weak to enable more than the demonstration of rudimentary quantum computing circuits remains to be seen. The employment of devices based on molecular transitions at the energy scale of visible light photons does open, however, the prospect for quantum gate operation at room temperature and with femtosecond switching speeds. Quantum computation through the assembly of dye aggregates via DNA-based self-assembly thus provides a reach goal to drive the development of high technology photonic devices in which the gates are of molecular scale, have high switching speeds, and the circuits have high component density.

Although the scope here is limited to dye aggregate architectures that can function as exciton-based quantum gates for universal quantum computation, with the implicit understanding that currently DNA self-assembly is the most promising means by which to assemble such aggregates, it worth noting that the study excitons in dye aggregates and molecular crystals has a long and extensive history [24]. This history predates the field of DNA assembly, beginning with theoretical work by Frenkel [25] in 1931 and the experimental work by Jelley [26] in 1936 and by Scheibe et al. [27–29] in 1937.

As indicated, here we explore the possibility of making information processing devices out of dye aggregates constructed by DNA self-assembly. Dye molecules exhibit color due to their ability to absorb light at specific wavelengths. The process occurs because the molecule has a transition from the ground state to an excited state that can be induced by the absorption of a visible light photon. Such a transition is said to be optically allowed in order to distinguish it from transitions that require a change in the total electron spin angular momentum, something that the absorption of a photon alone cannot do, or transitions that a photon cannot induce for symmetry or other reasons.

The bundle of energy stored in the molecule upon absorption of a photon can be transferred from one dye molecule to a neighboring dye molecule. In this process one dye molecule returns to its ground state while simultaneously the other is promoted from its ground state to its optically allowed excited state. This bundle of energy that once was a photon can thus propagate from dye molecule to dye molecule throughout an aggregate of dye molecules. In this manner the bundle of energy behaves like a particle and could be used as a carrier of information much like an electron.

This bundle of energy, which resides on one dye molecule at a time, is referred to as an exciton. Technically, this bundle of energy is referred to as a Frenkel exciton to distinguish it from a Wannier–Mott exciton which is an electron–hole pair residing in a semiconductor material. Either is simply referred to as an exciton, when it is clear from context which type of exciton is meant.

When the spacing between dyes becomes 2 nm or less, the hopping occurs in a coherent manner, which enables the exciton to behave like a quantum mechanical particle exhibiting wave-like behavior. This is where quantum mechanical wave-particle duality enters. Even though at any instant of time the exciton resides on only one dye, the exciton behaves as if it is a wave spread out over the entire dye aggregate.

When a pair of neighboring dyes are both in their excited state, the electrostatic energy between the dyes will be different from that when only one dye is excited due to the change in a dye's electron density distribution that occurs when the dye transitions from the ground state to the excited state. This gives rise to a two-body or exciton–exciton interaction in which excitons scatter off of each other.

These two properties, the ability of an exciton to coherently hop from dye to dye and the ability of two excitons to scatter off of each other, in principle, enable suitably structured dye aggregates to function as quantum gates, information processing systems and maybe even as quantum computers.

In the following sections a set of dye configurations will be described that form a complete set of gates for universal quantum computation. Also described is how these gates can be assembled into functioning information processing systems and quantum computers. Nonidealities that must be overcome to realize such devices are also discussed.

2 The Mathematical Structure of Reality

Here a brief introduction to quantum mechanics is provided with the aim to give some indication of where the power of quantum computing resides. Quantum theory has survived 100 years of rigorous testing and, as near as can be discerned, provides the foundational description of all physical phenomena. This includes physical computation processes. In this sense all computers are quantum computers. But not all computers take advantage of the additional computing resources that quantum mechanics provides beyond those utilized by computers relying on classical physics.

As an indication of how reality, by which we mean the state of a physical system, is represented in quantum mechanics, consider first a classical mechanical system.

At any given instant of time, such a system will possess a number of attributes that can be measured such as its position, its momentum, angular momentum and energy. The list of numbers needed to specify the state of the system can be shorter than the complete list of attributes. For example, for a point particle the energy and angular momentum can be computed if the position and momentum are known. Thus, the particle's position and momentum provide a complete specification of its state. In contrast to classical systems, in which the attributes can take on any real number value, in quantum mechanical systems, a measurement of an attribute often will only yield one of a discrete list of values: that is, the attribute is quantized. For example, for an electron in a hydrogen atom, the total angular momentum J and the z -component of the total angular momentum J_z , if measured, take on the discrete values $\hbar\sqrt{j(j+1)}$ and $\hbar m_j$, respectively, where \hbar is Planck's constant, j is a half integer greater than zero, that is, $j \in \{1/2, 3/2, 5/2, \dots\}$, and m_j is a half integer in the range $-j \leq m_j \leq j$. Given j and m_j , the energy of the electron can be computed. Indeed, j and m_j provide a complete specification of the state of the electron for which j and m_j have been measured and this state is often denoted by $|j, m_j\rangle$. Such states, however, do not exhaust the states that the electron can be in. The electron can be in any state $|\psi\rangle$ of the form

$$|\psi\rangle = \sum_{j=1/2}^{\infty} \sum_{m_j=-j}^j \alpha_{j,m_j} |j, m_j\rangle. \quad (1)$$

where the $\alpha_{j,m}$ are complex numbers subject to the constraint

$$\sum_{j=1/2}^{\infty} \sum_{m_j=-j}^j |\alpha_{j,m_j}|^2 = 1. \quad (2)$$

Note that the state $|\psi\rangle$ can be represented as a column vector listing all the α_{j,m_j} :

$$|\psi\rangle = \begin{bmatrix} \alpha_{1/2,-1/2} \\ \alpha_{1/2,1/2} \\ \alpha_{3/2,-3/2} \\ \alpha_{3/2,-1/2} \\ \alpha_{3/2,1/2} \\ \alpha_{3/2,3/2} \\ \vdots \\ \vdots \\ \vdots \end{bmatrix}. \quad (3)$$

If one performs a measurement of J and J_z on this state, the probability of obtaining the quantum numbers j and m_j from the measurement is $|\alpha_{j,m_j}|^2$. The measurement of J and J_z must yield some value for j and m_j . Hence, Eq. (2) is simply the statement that the sum of the probabilities of measurement outcomes over all

possible measurement outcomes must be 1. It is tempting to interpret $|\psi\rangle$ of Eq. (1) as representing a statistical ensemble for which $|\alpha_{j,m_j}|^2$ is the fraction of systems in the state $|j, m_j\rangle$; however, with more than one nonzero α_{j,m_j} , states having the from Eq. (1) can give rise to observable interference effects if the x -component J_x or y -component J_y of the angular momentum is measured, thereby making an ensemble interpretation untenable.

Having introduced quantum mechanical state vectors via the example of an electron in a hydrogen atom, the discussion will now proceed on a more general and abstract level. The state $|\psi\rangle$ of a quantum mechanical system consists of an N -dimensional vector where N could be ∞ . The state can be represented by a column vector

$$|\psi\rangle = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \cdot \\ \cdot \\ \cdot \\ \alpha_N \end{bmatrix} \quad (4)$$

where the α_i are complex functions of time. The Hermitian adjoint (the matrix that results when one interchanges rows and columns of a matrix and takes the complex conjugate of the matrix elements) of $|\psi\rangle$ is given by

$$\langle\psi| = [\alpha_1^* \alpha_2^* \cdot \cdot \cdot \alpha_N^*]. \quad (5)$$

To provide quantum mechanics with a probability interpretation, a norm of the state vector $|\psi\rangle$ is introduced and denoted by $\langle\psi|\psi\rangle$ and defined by the matrix product

$$\langle\psi|\psi\rangle \equiv [\alpha_1^* \alpha_2^* \cdot \cdot \cdot \alpha_N^*] \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \cdot \\ \cdot \\ \cdot \\ \alpha_N \end{bmatrix} = \sum_{m=1}^N |\alpha_m|^2. \quad (6)$$

The probability interpretation is imposed by requiring the state vector to have unit norm $\langle\psi|\psi\rangle = 1$. Then $|\alpha_m|^2$ is the probability of finding the system in the state represented by the m th position of the state vector.

The dynamics of a quantum mechanical system is governed by the Schrödinger equation

$$i\hbar \frac{\partial}{\partial t} |\psi\rangle = H|\psi\rangle, \quad (7)$$

where t is the time and H is the Hamiltonian. The Hamiltonian is an $N \times N$ array. The values for the elements of this array depend on the system under consideration and are ultimately determined by experiment, though procedures, such as canoni-

cal quantization, are available that enable one to generate the Hamiltonian from a classical mechanical description of the system. The Hamiltonian is required to be Hermitian, that is, if the rows and columns are interchanged and the complex conjugate of all the matrix elements is taken, one winds up with the same matrix. This ensures that the Hamiltonian only has real eigenvalues, which turn out to be the energy eigenvalues of the system.

Equation (7) can be formally integrated to yield

$$|\psi(t)\rangle = U(t, t_0)|\psi(t_0)\rangle \quad (8)$$

where

$$U(t, t_0) = \exp\left(-\frac{i}{\hbar} \int_{t_0}^t H dt\right). \quad (9)$$

Since H is Hermitian, the Hermitian conjugate of $U(t, t_0)$ is

$$U^\dagger(t, t_0) = \exp\left(\frac{i}{\hbar} \int_{t_0}^t H dt\right). \quad (10)$$

It then follows that $U^\dagger(t, t_0)$ is unitary, that is

$$U^\dagger(t, t_0)U(t, t_0) = U(t, t_0)U(t, t_0)^\dagger = 1. \quad (11)$$

Thus, governed by the Schrödinger equation a system undergoes unitary time evolution. An important consequence of this is that the norm of $|\psi\rangle$ does not change with time, that is, probability is conserved in the sense that the sum of the probabilities of outcomes over all possible outcomes is 1, as required for a consistent probabilistic interpretation.

3 Quantum Computers

Having outlined the mathematical structure of reality, an indication is now provided of how the power of quantum computation arises. Apart from unitarity, quantum mechanics places no further restrictions on the unitary transformations that can be physically realized. Of course, the challenge is to find or engineer physical systems that can perform the unitary transformations that carry out the desired computation. Here we take for granted that for any desired unitary transformation a physical system can be engineered that carries out that unitary transformation.

It is convenient to introduce some further notation. The state vector Eq. (4) can be written as

$$|\psi\rangle = \sum_{m=1}^N \alpha_m |m\rangle, \quad (12)$$

where the $|m\rangle$ are time-independent unit basis vectors satisfying the orthonormality condition

$$\langle m|m'\rangle = \delta_{m,m'} \quad (13)$$

where $\delta_{m,m'}$ is the Kronecker delta function. The Kronecker delta function is zero when $m \neq m'$ and one when $m = m'$.

Consider now a two-state system where one basis state has been arbitrarily chosen to represent a logical 0, whereas the orthogonal basis state is chosen to represent a logical 1. Denoting these states by $|0\rangle$ and $|1\rangle$, respectively, an arbitrary state of the system is then given by

$$|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle = \begin{bmatrix} \alpha_0 \\ \alpha_1 \end{bmatrix}. \quad (14)$$

This $|\psi\rangle$ could be the state of a binary memory register. A classical binary memory register can be in only one of two mutually exclusive states, a logical 0 or a logical 1. In contrast, a quantum mechanical binary memory register can be in any state of the form of Eq. (14). In the general case, the contents of the binary memory register must be specified by two complex numbers, α_0 and α_1 with the restriction $|\alpha_0|^2 + |\alpha_1|^2 = 1$. Although $|\alpha_0|^2$ and $|\alpha_1|^2$ are the probabilities of finding the register content to be a logical 0 or a logical 1 if one looks at the contents, this does not represent a statistical mixture in which the memory register is in one state or the other. Otherwise a single number, $|\alpha_0|^2$, would be sufficient to describe the state of the system rather than two complex numbers with a constraint on the sum of their norms. The information encoded by the state Eq. (14) is referred to as a qubit. Its logical value is specified by the complex numbers α_1 and α_2 .

A general unitary transformation of the state Eq. (14) has the form

$$\begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = U \begin{bmatrix} \alpha_0 \\ \alpha_1 \end{bmatrix} \quad (15)$$

where

$$U = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}. \quad (16)$$

In general, the $u_{m,n}$ are complex numbers subject to the constraint that U be unitary, that is, $U^\dagger U = U U^\dagger = 1$. Here a quantum mechanical logic gate performing such a transformation is referred to as a basis change gate. Note that this gate is a one input and one output gate and consequently could be regarded as a generalization of a classical NOT gate. A special case of the basis change gate is the Hadamard gate, which is given by

$$U_H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \quad (17)$$

This gate frequently shows up in quantum computing circuits. Suppose that initially the memory register is in the logical 0 state $|0\rangle$, then, subjected to a physical system that transforms the memory contents according to Eq. (17), the final state of the memory register will be $|\psi\rangle = (1/\sqrt{2})(|0\rangle + |1\rangle)$. A basis change gate thus provides a means for putting a memory element into a superposition state when initially it is in the logical 0 or logical 1 state. Another special case of the one input gate Eq. (15) is the phase gate:

$$U_P = \begin{bmatrix} e^{i\phi_0} & 0 \\ 0 & e^{i\phi_1} \end{bmatrix}. \quad (18)$$

This gate transforms the phases of the elements of a superposition state. For example, if the initial state of the memory element is given by $|\psi(t_0)\rangle = (1/\sqrt{2})(|0\rangle + |1\rangle)$, a physical system performing the operation Eq. (15) on the memory element will put it in the state $|\psi(t_1)\rangle = (1/\sqrt{2})(e^{i\phi_0}|0\rangle + e^{i\phi_1}|1\rangle)$.

3.1 The Controlled NOT Gate

It is noted that the phenomenon of classical wave interference is sufficient to realize any unitary transformation. For example, any unitary transformation on an N dimensional vector can be realized by an array of $N(N - 1)/2$ optical beam splitters [30]. However, for certain unitary transformations, quantum mechanics provides a means to greatly reduce the number of parts needed to implement the unitary transformation. To indicate how this works the controlled not (CNOT) gate will now be considered.

The CNOT gate is a two-input two-output gate that is a generalization of a classical exclusive or (XOR) operation. Since this is a two-input gate, two interacting two-level systems are required to implement the gate, the control system and the target system. Let the Boolean bases states of the control system be denoted by $|0\rangle_C$ and $|1\rangle_C$ and the Boolean bases states of the target system be denoted by $|0\rangle_T$ and $|1\rangle_T$. The state space for the complete system is the outer product of the state space for the two subsystems. Hence, the basis states for the complete system can be written as

$$\begin{aligned} |1\rangle &= |0\rangle_C |0\rangle_T = |0, 0\rangle \\ |2\rangle &= |0\rangle_C |1\rangle_T = |1, 0\rangle \\ |3\rangle &= |1\rangle_C |0\rangle_T = |0, 1\rangle \\ |4\rangle &= |1\rangle_C |1\rangle_T = |1, 1\rangle. \end{aligned} \quad (19)$$

In the rightmost equalities the states have been written in the form $|x, y\rangle$ where x is the Boolean value of the control basis state and y is the Boolean value of the target basis state. In this basis, the unitary transformation performed by the CNOT gate has the matrix representation

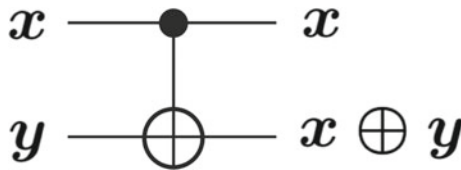


Fig. 1 Symbol for a CNOT gate. The input and output line labeled x is the control line. The target input is y . The target output is the XOR of x and y

$$U_{\text{CNOT}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \quad (20)$$

On the Boolean basis states this CNOT operation performs the following transformations:

$$\begin{aligned} |0, 0\rangle &\rightarrow |0, 0\rangle \\ |0, 1\rangle &\rightarrow |0, 1\rangle \\ |1, 0\rangle &\rightarrow |1, 1\rangle \\ |1, 1\rangle &\rightarrow |1, 0\rangle \end{aligned} \quad (21)$$

From this it is apparent that the control state x remains unchanged, whereas the target state y is transformed to $x \oplus y$, the XOR operation. The symbol for a CNOT gate in quantum computer diagrams is given in Fig. 1.

In general, the initial state of our two two-state system has the form

$$|\psi(t_0)\rangle = \begin{bmatrix} \alpha_{0,0} \\ \alpha_{0,1} \\ \alpha_{1,0} \\ \alpha_{1,1} \end{bmatrix} = \alpha_{0,0}|0, 0\rangle + \alpha_{0,1}|0, 1\rangle + \alpha_{1,0}|1, 0\rangle + \alpha_{1,1}|1, 1\rangle, \quad (22)$$

where the state vector has been written in two different forms, that of a column vector and that in which basis vectors in ket notation are employed. The result of operation of the CNOT gate on this state is obtained by multiplying the column vector form of the state by the matrix Eq. (20) representing the unitary transformation performed by the CNOT gate, as Eq. (15) indicates. The resulting state is

$$\begin{aligned} |\psi(t_1)\rangle &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha_{0,0} \\ \alpha_{0,1} \\ \alpha_{1,0} \\ \alpha_{1,1} \end{bmatrix} = \begin{bmatrix} \alpha_{0,0} \\ \alpha_{0,1} \\ \alpha_{1,1} \\ \alpha_{1,0} \end{bmatrix} \\ &= \alpha_{0,0}|0, 0\rangle + \alpha_{0,1}|0, 1\rangle + \alpha_{1,0}|1, 1\rangle + \alpha_{1,1}|1, 0\rangle. \end{aligned} \quad (23)$$

Comparing these last two equations one sees that the CNOT gate simultaneously performs the correct logical operation on each Boolean component of the input state. This is an example of quantum parallelism.

3.2 Quantum Parallelism

To generalize the discussion of CNOT gate operation, consider the case of a memory register consisting of N two-state systems. For a classical memory register, each bit element is either in a logical 0 or logical 1 state. As a consequence, the register can store only one N bit binary number at a time. Quantum mechanically the register can be in any state of the form

$$|\psi(t_0)\rangle = \sum_{m_1=0}^1 \sum_{m_2=0}^1 \dots \sum_{m_N=0}^1 \alpha_{m_1, m_2, \dots, m_N} |m_1, m_2, \dots, m_N\rangle, \quad (24)$$

where, as indicated by the subscripts and superscripts of the sums, $m_i \in \{0, 1\}$. Each state label m_1, m_2, \dots, m_N is, thus, a binary number and the sum is over all possible N bit binary numbers. Equation (24) is a generalization of Eq. (22). Thus, a quantum mechanical memory register consisting of N two-level systems behaves as if it is simultaneously storing 2^N complex numbers subject to the constraint that the sum of the norm-squares of these complex numbers is 1. Singling out memory elements i and a j for operation on by a CNOT gate, Eq. (24) can be written as

$$\begin{aligned} |\psi(t_0)\rangle = & \sum_{\{m_n | n \neq \{i, j\}\}} \alpha_{m_1, m_2, \dots, 0_i \dots 0_j \dots m_N} |m_1, m_2, \dots, 0_i \dots 0_j \dots m_N\rangle \\ & + \sum_{\{m_n | n \neq \{i, j\}\}} \alpha_{m_1, m_2, \dots, 0_i \dots 1_j \dots m_N} |m_1, m_2, \dots, 0_i \dots 1_j \dots m_N\rangle \\ & + \sum_{\{m_n | n \neq \{i, j\}\}} \alpha_{m_1, m_2, \dots, 1_i \dots 0_j \dots m_N} |m_1, m_2, \dots, 1_i \dots 0_j \dots m_N\rangle \\ & + \sum_{\{m_n | n \neq \{i, j\}\}} \alpha_{m_1, m_2, \dots, 1_i \dots 1_j \dots m_N} |m_1, m_2, \dots, 1_i \dots 1_j \dots m_N\rangle, \end{aligned} \quad (25)$$

where the subscript on the sums indicates summation over all m_n as in Eq. (24), excluding the sum over m_i and m_j . Upon operation by the CNOT gate, the contents of the memory register, where i is the control qubit and j is the target qubit, are changed to

$$\begin{aligned}
|\psi(t_1)\rangle = & \sum_{\{m_n | n \notin \{i, j\}\}} \alpha_{m_1, m_2, \dots, 0_i \dots 0_j \dots m_N} |m_1, m_2, \dots, 0_i \dots 0_j \dots m_N\rangle \\
& + \sum_{\{m_n | n \notin \{i, j\}\}} \alpha_{m_1, m_2, \dots, 0_i \dots 1_j \dots m_N} |m_1, m_2, \dots, 0_i \dots 1_j \dots m_N\rangle \\
& + \sum_{\{m_n | n \notin \{i, j\}\}} \alpha_{m_1, m_2, \dots, 1_i \dots 1_j \dots m_N} |m_1, m_2, \dots, 1_i \dots 1_j \dots m_N\rangle \\
& + \sum_{\{m_n | n \notin \{i, j\}\}} \alpha_{m_1, m_2, \dots, 1_i \dots 0_j \dots m_N} |m_1, m_2, \dots, 1_i \dots 0_j \dots m_N\rangle.
\end{aligned} \tag{26}$$

This equation is a generalization of Eq. (23). Thus, the CNOT gate simultaneously performs the correct Boolean operation on each of the 2^N Boolean basis states of the superposition.

Not all computations can take advantage of this quantum parallelism. The identification of computational tasks of practical interest that can benefit from quantum speedup has been slow in coming but include important problems such as factoring large numbers and doing database searches. The quantum computing algorithms for these two problems were discovered by Shor [31] and Grover [32] respectively.

The CNOT gate, the Hadamard gate, together with the phase gates, form a complete set of gates for universal quantum computation [33]. Thus, this set of gates plays a role similar to that of the NAND gate of classical electronic circuit design, the gate with which any Boolean function can be implemented.

4 The Frenkel Exciton Hamiltonian

Having displayed a complete set of gates that enable universal quantum computation, we now work toward showing how these gates might be realized by dye aggregates. To this end, a Hamiltonian governing the dynamics of excitons is now introduced, the Frenkel exciton Hamiltonian. This is a phenomenological or reduced Hamiltonian, in that it contains parameters that must be determined by experiment or by calculation methods, such as time-dependent density functional theory, that are closer to first principles calculations. Here interaction of excitons with molecular vibrations is neglected. How these may be included is discussed in Sect. 16.

The Frenkel exciton Hamiltonian is given by Abramavicius et al. [34], Renger et al. [35]

$$\begin{aligned}
H = & \sum_{m=1}^N E_m^e B_m^\dagger B_m + \sum_{(m,n)} J_{(m,n)} (B_m^\dagger B_n + B_n^\dagger B_m) \\
& + \frac{1}{2} \sum_{m=1}^N \Delta_m B_m^\dagger B_m^\dagger B_m B_m + \sum_{(m,n)} K_{(m,n)} B_m^\dagger B_n^\dagger B_n B_m.
\end{aligned} \tag{27}$$

Here the summation index (m, n) denotes summation over all distinct pairs dye molecules, and N is the total number of dyes in the aggregate.

E_m^e is the transition energy from the ground state to the first optically allowed excited state of molecule m . $J_{(m,n)}$ is the exciton exchange energy, which arises from the Coulomb interaction between the transition charge densities [34] of dye pair m and n . Δ_m is the anharmonicity parameter [35] that quantifies the energy cost with having two excitons occupy the same dye. It can be understood as follows. Let S_0 denote the ground state of the dye. The lowest optically allowed excited state is denoted by S_1 and is the one-exciton state of the dye. The transition energy between these two states is E_m^e . Let S_n denote the excited state of the dye having the allowed optical transition from the state S_1 whose energy lies closest to $2E_m^e$. Its energy can be written as $2E_m^e + \Delta_m$, where Δ_m accounts for the energy mismatch. Because the energy of this state is approximately twice that of the state S_1 , it can be regarded as the state for which two excitons reside on the dye. The anharmonicity parameter thus provides a simple means to account for the existence of dye energy levels above the first excited state that play a role when more than one exciton is present in the aggregate. $K_{(m,n)}$ is the exciton–exciton interaction energy between an exciton residing on dye m and an exciton residing on dye n . This interaction results from the difference in the charge density of a dye molecule when it is in its excited state compared to when it is in its ground state [34]. This results in a difference in the total Coulomb energy of the aggregate when two excitons reside on neighboring dyes or are farther apart.

These phenomenological parameters are amenable to engineering. The value of E_m^e depends on the dye structure and can be varied by changing substituents on the dye. $J_{(m,n)}$ and $K_{(m,n)}$ depend on the structure of dyes m and n and how the dyes are positioned and oriented with respect to each other. As indicated in the **Appendix**, dye pairs exist for which $J_{(m,n)}$ and $K_{(m,n)}$ can be adjusted independently of each other by reorienting the dyes. In what follows it is assumed that dye aggregate systems can be engineered in which these phenomenological parameters, for nearest neighbor dyes pairs, can take on any desired value within the maximal limits of these quantities for available dyes. Values of E_m^e are in the several electron volt (eV) range. For dye pairs in close proximity, $J_{(m,n)}$ can be in the 100 meV range [10]. On dimensional grounds, one expects that $K_{(m,n)}$ can achieve similar strength. As discussed in the **Appendix**, the strengths of $J_{(m,n)}$ and $K_{(m,n)}$ can be estimated from dipole approximations. In this approximation, $J_{(m,n)}$ is proportional to the product of the transition dipole moments μ_m and μ_n for the two dyes m and n , whereas $K_{(m,n)}$ is proportional to the product of the dipole moments Δd_m and Δd_n . These latter dipoles are referred to as excess dipoles in reference [36] and represent the difference between the excited-state and ground-state charge densities for the two molecules. μ_m and Δd_m can both range as high as 16 debye [36–38]. The values of $J_{(m,n)}$ and $K_{(m,n)}$ can be larger than the characteristic room-temperature thermal energy $k_B T = 26$ meV, where k_B is the Boltzmann constant and T is the absolute temperature of ~ 300 K. This means that exciton-based quantum gates should function at room temperature, in contrast to superconducting device-based quantum computers that require millikelvin temperatures to operate.

The B_m^\dagger and B_m are not numerical quantities but operators referred to as exciton creation operators and exciton annihilation operators, respectively. These are best viewed as part of a clever and economical bookkeeping formalism that enables the construction of the exciton state space and aids in keeping track of how the state vector changes with time.

At this point it is useful to introduce the notion of an energy eigenstate and an energy eigenvalue. By direct substitution it can be shown that states of the form

$$|\psi\rangle = e^{-iE_k t/\hbar}|E_k\rangle, \quad (28)$$

where $|E_k\rangle$ is time independent, are solutions to the Schrödinger equation Eq. (1) provided the equation

$$E_k|E_k\rangle = H|E_k\rangle \quad (29)$$

is satisfied. A state satisfying Eq. (29) is said to be an energy eigenstate and E_k is said to be its energy eigenvalue. If the state space is N dimensional N orthogonal energy eigenstates will exist, enumerated by the integer subscripts k .

An aggregate system will have a lowest energy state in which all the dye molecules are in their ground state. This state is denoted by $|0\rangle$ and is taken to have unit norm $\langle 0|0\rangle = 1$. The annihilation operator B_m has the property that

$$B_m|0\rangle = 0. \quad (30)$$

Using this equation one immediately obtains

$$H|0\rangle = 0. \quad (31)$$

It follows that $|0\rangle$ is an energy eigenstate having energy eigenvalue 0. Hence, the Hamiltonian Eq. (27) has been constructed so that the zero of its energy scale matches the ground-state energy of the aggregate.

It is now useful to introduce the notion of a commutator. Given any two operators A and B , their commutator is defined by

$$[A, B] \equiv AB - BA. \quad (32)$$

The exciton creation and annihilation operators satisfy the commutation relations

$$[B_m, B_n] = [B_m^\dagger, B_n^\dagger] = 0 \quad (33)$$

and

$$[B_m, B_n^\dagger] = \delta_{m,n} \quad (34)$$

where $\delta_{m,n}$ is the Kronecker delta function. The Kronecker delta function has the value 0 when $m \neq n$ and the value 1 when $m = n$.

A complete set of orthonormal basis vectors can be constructed by applying products of creation operators to $|0\rangle$. The general state will have the form

$$|n_1, n_2, \dots, n_N\rangle = \prod_{m=1}^N \frac{(B_m^\dagger)^{n_m}}{\sqrt{n_m!}} |0\rangle. \quad (35)$$

The number n_m is the number of excitons residing on dye m .

As the notation for the creation and annihilation operators suggest, B_m^\dagger is regarded as the Hermitian adjoint of B_m . In addition, the Hermitian adjoint of $|0\rangle$ is denoted by $\langle 0|$. Hence, the Hermitian adjoint of the state vector Eq. (35) is given by

$$\langle n_1, n_2, \dots, n_N| = \langle 0| \prod_{m=1}^N \frac{(B_m)^{n_m}}{\sqrt{n_m!}} \quad (36)$$

where use has been made of the commutation relation Eq. (33), allowing reordering of annihilation operators among themselves. Using the commutation relations for the creation and annihilation operators, one can show that these states satisfy the orthonormality condition

$$\langle n_1, n_2, \dots, n_N | n'_1, n'_2, \dots, n'_N \rangle = \prod_{k=1}^N \delta_{n_k, n'_k}. \quad (37)$$

The Hamiltonian and the state space have now been described. This system belongs to a class of Hamiltonians referred to as Bose–Hubbard models [20]. It has been shown that universal quantum computation can be performed by a many-particle quantum walk in such systems [20]. The proof consists of showing how a universal set of quantum gates can be implemented in such systems. A similar analysis will be presented here by exhibiting a set of gates that may be easier to implement in dye aggregate systems.

Before doing so, an analysis of a two-dye aggregate is performed to illustrate how computations are carried out with this formalism and to provide some insight into the quantum behavior of excitons.

5 Energy Eigenvalues of a Homodimer Dye Aggregate and Davydov Splitting

Here, as an example of how Frenkel exciton computations are carried out, the energy eigenvalues and eigenvectors of a dye aggregate consisting of two identical dyes (homodimer) are solved for the case when a dye aggregate contains one exciton. It will be shown that as a result of exciton exchange, the absorption spectrum of the dimer exhibits peak splitting, a phenomenon referred to as exciton splitting [39] or Davydov splitting [40].

Consider the case when the dye aggregate consists of two identical dye molecules. The Hamiltonian Eq. (27) then reduces to [41]

$$H = E^e \left(B_1^\dagger B_1 + B_2^\dagger B_2 \right) + J \left(B_1^\dagger B_2 + B_2^\dagger B_1 \right) + \frac{\Delta}{2} \left(B_1^\dagger B_1^\dagger B_1 B_1 + B_2^\dagger B_2^\dagger B_2 B_2 \right) + K B_1^\dagger B_2^\dagger B_2 B_1. \quad (38)$$

The Hamiltonian of Eq. (27) and, consequently, of Eq. (38) is exciton number conserving. That these cannot change the number of excitons follows from the fact that in each term of the Hamiltonian each creation operator is paired with an annihilation operator and from the commutation relations for the exciton creation and annihilation operators. As a consequence, the energy eigenvalues can be found by working with state spaces having a fixed number of excitons. The system ground state $|0\rangle$ is the zero exciton example of this as it is an energy eigenstate.

From Eq. (35), the set of one-exciton states in the site basis is

$$\mathbf{B}_1 = \left\{ B_1^\dagger |0\rangle, B_2^\dagger |0\rangle \right\}. \quad (39)$$

These two states are the states in which the exciton occupies molecule 1 or molecule 2, respectively. The set of two-exciton states in the site basis is

$$\mathbf{B}_2 = \left\{ \frac{B_1^\dagger B_1^\dagger}{\sqrt{2}} |0\rangle, B_1^\dagger B_2^\dagger |0\rangle, \frac{B_2^\dagger B_2^\dagger}{\sqrt{2}} |0\rangle \right\}. \quad (40)$$

The leftmost state of this set is that for which both excitons reside on molecule 1. The middle state is that for which one exciton resides on molecule 1 and the other on molecule 2. The rightmost state is that for which both excitons reside on molecule 2. Note that due to the commutation relation Eq. (33), the state $B_1^\dagger B_2^\dagger |0\rangle$ and the state $B_2^\dagger B_1^\dagger |0\rangle$ are the same state. Thus, only one appears in the set. The formalism via the commutation relations Eqs. (33) and (34), thus, has the indistinguishability of excitons built into it. Indistinguishable particles for which these commutation relations apply satisfy Bose statistics and are referred to as Bosons.

To determine the energy eigenstates and eigenvalues of Hamiltonian Eq. (38) it is useful to evaluate the expression $B_r^\dagger B_s B_t^\dagger |0\rangle$ and $B_s^\dagger B_r^\dagger B_r B_s B_t^\dagger |0\rangle$, where r, s and t are integer site labels. From the commutation relation Eq. (34) one has

$$B_s B_t^\dagger - B_t^\dagger B_s = \delta_{s,t} \quad (41)$$

or

$$B_s B_t^\dagger = \delta_{s,t} + B_t^\dagger B_s. \quad (42)$$

One thus has

$$B_r^\dagger B_s B_t^\dagger |0\rangle = B_r^\dagger [\delta_{s,t} + B_t^\dagger B_s] |0\rangle = \delta_{s,t} B_r^\dagger |0\rangle \quad (43)$$

and

$$B_s^\dagger B_r^\dagger B_r B_s B_t^\dagger |0\rangle = B_s^\dagger B_r^\dagger B_r [\delta_{s,t} + B_t^\dagger B_s] |0\rangle = 0 \quad (44)$$

where in each case the last equality follows from Eq. (30). A consequence of the last equation is that terms with the Δ and K coefficients of Eq. (38) are zero in the one-exciton sector of the state space, as one would expect given these terms account for exciton–exciton interactions. Using these last two equations one finds

$$H B_1^\dagger |0\rangle = E^e B_1^\dagger |0\rangle + J B_2^\dagger |0\rangle \quad (45)$$

and

$$H B_2^\dagger |0\rangle = J B_1^\dagger |0\rangle + E^e B_2^\dagger |0\rangle. \quad (46)$$

The general one-exciton state vector has the form

$$|\psi\rangle = \alpha_1 B_1^\dagger |0\rangle + \alpha_2 B_2^\dagger |0\rangle. \quad (47)$$

See Eqs. (4) and (12). Hence, in matrix form one has

$$H|\psi\rangle = \begin{bmatrix} E^e & J \\ J & E^e \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}. \quad (48)$$

Equation (29) becomes

$$E_k \begin{bmatrix} \alpha_{k1} \\ \alpha_{k2} \end{bmatrix} = \begin{bmatrix} E^e & J \\ J & E^e \end{bmatrix} \begin{bmatrix} \alpha_{k1} \\ \alpha_{k2} \end{bmatrix}. \quad (49)$$

The solutions to this eigenvalue–eigenvector equation can be solved by brute force using general linear algebra techniques. The clever approach, however, is to note the Hamiltonian Eq. (38) remains unchanged (is symmetric) under the interchange of subscripts 1 and 2 on the creation and annihilation operators, a consequence of having chosen the two dyes to be identical. In this case it follows from group representation theory of the permutation group that the eigenstates must be symmetric or antisymmetric (changes sign) under the interchange of the site subscripts. Hence, the eigenstates can be immediately written down:

$$|S\rangle = \frac{1}{\sqrt{2}} (B_1^\dagger |0\rangle + B_2^\dagger |0\rangle) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad (50)$$

and

$$|A\rangle = \frac{1}{\sqrt{2}} (B_1^\dagger |0\rangle - B_2^\dagger |0\rangle) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad (51)$$

where the state labels S and A indicate whether the state is symmetric or antisymmetric under interchange of the site labels. The corresponding energy eigenvalues are then immediately obtained by substitution of the eigenstates into Eq. (49). One finds

$$E_S = E^e + J \quad (52)$$

and

$$E_A = E^e - J. \quad (53)$$

When the exchange energy J is zero, such as when the dyes are far apart, the energies become degenerate with both eigenstates having the energy E^e . This is the energy of the photon that by absorption in a dye molecule induces a transition from the ground state to its lowest optically allowed excited state. When the exchange energy is nonzero, the energy of the two eigenstates differs by $2J$. The transition from the ground state $|0\rangle$ to the excited state $|S\rangle$ is induced by a photon having the energy $E^e + J$. The transition from the ground state to the excited state $|A\rangle$ is induced by a photon having the energy $E^e - J$. The absorption spectrum of the dimer will thus exhibit two peaks in the absorption spectrum, whereas a single dye molecule would exhibit a single absorption peak. This peak splitting is called Davydov splitting and is of size $2J$. Figure 2 provides an experimental example of Davydov splitting observed for two “squaraine-rotaxane” dyes confined to the core of a DNA Holliday junction by covalent linkages [42]. One sees that the absorbance peak of this dimer aggregate is split into two peaks, one on either side of the monomer (single dye) absorbance peak.

Note that for the energy eigenstates Eqs. (50) and (51), the probability amplitude is of equal magnitude for the exciton to reside on dye 1 and dye 2. The exciton acts as if it has a simultaneous existence on both dyes. This is referred to as exciton delocalization. A classical particle would not be able to do this because it can only have one position at a time.

6 Coherent Exciton Hopping

The energy eigenstates at any instant of time make a perfectly good basis set with which to express a quantum state. This basis set is particularly convenient to work with when considering the time evolution of a system. Any state $|\psi(0)\rangle$ can be expressed in the form

$$|\psi(0)\rangle = \sum_{k=1}^N \alpha_k |E_k\rangle, \quad (54)$$

where $|E_k\rangle$ are the energy eigenstates obtained by solving Eq. (29). Using this as the initial condition for the state $|\psi\rangle$ appearing in the Schrödinger equation Eq. (7) the solution to the Schrödinger equation is given by

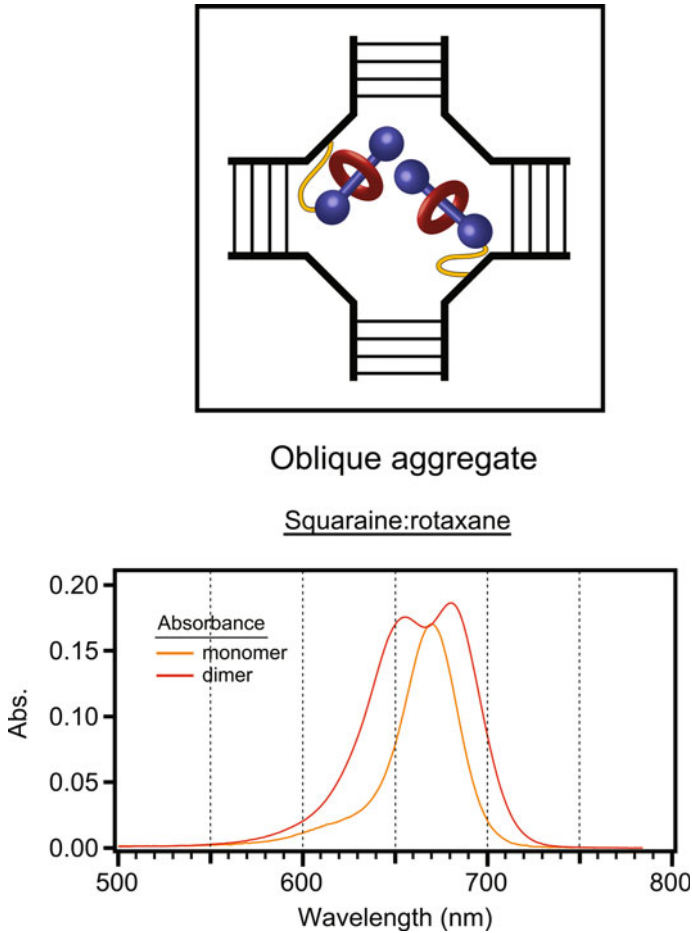


Fig. 2 A dimer dye aggregate whose absorbance spectrum exhibits Davydov splitting. The aggregate consists of two “squaraine-rotaxane” dyes (SeTau-670 from SETA BioMedicals) confined to the core of a DNA Holliday junction by covalent linkages, as shown schematically in the top panel. In the bottom panel, the absorbance spectrum of the monomer dye shows a single peak, whereas that of the dimer aggregate shows two peaks, one on either side of the monomer absorbance peak. Based on absorbance and circular dichroism data it was determined that the dyes make an angle of about 85° with respect to each other, a configuration referred to as oblique. Figure panels modified are from Barclay et al. [42]

$$|\psi(t)\rangle = \sum_{k=1}^N \alpha_k e^{-iE_k t/\hbar} |E_k\rangle. \quad (55)$$

As an example of the time dependence that a homodimer aggregate may exhibit, consider the case when at $t = 0$ the state vector is given by

$$|\psi(0)\rangle = \frac{1}{\sqrt{2}}(|S\rangle + |A\rangle). \quad (56)$$

Substituting Eqs. (50) and (51) into this equation yields

$$|\psi(0)\rangle = B_1^\dagger|0\rangle. \quad (57)$$

The initial state Eq. (56) is the state in which the exciton resides only on dye 1. Such a state can be prepared for a dye aggregate in which the two dye molecules have a different orientation. In this case, the polarization of a femtosecond laser light pulse can be oriented so that it is orthogonal to the dipole component of the transition charge density of molecule 2 but not that of dye 1. In this manner, the laser light pulse can only induce an optical transition from the ground state to the lowest optically allowed excited state in dye 1 [18].

As indicated by Eq. (55), the time evolution of this state is given by

$$|\psi(t)\rangle = \frac{1}{\sqrt{2}} \left(e^{-iE_S t/\hbar} |S\rangle + e^{-iE_A t/\hbar} |A\rangle \right). \quad (58)$$

Substitution of Eqs. (50) through (53) into this equation yields

$$|\psi(t)\rangle = e^{-iE^\epsilon t/\hbar} \left[\cos\left(\frac{Jt}{\hbar}\right) B_1^\dagger|0\rangle - 2i \sin\left(\frac{Jt}{\hbar}\right) B_2^\dagger|0\rangle \right]. \quad (59)$$

From this, it is evident that the probability of finding the exciton on dye 1 as a function of time is

$$P_1(t) = \cos^2\left(\frac{Jt}{\hbar}\right), \quad (60)$$

whereas the probability of finding the exciton on dye 2 is given by

$$P_2(t) = \sin^2\left(\frac{Jt}{\hbar}\right). \quad (61)$$

Hence, at the instances of time $t = n\pi\hbar/J$, where n is an integer, the exciton resides entirely on dye 1; at the instances of time $(n + 1/2)\pi\hbar/J$, the exciton resides entirely on dye 2. The exciton thus hops back and forth between the two dyes with a frequency of $\pi\hbar/J$. Optical experiments using femtosecond light pulses are able to reveal such coherent oscillations. This phenomenon is referred to as exciton coherence.

Having discussed exciton delocalization and exciton coherence, we now move on to exciton devices.

7 Exciton Transmission Lines

For a many-particle quantum walk-based quantum computer, a means is required to transport particles from the output of one gate to the input of the next. Here it is shown that a dye aggregate consisting of a linear array of molecules can function as an exciton transmission line and thus can serve as a wire connecting gates [43].

For simplicity, consider an infinite array of identical dye molecules equally spaced and the case when only one exciton is present. One can then drop the exciton–exciton interaction terms of the Hamiltonian Eq. (27) and the Hamiltonian becomes

$$H = E^e \sum_{r=-\infty}^{\infty} B_r^\dagger B_r + J \sum_{m=-\infty}^{\infty} (B_{r+1}^\dagger B_r + B_r^\dagger B_{r+1}) \quad (62)$$

where, for simplicity, all but nearest neighbor interactions have also been neglected, an approximation that is generally satisfactory because $J_{m,n}$ falls off as the reciprocal of the cube of the distance between dyes m and n .

The system is invariant under translation by the lattice spacing. Group representation theory then indicates that the one-exciton energy eigenstates must have the Bloch form

$$|k\rangle = \frac{1}{\sqrt{2\pi}} \sum_{r=-\infty}^{\infty} e^{ikr} B_r^\dagger |0\rangle, \quad (63)$$

where k is a real number restricted to the range $-\pi < k \leq \pi$ due to the periodic nature of the functions e^{ikr} . Applying this state to the Hamiltonian Eq. (62) yields

$$H|k\rangle = [E^e + 2J \cos(k)] |k\rangle, \quad (64)$$

demonstrating directly that $|k\rangle$ is an eigenstate of the Hamiltonian Eq. (62) having the energy eigenvalue

$$E_k = E^e + 2J \cos(k). \quad (65)$$

The general one-exciton state for an exciton residing on the dye array has the form

$$|\psi(t)\rangle = \frac{1}{\sqrt{2\pi}} \int_{-\pi}^{\pi} dk f(k) e^{-iE_k t/\hbar} |k\rangle, \quad (66)$$

where $f(k)$ is a complex function of k . This is a generalization of Eq. (55) for the case when the state label is a continuous variable rather than a member of a discrete set. Introducing the frequencies

$$\omega_k = \frac{E_k}{\hbar}; \quad \omega_e = \frac{E^e}{\hbar}; \quad \text{and} \quad \omega_J = \frac{J}{\hbar}, \quad (67)$$

the dispersion relation Eq. (65) can be written as

$$\omega_k = \omega_e + 2\omega_J \cos(k) \quad (68)$$

and Eq. (66), with the aid of Eq. (63), can be put into the form

$$|\psi(t)\rangle = \frac{1}{2\pi} \sum_{r=-\infty}^{\infty} \int_{-\pi}^{\pi} f(k) e^{-i(\omega_k t - kr)} dk B_r^\dagger |0\rangle. \quad (69)$$

In this form, it is evident that the general one-exciton state consists of a wave packet of waves with an oscillatory function of time and position along the dye array given by $e^{-i(\omega_k t - kr)}$. Thus, the exciton propagates along the transmission line in a wave-like manner.

When $f(k)$ is strongly peaked with a narrow width about a particular k , the concept of wave packet velocity (known as group velocity) becomes meaningful and is given by

$$v_g \equiv a \frac{\partial \omega_k}{\partial k}, \quad (70)$$

where a is the lattice spacing (the nearest neighbor distance between the dyes). From Eq. (68) one obtains

$$v_g = -2\omega_J a \sin(k). \quad (71)$$

The magnitude of the group velocity is greatest when

$$k = \pm \frac{\pi}{2} \quad (72)$$

and has the value $2\omega_J a$. This is the speed limit for signals propagating along the transmission line.

Because an exciton wave with definite k has the oscillatory form $e^{-i(\omega_k t - kr)}$, one sees that at the group velocity maximum the wavelength of the exciton is four lattice units long. A quarter wavelength or $\pi/2$ phase shift is present between two neighboring dyes at any instant of time. This observation will play a significant role in the discussion of exciton-based basis change gates.

8 Representation of an Exciton Qubit

In the design of quantum computers, a decision needs to be made on how information is to be encoded in the physical hardware. An obvious choice for excitons undergoing a many-particle quantum walk over a dye aggregate would be to let the absence of an exciton denote a logical 0 and the presence of an exciton denote a logical 1. Because

the exciton can exist in a superposition of being present or absent, this is a qubit. This is the representation that Childs et al. [20] chose in their proof that universal quantum computation can be implemented in Bose–Hubbard systems.

Here an alternative encoding will be employed where a qubit is carried by two transmission lines, say lines 1 and 2. If the exciton is on line 1 that is to be regarded as a logical 0. If it is on line 2 that is to be regarded as a logical 1. This is referred to as the “dual-rail” mode of operation. The exciton can be in a superposition state for which there is a probability amplitude α_1 of its being on line 1 and a probability amplitude α_2 of its being on line 2, so this coding indeed implements a qubit. This dual-rail representation of a qubit enables the simplification of gate design but at the cost of having twice as many wires (exciton transmission lines) connecting the gates.

9 Basis Change Gates

We now consider aggregates that function as quantum gates. These aggregates are connected to exciton transmission lines that supply the input signals and deliver the output. A general mathematical framework is developed before considering specific gates.

Equation (63) suggests that one introduces the annihilation operator

$$B(k) = \frac{1}{\sqrt{2\pi}} \sum_{r=-\infty}^{\infty} e^{ikr} B_r. \quad (73)$$

Its Hermitian conjugate is a creation operator which, when operating on $|0\rangle$, produces a one-exciton state with wave vector k and frequency ω_k . It is an energy eigenstate with the energy eigenvalue $E_k = \hbar\omega_k$ given by Eq. (65). A consequence of the infinite sum in Eq. (73) and the continuous nature of the index k is that these creation and annihilation operators satisfy the commutation relations

$$[B(k), B(k')] = 0 \quad (74)$$

and

$$[B(k), B^\dagger(k')] = \delta(k - k'), \quad (75)$$

where $\delta(k - k')$ is a functional referred to as the Dirac delta function.

The exciton of state $B^\dagger(-k)|0\rangle$ propagates in the opposite direction from the exciton of state $B^\dagger(k)|0\rangle$.

We now consider the case of a one-qubit gate. In the dual-rail representation, this gate will have two transmission lines carrying the input qubit and two transmission lines carrying the output qubit. The situation is depicted in Fig. 3. Because signals can propagate both ways on each transmission line, we must consider eight annihilation operators. We employ the labelings $B_\beta^\alpha(k)$ where the superscript $\alpha \in \{\text{in}, \text{out}\}$

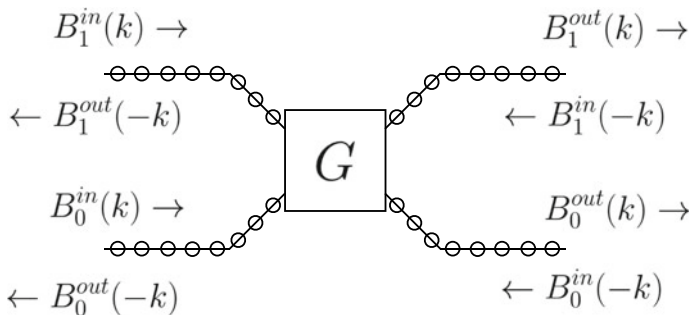


Fig. 3 A general one qubit gate G connected to four exciton transmission lines. The transmission lines consist of arrays of dyes that are here represented by circles placed along a line

indicates whether the exciton is propagating “in” toward the gate G or “out” away from the gate. The subscript $\beta \in \{0, 1\}$ indicates whether the exciton resides on the logical 0 or the logical 1 transmission line of the qubit, and the argument k indicates the value of the wave vector.

Consider the case when all the transmission lines have the same E^e and J . The exciton associated with the exciton creation operator $B_1^{\text{in}}(k)$, after arriving at the gate G , will exit one of the four outputs $B_0^{\text{out}}(k)$, $B_1^{\text{out}}(k)$, $B_0^{\text{out}}(-k)$ and $B_1^{\text{out}}(-k)$. Because energy is conserved and the transmission lines are identical, if the incoming exciton has wave vector k , the outgoing exciton can only have the wave vector k or $-k$. Hence, one need only consider the annihilation operators shown in Fig. 3. One has a similar situation with excitons entering the other input ports. The relation between the input and output annihilation operators is given by

$$\begin{bmatrix} B_1^{\text{out}}(k) \\ B_0^{\text{out}}(k) \\ B_1^{\text{out}}(-k) \\ B_0^{\text{out}}(-k) \end{bmatrix} = \begin{bmatrix} S_{11} & S_{12} & S_{13} & S_{14} \\ S_{21} & S_{22} & S_{23} & S_{24} \\ S_{31} & S_{32} & S_{33} & S_{34} \\ S_{41} & S_{42} & S_{43} & S_{44} \end{bmatrix} \begin{bmatrix} B_1^{\text{in}}(k) \\ B_0^{\text{in}}(k) \\ B_1^{\text{in}}(-k) \\ B_0^{\text{in}}(-k) \end{bmatrix}. \quad (76)$$

The matrix elements $S_{m,n}$ are, in general, complex numbers. The square matrix containing the $S_{m,n}$ is referred to as a scattering matrix. Let this matrix be denoted by \mathbf{S} . Because the incoming signals are independent, the “in” creation and annihilation operators must satisfy commutation relations similar to those of Eqs. (71) and (72)

$$[B_\beta^{\text{in}}(k), B_{\beta'}^{\text{in}}(k')] = 0 \quad (77)$$

and

$$[B_\beta^{\text{in}}(k), B_{\beta'}^{\text{in}\dagger}(k')] = \delta_{\beta,\beta'} \delta(k - k'). \quad (78)$$

Similarly the outgoing singles are linearly independent of each other and consequently the “out” creation and annihilation operators also satisfy commutation rela-

tions of the form Eqs. (77) and (78). For all these commutation relations to be satisfied, the matrix \mathbf{S} must be unitary. That \mathbf{S} be unitary is also required by the conservation of energy and is a manifestation of the unitary evolution imposed by the Schrödinger equation.

When the device simply consists of two parallel transmission lines that are sufficiently far apart that exciton hopping from one transmission line to the other does not occur there is no scattering and the \mathbf{S} matrix is diagonal

$$\mathbf{S} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (79)$$

When two exciton transmission lines are brought sufficiently close so that the oscillatory Coulomb interaction can occur between dye pairs, with one dye located on each transmission line, an exciton can hop from one transmission line to the other. Thereby nondiagonal entries of the \mathbf{S} array become nonzero.

To make a specific quantum gate, such as a Hadamard gate, the challenge is to engineer a dye aggregate that produces the desired matrix elements $S_{m,n}$. For this, the electrical engineering literature on distributed element circuits serves as a useful guide. These radio frequency and microwave circuits are based on wave interference. In these circuits, transmission line segments a quarter of a wavelength long play a prominent role. One such device is shown in Fig. 4a. It is an example of a branch line coupler [44]. Signals entering port 1 of this coupler exit ports 2 and 3, and no signal exits port 4. For the case shown, the transmission line segments that are narrow have a transmission line impedance of Z_0 , whereas the two transmission line segments that are wide have a transmission line impedance of $Z_0/\sqrt{2}$. With these impedance values, the signal entering port 1 is split evenly between ports 2 and 3, that is, the device functions as a 50/50 beam splitter. Because distributed element circuits rely on wave interference they generally function well over a limited range of wavelengths (or frequencies) centered about a midband wavelength (or frequency) determined by the length of the transmission line segments.

The device shown in Fig. 4b is a direct translation of the device of Fig. 4a into an exciton device. Because there is a quarter wavelength shift in the phase between neighboring dyes at the band center $k = \pi/2$, the distance between the dyes in effect serves as a quarter wavelength section of transmission line. The energy parameter J plays the role of the reciprocal of impedance. Because the value of J depends on the spacing between dyes, the dye spacing can be adjusted to yield the desired value. In this case, the coupling between all nearest neighbor dyes is J except for the two having the value $\sqrt{2}J$, as indicated in the figure by arrows.

At band center, the scattering matrix Eq. (73) for this device is

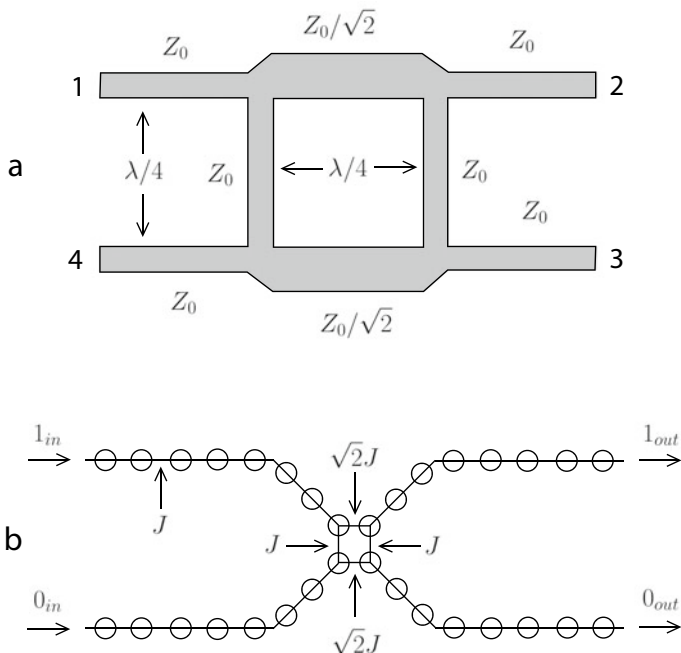


Fig. 4 A branch line coupler as implemented in radio and microwave frequency electronics (a) and as an exciton device implemented using dye molecules (b). For the values of the impedance or exciton exchange energies shown, these devices act as 50/50 power dividers. The exciton device (b) functions as a basis change gate

$$\mathbf{S} = \frac{1}{\sqrt{2}} \begin{bmatrix} i & 1 & 0 & 0 \\ 1 & i & 0 & 0 \\ 0 & 0 & i & 1 \\ 0 & 0 & 1 & i \end{bmatrix}. \quad (80)$$

From the zeros in this matrix, it is evident that signals entering ports B_1^{in} and B_0^{in} are only delivered to ports B_1^{out} and B_0^{out} . Hence, the device can be regarded as a one-qubit gate in which the input enters the left side of the device and the output exits the right side as shown in Fig. 4b, where the logical 1 and logical 0 lines have been indicated. The transformation performed on the annihilation operators is

$$\begin{bmatrix} B_1^{\text{out}} \\ B_0^{\text{out}} \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} i & 1 \\ 1 & i \end{bmatrix} \begin{bmatrix} B_1^{\text{in}} \\ B_0^{\text{in}} \end{bmatrix}. \quad (81)$$

This transformation can be inverted to express the annihilation operators of the incoming signals in terms of the annihilation operators of the outgoing signals.

$$\begin{bmatrix} B_1^{\text{in}} \\ B_0^{\text{in}} \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} -i & 1 \\ 1 & -i \end{bmatrix} \begin{bmatrix} B_1^{\text{out}} \\ B_0^{\text{out}} \end{bmatrix}. \quad (82)$$

Consider the case when the state of the system $|\psi_S\rangle$ is that for which the incoming signal is a logical 1:

$$|\psi_S\rangle = B_1^{\text{in}\dagger}|0\rangle. \quad (83)$$

From Eq. (82), one has

$$B_1^{\text{in}} = \frac{1}{\sqrt{2}} (B_1^{\text{out}} - i B_0^{\text{out}}). \quad (84)$$

Substituting this into Eq. (83) and remembering that taking the Hermitian adjoint involves complex conjugation one obtains

$$|\psi_S\rangle = \frac{1}{\sqrt{2}} \left(B_1^{\text{out}\dagger}|0\rangle + i B_0^{\text{out}\dagger}|0\rangle \right). \quad (85)$$

Hence, this system state is one for which the exciton exits the gate as a superposition state in which the probability amplitude that the exciton is on the logical 1 line is $1/\sqrt{2}$ and the probability amplitude that the exciton is on the logical 0 line is $i/\sqrt{2}$.

The gate of Fig. 4b is capable of exhibiting the ideal performance of Eq. (81) at the transmission line band center $k = \pm\pi/2$; however, its performance degrades away from band center. But the degradation is graceful in that there is a finite bandwidth over which the device functions satisfactorily for any specified tolerance level. A full analysis of this gate is presented in [43]. Greater bandwidth than that exhibited by the gates of Fig. 4 can be achieved with more complex gate designs, the theory of which is well developed for distributed element circuits.

10 Phase Gates

Phase shifts can be implemented as propagation delays. Figure 5 illustrates two transmission lines along which a qubit in the dual-rail representation propagates. For the phase gate shown in Fig. 5, one line has been made one dye longer than the other. For signals propagating at the midband of the transmission line, this extra length induces a quarter wavelength propagation delay (a $\pi/2$ phase shift) with respect to the shorter transmission line. The transformation performed by this gate is given by

$$\begin{bmatrix} B_1^{\text{out}} \\ B_0^{\text{out}} \end{bmatrix} = \begin{bmatrix} e^{i\pi/2} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} B_1^{\text{in}} \\ B_0^{\text{in}} \end{bmatrix}. \quad (86)$$

Propagation delays can be induced by other means. From Eq. (63), it is evident that the phase factor between neighboring dyes is e^{ik} , that is, the phase shift is k . Taking the energy E_k of the exciton (or the carrier frequency ω_k of the signal) to

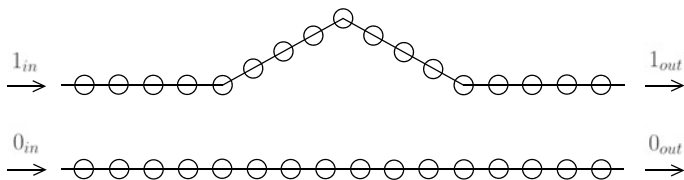


Fig. 5 A phase gate consisting of two transmission lines. The phase shift results from propagation a delay of the signal traveling over the longer transmission line. For the case shown, where one transmission line is one dye longer than the other, the phase shift is $\pi/2$ at the transmission line midband

be the controlling variable, the value of k is obtained by solving Eq. (65). One sees that it depends on E^e and J . Hence, phase shifts can be induced by having E^e or J differ in sections of one transmission line by employing different dyes (changes E^e and J) or by changing the spacing between neighboring dyes (changes J). We thus posit that any phase gate of the form of Eq. (18) can be engineered.

It is now shown how the basis change gate exhibited in Eq. (81),

$$U_B = \frac{1}{\sqrt{2}} \begin{bmatrix} i & 1 \\ 1 & i \end{bmatrix}, \quad (87)$$

can be converted into a Hadamard gate by sandwiching it between two phase gates of the form

$$U_P = \begin{bmatrix} e^{-i\pi/4} & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}. \quad (88)$$

Carrying out the matrix multiplication

$$U_H = U_P U_B U_P \quad (89)$$

one finds that U_H is the Hadamard gate Eq. (17).

An alternative means of implementing a Hadamard gate would be to translate a hybrid ring coupler, also called a rat-race coupler [44], (a distributed element circuit device) into an exciton device as was done for the branch line coupler of Fig. 4.

11 An Exciton Interferometer

As an illustration of how single qubit gates can be composed to produce new single qubit gates, exciton interferometers will now be discussed. This will also lay the foundation for a discussion of the CNOT gate. An exciton interferometer can be constructed as a phase gate sandwiched between two basis change gates. As an

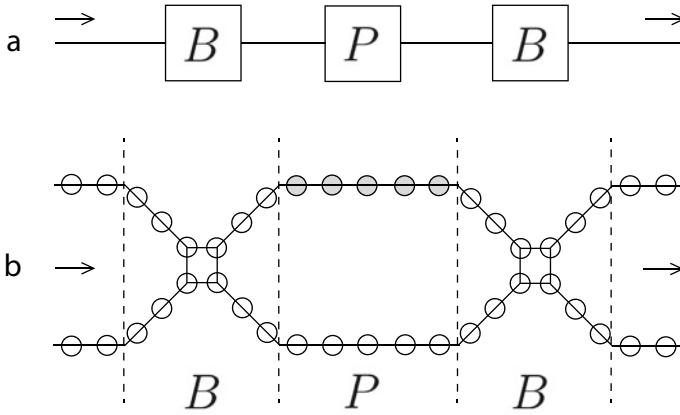


Fig. 6 An exciton interferometer. **a** A schematic representation of the device as a cascade of a branch line coupler gate, a phase gate and a second branch line coupler. **b** The physical layout of the dyes forming an interferometer. The upper transmission line of the phase gate has been shaded to indicate that its propagation delay (phase shift) may be different from that of the lower transmission line

example, consider the phase gate of Eq. (18) sandwiched between two basis change gates given by Eq. (87). The configuration is illustrated schematically in Fig. 6a as a cascade of gates, and the physical layout of the dyes is shown in Fig. 6b.

The overall transformation is given by

$$U_I = U_B U_P U_B \quad (90)$$

Carrying out the matrix multiplication yields

$$U_I = i e^{i(\phi_1 + \phi_2)/2} \begin{bmatrix} -\sin[(\phi_1 - \phi_2)/2] & \cos[(\phi_1 - \phi_2)/2] \\ \cos[(\phi_1 - \phi_2)/2] & \sin[(\phi_1 - \phi_2)/2] \end{bmatrix}. \quad (91)$$

From this, it is evident that how an exciton entering one port of the interferometer is distributed among the output ports depends sinusoidally on the phase difference $\phi_1 - \phi_2$ of the phases of the phase gate. This composition of gates thus functions as an interferometer that is sensitive to the phase difference between the two arms (transmission lines) internal to the interferometer.

To simplify the discussion, consider the case when ϕ_2 has the fixed value π . Then Eq. (91) reduces to

$$U_I(\phi) = -e^{i\phi/2} \begin{bmatrix} \cos(\phi/2) & \sin(\phi/2) \\ \sin(\phi/2) & -\cos(\phi/2) \end{bmatrix}, \quad (92)$$

where we have set $\phi_1 = \phi$.

When $\phi = 0$, this matrix reduces to

$$U(0) = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}. \quad (93)$$

In this case, an exciton entering on the logical 1 line exits on the logical 1 line and an exciton entering on the logical 0 line exits on the logical 0 line. Hence, for Boolean inputs, the Boolean value remains unchanged as the qubit passes through the interferometer with the setting $\phi = 0$.

When $\phi = \pi$, Eq. (92) reduces to

$$U(\pi) = -i \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \quad (94)$$

Now a qubit entering as a logical 1 exits as a logical 0 and a qubit entering as a logical 0 exits as a logical 1. Thus, with the phase ϕ set at π the interferometer acts as a NOT gate. Now, if one had a means to switch ϕ between 0 and π , one would have a controlled NOT gate. How such a switching element can be constructed is discussed next.

12 A Controlled Phase Shift

To convert the interferometer of Fig. 6 into a controlled gate, a controlled phase shifting element is needed in which one exciton controls the phase shift of another. This requires an exciton–exciton interaction. Here the changes in the static Coulomb interaction between dyes resulting from changes in the charge distribution, when a dye transitions from the ground state to the lowest optically allowed excited state, are utilized, that is, the $K_{m,n}$ interactions of Eq. (27) are employed.

A means is required to enable two excitons to interact in a controlled manner. A way to accomplish this is shown in Fig. 7. Shown are two parallel transmission lines that differ such that the group velocity of the upper transmission line is less than that of the lower transmission line. The dyes of the two transmission lines are oriented such that the exchange energy between dyes on separate transmission lines is zero. This prevents the transitioning of an exciton from one transmission line to the other. As discussed in the **Appendix**, since the exciton–exciton interaction energy arises from a different mechanism than of the exchange energy, the inter-transmission line exciton–exciton interaction energy need not be zero even though the exciton exchange energy is. By running the two transmission lines close to each other the inter-transmission line strength K of the exciton–exciton interaction can be made large.

For operation, exciton wave packets are introduced on both transmission lines; however, the packet on the upper transmission line is introduced first to give it a head start as shown in Fig. 7a. The exciton wave packet on the lower transmission line

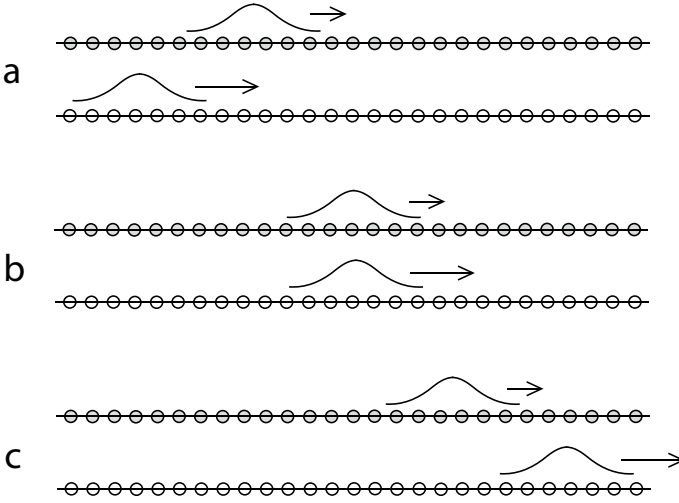


Fig. 7 A controlled phase shifting element. The device consists of two transmission lines which have no inter-transmission line J coupling, but inter-transmission line K coupling, enabling an exciton on one transmission line to change the phase of an exciton on the other transmission line. The dyes of the upper transmission line are shaded to indicate that the signal propagation speed is slower on that transmission line. That enables an exciton wave packet of the lower transmission line to overtake an exciton wave packet of the upper transmission line, as shown at successive time snapshots (a), (b) and (c). Thereby, the two excitons are ensured to interact regardless of where each resides in its wave packet

catches up with the wave packet of the upper transmission line as shown in Fig. 7b and then surpasses it as shown in Fig. 7c. Where the excitons are located in each wave packet is unknown, but, because one wave packet completely overtakes the other, the two excitons are guaranteed to interact. The interaction energy is short range but has the value K when the two excitons are directly across from each other. The phase winding induced by the interaction can be estimated using $e^{-iKt_I/\hbar}$, where t_I is the time interval over which the excitons interact. Let J_1 and J_2 denote the strength of the hopping interaction for the upper and lower transmission line, respectively, then from the expression for the group velocity at midband Eq. (71) and from Eq. (67), the magnitude of the group velocity difference is

$$\Delta v_g = \frac{2|J_2 - J_1|a}{\hbar}. \quad (95)$$

Because the excitons strongly interact only when they are within a lattice unit a of each other, an estimate of the interaction time is

$$t_I = \frac{a}{|\Delta v_g|} = \frac{\hbar}{2|J_2 - J_1|}. \quad (96)$$

Hence, the phase winding is $e^{-iK/2|J_2-J_1|}$, that is, the phase accumulated during the interaction is

$$\phi_I = -\frac{K}{2|J_2 - J_1|}. \quad (97)$$

The value of ϕ_I can be engineered through choice of K , J_1 or J_2 . Let $|\psi_{\text{in}}\rangle$ denote the state of the system before the interaction and $|\psi_{\text{out}}\rangle$ the state of the system after the interaction. The relation between these two states is

$$|\psi_{\text{out}}\rangle = e^{i\phi_I} |\psi_{\text{in}}\rangle. \quad (98)$$

Because the incoming and outgoing states are both two-exciton states and the excitons cannot transition from one transmission line to the other, the exciton–exciton interaction can be expressed in terms of the incoming and outgoing annihilation operators by

$$B_1^{\text{out}} B_2^{\text{out}} = e^{i\phi_I} B_1^{\text{in}} B_2^{\text{in}}. \quad (99)$$

13 A CNOT Gate

Here we consider the CNOT gate shown in Fig. 8. This device is implemented by adding to the interferometer circuit of Fig. 6 two more transmission lines that carry the control qubit. The lower of these two transmission lines comes in close proximity to the transmission line of the upper arm of the interferometer to form the controlled phase shifter discussed in Sect. 12. Here A_m , B_m , C_m and D_m denote the annihilation operators for the right propagating exciton modes at various points in the device. They also serve as position markers within the device. The input exciton modes are denoted by A_m^{in} and the output exciton modes are denoted by D_m^{out} . The controlled phase shifter consists of the parallel transmission line segments B_2 – C_2 and B_3 – C_3 .

The core of the CNOT gate is analyzed first. Because a dual-rail representation is employed, when the control qubit and target qubit enter the device, two excitons are present in device, one residing in the control transmission lines and the other residing in the interferometer. Hence, to analyze the performance of the core of this device, one can restrict the analysis to the state space spanned by the basis set:

$$\mathbf{B}_{CT} = \left\{ A_1^\dagger A_3^\dagger |0\rangle, A_1^\dagger A_4^\dagger |0\rangle, A_2^\dagger A_3^\dagger |0\rangle, A_2^\dagger A_4^\dagger |0\rangle \right\}. \quad (100)$$

Expressed in terms of the B_m^\dagger creation operators, the A_m^\dagger creation operators are given by

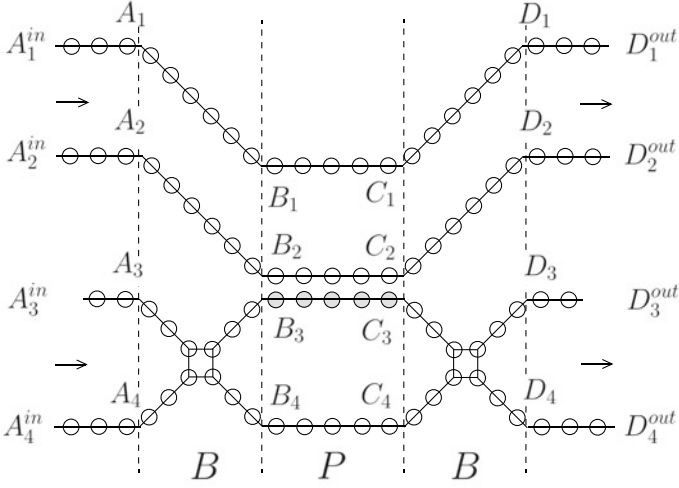


Fig. 8 A controlled NOT gate (CNOT) consisting of an interferometer with one arm coupled to one transmission line of the control qubit transmission line pair to form a controlled phase shift element. The interferometer consists of a branch line coupler B , followed by the phase shifting elements P , followed in turn by a second branch line coupler. Note that the input and output transmission line 3 has been shortened by one dye relative to the other input and output transmission lines. This implements propagation delays that put the scattering matrix of the device in standard form for that of a CNOT gate

$$A_1^\dagger = B_1^\dagger \quad (101)$$

$$A_2^\dagger = B_2^\dagger \quad (102)$$

$$A_3^\dagger = \frac{1}{\sqrt{2}} \left(-iB_3^\dagger + B_4^\dagger \right) \quad (103)$$

$$A_4^\dagger = \frac{1}{\sqrt{2}} \left(B_3^\dagger - iB_4^\dagger \right). \quad (104)$$

The first two equations express propagation along the control qubit transmission lines. The second two equations express the basis change transformation performed by the first branch line coupler, see Eq. (82). With relationships Eq. (101) through (104), one obtains

$$\begin{aligned} A_1^\dagger A_3^\dagger &= \frac{1}{\sqrt{2}} \left(-iB_1^\dagger B_3^\dagger + B_1^\dagger B_4^\dagger \right) \\ A_1^\dagger A_4^\dagger &= \frac{1}{\sqrt{2}} \left(B_1^\dagger B_3^\dagger - iB_1^\dagger B_4^\dagger \right) \\ A_2^\dagger A_3^\dagger &= \frac{1}{\sqrt{2}} \left(-iB_2^\dagger B_3^\dagger + B_2^\dagger B_4^\dagger \right) \\ A_2^\dagger A_4^\dagger &= \frac{1}{\sqrt{2}} \left(B_2^\dagger B_3^\dagger - iB_2^\dagger B_4^\dagger \right). \end{aligned} \quad (105)$$

The relationships between B_m^\dagger and C_m^\dagger are now established. Because excitons on transmission lines 1 and 4 do not interact with excitons on the other transmission lines in the region P , one has

$$B_1^\dagger = C_1^\dagger \quad (106)$$

and

$$B_4^\dagger = e^{i\phi_F} C_4^\dagger \quad (107)$$

where, in writing the last equation, allowance has been made for a fixed propagation delay characterized by the fixed phase ϕ_F that can be engineered to have a desired value. When no exciton exists on transmission line 3, the exciton on transmission line 2 does not interact with excitons on the other transmission lines. In this case

$$B_2^\dagger = C_2^\dagger. \quad (108)$$

One thus has the relations

$$B_1^\dagger B_3^\dagger = C_1^\dagger C_3^\dagger \quad (109)$$

$$B_1^\dagger B_4^\dagger = e^{i\phi_F} C_1^\dagger C_4^\dagger \quad (110)$$

$$B_2^\dagger B_4^\dagger = e^{i\phi_F} C_2^\dagger C_4^\dagger. \quad (111)$$

When an exciton exists on each of transmission lines 2 and 3, the two excitons interact, and from Eq. (99) one has

$$B_2^\dagger B_3^\dagger = e^{i\phi_I} C_2^\dagger C_3^\dagger. \quad (112)$$

The equations of (105) now yield

$$\begin{aligned} A_1^\dagger A_3^\dagger &= \frac{1}{\sqrt{2}} \left(-i C_1^\dagger C_3^\dagger + e^{i\phi_F} C_1^\dagger C_4^\dagger \right) \\ A_1^\dagger A_4^\dagger &= \frac{1}{\sqrt{2}} \left(C_1^\dagger C_3^\dagger - i e^{i\phi_F} C_1^\dagger C_4^\dagger \right) \\ A_2^\dagger A_3^\dagger &= \frac{1}{\sqrt{2}} \left(-i e^{i\phi_I} C_2^\dagger C_3^\dagger + e^{i\phi_F} C_2^\dagger C_4^\dagger \right) \\ A_2^\dagger A_4^\dagger &= \frac{1}{\sqrt{2}} \left(e^{i\phi_I} C_2^\dagger C_3^\dagger - i e^{i\phi_F} C_2^\dagger C_4^\dagger \right). \end{aligned} \quad (113)$$

The mode transformations between C_m^\dagger and D_m^\dagger are similar to those of Eqs. (101) through (104) and will not be presented here. One finds

$$\begin{aligned}
A_1^\dagger A_3^\dagger &= \frac{1}{2} \left[(e^{i\phi_F} - 1) D_1^\dagger D_3^\dagger - i (e^{i\phi_F} + 1) D_1^\dagger D_4^\dagger \right] \\
A_1^\dagger A_4^\dagger &= \frac{1}{2} \left[-i (e^{i\phi_F} + 1) D_1^\dagger D_3^\dagger - (e^{i\phi_F} - 1) D_1^\dagger D_4^\dagger \right] \\
A_2^\dagger A_3^\dagger &= \frac{1}{2} \left[(e^{i\phi_F} - e^{i\phi_I}) D_2^\dagger D_3^\dagger - i (e^{i\phi_F} + e^{i\phi_I}) D_2^\dagger D_4^\dagger \right] \\
A_2^\dagger A_4^\dagger &= \frac{1}{2} \left[-i (e^{i\phi_F} + e^{i\phi_I}) D_2^\dagger D_3^\dagger - (e^{i\phi_F} - e^{i\phi_I}) D_2^\dagger D_4^\dagger \right]. \quad (114)
\end{aligned}$$

Now the fixed phase on transmission line 4 is chosen to be $\phi_F = \pi$ and the interaction-induced phase shift, Eq. (97), is chosen to be $\phi_I = \pi$. The transformation (114) then becomes

$$\begin{aligned}
A_1^\dagger A_3^\dagger |0\rangle &= -D_1^\dagger D_3^\dagger |0\rangle \\
A_1^\dagger A_4^\dagger |0\rangle &= D_1^\dagger D_4^\dagger |0\rangle \\
A_2^\dagger A_3^\dagger |0\rangle &= i D_2^\dagger D_4^\dagger |0\rangle \\
A_2^\dagger A_4^\dagger |0\rangle &= i D_2^\dagger D_3^\dagger |0\rangle. \quad (115)
\end{aligned}$$

By identifying an exciton on transmission lines 1 and 3 to correspond to a Boolean 0 and an exciton on transmission lines 2 and 4 to correspond to a Boolean 1, the scattering matrix corresponding to the basis transformation Eq. (115) is given by

$$\begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -i \\ 0 & 0 & -i & 0 \end{bmatrix}. \quad (116)$$

Comparing this with Eq. (20) one sees that this transformation is not quite the standard CNOT gate transformation, but it does exhibit CNOT functionality.

The standard CNOT gate can be obtained from this core by introducing a phase shift of $-\pi/2$ at the transmission line 3 input port and the output port. This is implemented in Fig. 8 by making the transmission line segments for input and output ports 3 one dye shorter than the corresponding transmission line segments for the other input and output ports. That is, one implements the transformations

$$\begin{aligned}
A_3^{\text{in}\dagger} &= -i A_3^\dagger \\
D_3^\dagger &= -i D_3^{\text{out}\dagger} \\
A_m^{\text{in}\dagger} &= A_m^\dagger \text{ if } m \in \{1, 2, 4\} \\
D_m^\dagger &= D_m^{\text{out}\dagger} \text{ if } m \in \{1, 2, 4\}. \quad (117)
\end{aligned}$$

With these equations, Eq. (115) yields

$$\begin{aligned}
 A_1^{\text{in}\dagger} A_3^{\text{in}\dagger} |0\rangle &= D_1^{\text{out}\dagger} D_3^{\text{out}\dagger} |0\rangle \\
 A_1^{\text{in}\dagger} A_4^{\text{in}\dagger} |0\rangle &= D_1^{\text{out}\dagger} D_4^{\text{out}\dagger} |0\rangle \\
 A_2^{\text{in}\dagger} A_3^{\text{in}\dagger} |0\rangle &= D_2^{\text{out}\dagger} D_4^{\text{out}\dagger} |0\rangle \\
 A_2^{\text{in}\dagger} A_4^{\text{in}\dagger} |0\rangle &= D_2^{\text{out}\dagger} D_3^{\text{out}\dagger} |0\rangle.
 \end{aligned}
 \tag{118}$$

The scattering matrix corresponding to this transformation is

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix},
 \tag{119}$$

which is the standard scattering matrix, Eq. (20), for the CNOT gate [33].

This completes the demonstration that a set of gates enabling universal quantum computation can be implemented as suitably constructed dye aggregates in which the exciton dynamics is governed by the Frenkel Hamiltonian Eq. (27).

14 Exciton-Based Quantum Computer Architecture

Having discussed individual gates, the overall architecture of an exciton-based quantum computer is presented. Figure 9 indicates what the quantum computer might look like, how it is initialized and how the result of the computation is delivered as output.

Between the two vertical dashed lines is the computer circuit itself. In this case, the circuit for a Fredkin gate was chosen as a stand-in for a general quantum computer circuit [45]. In this case, three qubit lines run parallel to each other from left to right. In the dual-rail representation each of these lines consists of two transmission lines. The boxes represent single qubit gates. The boxes labeled *H* are Hadamard gates that perform the transformation given by Eq. (17). The boxes labeled *T* are phase gates performing the transformation

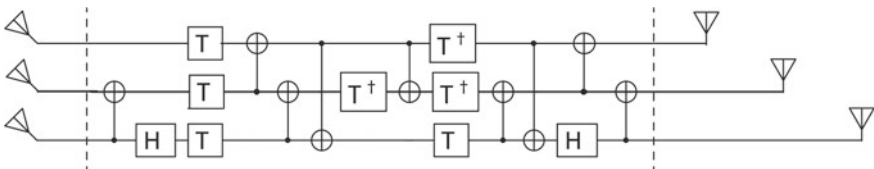


Fig. 9 A schematic indicating the architecture of an exciton-based many-body quantum walk quantum computer. See text for details

$$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/2} \end{bmatrix}.$$

Coupling between qubit wires are through the CNOT gates (Fig. 1).

The input ports are represented by the antennas at the left end of each qubit line. Each of these antennas would couple to a separate electromagnetic field mode. On-demand single photon sources would be coupled to these antennas to generate the exciton input state. In principle the conversion of a photon into an exciton can be done with unit quantum efficiency, but in practice the coupling optics and antenna design could be quite challenging. In addition, the single photon sources would need to emit short optical pulses all timed to simultaneously initiate each qubit. The excitons propagate ballistically (in sync) through the gate network to the output. Note that the number of output lines is the same as the number of input lines. This is a consequence of unitary evolution. The output can be delivered to photodetectors by antennas at the output side (right side of the circuit). These antennas would have the same structure as the input antennas and would, with unit quantum efficiency, convert an exciton to a photon that would then be detected with a unit quantum efficiency photodetector. The output qubit lines (to the right side of the right most vertical dashed line) have been drawn to different lengths. Each qubit will thus arrive at its detector at a different time due to propagation delays. In this manner, the output of the quantum computer is time multiplexed so that the output can be read by noting the arrival time of each photon at the photodetector. This scheme has the advantage that only a single photodetector need be employed, which should simplify the optics that delivers the photons to the photodetector.

The quantum computer, as described, is a special purpose device. For each problem to be addressed by quantum computation, a special purpose circuit would be assembled to carry out that computation. In principle, a general purpose quantum computer could be implemented that uses classical switches to reconfigure the circuits. With DNA nanotechnology this might be done using strand displacement techniques to reconfigure the circuit [46].

15 But Isn't a Quantum Computer Just an Analog Computer?

The question posed in this section heading is often expressed. At first sight, constructing a quantum computer is simply a matter of assembling a physical system that implements a desired unitary transformation. Wave interference effects alone enable the implementation of an arbitrary unitary transformation, and this can be done at the classical level with a collection of optical beam splitters [30]. A quantum computer differs in two crucial ways from a classical computer. First, quantum superposition, in effect, enables the same gate to carry out multiple operations simultaneously, which greatly reduces the parts count or the number of steps needed to carry out a computation for those tasks amenable to quantum speedup. Second, if the

gate error rate is below a certain threshold value, quantum error correction can be implemented, enabling scalable quantum computation with imperfect gates [47, 48]. In contrast, the precision and accuracy of analog computers is generally limited by the precision and accuracy of their components. Error correcting quantum computing schemes have been devised for which the tolerance for errors is about 1% per gate operation, which is still quite demanding. Nevertheless, this is a good reach goal to drive technology and the resulting information processing technology is likely to have applications even if full-scale quantum computing is not achieved, particularly because of the compact size (molecular scale) of the gates and the femtosecond switching time for gates employing optical transitions.

16 Molecular Vibrations

A number of imperfections that occur at the molecular level can give rise to gate errors or present challenges that must be overcome to construct viable gates. These include the dispersiveness inherent in exciton transmission lines consisting of an array of dyes, errors in DNA assembly, and Brownian motion that modulates the gate parameters as a function of time. The most serious “imperfections” result from the interaction of excitons with molecular vibrations. How this interaction is modeled is discussed here.

A molecule in its ground state has a structural configuration characterized by the equilibrium position of the nucleus of each atom. Should the molecule absorb a photon and transition to its lowest optically allowed excited state, the position of the atomic nuclei at the instance of the transition will still be in the ground-state equilibrium configuration, as optical transitions occur on a shorter time scale than that required for the atomic nuclei to readjust their positions. As a consequence, the atomic nuclei are displaced from their excited state equilibrium position. The system responds by converting this potential energy into kinetic energy as the nuclei accelerate toward their excited state equilibrium positions. The result is that the molecule undergoes molecular vibrations. These vibrations couple to the environment thereby providing a means of energy exchange between the molecule and the environment. The result is a scrambling of phases that washes out interference effects. This process is referred to as decoherence. This process degrades quantum gate performance, as these gates rely on quantum interference effects.

The exciton–vibration coupling is well modeled by what is often referred to as the Frenkel–Holstein Hamiltonian [49, 50]. This Hamiltonian can be written as the sum of the Frenkel Hamiltonian H_F of Eq. (27) with a Hamiltonian H_V (the Holstein part) characterizing the dynamics of the molecular vibrations and their coupling to the excitons

$$H = H_F + H_V. \quad (120)$$

The Hamiltonian H_V is given by

$$H_V = \sum_{m=1}^N \sum_{\alpha} E_{m,\alpha}^v a_{m,\alpha}^{\dagger} a_{m,\alpha} + \sum_{m=1}^N \sum_{\alpha} E_{m,\alpha}^v \lambda_{m,\alpha} B_m^{\dagger} B_m (a_{m,\alpha} + a_{m,\alpha}^{\dagger}) \quad (121)$$

where $a_{m,\alpha}$ is the annihilation operator for a quantum of vibration for the α th vibration mode on molecule m and the Hermitian adjoint $a_{m,\alpha}^{\dagger}$ is the corresponding creation operator. These satisfy Bose commutation relations similar to those of the excitons Eqs. (33) and (34). $E_{m,\alpha}^v$ is the energy of a quantum of vibration for the α th vibration mode on molecule m , and $\lambda_{m,\alpha}$ is the displacement between the equilibrium position of the ground and excited electronic state for the α th vibration mode on the m th molecule. The sums are over all vibration modes of the molecule and over all molecules.

Some insight into the consequences of the exciton–vibration coupling can be obtained by considering the Heisenberg equation of motion for the exciton annihilation operator B for a single molecule. For a single molecule the full Hamiltonian, neglecting two-exciton terms, becomes

$$H = E^e B^{\dagger} B + \sum_{\alpha} E_{\alpha}^v a_{\alpha}^{\dagger} a_{\alpha} + \sum_{\alpha} E_{\alpha}^v \lambda_{\alpha} B^{\dagger} B (a_{\alpha} + a_{\alpha}^{\dagger}), \quad (122)$$

where, because we are dealing with only one molecule, the molecule index has been suppressed. The Heisenberg equation of motion for B is given by

$$\frac{dB}{dt} = \frac{i}{\hbar} [H, B]. \quad (123)$$

For the Hamiltonian Eq. (122) this yields

$$\frac{dB}{dt} = -i \left[\omega^e + \sum_{\alpha} \omega_{\alpha}^v \lambda_{\alpha} (a_{\alpha} + a_{\alpha}^{\dagger}) \right] B, \quad (124)$$

where ω_{α}^v is the vibration frequency of the α th vibration mode and is related to the energy of a quantum of vibration by $E_{\alpha}^v = \hbar \omega_{\alpha}^v$. The operator $a_{\alpha} + a_{\alpha}^{\dagger}$ is the position coordinate for the vibrational mode α . Treating this as a classical variable, Eq. (124) can be integrated to yield

$$B(t) = B(0) e^{-i[\omega^e t + \phi_v(t)]}, \quad (125)$$

where $\phi_v(t)$ is a fluctuating phase arising from the motion of all the molecular vibration modes

$$\phi_v(t) = \int_0^t \sum_{\alpha} \omega_{\alpha}^v \lambda_{\alpha} (a_{\alpha}(t') + a_{\alpha}^{\dagger}(t')) dt'. \quad (126)$$

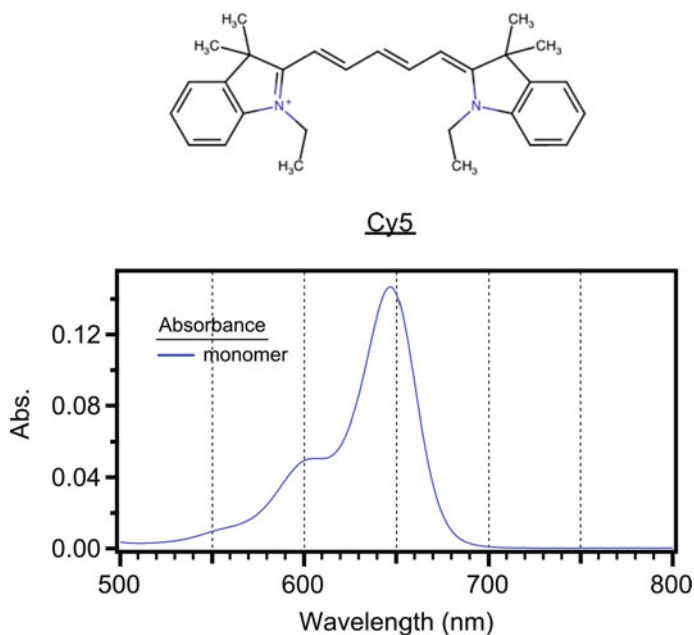


Fig. 10 An example of a dye whose aggregate absorbance and circular dichroism spectra can be well modeled by including only one vibrational mode of the dye. Shown is the cyanine dye Cy5 structure (top panel) and the monomer absorbance spectrum (bottom panel). The shoulder at 600 nm in the absorbance spectrum is due to the dominant vibrational mode. The absorbance units are $10^6 \text{ M}^{-1} \text{ cm}^{-1}$

This random phase causes the phase winding of B to deviate from that for pure sinusoidal oscillation $e^{i\omega t}$. This spoils interference effects and gives a width to the spectral lines of the dye absorption and emission spectra.

It is often the case that a coupling between the exciton and vibration is particularly strong for a single or small group of vibration modes [51]. An example of this is the Cyanine dye Cy5 whose structure and absorbance spectrum [42] are shown in Fig. 10. The absorbance spectrum shows an absorbance maximum at about 650 nm. This corresponds to the optical transition from the ground electronic state with no vibrational quanta to the lowest excited electronic state with no vibrational quanta. The shoulder at 600 nm on the short wave-length side of the peak corresponds to the optical transition from the ground electronic state with no vibrational quanta to the lowest excited electronic state with one vibrational quantum for a dominant vibrational mode. In the case when one dominant vibration mode occurs the absorbance spectrum can be well modeled by including only this one vibrational mode for each molecule. In this case the Hamiltonian Eq. (121) reduces to

$$H_V = \sum_{m=1}^N E_m^v a_m^\dagger a_m + \sum_{m=1}^N E_m^v \lambda_m B_m^\dagger B_m (a_m + a_m^\dagger). \quad (127)$$

This Hamiltonian is simple enough that the energy eigenstates and eigenvalues can be solved to a high degree of accuracy by numerical methods while still adequately accounting for the spectral features in optical absorption spectra of aggregates.

When an exciton moves from one dye molecule to another, the dye it leaves undergoes a transition from the excited state to the ground state while the dye it moves to undergoes a transition from the ground state to the excited state. Both of these transitions induce molecular vibrations; however, due to conservation of energy the exciton cannot endlessly shed vibrations. As a result, the exciton carries a cloud of vibrations (or molecular distortions) with it [52]. The exciton becomes a composite particle, a bundle of electronic and vibrational energy. This modifies the dispersion relation for an exciton propagating along an exciton transmission line. Nevertheless, it should still be possible to engineer exciton quantum gates that work in spite of the composite nature of the exciton.

It is also noted that molecular vibrations exhibit longer decoherence times than excitons [53]. The exciton hopping interaction provides a means of coupling vibrations between dyes. It is thus an interesting question as to whether quantum computing with dye aggregates could be implemented using quanta of molecular vibrations in which the excitons simply provide a means to control the flow of quantum information from dye molecule to dye molecule.

17 Conclusion

It has been shown that exciton-based quantum gates can be constructed from dye aggregates; and the aggregate configurations for a set of dyes sufficient to enable universal quantum computation have been presented. How these could be assembled into circuits with inputs and outputs has been discussed. In addition, nonidealities resulting from the coupling between excitons and molecular vibrations have been treated.

The question arises: What are the prospects for realizing such devices in practice? Exciton delocalization extending over 30–100 dyes has been reported in the literature [24]. The transmission lines in the largest gate presented, Fig. 8, are 23 dyes long. This suggests that it should be possible to experimentally demonstrate the gates that have been presented, as well as small circuits assembled from such gates. Constructing quantum computers that would be competitive with conventional computers would require that workarounds be devised for a number of nonidealities that dye molecules exhibit.

Although DNA nanotechnology currently offers the most promising means by which to assemble dye aggregates into functioning quantum gates and circuits, the technology still has limitations that make the construction of these gates challenging. It is desirable that the $J_{m,n}$ and $K_{m,n}$ be as large as possible to enable the gate or circuit to complete its operation before coherence is lost. Ideally, the dyes would be stacked as closely as possible. That is, one would like the spacing between dyes to be comparable to the base stacking distance of DNA. The helical twist of DNA, for

which a full turn occurs over roughly 3.4 nm, however, makes it difficult to take advantage of the base stacking to pack dyes closely together. One would like to lay out the gates in a two-dimensional or three-dimensional arrangement. Here the 2 nm diameter of duplex DNA complicates the stacking of dyes close together in the direction orthogonal to the DNA helix direction. The size mismatches between the DNA structure and the desired spacing between dyes could be ameliorated if dyes were covalently linked into transmission-line lengths that could span the distance of a helical turn of DNA or the distance between neighboring duplex strands. Covalently linked aggregates forming gates would also help. DNA assembly would then be used to arrange these larger components into a desired circuit.

Finally, the quantum computer architecture proposed here is not necessarily optimum. Childs et al. [20] provide a different set of gates that could be implemented with dye aggregates in which the exciton–exciton interaction characterized by the anharmonicity parameter Δ_m rather than the exciton–exciton interaction characterized by $K_{m,n}$ provides the exciton–exciton interaction needed to implement controlled basis change gates. A search for alternative means with which to do information processing or quantum computing using dye aggregates could be productive. In this regard, it is noted that molecular vibrations exhibit coherence times that are longer than those of excitons. This suggests that it may be worth considering how molecular vibration quanta might be used to process and store quantum information.

Acknowledgements The author thanks the Boise State University qDNA Device Group for stimulating discussions and is especially appreciative of William B. Knowlton, Joseph Melinger and anonymous reviewers for comments helpful in the preparation of this manuscript. This research was supported wholly by the Department of Navy award No. N00014-19-1-2615 issued by the Office of Naval Research.

Appendix

Here expressions for $J_{m,n}$ and $K_{m,n}$ are presented. Both of these quantities represent Coulomb interaction energies between pairs of dyes. The first results from the Coulomb interaction between the transition charge densities of a pair of dyes, whereas the latter arises from differences in the Coulomb interaction between a pair of dyes resulting from differences in the ground-state and excited-state charge densities of the dyes [34]. Expressions for these energies greatly simplify when the distance between the dye molecules is much greater than the size of the molecule. In this case an approximation can be made in which the charge distribution on a dye is represented by a dipole moment. Even when the distance between the dyes is less than their lengths, the dipole approximation often provides a factor of two estimate for $J_{m,n}$ and $K_{m,n}$.

Consider first $J_{m,n}$. The dipole component of the transition charge density is referred to as the transition dipole. It is a vector quantity whose magnitude for dye m is here denoted by μ_m . The dipole vector generally is parallel to the long axis of

the dye molecule. In the dipole approximation $J_{m,n}$ is given by

$$J_{m,n} = \frac{\mu_m \mu_n}{4\pi \epsilon \epsilon_0 R_{m,n}^3} [\cos(\theta_1) - 3 \cos(\theta_2) \cos(\theta_3)], \quad (128)$$

where $R_{m,n}$ is the distance between the centers of the two dyes, ϵ_0 is the permittivity of free space and ϵ is the relative dielectric constant of the medium in which the dyes reside. θ_1 is the angle the two dyes make with respect to each other, θ_2 is the angle between dye m and the line between the centers of the two dyes. Similarly, θ_3 is the angle between dye n and the line between the centers of the two dyes. Thus four quantities $R_{m,n}$, θ_1 , θ_2 and θ_3 that can be adjusted to fine-tune $J_{m,n}$ to a desired value.

Consider now $K_{m,n}$. Let Δd_m denote the magnitude of the dipole component of the difference between the excited-state and ground-state charge densities of molecule m . The direction of this dipole also generally lies along the long axis of the dye molecule. But if the dye molecule has a bent shape or has a width comparable to its length then it need not lie along the long axis of the molecule. It could even be perpendicular to the transition dipole. In general it will have some fixed angle with respect to the transition dipole. In the dipole–dipole approximation one has

$$K_{m,n} = \frac{\Delta d_m \Delta d_n}{4\pi \epsilon \epsilon_0 R_{m,n}^3} [\cos(\phi_1) - 3 \cos(\phi_2) \cos(\phi_3)], \quad (129)$$

As in the case for $J_{m,n}$, four quantities can be varied by adjusting the position or orientation of the two dyes. These are $R_{m,n}$, ϕ_1 , ϕ_2 and ϕ_3 . Because $R_{m,n}$ is present in both Eqs. (128) and (129) and the transition dipole and the difference static dipole make a fixed angle with respect to each other, the degree to which $J_{m,n}$ and $K_{m,n}$ can be adjusted independently is constrained. When the transition dipoles and difference static dipoles are not parallel, however, $J_{m,n}$ and $K_{m,n}$ can still be adjusted independently of each other over a range of values.

General expressions for $J_{m,n}$ and $K_{m,n}$ are given in [34]; however, their evaluation requires that one obtain the transition and ground-state and excited state-static charge densities for the molecules by an ab initio calculation, such as through density functional theory and time-dependent density functional theory.

References

1. B.H. Robinson, N.C. Seeman, The design of a biochip: a self-assembling molecular-scale memory device. *Protein Eng.* **1**, 295–300 (1987)
2. X. Yan, S. Huang, Y. Wang, Y. Tang, Y. Tian, Bottom-up self-assembly based on DNA nanotechnology. *Nanomaterials* **10**, 2047 (2020)
3. Y. Chen, W. Sun, C. Yang, Z. Zhu, Scaling up DNA self-assembly. *ACS Appl. Bio Mater.* **3**, 2805–2815 (2020)

- M. Massey, I.L. Medintz, M.G. Ancona, W.R. Algar, Time-gated FRET and DNA-based photonic molecular logic gates AND, OR, NAND, and NOR, *ACS Sens.* **2**, 1205–1214 (2017)
- X. Lan, Q. Wang, DNA-programmed self-assembly of photonic nanoarchitectures. *NPG Asia Materials* **6**, e97 (2014)
- Buckhout-White S. Spillmann C.M., Algar W.R., Khachatryan A., Melinger J.S. Goldman E.R., Ancona M.G., Medintz I.: Assembling programmable FRET-based photonic networks using designer DNA scaffolds, *Nature Com.* **5**, 5615 (2014)
- Z.-G. Wang, C. Song, B. Ding, Functional DNA nanostructures for photonic and biomedical applications. *Small* **9**, 2210–2222 (2013)
- B. Albinsson, J.K. Hannestad, K. Börjesson, Functionalized DNA nanostructures for light harvesting and charge separation. *Coordination Chem. Rev.* **256**, 2399–2413 (2012)
- S. Su, V. Bonnard, G.A. Burley, DNA-templated photonic arrays and assemblies: Design principles and future opportunities. *Chem. Eur. J.* **17**, 7982–7991 (2011)
- J.S. Huff, D.B. Turner, O.A. Mass, L.K. Patten, C.K. Wilson, S.K. Roy, M.S. Barclay, B. Yurke, W.B. Knowlton, P.H. Davis, R.D. Pensack, Excited-state lifetimes of DNA-templated cyanine dimer, trimer, and tetramer aggregates: The role of exciton delocalization, dye separation, and DNA heterogeneity. *J. Phys. Chem. B* **125**, 10240–10259 (2021)
- O.A. Mass, C.K. Wilson, S.K. Roy, M.S. Barclay, L.K. Patten, E.A. Terpetschnig, J. Lee, R.D. Pensack, B. Yurke, W.B. Knowlton, Exciton delocalization in indolenine squaraine aggregates templated by DNA Holliday junction scaffolds. *J. Phys. Chem. B* **124**, 9636–9647 (2020)
- Rukin P.S., Komarova K.G., Fresch B., Collini E., Remacle F.: Chirality of a rhodamine heterodimer linked to a DNA scaffold: an experimental and computational study, *Phys. Chem. Chem. Phys.* **22**, 7516–7523 (2020)
- S. Mandal, X. Zhou, S. Lin, H. Yan, N. Woodbury, Directed energy transfer through DNA-templated J-aggregates. *Bioconjugate Chem.* **30**, 1870–1879 (2019)
- B.L. Cannon, L.K. Patten, D.L. Kellis, P.H. Davis, J. Lee, E. Graugnard, B. Yurke, W.B. Knowlton, Large Davydov splitting and strong fluorescence suppression: An investigation of exciton delocalization in DNA-templated Holliday junction dye aggregates. *J. Phys. Chem. A* **122**, 2086–2095 (2018)
- B.L. Cannon, D.L. Kellis, L.K. Patten, P.H. Davis, J. Lee, E. Graugnard, B. Yurke, W.B. Knowlton, Coherent exciton delocalization in a two-state DNA-templated dye aggregate system. *J. Phys. Chem. A* **121**, 6905–6916 (2017)
- H. Asanuma, T. Fujii, T. Kato, H. Kashida, Coherent interactions of dyes assembled on DNA. *J. Photochem. Photobiol. C: Photochem. Rev.* **13**, 124–135 (2012)
- M. Wang, G.L. Silva, B.A. Armitage, DNA-templated formation of a helical cyanine dye J-aggregate. *J. Am. Chem. Soc.* **122**, 9977–9986 (2000)
- M.A. Castellanos, A. Dodin, A.P. Willard, On the design of molecular excitonic circuits for quantum computing: the universal quantum gates. *Phys. Chem. Chem. Phys.* **22**, 3048–3057 (2020)
- M.A. Castellanos, A.P. Willard, Designing excitonic circuits for the Deutsch-Jozsa algorithm: mitigating fidelity loss by merging gate operations. *Phys. Chem. Chem. Phys.* **23**, 15196–15208 (2021)
- A.M. Childs, D. Gosset, Z. Webb, Universal computation by multiparticle quantum walk. *Science* **339**, 791–794 (2013)
- M. Kjaergaard, M.E. Schwartz, J. Braumüller, P. Krantz, J.I.-J. Wang, S. Gustavsson, W.D. Oliver, Superconducting qubits: Current state of play. *Annu. Rev. Condens. Matter Phys.* **11**, 369–395 (2020)
- L. Gyongyosi, S. Imre, A survey on quantum computing technology. *Comp. Sci. Rev.* **31**, 51–71 (2019)
- J.N. Eckstein, J. Levy, Materials issues for quantum computation. *MRS Bulletin* **38**, 783–789 (2013)
- F. Würthner, T.E. Kaiser, C.R. Saha-Möller, J-aggregates: From serendipitous discovery to supramolecular engineering of functional dye materials. *Angew. Chem. Int. Ed.* **50**, 3376–3410 (2011)

25. Y. Frenkel, On the transformation of light into heat in solids. *Phys. Rev.* **37**, 17–44 (1931)
26. E.E. Jelley, Spectral absorption and fluorescence of dyes in the molecular state. *Nature* **138**, 1009–1010 (1936)
27. G. Scheibe, L. Kandler, H. Ecker, Polymerisation and polymere adsorption as a cause of novel absorption bands of organic pigments. *Naturwissenschaften* **25**, 75–75 (1937)
28. G. Scheibe, A. Mareis, H. Ecker, The reversible polymerisation as a cause of unusual absorption bands III. *Naturwissenschaften* **25**, 474–475 (1937)
29. G. Scheibe, Über die Veränderlichkeit der Absorptionsspektren in Lösungen und die Nebenvalenzen als ihre Ursache. *Angew. Chem.* **50**, 212–219 (1937)
30. M. Reck, A. Zeilinger, H.J. Bernstein, P. Bertani, Experimental realization of any discrete unitary operator. *Phy. Rev. Lett.* **73**, 58–61 (1994)
31. P.W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.* **26**, 1484–1509 (1997)
32. Grover L.K.: A fast quantum mechanical algorithm for database search, Proceedings, 28th Annual ACM Symposium of the Theory of Computing (May 1996), 212–219 (1996)
33. C.P. Williams, *Explorations in Quantum Computing* (Springer, London, 2011), pp.51–122
34. D. Abramavicius, B. Palmieri, S. Mukamel, Extracting single and two-exciton couplings in photosynthetic complexes by coherent two-dimensional electronic spectra. *Chem. Phys.* **357**, 79–84 (2009)
35. T. Renger, V. May, V. Sudström, O. Kühn, Anharmonic oscillator approach to the exciton-exciton annihilation dynamics in molecular aggregates. *J. Chinese Che. Soc.* **47**, 807–819 (2000)
36. Jacquemin D.: Excited-state dipole and quadrupole moments: TD-DFT vers CC2, **12** 3993-4003 (2016)
37. G. Barcenas, A. Biagne, O.A. Mass, C.K. Wilson, O.M. Obukhova, O.S. Kolosova, A.L. Tatarets, E. Terpetschnig, R.D. Pensack, J. Lee, W.B. Knowlton, B. Yurke, L.I.L: First-principles studies of substituent effects on squaraine dyes. *RSC Adv.* **11**, 19029–19040 (2021)
38. A. Biagne, W.B. Knowlton, B. Yurke, J. Lee, L. Li, Substituent effects on the solubility and electronic properties of the cyanine dye Cy5: Density Functional and Time-Dependent Density Functional Theory calculations. *Molecules* **26**, 524 (2021)
39. M. Kasha, H.R. Rawls, A. El-Bayoumi, The exciton model in molecular spectroscopy. *Pure Appl. Chem.* **11**, 371–392 (1965)
40. C. Zhong, D. Bialas, C.J. Collison, F.C. Spano, Davydov splitting in squaraine dimers. *J. Phys. Chem. C* **123**, 18734–18745 (2019)
41. P.D. Cunningham, S.A. Dí, B. Yurke, I.L. Medintz, J.S. Melinger, Delocalized two-exciton states in DNA scaffolded cyanine dimers. *J. Phys. Chem. B* **124**, 8042–8049 (2020)
42. M.S. Barclay, S.K. Roy, J.S. Huff, O.A. Mass, D.B. Turner, C.K. Wilson, D.L. Kellis, E.A. Terpetschnig, J. Lee, P.H. Davis, B. Yurke, W.B. Knowlton, R.D. Pensack, Rotaxane rings promote oblique packing and extended lifetimes in DNA-templated molecular dye aggregates. *Commun. Chem.* **4**, 19 (2021)
43. B. Yurke, W. Kuang, Passive linear nanoscale optical and molecular electronics device synthesis from nanoparticles. *Phy. Rev. A* **81**, 033814 (2010)
44. K.W. Eccleston, S.H.M. Ong, Compact planar microstripline branch-line and rat-race couplers. *IEEE Trans. Microwave Theory Tech.* **51**, 2119–2125 (2003)
45. T.S. Humble, H. Thapliyal, E. Muñoz-Coreas, F.A. Mohiyaddin, R.S. Bennink, Quantum computing circuits and devices. *IEEE Design & Test* **36**, 69–94 (2019)
46. F.C. Simmel, B. Yurke, H.R. Singh, Principles and applications of nucleic acid strand displacement reactions. *Chem. Rev.* **119**, 6326–6369 (2019)
47. A.M. Stephens, Fault-tolerant thresholds for quantum error correction with the surface code. *Phys. Rev. A* **89**, 022321 (2014)
48. A.G. Fowler, Surface codes: Towards practical large-scale quantum computation. *Phys. Rev. A* **86**, 032324 (2012)
49. Barford W., Marcus M.: Theory of optical transitions in conjugated polymers. I. Ideal systems, *J. Chem. Phys.* **141**, 164101 (2014)

50. W. Barford, O.R. Tozer, Theory of exciton transfer and diffusion in conjugated polymers. *J. Chem. Phys.* **141**, 164103 (2014)
51. H. Muroph, A. Towns, Fine structure in electronic spectra of cyanine dyes: Are sub-bands largely determined by a dominant vibration or a collection of singly excited vibrations? *Chem. Phys. Chem.* **19**, 1016–1023 (2018)
52. F.C. Spano, The spectral signatures of Frenkel polarons in H- and J- aggregates. *Accounts of Chem. Res.* **43**, 429–439 (2010)
53. J. Lim, D. Paleček, F. Caycedo-Soler, C.N. Lincoln, J. Prior, H. von Berlepsch, S.F. Huelga, M.B. Plenio, D. Zigmantas, J. Hauer, Vibronic origin of long-lived coherence in an artificial molecular light harvester. *Nat. Commun.* **6**, 7755 (2015)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Structures

Building with DNA: From Curiosity-Driven Research to Practice



Fei Zhang

Abstract The origins of DNA nanotechnology can be traced back to 1982, when Dr. Ned Seeman proposed assembling branched junctions as 3D lattices to facilitate protein crystallization. Over the past four decades, this concept has evolved into a multidisciplinary research field with vast potential for applications. In this mini review, we present a brief introduction of selected topics in nucleic acid nanotechnology, focusing on scaling up DNA assembly, achieving higher resolutions, and transferring to RNA structural design. We discuss the advantages and challenges of each topic, aiming to shed light on the enormous potential of nucleic acid nanotechnology.

1 Introduction

In natural systems, biopolymers cooperatively assemble and interact at the nanoscale to form microscale structures. For example, the diploid human genome contains approximately 3 billion DNA base pairs. Based on the canonical B-form DNA duplex (0.34 nm per base pair), the total length of DNA in each cell is about 2 m. How does such long DNA fit into the nucleus which is about 10 μm in diameter? The answer is DNA packaging. The assembly of chromosomal DNA is a highly regulated and hierarchical condensation involving many proteins. In eukaryotic cells, DNA packaging is an important process of wrapping DNA around histone proteins resulting in a hierarchical well-defined structure of compact DNA–protein complexes. This assembly procedure occurs at a broad of length scales from nanometer to micron scales, displaying organizational precision down to the angstrom level (Fig. 1).

Towards the goal of engineering bioinspired systems that rival natural systems, information-coding biopolymers such as nucleic acids [1–3], proteins [4–6], and lipids [7] have been used as building blocks in the assembly of designer nanoarchitectures and nanodevices. Among them, DNA self-assembly has been broadly exploited [8–13], and diverse design techniques [3, 14–17] and computational

F. Zhang (✉)

Department of Chemistry, Rutgers University, Newark, NJ, USA

e-mail: fei.zhang@rutgers.edu

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*, Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_10

173

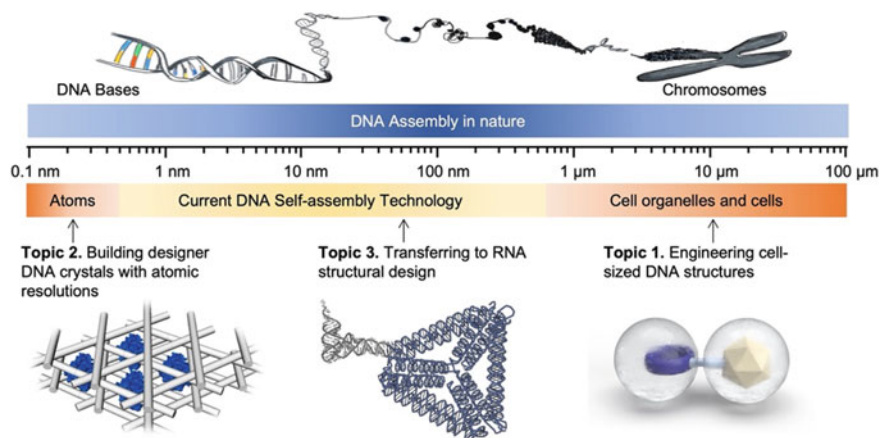


Fig. 1 DNA assembly happens at multiple scales in nature. Current DNA nanotechnology has several challenges and opportunities to be expanded in ‘depth and breadth.’ Three selected topics will be discussed here: (1) Engineering cell-sized DNA structures; (2) Building self-assembled DNA crystals with atomic resolutions; (3) Transferring to RNA structural design

tools [18–24] have been used, resulting in increased knowledge of building things with DNA as well as a wide variety of nanostructures. The success of DNA self-assembly in constructing various architectures is attributed to several reasons. Firstly, Watson–Crick base pairing between complementary DNA strands is simple and highly predictable and thus makes the four-letter polymeric strands convenient units with designed pairing rules between each other. Secondly, the geometric features of DNA double helices are well understood, with a diameter of about 2 nm and 3.4 nm per helical repeat for canonical B-form DNA [25]. Additionally, the development of several user-friendly software interfaces has facilitated the designing and viewing of even the most intricate DNA nanostructures before experimental testing [18–24]. Thirdly, modern organic chemistry and molecular biology provide a diverse toolbox to readily synthesize, modify, and replicate DNA molecules at a relatively low cost. Finally, the biocompatibility of DNA makes it suitable for constructing multicomponent nanostructures made from hetero-biomaterials with designed functions.

Inspired by nature, scientists keep turning to DNA nanotechnology to create structural designs that can also operate at various length scales. Here, I will share three examples that show how nanotechnology is moving closer to the goal of achieving precise structural control at the angstrom level within cells, as well as discuss some of the challenges that are involved. The first example focuses on the challenge of engineering programmable structures at the scale of entire cells. This is a significant obstacle, as it requires creating something microns in size from many individual nanoscale components through programmable interactions. The second example highlights the challenge of engineering the designer DNA crystals, which is the original vision for DNA nanotechnology. It has taken decades to achieve, but

the process of its development has greatly expanded our understanding of molecular construction. Lastly, toward producing precision nanostructures inside of living cells, RNA is highly advantageous because it can be folded cotranscriptionally. Fortunately, the great progress in developing DNA nanostructures has accelerated the development of RNA nanostructures. Nucleic acid nanotechnology, originally driven by curiosity, has greatly informed our understanding of structural biology and now has produced many exciting future applications.

2 Engineering Cell-Sized DNA Structures

It is a challenging yet rewarding goal in DNA nanotechnology to make cell-sized structures because such materials have several potential applications that cannot be realized by smaller structures. For example, DNA origami [3] is a technique to fold a long single-stranded ‘scaffold’ DNA to a target object using hundreds of short ‘staple’ DNA strands. This technique was first introduced in 2006 and has been employed in many works. The DNA origami nanostructures generally show 10–100 nm in diameter and are great scaffolds to host a few enzymes or quantum dots due to their compatible sizes. Similarly, micron-sized constructs with defined shapes and finite sizes will provide a fully addressable canvas to organize larger guests with nanoscale precision. Those constructs will be able to organize larger targets, such as cells, and offer long-range interactions between guest molecules that cannot be achieved by previous small structures. The micron-sized 2D arrays can serve as high-throughput biological nanopore arrays for protein sequencing and biosensing. They can also be used as templates for creating long-range enzyme cascades/signaling, modulating cell-free membrane, and eventually facilitating artificial cell engineering.

2.1 Challenges

Although various strategies have been developed to produce synthetic DNA architectures that exhibit significant geometric complexity, most current techniques are still limited to produce DNA structures smaller than 1 micron at few nanometer resolutions. To create larger DNA constructs with defined shapes and finite sizes, different strategies have been reported. Qian and coworkers advanced higher-order assemblies by using 2D square-shaped DNA origamis with surface patterns and short sticky ends hybridization to create a finite-sized 2D DNA canvas of sizes up to half micron [26]. 3D DNA origami higher-order structures are achieved by using an angle-controllable V-shaped DNA origami object to form several polyhedrons that are up to 450 nm in diameter [27]. The single-stranded tile (SST) DNA structures are also advanced to larger assemblies employing a brick design of 52-nt long with four 13-nt domains [28]. The 2D and 3D origami-based strategies seek weak interactions between building blocks and employ a step-by-step hierarchical assembly

process that typically involves the formation and purification of individual origami, assembly of sub-units from a set of origami blocks, and the addition of origami blocks for final constructions. However, there are challenges associated with these methods. The first challenge lies in designing DNA origami that can self-assemble into micron-sized structures. The individual origami needs to be accurately designed because any twist or distortion in these units will accumulate and become amplified during the higher-order assembly process. Another challenge is the low formation yield. Potential reasons for the low yield include the possible defects in individual origami building blocks and the slow kinetics of origami higher-order assembly. For the SST method, a new challenge is the importance of sequence besides the accurate geometric design since there is no scaffold sequence as a guidance in the structures. In addition, the low formation yield, as well as the high synthesis cost, also contributes to the difficulty of using this method to create even larger structures.

For the formation of 2D and 3D DNA superlattices, DNA origami nanostructures serve as repeating units to hierarchically connect into higher-order assemblies. Inspired by the first rationally designed DNA crystal structure [16], Liedl and coworkers created an origami version of the tensegrity triangle and demonstrated the assembled rhombohedral crystalline lattices [29]. Several examples showed that weak interaction will help the mismatched DNA origami units to dissociate from each other during assembly and thus promote the formation of correct target patterns [26, 30]. One challenge is the accurate estimation of the mechanical properties of DNA structures. For instance, the formation of large 2D arrays generally suffers from the inherent flexibility of individual building blocks. It is important to develop methods for confining the assembly process to one plane, rather than in 3D, to grow 2D crystalline assemblies. Surface-mediated growth is also a helpful strategy to encourage the 2D arrays development including using lipid [31, 32] and mica surfaces [33]. Recently, Gang and colleagues reported a novel technique of creating vertex-to-vertex hybridization between polyhedral DNA origami building blocks [34]. Instead of avoiding flexibility, they explored single-stranded loop linkers between origami, relied on the geometric restriction, and harnessed the flexibility of the loops. This vertex-to-vertex hybridization is remarkably robust and programmable being able to form a wide variety of polyhedral geometries. Although the DNA origami crystalline lattices do not have the similar atomic resolution as designer DNA crystals, these higher-ordered arrays provide much larger cavities for hosting various guest molecules such as large enzyme complexes.

2.2 Opportunities

Although a variety of design techniques have been developed, it is desirable to further enrich the types of DNA building blocks with novel structural properties that enable efficient scale-up of DNA assemblies. New design strategies combined with mathematical graphics will provide additional opportunities for DNA nanoconstruction

and enhance our understanding of DNA self-assembly as a programmable biomaterial. Advanced experimental validation techniques are also valuable for revealing the assembly dynamics at the individual unit level. In addition, it is highly desirable to develop simple yet robust procedures to improve the mechanical properties of DNA nanostructures, while preserving their nanometer precision and dynamic features such as strand displacement reactions [35, 36].

A key opportunity is to utilize hybrid assembly methods, which involve combining multiple types of building block materials and leveraging different interactions between building blocks. The co-assembly of DNA polyhedral origami and gold nanoparticle is a great example of this approach, showing the potentials for superlattice construction. In order to ensure appropriate binding forces between assembly building blocks for orthogonal regulation of assembly structures, a wide variety of binding forces should be investigated, ranging from weak interactions such as base stacking and sequence recognition to strong chemical bonds through click chemistry and enzyme ligation. The utilization of different binding forces will help to create a local energetic maximization of stability, while maintaining weak interactions between assembly units. The local optimization of the recognition between building blocks will promote the formation of target structures, and the weak interactions will minimize any possible mismatch. More importantly, integration of orthogonal interaction forces between assembly units could potentially facilitate sophisticated assembly behaviors such as dynamic or developmental self-assembly behaviors observed in living cells [37].

The possible successful formation of micron-sized programmable DNA structures with nanometer precision promises many new opportunities for both fundamental and applied research. Micron-sized DNA assemblies with fully addressable surfaces will enable various types of synthetic cell engineering by employing structural DNA assemblies as spatial frameworks. DNA nanostructures equipped with membrane-interacting molecules have been demonstrated as nanoscale mechanical tools to scaffold and sculpt lipid bilayer [38, 39]. Large DNA assemblies will contribute to the creation of artificial cells with comparable sizes as living cells and deliver a unique interface between cell biology and biomolecular engineering.

3 Building Designer DNA Crystals with Atomic Resolutions

Crystal lattices with atomic resolution are useful materials for positioning guest molecules at specific locations within three-dimensional space to achieve various functions. Obtaining self-assembled nucleic acid crystals with programmable space groups and cavity sizes is not only a long-standing challenge in DNA nanotechnology but also one of the important research frontiers. As the original idea proposed by Nadrian Seeman in 1982 [1], introducing target molecules into the DNA scaffolds will facilitate structure determination of guest molecules such as RNA, peptide, and protein. Rational design and synthesis of 3D DNA crystals provide both precisely designed symmetry and functions. Such crystals can serve as porous scaffolds to

arrange guest molecules at specific positions, and thus the guest molecules could be integral parts of the crystalline lattices. Those hybrid crystalline structures can be selected as novel candidates for the determination of protein structures, especially for membrane proteins. Self-assembling DNA crystals have also been treated as 3D templates for molecular electronics, such as information storage devices and zeolite-like nanoporous materials that are capable of catalysis [40] and molecular separations [41].

3.1 Challenges

However, after four decades of the initial proposal [1], only a few self-assembled DNA crystals have been reported displacing rationally designed 3D crystalline structures. In 2009, Seeman, Mao, and their coworkers created the first rationally designed DNA crystals that were based on a tensegrity triangle DNA motif, demonstrating a set of crystals in the rhombohedral space group R3 [16]. These are the first-ever designed self-assembled DNA crystals, which are published 29 years after Dr. Seeman's initial inspiration [42]. In 2016, Yan, Seeman, and coworkers reported a new designer DNA crystal based on a layered Holliday junction design with three distinct strands, solving the structure by X-ray crystallography to ~ 3 Å [43]. Later, a rationally designed and self-assembled 3D DNA crystal lattice with hexagonal symmetry was successfully created by using only two DNA strands. The six-fold symmetry, as well as the chirality of the crystal lattices, is directed by the Holliday junctions formed between the duplex motifs. Native crystals were measured and analyzed to ~ 3 Å resolution with the hexagonal space group P6 [44].

The main scientific challenges for creating designer DNA crystals lie in both fundamental design and experimental growth of such crystals. Three questions will be discussed including how to develop robust design methods for creating a wide variety of designer DNA crystals with prescribed lattices, how to introduce guest molecules into crystals at specific locations, and how to enable DNA crystals to have better chemical/physical/mechanical properties better suited to broader applications.

To create 3D DNA motifs that can be connected by rationally designed sticky end connections rather than through nonspecific stacking, various factors should be examined deliberately. The designer DNA crystals should be assembled from rigid tiles/motifs with appropriate numbers of sticky ends. The individual DNA tile needs to be designed with sufficient connections to define the structural frame, and the interactions between tiles should be encoded into sticky ends. The sequence design of DNA tiles should be investigated thoughtfully. The DNA motifs are formed through an annealing process, where a slow temperature ramp is needed. A systematic crystallization protocol needs to be established to incorporate the annealing process with the crystallization process as well as the experimental conditions to improve the yield of large crystals, including buffer conditions, reservoir conditions, and annealing temperatures. X-ray crystallography will be the major technique used to determine the crystalline structures of DNA crystals. As X-ray requires relatively

large sizes of crystals, other techniques, such as micro-electron diffraction (Micro-ED), can be employed to solve the structures of nanocrystals.

Self-assembled DNA crystals contain large cavities that make them excellent scaffolds for attaching various biomolecules (e.g., peptides, proteins, and RNAs) to achieve different functional applications. The major hurdles to adding guest materials into designer DNA crystals are two-fold. First, the binding of the DNA and guest molecules shouldn't change the space groups of the predefined 3D crystal lattices, and the guest molecules can be only arranged in the appropriate cavities of the designer crystals. Second, the interaction between the DNA lattices with the guest molecules should be robust and specific, so that the guest molecules can be treated as an internal part of the crystal lattices with atomic-level precision for further applications. For these two main challenges, one simple solution is to use sequence-specific DNA binding peptides or proteins as the guest molecule. The interactions between DNA and peptides/proteins are highly specific and strong, which exists naturally and doesn't need artificial linkers between DNA and proteins. Another advantage of using DNA binding peptides as the guest molecules is that these types of proteins generally have small sizes and can easily diffuse into the cavities of a crystal. Therefore, we can separate the experimental steps between the growth of the designer DNA crystals with the soaking of target DNA binding peptides/proteins, so that the crystal lattices can be preserved after introducing guest proteins. Another exciting prospect is constructing DNA/RNA and RNA/RNA self-assembled crystals by incorporating RNA strands into the DNA crystals and adapting a DNA motif design to create RNA motifs. These hybrid crystalline materials will provide new applications in many areas, such as assisting the structural determination of small RNA structures, building molecular devices from functional RNA motifs, and studying the RNA-protein interactions.

3.2 *Opportunities*

In addition to expanding the structural diversity of designer DNA crystals, one interesting topic is generating unconventional states of materials [45], such as quasicrystals and disordered hyperuniform DNA structures, in a controllable and programmable fashion. The mathematic description and simulation of using patch particles to create quasicrystal patterns were reported [46] before the experimental realizations [47]. The well-studied DNA multi-arm junction tiles make them ideal model systems to conduct programmable self-assembly in 2D. Rational designed 3D quasicrystals using DNA tiles remain a challenge. Computational simulations using 3D particles were reported [48], and designer DNA quasicrystals are likely to be achieved in the foreseeable future. Other states of matter that have been introduced in recent years can be new design targets for engineering bioinspired and biomimetic systems in DNA nanotechnology. Such studies will enrich the types of programmable

functional biomaterials, establish the structure–function relationship for unconventional states of materials, and gain knowledge of fundamental self-assembly in such systems.

The creation of DNA crystalline scaffolds with atomic resolution will provide precise spatial programmability of molecules, offering many application opportunities for various research areas. For example, it is useful to create a 3D network of enzyme cascades or energy transfer pathways based on chromophores with accurate location and orientation by employing DNA crystal templates. 3D DNA crystals also serve as zeolite-like/nanoporous materials that are capable of catalysis and molecular separations. In addition, some applications require intimate knowledge of the atomic-level details of DNA structures, and the designer DNA crystals can provide such information. Although researchers have obtained cryo-EM reconstructions for various DNA complexes and revealed the structures of DNA complexes with nucleotide resolution [49, 50], universal strategies are not established yet for creating different types of designer DNA crystals. For instance, there is no crystal structure of the double crossover (DX) tile that existed in most DNA nanostructures including DNA origami. Consequently, the deformation near crossovers introduced by adding the crossovers, and the base-level sequence impact of DNA structures are still unknown. Angstrom-level positioning accuracy at the scale of individual DNA bases will enable new opportunities such as chromophore placement on DNA-dye assemblies. Structural information gained from crystals can be used to improving DNA designs, leading to a greater understanding of DNA as a biomaterial.

The potential applications of DNA crystals extend beyond their use as scaffolds for crystallization. One example is using crystalline complexes as molecular ‘sponges’ to detect or separate target molecules by trapping or binding them in the cavities. To expand the scope of DNA lattices to other fields—especially ones where biological materials may not be stable—integrating designer DNA crystals with other materials will be highly desirable. For example, it is possible to introduce inorganic materials growing along DNA helices, so that the resulting products will inherit the programmable geometric feature of DNA crystals. Like the biomineralization process, inorganic materials, such as silica, can be deposited on the surface of DNA. Our recent study revealed that silica-DNA composited structural nanomaterials with a set of controllable morphologies can be created using chemical reactions [51]. This technique was further developed to cost silica shell on 3D origami lattices [52] and to create superconducting 3D materials with another layer of niobium coating on top of silica layer [53]. This technique has the potential to be adopted into designer DNA crystals as well. One challenge is how to allow the depositing materials to permeate the 3D crystals through their relatively small cavities. The resulting porous materials should exhibit significantly improved mechanical properties due to the inorganic coating. By transferring the structural programmability of designer DNA crystals to inorganic materials, the nucleic acid-based concreting strategy will open exciting opportunities for novel nanofabrication with various application potentials.

4 Transferring to RNA Structural Design

An important landmark in the development of nanotechnology was the use of biological molecules as building blocks to construct devices with control of structures and function at the nanometer scale. RNA, which shares some general features with DNA, has played a unique role. Unlike DNA, RNA has an inherent architectural potential to form a wide variety of interactions far beyond the Watson–Crick base pairing [54, 55]. As opposed to DNA, naturally existing 3D RNA molecules and man-made RNA building blocks/tiles [56, 57] at an atomic resolution can be modified and provide potentially a much larger toolkit to readily build a variety of structures with high complexity. In addition, functionalities associated with RNA molecules, such as catalysis [58], gene regulation [59], and organization of proteins in large machineries [60], enable their use in material and biomedical sciences [61]. Most importantly, RNA molecules can be readily synthesized in cells through transcription [62]. Therefore, RNA nanostructures have a great potential to build self-assembling nanodevices inside cells by utilizing the cellular nucleic acid synthesis pathway.

4.1 Challenges

Although RNA shares many general geometric and chemical features with DNA, the construction of custom RNA nanostructures has been hindered. For instance, it is still challenging to rationally designed RNA objects with comparable size and complexity to natural RNA machineries or current highly sophisticated DNA nanostructures. The emerging field of RNA nanotechnology has attracted increasing attention from diverse research areas in recent years, and many RNA nanostructures have been constructed, including squares [63, 64], tubes [65], arrays [66, 67], and 3D objects [68, 69]. In most studies, the use of conserved, naturally evolved motifs with predictable tertiary structures dominates the current RNA self-assembly methods. Natural RNA building blocks, called structural modules, can be combined and rearranged in a large number of ways into target shapes. But currently by using this method, the sizes of constructed RNA nanostructures are generally smaller than 200 nucleotides, and the complexity of RNA designs has been limited as well [70]. As a complementary approach, a *de novo* design strategy offers higher versatility, which allows us to build structures/functions that do not exist in nature but fulfill our needs. *De novo* designed RNA nanostructures have emerged recently. One example is a designed RNA nano-prism that self-assembled from eight T-shaped RNA motifs [71]. Those motifs have well-defined configurations and were created following the example of DNA T-junction tiles [72]. As compared with previous works that used naturally existing 3-way RNA junctions to construct polyhedrons [71], this *de novo* design method provides a unique way to control synthetic RNA structures, such as the angle, the length, and the sequence of target structures. A recent example of the T-shaped RNA tile is a single-stranded version named branched kissing loops [73].

16 different linear and circular assemblies have been constructed from the tile by adjusting the tile geometry.

Self-folding of a single RNA chain into a defined complex nanostructure represents a new era in nucleic acid nanotechnology. Two-dimensional and unknotted folding of single-stranded RNA with 6300 bases long were demonstrated using thermal anneals [74]. Single-stranded RNA knots with 1000 nucleotides were reported by hierarchical folding in prescribed orders, exhibiting an unprecedented amount of complicated topological features [75]. One most exciting feature of RNA molecules is their cotranscriptional folding ability, which offers attractive potential applications for synthetic biology. The single chain of RNA folds itself during the transcription process in isothermal conditions in vitro and in cells. The rationally designed RNA assemblies with cotranscriptional folding provide a new avenue that could create self-assembled RNA scaffolds interfacing with synthetic biology and nanomedicine. Geary, Rothmund, and Andersen pioneered the RNA origami approach [67], enabling cotranscriptional assembly by arranging RNA helices parallelly through crossovers and kissing loops. Recently, the same team introduced RNA origami automated design software and extended the creation of large RNA tiles up to 2360 nucleotides [76], which represents the largest synthetic RNA structure that can be folded cotranscriptionally to date.

4.2 Opportunities

Since RNA structures began to be characterized in higher resolution, the more we learn about RNA molecules, the more we realize how much we don't know. There are a tremendous amount of knowledge and undiscovered rules for RNA structural and functional design. DNA nanostructures generally need annealing, a temperature cooling process, to promote the formation of designed Watson-Crick base pairs, leading to the minimal free energy (MFE) configurations. While natural RNA molecules don't always fold into the MFE conformation [77, 78], the single-stranded RNA origami strategy [76] employs localized domain modules, optimizes MFE domains, and uses a multi-stage sequence optimization procedure to facilitate the isothermal folding. Considering the complexity of natural RNA structures and RNA folding, it is promising yet challenging to discover new design strategies. Inspirations could come from the unique features of RNA. For instance, RNA structural design allows us to harness the kinetic energy in RNA assembly. Near MFE configurations could be possibly achieved and stabilized through inserting local kinetic traps, adding protein binding regions, and topological constraints. Incorporating computational components into the RNA assembly process may also enable the creation of programmable dynamics. Particularly, in single-stranded RNA folding, we can imagine that inserting intramolecular strand displacement reactions in RNA sequence design could contribute to sequential and spatiotemporal control of dynamic assembly, such as on-the-fly single-chain folding. Thanks to the development of data science and machine learning, data-driven and data-based strategies have

started to show the power in RNA folding prediction as well as sequence generation for target structures. These computational tools offer an efficient means of exploring a sufficient number of RNA sequence-structure pairs to provide insights into what may work, thereby reducing the need for extensive wet-lab experimentation.

5 At the End

The Olympic motto, proposed in 1894, is the hendiatriis *Citius, Altius, Fortius*, which means faster, higher, and stronger. The design, building, and assembly of DNA at nanoscales share a similar ethos with the Olympic games, striving for advancements in areas such as size, accuracy, versatility, adaptability and cost-effectiveness. Through DNA research, a single question can lead to many answers, and DNA assembly opens up endless possibilities for exploration. Nucleic acid nanostructures play the roles from nanomaterials to molecular devices and tools. As it continues to progress, it is only a matter of time before we witness the widely adopted real-world applications of nucleic acid nanotechnology. The question now is which one will be the first.

Examples of exploiting DNA nanostructures as drug delivery vehicles have been demonstrated for targeted delivery with precise control of the structures, components, and reconfigurations [79–83]. Moreover, nucleic acid nanostructures have been investigated for other biomedical applications beyond drug delivery. For example, the cotranscriptionally folded single-stranded RNA origami showed significant anticoagulant activity which was sevenfold greater than free aptamer [84]. Researchers also developed a set of aptamer-decorated RNA origami with the ability to reverse the anticoagulation activity [85]. Single-stranded RNA origami has been used as an immunostimulatory reagent to stimulate innate response for cancer immunotherapy [86]. In a recent case, a half polyhedron shell formed from higher-ordered DNA origami assemblies works as a virus trap, which was decorated 90 sites of virus-binding moieties in its interior surface [87]. More innovative usages of nucleic acid nanostructures should be explored to fully benefit from their structural, dynamic, and functional features.

Individual 2D DNA origami has been placed onto lithographical patterns precisely for nanophotonic applications [88]. Arrays of DNA origami patterns with sizes larger than few micrometers can be integrated with top-down lithographic patterning techniques to access even larger length scales. One can imagine that the future 3D integration between DNA assemblies, such as 3D DNA superlattices, and top-down lithography could be implemented to address the needs of extending to the third dimension in nanofabrication.

The key to providing a vast library of described functionalities lies in creating hybrid systems that incorporate nucleic acid, proteins, lipid, quantum dots, and other functional materials. Adaptability issues are crucial for the successful creation of hybrid materials. With the consideration of efficacy, scalability, robustness, and safety, nucleic acid-directed assemblies and devices provide a multitude of possible

applications that may produce novel materials with unique functionalities beyond our wildest imagination today.

Acknowledgements I thank Cody Geary for critical reading, providing constructive writing suggestions, and kindly offering writing examples. I thank Tim Liedl for reviewing and commenting on the manuscript. This work is supported by a US National Science Foundation (NSF) Faculty Early Career Development Award (DMR-2046835), an NSF grant (CCF-2007821), a Busch Biomedical Grant, and a faculty Startup Fund from Rutgers University.

References

1. N.C. Seeman, Nucleic-acid junctions and lattices. *J. Theor. Biol.* **99**(2), 237–247 (1982). [https://doi.org/10.1016/0022-5193\(82\)90002-9](https://doi.org/10.1016/0022-5193(82)90002-9)
2. P.X. Guo, C.L. Zhang, C.P. Chen, K. Garver, M. Trottier, Inter-RNA interaction of phage phi 29 pRNA to form a hexameric complex for viral DNA transportation. *Mol. Cell.* **2**(1), 149–155 (1998). [https://doi.org/10.1016/S1097-2765\(00\)80124-0](https://doi.org/10.1016/S1097-2765(00)80124-0)
3. P.W.K. Rothmund, Folding DNA to create nanoscale shapes and patterns. *Nature* **440**(7082), 297–302 (2006). <https://doi.org/10.1038/Nature04586>
4. J.S. Richardson, D.C. Richardson, The Denovo design of protein structures. *Trends Biochem. Sci.* **14**(7), 304–309 (1989). [https://doi.org/10.1016/0968-0004\(89\)90070-4](https://doi.org/10.1016/0968-0004(89)90070-4)
5. N.P. King et al., Computational design of self-assembling protein nanomaterials with atomic level accuracy. *Science* **336**(6085), 1171–1174 (2012). <https://doi.org/10.1126/science.1219364>
6. Y.T. Lai, D. Cascio, T.O. Yeates, Structure of a 16-nm cage designed by using protein oligomers. *Science* **336**(6085), 1129–1129 (2012). <https://doi.org/10.1126/science.1219351>
7. Y. Misquitta et al., Rational design of lipid for membrane protein crystallization. *J. Struct. Biol.* **148**(2), 169–175 (2004). <https://doi.org/10.1016/j.jsb.2004.06.008>
8. F.A. Aldaye, A.L. Palmer, H.F. Sleiman, Assembling materials with DNA as the guide. *Science* **321**(5897), 1795–1799 (2008). <https://doi.org/10.1126/science.1154533>
9. N.C. Seeman, Nanomaterials based on DNA. *Annu. Rev. Biochem.* **79**(1), 65–87 (2010). <https://doi.org/10.1146/annurev-biochem-060308-102244>
10. R.F. Service, DNA nanotechnology grows up. *Science* **332**(6034), 1140–1142 (2011). <https://doi.org/10.1126/science.332.6034.1140>
11. [11]A.V. Pinheiro, D.R. Han, W.M. Shih, H. Yan, Challenges and opportunities for structural DNA nanotechnology. *Nat. Nanotech.* **6**(12), 763–772 (2011). <https://doi.org/10.1038/Nnano.2011.187>
12. R.M. Zedegan, M.L. Norton, Structural DNA nanotechnology: from design to applications. *Int. J. Mol. Sci.* **13**(6), 7149–7162 (2012). <https://doi.org/10.3390/Ijms13067149>
13. F. Zhang, J. Nangreave, Y. Liu, H. Yan, Structural DNA nanotechnology: state of the art and future perspective. *J. Am. Chem. Soc.* **136**(32), 11198–11211 (2014). <https://doi.org/10.1021/Ja505101a>
14. E. Winfree, F.R. Liu, L.A. Wenzler, N.C. Seeman, Design and self-assembly of two-dimensional DNA crystals. *Nature* **394**(6693), 539–544 (1998). <https://doi.org/10.1038/28998>
15. H. Yan, S.H. Park, G. Finkelstein, J.H. Reif, T.H. LaBean, DNA-templated self-assembly of protein arrays and highly conductive nanowires. *Science* **301**(5641), 1882–1884 (2003). <https://doi.org/10.1126/science.1089389>
16. J.P. Zheng et al., From molecular to macroscopic via the rational design of a self-assembled 3D DNA crystal. *Nature* **461**(7260), 74–77 (2009). <https://doi.org/10.1038/Nature08274>
17. B. Wei, M.J. Dai, P. Yin, Complex shapes self-assembled from single-stranded DNA tiles. *Nature* **485**(7400), 623, (2012). <https://doi.org/10.1038/Nature11075>

18. N.C. Seeman, De Novo design of sequences for nucleic acid structural engineering. *J. Biomol. Struct. Dyn.* **8**(3), 573–581 (1990). <https://doi.org/10.1080/07391102.1990.10507829>
19. E.S. Andersen et al., DNA origami design of dolphin-shaped structures with flexible tails. *ACS Nano.* **2**(6), 1213–1218 (2008). <https://doi.org/10.1021/Nn800215j>
20. S.M. Douglas, A.H. Marblestone, S. Teerapittayanon, A. Vazquez, G.M. Church, W.M. Shih, Rapid prototyping of 3D DNA-origami shapes with caDNA. *Nucleic. Acids Res.* **37**(15), 5001–5006 (2009). <https://doi.org/10.1093/Nar/Gkp436>
21. S. Williams, K. Lund, C. Lin, P. Wonka, S. Lindsay, H. Yan, Tiamat: a three-dimensional editing tool for complex DNA structures. **5347**, 90–101 (2009). https://doi.org/10.1007/978-3-642-03076-5_8
22. J.N. Zadeh et al., NUPACK: analysis and design of nucleic acid systems. *J. Comput. Chem.* **32**(1), 170–173 (2011). <https://doi.org/10.1002/Jcc.21596>
23. D.N. Kim, F. Kilchherr, H. Dietz, M. Bathe, Quantitative prediction of 3D solution shape and flexibility of nucleic acid nanostructures. *Nucleic Acids Res.* **40**(7), 2862–2868 (2012). <https://doi.org/10.1093/Nar/Gkr1173>
24. E. Benson et al., DNA rendering of polyhedral meshes at the nanoscale. *Nature* **523**(7561), 441–U139 (2015). <https://doi.org/10.1038/nature14586>
25. J.D. Watson, F.H. Crick, Molecular structure of nucleic acids; a structure for deoxyribose nucleic acid. *Nature* **171**(4356), 737–738 (1953). <https://doi.org/10.1038/171737a0>
26. G. Tikhomirov, P. Petersen, L. Qian, Fractal assembly of micrometre-scale DNA origami arrays with arbitrary patterns. *Nature* **552**(7683), 67–71 (2017). <https://doi.org/10.1038/nature24655>
27. K.F. Wagenbauer, C. Sigl, H. Dietz, Gigadalton-scale shape-programmable DNA assemblies. *Nature* **552**(7683), 78–83 (2017). <https://doi.org/10.1038/nature24651>
28. L.L. Ong et al., Programmable self-assembly of three-dimensional nanostructures from 10,000 unique components. *Nature* **552**(7683), 72–77 (2017). <https://doi.org/10.1038/nature24648>
29. T. Zhang, et al., 3D DNA origami crystals. *Adv. Mat.* **30**(28) (2018). <https://doi.org/10.1002/adma.201800273>
30. P. Wang, S. Gaitanaros, S. Lee, M. Bathe, W.M. Shih, Y. Ke, Programming self-assembly of DNA origami honeycomb two-dimensional lattices and plasmonic metamaterials. *J Am Chem Soc* **138**(24), 7733–7740 (2016). <https://doi.org/10.1021/jacs.6b03966>
31. Y. Suzuki, M. Endo, H. Sugiyama, Lipid-bilayer-assisted two-dimensional self-assembly of DNA origami nanostructures. *Nat. Commun.* **6**, 8052 (2015). <https://doi.org/10.1038/ncomms9052>
32. S. Kocabay et al., Membrane-assisted growth of DNA origami nanostructure arrays. *ACS Nano.* **9**(4), 3530–3539 (2015). <https://doi.org/10.1021/acs.nano.5b00161>
33. A. Aghebat Rafat, T. Pirzer, M.B. Scheible, A. Kostina, F.C. Simmel, Surface-assisted large-scale ordering of DNA origami tiles. *Angew Chem. Int. Ed. Engl.* **53**(29), 7665–7668 (2014). <https://doi.org/10.1002/anie.201403965>
34. Y. Tian et al., Ordered three-dimensional nanomaterials using DNA-prescribed and valence-controlled material voxels. *Nat. Mater.* **19**(7), 789–796 (2020). <https://doi.org/10.1038/s41563-019-0550-x>
35. D.Y. Zhang, E. Winfree, Control of DNA strand displacement kinetics using Toehold exchange. *J. Am. Chem. Soc.* **131**(47), 17303–17314 (2009). <https://doi.org/10.1021/ja906987s>
36. B. Yurke, A.J. Turberfield, A.P. Mills, F.C. Simmel, J.L. Neumann, A DNA-fuelled molecular machine made of DNA. *Nature* **406**(6796), 605–608 (2000). <https://doi.org/10.1038/35020524>
37. K.G. Campellone, M.D. Welch, A nucleator arms race: cellular control of actin assembly. *Nat. Rev. Mol. Cell. Biol.* **11**(4), 237–251 (2010). <https://doi.org/10.1038/nrm2867>
38. Y. Yang et al., Self-assembly of size-controlled liposomes on DNA nanotemplates. *Nat Chem* **8**(5), 476–483 (2016). <https://doi.org/10.1038/nchem.2472>
39. X. Bian, Z. Zhang, Q. Xiong, P. De Camilli, C. Lin, A programmable DNA-origami platform for studying lipid transfer between bilayers. *Nat. Chem. Biol.* **15**(8), 830–837 (2019). <https://doi.org/10.1038/s41589-019-0325-3>
40. C. Geng, P.J. Paukstelis, DNA crystals as vehicles for biocatalysis. *J Am Chem Soc* **136**(22), 7817–7820 (2014). <https://doi.org/10.1021/ja502356m>

41. P.J. Paukstelis, Three-dimensional DNA crystals as molecular sieves. *J Am Chem Soc* **128**(21), 6794–6795 (2006). <https://doi.org/10.1021/ja061322r>
42. N.C. Seeman, DNA Nanotechnology at 40. *Nano Lett.* **20**(3), 1477–1478 (2020). <https://doi.org/10.1021/acs.nanolett.0c00325>
43. C.R. Simmons et al., Construction and structure determination of a three-dimensional DNA crystal. *J. Am. Chem. Soc.* **138**(31), 10047–10054 (2016). <https://doi.org/10.1021/jacs.6b06508>
44. F. Zhang, C.R. Simmons, J. Gates, Y. Liu, H. Yan, Self-assembly of a 3D DNA crystal structure with rationally designed six-fold symmetry. *Angew. Chem.* **130**(38), 12684–12687 (2018). <https://doi.org/10.1002/ange.201807223>
45. S. Torquato, F.H. Stillinger, Local density fluctuations, hyperuniformity, and order metrics. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.* **68**(4 Pt 1), 041113 (2003). <https://doi.org/10.1103/PhysRevE.68.041113>
46. M.N. van der Linden, J.P.K. Doye, A.A. Louis, Formation of dodecagonal quasicrystals in two-dimensional systems of patchy particles. *J. Chem. Phys.* **136**(5) (2012). <https://doi.org/10.1063/1.3679653>
47. L. Liu, Z. Li, Y. Li, C. Mao, Rational design and self-assembly of two-dimensional, Dodecagonal DNA quasicrystals. *J Am Chem Soc* **141**(10), 4248–4251 (2019). <https://doi.org/10.1021/jacs.9b00843>
48. D.F. Tracey, E.G. Noya, J.P.K. Doye, Programming patchy particles to form three-dimensional dodecagonal quasicrystals. *J. Chem. Phys.* **154**(19) (2021). <https://doi.org/10.1063/5.0049805>
49. M. Kube, et al., Revealing the structures of megadalton-scale DNA complexes with nucleotide resolution. *Nat. Commun.* **11**(1), (2020). <https://doi.org/10.1038/s41467-020-20020-7>
50. X.C. Bai, T.G. Martin, S.H. Scheres, H. Dietz, Cryo-EM structure of a 3D DNA-origami object. *Proc. Natl. Acad. Sci. USA* **109**(49), 20012–20017 (2012). <https://doi.org/10.1073/pnas.1215713109>
51. X. Liu et al., Complex silica composite nanomaterials templated with DNA origami. *Nature* **559**(7715), 593–598 (2018). <https://doi.org/10.1038/s41586-018-0332-7>
52. L. Nguyen, M. Doblinger, T. Liedl, A. Heuer-Jungemann, DNA-origami-templated silica growth by sol-gel chemistry. *Angew Chem. Int. Ed. Engl.* **58**(3), 912–916 (2019). <https://doi.org/10.1002/anie.201811323>
53. L. Shani, et al., DNA-assembled superconducting 3D nanoscale architectures. *Nat. Commun.* **11**(1), 5697 (2020). <https://doi.org/10.1038/s41467-020-19439-9>
54. N.B. Leontis, E. Westhof, Geometric nomenclature and classification of RNA base pairs. *RNA* **7**(4), 499–512 (2001). <https://doi.org/10.1017/S1355838201002515>
55. S.E. Butcher, A.M. Pyle, The molecular interactions that stabilize RNA tertiary structure: RNA motifs, patterns, and networks. *Acc. Chem. Res.* **44**(12), 1302–1311 (2011). <https://doi.org/10.1021/ar200098t>
56. L. Jaeger, N.B. Leontis, “Tecto-RNA: one-dimensional self-assembly through tertiary interactions. *Angew Chem. Int. Edit.* **39**(14), 2521 (2000). [https://doi.org/10.1002/1521-3773\(20000717\)39:14<2521::Aid-Anie2521>3.0.Co;2-P](https://doi.org/10.1002/1521-3773(20000717)39:14<2521::Aid-Anie2521>3.0.Co;2-P)
57. L. Jaeger, E. Westhof, N.B. Leontis, TectoRNA: modular assembly units for the construction of RNA nano-objects. *Nucleic Acids Res.* **29**(2), 455–463 (2001). <https://doi.org/10.1093/nar/29.2.455>
58. E.A. Doherty, J.A. Doudna, Ribozyme structures and mechanisms. *Annu. Rev. Biophys. Biomol. Struct.* **30**, 457–475 (2001). <https://doi.org/10.1146/annurev.biophys.30.1.457>
59. S.M. Elbashir, J. Harborth, W. Lendeckel, A. Yalcin, K. Weber, T. Tuschl, Duplexes of 21-nucleotide RNAs mediate RNA interference in cultured mammalian cells. *Nature* **411**(6836), 494–498 (2001). <https://doi.org/10.1038/35078107>
60. Z. Shajani, M.T. Sykes, J.R. Williamson, Assembly of bacterial ribosomes. *Ann. Rev. Biochem.* **80**, 501–526 (2011). <https://doi.org/10.1146/annurev-biochem-062608-160432>
61. J.C. Burnett, J.J. Rossi, RNA-based therapeutics: current progress and future prospects. *Chem. Biol.* **19**(1), 60–71 (2012). <https://doi.org/10.1016/j.chembiol.2011.12.008>

62. E. Laurenti et al., Inducible gene and shRNA expression in resident hematopoietic stem cells *In Vivo. Stem Cell.* **28**(8), 1390–1398 (2010). <https://doi.org/10.1002/stem.460>
63. A. Chworos et al., Building programmable jigsaw puzzles with RNA. *Science* **306**(5704), 2068–2072 (2004). <https://doi.org/10.1126/science.1104686>
64. S.M. Dibrov, J. McLean, J. Parsons, T. Hermann, Self-assembling RNA square. *Proc. Natl. Acad. Sci. USA* **108**(16), 6405–6408 (2011). <https://doi.org/10.1073/pnas.1017999108>
65. M. Endo, Y. Takeuchi, T. Emura, K. Hidaka, H. Sugiyama, Preparation of Chemically Modified RNA Origami Nanostructures. *Chem.-A Europ. J.* **20**(47), 15330–15333 (2014). <https://doi.org/10.1002/chem.201404084>
66. E.F. Khisamutdinov, D.L. Jasinski, P.X. Guo, RNA as a boiling-resistant anionic polymer material to build robust structures with defined shape and stoichiometry. *ACS Nano.* **8**(5), 4771–4781 (2014). <https://doi.org/10.1021/nn5006254>
67. C. Geary, P.W.K. Rothmund, E.S. Andersen, A single-stranded architecture for cotranscriptional folding of RNA nanostructures. *Science* **345**(6198), 799–804 (2014). <https://doi.org/10.1126/science.1253920>
68. K.A. Afonin et al., In vitro assembly of cubic RNA-based scaffolds designed in silico. *Nature Nanotech.* **5**(9), 676–682 (2010). <https://doi.org/10.1038/Nnano.2010.160>
69. I. Severcan, C. Geary, A. Chworos, N. Voss, E. Jacovetty, L. Jaeger, A polyhedron made of tRNAs. *Nat. Chem.* **2**(9), 772–779 (2010). <https://doi.org/10.1038/nchem.733>
70. J. Lee et al., RNA design rules from a massive open laboratory. *Proc. Natl. Acad. Sci. USA* **111**(6), 2122–2127 (2014). <https://doi.org/10.1073/pnas.1313039111>
71. J.W. Yu, Z.Y. Liu, W. Jiang, G.S. Wang, C.D. Mao, De novo design of an RNA tile that self-assembles into a homo-octameric nanoprism. *Nat. Commun.* **6** (2015). <https://doi.org/10.1038/ncomms6724>
72. S. Hamada, S. Murata, Substrate-assisted assembly of interconnected single-duplex DNA nanostructures. *Angew Chem. Int. Ed.* **48**(37), 6820–6823 (2009). <https://doi.org/10.1002/anie.200902662>
73. D. Liu et al., Branched kissing loops for the construction of diverse RNA homooligomeric nanostructures. *Nat. Chem.* **12**(3), 249–259 (2020). <https://doi.org/10.1038/s41557-019-0406-7>
74. D. Han et al., Single-stranded DNA and RNA origami. *Science* **358**(6369) (2017). <https://doi.org/10.1126/science.aao2648>
75. X. Qi, et al. Programming molecular topologies from single-stranded nucleic acids. *Nat. Commun.* **9**(1) (2018). <https://doi.org/10.1038/s41467-018-07039-7>
76. C. Geary, G. Grossi, E.K.S. McRae, P.W.K. Rothmund, E.S. Andersen, RNA origami design tools enable cotranscriptional folding of kilobase-sized nanoscaffolds. *Nat. Chem.* (2021). <https://doi.org/10.1038/s41557-021-00679-1>
77. S. Wuchty, W. Fontana, I.L. Hofacker, P. Schuster, Complete suboptimal folding of RNA and the stability of secondary structures. *Biopoly.* **49**(2), 145–165 (1999). [https://doi.org/10.1002/\(SICI\)1097-0282\(199902\)49:2%3c145::AID-BIP4%3e3.0.CO;2-G](https://doi.org/10.1002/(SICI)1097-0282(199902)49:2%3c145::AID-BIP4%3e3.0.CO;2-G)
78. C. Su, J.D. Weir, F. Zhang, H. Yan, T. Wu, ENTRNA: a framework to predict RNA foldability. *BMC Bioinfo.* **20**(1), 373 (2019). <https://doi.org/10.1186/s12859-019-2948-5>
79. S.M. Douglas, I. Bachelet, G.M. Church, A logic-gated nanorobot for targeted transport of molecular payloads. *Science* **335**(6070), 831–834 (2012). <https://doi.org/10.1126/science.1214081>
80. H. Lee et al., Molecularly self-assembled nucleic acid nanoparticles for targeted in vivo siRNA delivery. *Nat. Nanotechnol.* **7**(6), 389–393 (2012). <https://doi.org/10.1038/nnano.2012.73>
81. Y. Amir, E. Ben-Ishay, D. Levner, S. Ittah, A. Abu-Horowitz, I. Bachelet, Universal computing by DNA origami robots in a living animal. *Nat. Nanotechnol.* **9**(5), 353–357 (2014). <https://doi.org/10.1038/nnano.2014.58>
82. C. Wiraja, et al., Framework nucleic acids as programmable carrier for transdermal drug delivery. *Nat. Commun.* **10**(1) (2019). <https://doi.org/10.1038/s41467-019-09029-9>
83. S. Li et al., A DNA nanorobot functions as a cancer therapeutic in response to a molecular trigger in vivo. *Nat. Biotechnol.* **36**(3), 258–264 (2018). <https://doi.org/10.1038/nbt.4071>

84. A. Krissanaprasit et al., Genetically encoded, functional single-strand RNA origami: anticoagulant. *Adv. Mater.* **31**(21), e1808262 (2019). <https://doi.org/10.1002/adma.201808262>
85. A. Krissanaprasit et al., Multivalent aptamer-functionalized single-strand RNA origami as effective, target-specific anticoagulants with corresponding reversal agents. *Adv. Health. Mater.* **10**(11), e2001826 (2021). <https://doi.org/10.1002/adhm.202001826>
86. X. Qi et al., RNA origami nanostructures for potent and safe anticancer immunotherapy. *ACS Nano.* **14**(4), 4727–4740 (2020). <https://doi.org/10.1021/acsnano.0c00602>
87. C. Sigl et al., Programmable icosahedral shell system for virus trapping. *Nat. Mater.* (2021). <https://doi.org/10.1038/s41563-021-01020-4>
88. A. Gopinath, E. Miyazono, A. Faraon, P.W.K. Rothmund, Engineering and mapping nanocavity emission via precision placement of DNA origami. *Nature* **535**(7612), 401–405 (2016). <https://doi.org/10.1038/nature18287>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



From Molecules to Mathematics



Joanna Ellis-Monaghan and Nataša Jonoska

Abstract To celebrate the 40th anniversary of bottom-up DNA nanotechnology we highlight the interaction of the field with mathematics. DNA self-assembly as a method to construct nanostructures gave impetus to an emerging branch of mathematics, called here ‘DNA mathematics’. DNA mathematics models and analyzes structures obtained as bottom-up assembly, as well as the process of self-assembly. Here we survey some of the new tools from DNA mathematics that can help advance the science of DNA self-assembly. The theory needed to develop these tools is now driving the field of mathematics in new and exciting directions. We describe some of these rich questions, focusing particularly on those related to knot theory, graph theory, and algebra.

1 Introduction

Seeman’s ground-breaking work in DNA self-assembly that initiated the field of DNA nanotechnology has had surprisingly broad impacts on many other fields. In particular it has impacted the field of nanotechnology more widely [1–4], the science of computing and molecular programming (e.g., [5–10] as well as articles in the proceedings of DNAX for over 26 years), bottom-up nano-assembly [11–19] including computer science and mathematics [20–28]. Advances in the sciences often drive the creation of completely new mathematical fields. The field of DNA nanotechnology is a prime example of this phenomenon. It has been steadily spawning a slew of new mathematical problems. These problems are now giving birth to a new field of mathematics that might be called ‘DNA-mathematics’.

We live in a world dominated by computers, mobile devices, advances in technology, and the internet. We have foodwebs, social networks, and contact tracing in

J. Ellis-Monaghan (✉)
University of Amsterdam, Amsterdam, The Netherlands
e-mail: j.a.ellismonaghan@uva.nl

N. Jonoska
University of South Florida, Florida, USA
e-mail: jonoska@mail.usf.edu

© The Author(s) 2023
N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*,
Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_11

epidemiological models. The mathematics of interconnections, in particular network and graph theory, became essential for better understanding of many modern life situations. For example, graph drawing tools led to effective computer chip layouts while random graphs are often used in modeling the worldwide web. We now see this same phenomenon of new mathematics emerging from questions about shapes and interconnections driven by nucleic acids structures in biology and DNA nanotechnology.

In his pioneering article on DNA self-assembly Ned Seeman proposed using DNA molecules, called branched junction molecules, to build complex nanostructures [29]. These molecules are shaped like starfish with anywhere from three to twelve arms and can attach to one another via sequences of complementary DNA bases at the ends of their arms (sticky ends), as though the starfish are holding hands. Thus, for example, four three-armed branched junction molecules can join together to form the outline of a tetrahedron.

In 2006, a major advance in DNA self-assembly was obtained by Rothemund's introduction of DNA origami [30], a method where a single-stranded DNA plasmid outlines a pre-designed shape and about 200–250 short strands of DNA complementary to different locations of the plasmid are used to assemble and secure the structure. Because of the potential applications of self-assembling DNA nanotechnology, especially in medicine, and also in nanoscale robotics, circuitry, and biosensors, hundreds of laboratories around the world today focus on it.

Many challenges arise in designing DNA molecules that are to self-assemble into a desired shape. While some of the challenges involve chemical processes, many others involve structural questions such as which arms of the branched junction molecules should attach to which other arms, or how to route the scaffolding strand and staples through the desired structure, so that the smaller molecules then self-assemble into exactly the desired larger shape. As the experimental advances evolve rapidly, mathematical foundations to address new and upcoming questions that arise from laboratory experiments are becoming more essential.

DNA self-assembly now involves novel mathematical approaches and tools that inform the design of the structure both to assemble the nanocomplex as well as to ease the analysis of the experimental results. It becomes particularly exciting, from a mathematical perspective, when these approaches diverge from the original stimulus to problems of intrinsic mathematical interest independent of the initial application, thus significantly expanding the scope of the mathematical investigations.

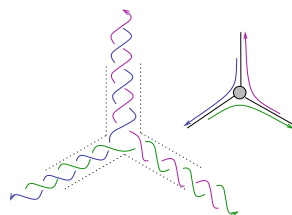
Fortunately, problems involving shapes and interconnections are at the heart of mathematics and mathematicians have become natural collaborators to DNA self-assembly researchers. New mathematical formalism is being developed to solve mathematical problems arising from DNA self-assembly. Many of the target molecular shapes have wireframe structures, such as the outlines of a cube [31] or octahedron [32], or 2D and 3D lattices [11, 14], or even triangular mesh bunny rabbits and lacy snowflakes [33]. Since the outlines of these wireframe structures correspond to the edges of a graph and the corners of the shape to the vertices of a graph, graph theory (in particular topological graph theory) and knot theory have emerged as an excellent platform to study assembly problems. However, existing graph and

knot theory lack sufficient descriptive properties to capture the essence of DNA self-assembly and cannot always address the new problems arising from the self-assembly application. Thus, a new subfield in mathematics is born, *DNA mathematics*, which develops new mathematical tools for, and arises from, bottom-up assembly.

As shared in the sections below, the mathematical problems driven by self-assembly processes are rich and prolific. They lead discrete mathematics and topology beyond current mainstream trends in these fields and hence break open new directions such as edge-outer embeddability, origami knotting, and new algebraic languages to describe structures. These theoretical directions are open-ended, generally scale independent, and will lay a broad foundation for future growth applicable to self-assembly in many settings, from nano to macro.

2 Flexible Tiles and New Graph Invariants

Fig. 1 Three-armed branched DNA molecule seen as three-valent vertex in a graph with three half-edges. The single-stranded extensions of the arms encode the bond types



The appearances of the first three-dimensional DNA structures such as the cube [31], and the truncated octahedron [32], arrived at the same time as the idea of using nucleic acids and biomolecules for computations when molecular-based information processing was initiated with Adleman's seminal paper [34]. If computation is to be performed with molecular structures, assemblies of arbitrary 3D wireframe or graph-like molecules without inherent symmetry may be necessary. The first such construct of a graph with six vertices using branched junction molecules representing vertices, and regular duplex molecules representing edges, was reported in [35, 36]. These molecules contained non-paired nucleotides throughout the duplexes making the arms of the molecules flexible such that, in the process of assembly, their sticky ends could easily join their respective complements. After ligating the nicks (breaks in the strands), the resulting (graph) structure consisted of a single cyclic DNA strand, which could conform in at least two knotted topologies [36]. These experimental results initiated a mathematical model that captures some of the design challenges of the flexible armed tiles used in construction of spatially embedded graph structures.

A combinatorial abstraction that consists of a vertex with half-edges labeled by the sticky-end types on the arms of the branched junction molecule is called a *tile* (Fig. 1). It can be denoted as a multi-set of *bond types* indicating the types of sticky-end types flanking the ends of the arms. The complementary sticky-end types are

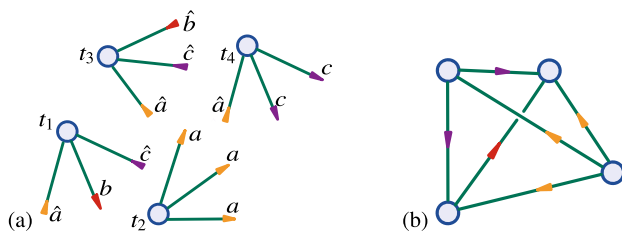


Fig. 2 **a** Four tiles with their corresponding bonding types in colored arrowheads appearing at the end of the half-edges. The complementary bond types are indicated with reversed arrowheads. **b** A tetrahedron realized by a pot containing the tiles in **(a)**

denoted by marking bond types with two complementary versions, indicated with a symbol from an alphabet (e.g., a, b, c, \dots) and a hatted version of the symbol (e.g., $\hat{a}, \hat{b}, \hat{c}, \dots$). Multiple entries of the same bond type are indicated by the exponent to the corresponding symbol. For example, Fig. 2a shows tiles $t_1 = \{\hat{a}, b, \hat{c}\}$, $t_2 = \{a^3\}$, $t_3 = \{\hat{a}, \hat{b}, \hat{c}\}$, and $t_4 = \{\hat{a}, c^2\}$.

A collection of tile types forms a *pot*, and a target graph G (or other 3D wireframe structure) is *realized* by a pot if it can be obtained by matching the vertices of G with tile types and identifying two half-edges of tiles having complementary versions of the same bond type with an edge of the graph. In the most basic setting G is an abstract graph, but other models also consider the geometry of the target graph. Each pair of half-edges forming an edge is subject to the restriction that a symbol is always paired with its hatted version and vice versa. Abstractly, a graph is considered assembled from tiles, if every vertex of the graph corresponds to a vertex of a tile and each edge corresponds to a pair of half edges with complementary sticky-ends joined together to form a bond edge. This is equivalent to finding an edge-labeled orientation of the graph, with the arrows pointing from the unhatted to the hatted half edges making up the labeled edge. In theory, any covering of a graph with a cycle realized by a pot can also be realized by the pot, but entropy disfavors these larger constructs. Further description of the model can be found in [28, 37–39].

There are two aspects to consider in such a ‘pot’ and ‘tile’ set-up. One aspect considers properties of the pot: what types of graphs can be realized by a pot, how many isomorphism classes are there, are all structures that are realized complete (i.e., there are no unbonded half-edges), or, can they be always completed, etc. The other aspect asks questions about the graphs and the structures: what is the most ‘efficient’ pot (in terms of minimal number of tile types needed and bond types needed) that can realize a given graph, how are the properties of the pot related to other invariants of the graph in question, etc.

The properties of pots are most suitably studied with methods from linear algebra by associating a matrix to each pot whose ij -entries are ratios of bond type i for a tile type j present in a pot. Using this matrix, one can determine the stoichiometry of the self-assembly to ensure certain types of pot properties [39] (see also [37]). Although useful, this linear algebra approach does not capture the difficult task of

better understanding of the process of assembly. As larger complexes form in a test tube, the thermodynamic properties can change and further assembly of larger structures may depend of the entropic conditions in the test tube, hence an expansion of the model to include thermodynamic properties may be necessary. One can also associate computational complexity classes with types of pots, where the number of tiles used in a structure that computes a problem is a base for the associated classes [23]. It was shown that 3-colorability of a graph, or other NP-hard¹ problems could be solved in $\mathcal{O}(1)$ bio-steps [23, 36] by physically assembling the structure that emulates the solution of the problem. This was also shown experimentally by Seeman's lab [7]. However, it seems we are yet to develop a good model that can encapsulate information processing by 3D structures, or a model that can tell us about computations by shapes.

Determining an 'efficient' pot for a target graph G means specifying a pot of tiles that realizes combinatorially, with a minimum number of tile types, and using a minimum number of bond-edge types. Recall that in the assembly process there are typically $\sim 10^{15}$ tiles of each type present in a tube, each tile can be used within a structure multiple times (or often taken, an arbitrarily large number of times). For the construction of G one can require an edge labeled orientation of the graph using a minimal alphabet for the labels. This becomes a new graph invariant $B(G)$, the *minimal bond alphabet*. In order to prevent the self-aggregation of tiles during initial synthesis, a natural experimental requirement can be added so that no two half-edges of the same tile have complementary labels. This corresponds to prohibiting loops in the target graph. One can further consider the *minimal set of vertex types* (the resulting molecular components) needed for construction of the graph G , an invariant $T(G)$. The invariants $T(G)$ and $B(G)$ are new graph invariants of intrinsic interest that have yet to be determined for most graphs. Except for a few special graph classes in some specific pot setting, these combinatorial questions are wide open. Familiar graph theoretical tools such as coloring, chromatic numbers, classical graph automorphism, etc., do not appear to determine $T(G)$ or $B(G)$ in any of the settings. The chromatic number provides a lower bound in some settings, although a poor one as it and $T(G)$ can be arbitrarily far apart [28].

The problem can further be confounded by the constraints of several different experimental settings, including the degree of flexibility or rigidity of the arms and the strength of the cohesive sites, as well as yield considerations such as whether or not the incidental creation of complete complexes smaller than the target graph is acceptable. In [40] it was confirmed that it is NP-hard to determine the output of a pot and to determine whether a pot that assembles a target structure will also assemble unwanted smaller structures.

Mathematical formalism, models, and design tools for types of self-assembly can be found here [28, 37, 38, 40, 41], with [42] further specifying the inter-arm angles and cohesive end rotation orientations for rigid tiles. These resources provide provably optimal design strategies, i.e., combinatorial specification for the minimum

¹ Loosely speaking, NP-hard problems are those shown to be at least as difficult as a large set of other problems for which no fast algorithms are known.

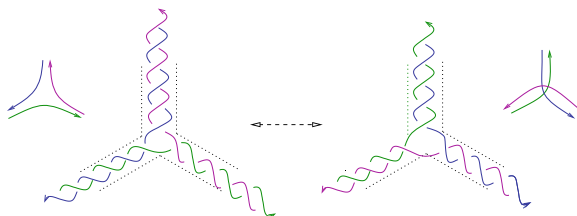


Fig. 3 Changing strand connections at a three-valent vertex in a graph. As DNA strands have polarity with a phosphate end (5') and a hydroxyl group on the other end (3'), the arrowheads indicate the 3' ends. The strands bind with opposite polarities. A three-armed junction molecule can exhibit two types of connections as depicted, e.g., the green strand has a direction left to right in the left figure, while it has direction left to top in the right figure

number of cohesive regions and the minimum number of branched junction molecule types, to assembly a given wireframe target structure, for several common graph classes. Further computations of these invariants for specific graphs such as Platonic and Archimedes solids and prisms can be found in [43]. A further extensive body of work can be found on DNA Wang tiles, which are the restrictive case of completely rigid four-armed tiles. See, among others, Refs. [5, 44–49].

3 DNA Strand Routing and Topological Graph Theory

Determining routes for a scaffolding strand throughout assembly targets is integral to both DNA origami [27, 50, 51] and experimental verification of graph constructs [7]. A vertex of a graph structure can be traced by DNA in multiple ways such that the resulting DNA structure is assembled by different sets of cross-hybridizing cyclic molecules.

For a three-valent vertex, the DNA structure representing such vertex can have two local strand configurations (vertex connections) as shown in Fig. 3. One configuration can be represented by ‘non-crossing’ strands at a vertex, and the other is obtained by ‘crossing strands’. By changing vertex connections, the number of cyclic molecules assembling the structure can vary. Figure 4a and b show two strand outlines of the same triangular prism graph. In Fig. 4a, the strand connections at each vertex in this planar representation do not ‘cross’ each other and the graph is outlined by five cyclic strands. By changing the strand connection at vertices v_1 and v_4 with the other configuration, the graph can be routed by a single circuit, hence outlined by a single cyclic molecule (as shown in Fig. 4b).

Early work focused on number of cyclic molecules in such double-strand covers, where the target graph is covered by a set of circuits so that every edge is covered exactly twice, as in Fig. 4a [52]. These circuits form the facial walks of an oriented embedding of the graph, and thus correspond to a strong version of the cycle double cover conjecture, as described below. Double-strand covers have re-emerged recently in novel origami methods such as [53].

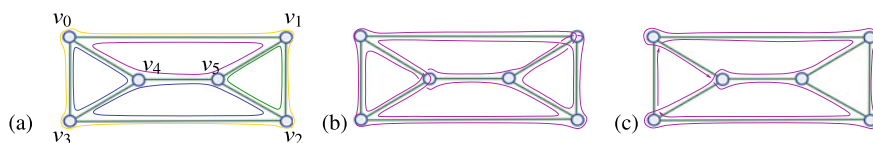


Fig. 4 Possible routes for DNA strands in a triangular prism graph: **a** following the faces of a plane embedding produces 5 faces, corresponding to a maximum strand number of 5. **b** by changing strand connections at two vertices, the graph is traced by a single strand, reflecting that it is upper-embeddable on a double torus (see Fig. 5a). **c** A minimal length reporter strand for the graph, giving an edge-outer embedding on the torus (see Fig. 5b)

In DNA settings, strand routings of structures require turns at vertices that are constrained, i.e., routes may not doubleback on an edge. Because DNA strands are oriented, any repeated edges in a strand route must be traversed in the opposite direction when revisited. Thus, by folding a DNA origami in a graph structure, the strand routing objective is to find a route in the graph which meets these constraints, traverses every edge, ideally, with a minimum number of repeated edges. A similar problem arises in determining a route for the reporter strand, where after an experiment has been conducted, a single strand traversing the structure is extracted from the assembled construct and analyzed to yield the experimental data (see [7]). Here again, the objective is a route covering the graph with a minimum number of repeated edges. Such a minimal strand routing for the triangular prism graph is shown in Fig. 4c. In this case, the connections at three vertices, v_0 , v_3 , and v_4 are changed relative to the connections in Fig. 4a.

The strand routing problem is in general intractable, even in the special case of an Eulerian surface mesh when an optimal route follows face boundaries. In this case the problem corresponds to finding *A*-trails in the surface, which are Eulerian circuits that turn either left or right at each vertex. This problem is known to be NP-hard (see [54]) even on the plane. In [51] it is proven that the strand routing problem remains NP-hard even for graphs of maximum degree 8. A survey of approaches to the strand routing problem is given in [27]. The problem can be translated to the traveling salesman problem (TSP), which makes the wealth of software available for solving the TSP available to DNA origami applications.

A notable alternative approach is given in [33]. Provided that the graph is an augmented triangulation of a surface topologically equivalent to a sphere, there is a fast algorithm for routing the scaffolding strand, albeit at the cost of duplicating a number of edges in the final nanostructure.

Mathematically, different routings correspond to different cellular embeddings of a graph in a surface (drawings of a graph in a surface such that its edges do not cross each other and such that the complement of the graph in the surface is a set of topological disks). In general, the number of strands outlining a graph is counter-proportional to the genus² of the surface in which the graph is embedded.

² Intuitively, the genus of an orientable surface is the number of holes, or handles it has, so that a torus has genus one, a double torus has genus two, and so forth.

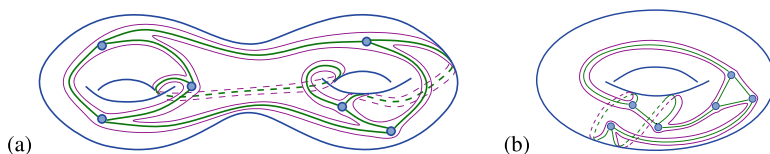


Fig. 5 Embeddings of the triangular prism graph in Fig. 4. **a** Embedding of the graph in a double torus. The graph can be realized as a tertiary structure of a single DNA strand, shown here tracing the single face. **b** An embedding of a graph in a torus with a minimum length reporter strand tracing face boundary that covers all edges at least once

As has repeatedly been the case with self-assembly applications, a wealth of new mathematical directions have emerged from the design objective, with *edge-outer embeddability* as described below being a particularly rich example expanding on topological graph theory.

An *edge-outer embedding* is a cellular embedding of a graph in an orientable surface where every edge lies on a special ‘outer’ face (there may be other faces too). Figure 5 shows an edge-outer embedding of the triangular prism graph in the double torus and the torus corresponding to the reporter strand routings in Fig. 4b, c. In [55] and [56] it was shown that the facial walk of an edge-outer face in an edge-outer embedding of the target graph exactly captures reporter strand routing constraints. While outer-planar and outer-projective-planar graphs, i.e., graphs in the plane or projective plane with all vertices on a single distinguished face, have been heavily studied [57], edge-outer embeddable graphs are an entirely new, yet very natural, construct.

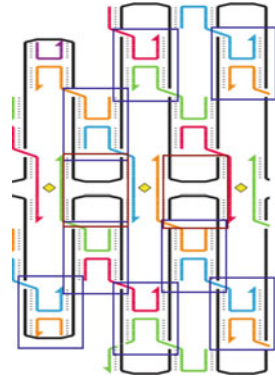
Conventional graph theoretical tools do not apply directly to edge-outer embeddability. For example, the Chinese postman problem lacks the bidirectional constraint while upper embeddability results require every edge to be covered exactly twice [58]. That every graph has such a reporter strand route, and hence an edge-outer embedding was shown in [55], with a short algorithmic proof given in [56]. At the heart of the proof is the reconfiguration shown in Fig. 3, which changes the cyclic order of the edges about a vertex. Finding a minimum such route, that is an embedding with a smallest possible edge-outer face, is in general NP-hard [56].

Such a computational complexity observation introduces a wealth of further open questions. If the graph is Eulerian, an optimal reporter strand route is simply an Euler circuit. Thus, in this case, Fleury’s algorithm provides a polynomial time solution to the problem. For what classes, other than Eulerian graphs, are there polynomial-time algorithms to find optimal reporter strand routes? Does there exist a polynomial-time algorithm guaranteed to return a route that is within $x\%$ of the optimal length for some reasonable x ?

It is also natural, both mathematically and for the application, to consider *maximum* length routes. If the graph is upper embeddable, i.e., has an orientable one-face embedding, then this embedding is a maximum edge-outer embedding with every edge covered twice in the facial walk (see Fig. 4b, c); however, not every graph is upper embeddable. As with the dichotomy in the complexity of maximum and minimum genus of the graph, it is possible that the maximum length problem is tractable.

4 DNA Origami and New Algebraic Structures

Fig. 6 A segment of a DNA origami structure with black scaffold and colored staples. The generators of the proposed monoid are boxed



Since their introduction to the scientific community, DNA origami structures [59] have become one of the most prevalent experimental substrates for a variety of 3D structures [12]. The method uses a single-stranded DNA plasmid vector as a scaffold that outlines the shape of the desired structure with help of so-called staple strands. The staple strands are short segments of single-stranded DNA (about 24 to 32 nucleotides long) that span across two or three segments of the scaffold strand to bind it in place. In order to achieve stability of the construct, it is necessary that the staple strands cross each other within the structure in an antiparallel way. An example of a portion of an origami structure is depicted in Fig. 6. (Figure adjusted from [30] with added boxes). Systematic mathematical methods to describe DNA origami structures have not been established. An algebraic language that describes DNA origami motivated by the Temperley-Lieb algebra that has been extensively used in physics and knot theory, particularly with the Jones polynomial and the Kauffman bracket, was introduced in [60, 61]. Such languages could provide a method for modifying (through strand displacements) a given design to achieve either a more effective and stable structure or a completely new geometric shape.

A well-studied Jones monoid \mathcal{J}_n has n generators are u_1, \dots, u_n . Each generator can be represented with $n + 1$ lines such that u_i has cap/cup connection of lines i and $i + 1$ as shown in Fig. 7a while all other lines are vertical. The product of generators is diagrammatically represented by concatenating the corresponding diagrams vertically. Figure 7b shows the product $u_i u_{i+1} u_i$ to the left of the equality where the vertical lines have not been depicted.

The set of relations of \mathcal{J}_n is depicted in Fig. 7b–d. For example, (b) represents the relation $u_i u_{i+1} u_i = u_i$. Although the generators and relations are symbolically written, they correspond directly to the depicted diagrams and planar isotopy. Taking this diagrammatic usage as an advantage, we use a monoid version that is a generalization of \mathcal{J}_n for describing DNA origami.

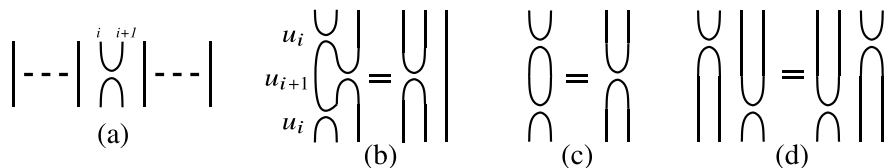


Fig. 7 Generator u_i in **(a)** and three types of relations in **(b, c, d)** of the Jones monoid. **b** Conjugacy relation $u_i u_{i+1} u_i = u_i$, **c** idempotent relation $u_i u_i = u_i$, **d** commuting relation $u_i u_j = u_j u_i$ for $|i - j| \geq 2$

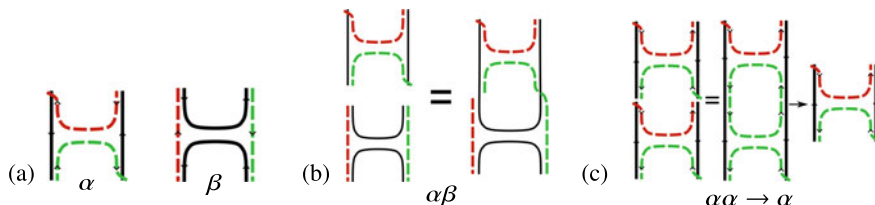


Fig. 8 **a** Two types of generators, cross-over staples connecting straight-line portions of the scaffold (denoted with α) and straight staples connecting two cross-over turning tips of the scaffold (denoted with β). **b** Multiplying generators is represented by structures connecting scaffold segments and respective staples, when they don't ‘cross-over’ the scaffold strand. **c** An idempotent rule; the product $\alpha\alpha$ has the top and bottom structure the same as α . The cyclic portion in the middle does not affect further products

The basis of the origami algebraic language is a monoid \mathcal{O}_n inspired by \mathcal{J}_n . It is defined by generators and relations whose diagrammatic representations and closures are parts of DNA origami with $n + 1$ scaffold strands tracing the structure (e.g., in Fig. 6 the structure is traced with 6 passings of the scaffold across top to bottom, and one can consider the structure as a representative of \mathcal{O}_5). The monoid has two types of generators, those that correspond to the scaffold strands connecting straight vertically in the structure (α 's) and those that correspond to staple strands connecting across (β 's). The generators α_i and β_i of the origami monoid \mathcal{O}_n of n strands are depicted in Fig. 8a, where the cap and cup are placed at the i -th position from the left, and the vertical lines surrounding these positions are not depicted.

We define a DNA origami monoid \mathcal{O}_n on $n + 1$ strands by generators and relations as follows. Generators of \mathcal{O}_n are α_i , for $i = 1, \dots, n$. The α_i 's represent local DNA foldings of a pair of staples of the form of cap and cup as indicated on the left in Fig. 8a. An additional set of generators β_i corresponds to pairs of caps and cups of the scaffold strand as indicated in the figure to the right. The index i indicates that the caps and cups are between the i th and $i + 1$ st strands of the structure. The set of relations is defined analogously to those of \mathcal{J}_n , while the closure of the diagram is defined in a manner similar to the so-called plat closure in knot theory [62].

To justify modeling DNA origami structures by words over the generators we make a correspondence between concatenations of generators α_i , β_i and connections of DNA segments. For a natural number $n \geq 2$, the set of generators of the monoid \mathcal{O}_n

is the set $\Sigma_n = \{\alpha_1, \alpha_2, \dots, \alpha_n, \beta_1, \beta_2, \dots, \beta_n\}$. For a product of two generators x_i and y_j in Σ_n , we place the diagram of the first generator above the second, lining up the scaffold strings of the two generators, and then we connect the respective scaffold strands. If the two generators are ‘far’ apart, that is, if for indices i and j we have $|i - j| \geq 2$, then no staple connection is performed. If the two generators are adjacent, that is, if for indices i and j it holds that $|i - j| \leq 1$, then we connect part of the staples. Since the staples are not too long within an origami structure, spanning only two or three segments of the scaffold, a convention of connecting staples representing a product of generators is motivated by the manner in which staples connect within the DNA origami structure. The staples of α -type generators protrude “outside” of the scaffold in Fig. 8a. The staples are connected everywhere *except* when two non-extending staple-ends would have to cross a scaffold to connect. With this convention it means that $\alpha_i\beta_i$ and $\beta_i\alpha_i$ are two distinct words, i.e., that α s and β s do not freely commute. The rules of connecting scaffold strands and staples assures that concatenation of three or more generators is associative. The graphical representation of a product $\alpha_i\beta_i$ is shown in Fig. 8b where the vertical strands surrounding the generators are not depicted.

The generators of the origami monoid satisfy a set of relations, according to the types of graphical structures they represent. One of the relations is a generator idempotent relation as shown in Fig. 8c. The relations are extensions of the relations of the Jones monoids. In [61] two scenarios of relations and graphical representations of the elements of the corresponding monoids are given. For each scenario, the number of all possible structures is provided through the number of equivalence classes of words (or elements in the corresponding origami monoid). Also a polynomial time algorithm exists that computes the shortest word for each equivalence class. A connection between the Green’s relations of an origami monoid and those of a direct product of Jones monoids [62] is given in [60]. In particular, it was shown that an epimorphism $p : \mathcal{O}_n \rightarrow \mathcal{J}_n \times \mathcal{J}_n$ induces the bijective correspondence on Green’s classes.

The definition of origami monoids is motivated by the Jones monoids, and the origami structure, concatenating two types of strands (scaffolds and staples), implies two types of generators with similar relations. This construction of doubling generators and imposing substitution relations can be generalized to other algebraic structures, or it can be generalized with more than two types of generators. These are algebraic problems directly arising from the experimental design of DNA origami.

5 DNA Origami and Origami Knots

DNA origami assembly can be confounded by knotting in the scaffolding strand. For example, in a preliminary experiment, an essentially planar target did not form well when a simply knotted (trefoil) scaffold was used [63]. Thus, tools are needed to avoid inadvertently knotted routes when designing the increasingly sophisticated targets of DNA origami. On the other hand, Seeman has shown it is possible to engineer single-

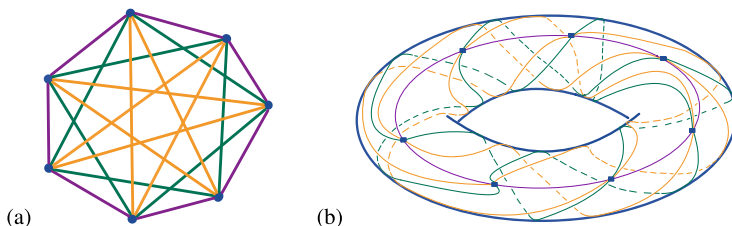


Fig. 9 **a** A complete graph with seven vertices, K_7 , with three Hamiltonian cycles colored distinctly. **b** An embedding of K_7 on a torus. The edges of the yellow and green cycles go around the torus handle. Both those cycles are knotted

stranded DNA with specified knotted topologies [64], and this capacity for controlled knotting may be used intentionally to design better (higher yield, larger, smaller, more symmetric, more robust, more topologically complex, etc.) nanostructures by deliberately exploiting the topology of the knotted scaffold.

When routing a single scaffold through a graph-like target structure, a typical design constraint is avoiding self-crossings [24, 33]. If the target structure is modeled by an Eulerian graph cellularly embedded on an oriented surface in 3-space, then some of these routes correspond to *A-trails*, which are Eulerian circuits that turn either ‘left’ or ‘right’ at each vertex. If the surface is a sphere, then all *A-trails* in the target structure are necessarily unknotted, but for higher-genus surfaces there are settings in which every *A-trail* is knotted [26, 27]. The complete graph on seven vertices, K_7 , is shown in Fig. 9a, and an embedding of K_7 on a torus is shown in Fig. 9b. Every vertex of K_7 has even valency (i.e., valency 6) and hence, the graph is Eulerian. Three distinct Hamiltonian cycles are depicted, purple, green and yellow. The embeddings of the yellow and the green cycles are knotted.

Since standard DNA origami scaffold strands are unknotted, the problems arise of determining whether there exists an unknotted route for a scaffolding strand in a given geometrically (or surface) embedded target graphs, and of characterizing graphs which have unknotted routes. Once again, a new area of mathematics emerges, as determining knotted and unknotted routing trails is fundamentally different from previously studied knots and links in graphs. Prior work focused on intrinsically knotted and linked cycles in graphs (see, for example, [65–67]), appearing as cycles in the graph. However, for this application, the knots are Eulerian circuits rather than cycles. For example, Conway and Gordon [67] showed that every embedding of K_7 in \mathbb{R}^3 has at least one knotted cycle. Figure 9 shows an embedding of K_7 on the torus. By Conway and Gordon [67], it contains at least one knotted Hamilton cycle (the green cycle, for example).

However, in contrast, [24, 26] have shown that every *A-trail* in the embedding of K_7 in the torus is unknotted. This follows since the embedding of K_7 in the torus is checkerboard colorable, and hence every *A-trail* bounds the union of the black regions, and hence a disk. It is consequently unknotted (Fig. 10). Moreover, in [24]

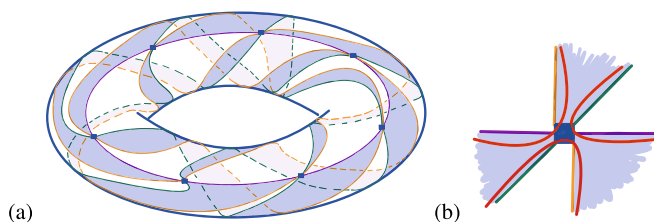


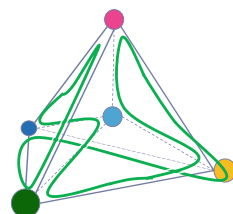
Fig. 10 **a** A checker-board coloring of the embedded K_7 , **b** A-trail transitions at a vertex of the embedding joining the shaded regions of the embedded K_7 . These A-trails bound a disk consisting of the shaded regions on the torus and therefore they outline K_7 with an unknotted routing

it is shown that every Eulerian circuit is knotted on a torus if and only if there is a non-checkerboard colorable embedding of the graph.

For geometrically embedded graphs that are not necessarily embedded on some surface, and hence for which A-trails are not necessarily defined, the initial challenge is formalizing the constraints on how the scaffolding strand may pass through vertices.

A-trails can be generalized to *O-trails*, which permits analyzing the knotting of strand routes in more general geometric embeddings of graphs in space, or embedding on a surface where ‘non-crossing’ smoothings at the vertices are defined [24]. For example, Fig. 11 shows K_5 embedded as a tetrahedron with a body-center vertex, together with an *O-trail* through it.

Fig. 11 An *O-trail* in a geometric embedding of K_5



O-trails are Eulerian circuits in geometrically embedded Eulerian graphs so that at each vertex the edge pairings determined by the circuit are all non-crossing (i.e., there is a topological disk containing the half edges about the vertex in which both the turnings determined by the circuit and the orthogonal turnings are non-crossing). For A-trails in a surface mesh, the disk is just a small neighborhood of the vertex, so A-trails are a special subclass of *O-trails*.

The DNA origami application needs to understand and control the behavior of *O-trails*, both knotted and unknotted, in fixed geometric graphs, as well as over all possible embeddings of an abstract graph. Finding *O-trails* is NP-complete in general, since finding A-trails is, even for plane graphs, and the problem of determining if a knot is unknotted is NP-hard.

Both unknotted O -trails and constrained knot embeddings are entirely new directions in knot theory. There has been extensive prior work on knots in graphs, but this focused on intrinsically linked cycles or knotted Hamiltonian cycles [66–68]. Here however, the DNA origami constraints lead to questions of knots in Eulerian circuits, which opens an analogous and equally rich line of inquiry, but now in a completely new setting. Although embedding a given knot on a standard surface is known, considering geometric knot invariants under requirements imposed by DNA conformation is novel. Furthermore, characterizing graphs with unknotted O -trails will help identify target structures amenable to origami self-assembly from unknotted scaffolding strands and complements current experimentation [15]. In most applications a specific geometric embedding of the target is sought, so a pragmatic goal is to characterize classes of embedded graphs for which determining unknotted O -trails is computationally tractable.

6 Where Next?

The mathematical models discussed here generally follow what is becoming a common pattern. The first step is always developing a theoretical formalism that simultaneously captures the essence of a design problem, while providing a foundation for the following theoretical work. The second step is developing preliminary results for specific experiments.

When moving from specific experiments to seeking general strategies, e.g., via fast computer algorithms, particular design problems often can be prohibitively difficult. Thus the third step frequently is providing fast algorithms or proving that fast algorithms for general solutions might not be possible, i.e., are NP-hard.

Discovering that a DNA self-assembly problem is NP-hard is exciting theoretically, because it immediately opens a plethora of related computational problems. These include seeking approximation algorithms, finding optimal solutions for particular families of problems, and devising pragmatic approaches for urgently needed special cases.

Although an NP-hardness result does not help a lab trying to conduct its next experiment, it does prevent wasted effort seeking general strategies. Furthermore, a specific NP-hard problem may be reduced to another known NP-hard problem, for example, the Traveling Salesman Problem (TSP), but for which there already exist robust tools such as fast approximation algorithms and algorithms optimized for special cases. These then may be adapted to the self-assembly problem, as with the TSP reduction for strand routing in [27].

DNA self-assembly is now a rich source of theoretical problems. These problems and their solutions advance both the mathematics and the self-assembly processes. The lab constraints lead to new mathematics and often breakthrough new directions, as we point out here. The mathematical foundations supporting innovations in science and industry are often seen years after they are developed, and are sometimes unacknowledged, because by the time the theory has been applied, it has entered

the mainstream of engineering. This emergent mathematical theory, this new ‘DNA mathematics’, lays the foundations that can support the further growth of DNA self-assembly technologies, and whose potential applications, although inspired by DNA self-assembly, may still be yet to come.

Acknowledgements The authors dedicate this paper to the memory of Nadrian C. Seeman in gratitude for longlasting collaboration and for his original and insightful thinking accompanied by a plethora of experimental results that initiated DNA mathematics. We also thank the E. Czeizler, H.J. Hoogeboom and G. Pangborn for their thoughtful suggestions. This research was supported by NSF grants CCF-2107267, DMS-1800443, and DMS-2054321, as well as the NSF-Simons Southeast Center for Mathematics and Biology (SCMB) through the grants National Science Foundation DMS1764406 and Simons Foundation/SFARI 594594.

References

1. Y. Kamiya, H. Asanuma, Light-driven DNA nanomachine with a photoresponsive molecular engine. *Acc. Chem. Res.* **47**(6), 1663–1672 (2014)
2. S. Jia, J. Wang, M. Xie, J. Sun, H. Liu, Y. Zhang, J. Chao, J. Li, L. Wang, J. Lin, K.V. Gothelf, C. Fan, Programming DNA origami patterning with non-canonical DNA-based metallization reactions. *Nat. Commun.* **10**(1), 5597 (2019)
3. C. Cui, D.H. Park, D.J. Ahn, Organic semiconductor-DNA hybrid assemblies. *Adv. Mater.* **32**(51), 2002213 (2020)
4. Y. Li, L. Song, B. Wang, J. He, Y. Li, Z. Deng, C. Mao, Universal pH-responsive and metal-ion-free self-assembly of DNA nanostructures. *Angew. Chem. Int. Ed.* **130**(23), 6892–6895 (2018)
5. D. Woods, D. Doty, C. Myhrvold, J. Hui, F. Zhou, P. Yin, E. Winfree, Diverse and robust molecular algorithms using reprogrammable DNA self-assembly. *Nature* **567**(7748), 366–372 (2019)
6. P.W.K. Rothmund, N. Papadakis, E. Winfree, Algorithmic self-assembly of DNA sierpinski triangles. *PLoS Biol.* **2**(12), e424 (2004)
7. G. Wu, N. Jonoska, N.C. Seeman, Construction of a DNA nano-object directly demonstrates computation. *Biosystems* **98**(2), 80–84 (2009)
8. B. Chakraborty, N. Jonoska, N.C. Seeman, A programmable transducer self-assembled from DNA. *Chem. Sci.* **3**(1), 168–176 (2012)
9. R.D. Barish, P.W.K. Rothmund, E. Winfree, Two computational primitives for algorithmic self-assembly: copying and counting. *NanoLetters* **5**, 2586–2592 (2005)
10. H. Gu, J. Chao, S.-J. Xiao, N.C. Seeman, A proximity-based programmable DNA nanoscale assembly line. *Nature* **465**, 202–205 (2010)
11. E. Winfree, F. Liu, L.A. Wenzler, N.C. Seeman, Design and self-assembly of two-dimensional DNA crystals. *Nature* **394**(6693), 539–544 (1998)
12. Y. Hao, M. Kristiansen, R. Sha, J.J. Birktoft, C. Hernandez, C. Mao, N.C. Seeman, A device that operates within a self-assembled 3D DNA crystal. *Nat. Chem.* **9**(8), 824–827 (2017)
13. X. Wang, R. Sha, M. Kristiansen, C. Hernandez, Y. Hao, C. Mao, J.W. Canary, N.C. Seeman, An organic semiconductor organized into 3D DNA arrays by “bottom-up” rational design. *Angew. Chem. Int. Ed.* **56**(23), 6445–6448 (2017)
14. J. Zheng, J.J. Birktoft, Y. Chen, T. Wang, R. Sha, P.E. Constantinou, S.L. Ginell, C. Mao, N.C. Seeman, From molecular to macroscopic via the rational design of a self-assembled 3D DNA crystal. *Nature* **461**(7260), 74–77 (2009)
15. F. Hong, F. Zhang, Y. Liu, H. Yan, DNA origami: Scaffolds for creating higher order structures. *Chem. Rev.* **117**(20), 12584–12640 (2017)

16. R. Inuma, Y. Ke, R. Jungmann, T. Schlichthaerle, J.B. Woehrstein, P. Yin, Polyhedra self-assembled from DNA tripods and characterized with 3d DNA-PAINT. *Science* **344**(6179), 65–69 (2014)
17. S. Douglas, A. Marblestone, S. Teerapittayanon, A. Vazquez, G. Church, W. Shih, Rapid prototyping of 3d DNA-origami shapes with caDNAno. *Nucleic Acids Res.* **15**, 5001–5006 (2009)
18. W. Xu, B. Nathwani, C. Lin, J. Wang, E. Karatekin, F. Pincet, W. Shih, J.E. Rothman, A programmable DNA origami platform to organize snares for membrane fusion. *J. Am. Chem. Soc.* **138**(13), 4439–4447 (2016). (PMID: 26938705)
19. D. Liu, Y. Shao, G. Chen, Y.C. Tse-Dinh, J.A. Piccirilli, Y. Weizmann, Synthesizing topological structures containing RNA. *Nat. Commun.* **8**(1), 14936 (2017)
20. E. Winfree, *Algorithmic Self-Assembly of DNA*. Ph.D thesis (California Institute of Technology, 1998)
21. D. Woods, Intrinsic universality and the computational power of self-assembly. *Philos. Trans. R. Soc. A. Math. Phys. Eng. Sci.* **373**(2046), 20140214 (2015)
22. M. Vasić, D. Soloveichik, S. Khurshid, Crn++: Molecular programming language. *Nat. Comput.* **19**(2), 391–407 (2020)
23. N. Jonoska, G.L. McColm, Complexity classes for self-assembling flexible tiles. *Theoret. Comput. Sci.* **410**(4–5):332–346 (2009). Computational Paradigms from Nature
24. A. Mohammed, N. Jonoska, M. Saito, The topology of Scaffold routings on non-spherical mesh wireframes, in *26th International Conference on DNA Computing and Molecular Programming (DNA 26)*. ed. by C. Geary, M. J. Patitz, vol. 174 of *Leibniz International Proceedings in Informatics (LIPIcs)*, (Dagstuhl, Germany), pp. 1:1–17 (Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020)
25. M.N. Ellingham, J.A. Ellis-Monaghan, Edge-outer graph embedding and the complexity of the DNA reporter strand problem. *Theoret. Comput. Sci.* **785**, 117–127 (2019)
26. A. Morse, W. Adkisson, J. Greene, D. Perry, B. Smith, J. Ellis-Monaghan, G. Pangborn, DNA origami and unknotted A-trails in torus graphs. *J. Knot Theor. Ramifications* **29**(07), 2050041 (2020)
27. J. Ellis-Monaghan, G. Pangborn, N. Seeman, S. Blakeley, C. Disher, M. Falcigno, B. Healy, A. Morse, M. Singh, B. Westland, Design tools for reporter strands and DNA origami scaffold strands. *Theor. Comput. Sci.* **671**(6), 69–78 (2017)
28. J. Ellis-Monaghan, G. Pangborn, L. Beaudin, D. Miller, N. Bruno, A. Hashimoto, Minimal tile and bond-edge types for self-assembling DNA graphs, in *Discrete and Topological Models in Molecular Biology*, ed. by N. Jonoska, M. Saito, pp. 241–270 (Springer, Berlin/Heidelberg, 2014)
29. N.C. Seeman, Nucleic acid junctions and lattices. *J. Theoret. Biol.* **99**(2), 237–47 (1982)
30. P.W. Rothmund, Folding DNA to create nanoscale shapes and patterns. *Nature* **440**(7082), 297–302 (2006)
31. J. Chen, N. Seeman, Synthesis from DNA of a molecule with the connectivity of a cube. *Nature* **350**, 631–633 (1991)
32. Y. Zhang, N. Seeman, The construction of a DNA truncated octahedron. *J. Am. Chem. Soc.* **160**, 1661–1669 (1994)
33. E. Benson, A. Mohammed, J. Gardell, S. Masich, E. Czeizler, P. Orponen, B. Högberg, DNA rendering of polyhedral meshes at the nanoscale. *Nature* **523**(7561), 441–444 (2015)
34. L. Adleman, Molecular computation of solutions to combinatorial problems. *Science* **266**(5187), 1021–1024 (1994)
35. N. Jonoska, P. Sa-Ardyen, N. Seeman, Computation by self-assembly of DNA graphs. *Genet. Program. Evolvable Mach.* **4**, 123–137 (2003)
36. P. Sa-Ardyen, N. Jonoska, N. Seeman, Self-assembling DNA graphs. *Nat. Comput.* **4**, 427–438 (2003)
37. J. Ellis-Monaghan, N. Jonoska, G. Pangborn, Tile-based DNA nanostructures: mathematical design and problem encoding, in *Algebraic and Combinatorial Computational Biology*, ed. by R. Robeva, M. Macauley, MSE/Mathematics in Science and Engineering, pp. 35–60 (Academic Press, 2019)

38. S. Bonvicini, M.M. Ferrari, On the minimum number of bond-edge types and tile types: an approach by edge-colorings of graphs. *Discrete Appl. Math.* **277**, 1–13 (2020)
39. N. Jonoska, G.L. McColm, A. Staninska, On stoichiometry for the assembly of flexible tile DNA complexes. *Nat. Comput.* **10**(3), 1121–1141 (2011)
40. L. Almodóvar, J. Ellis-Monaghan, A. Harsy, C. Johnson, J. Sorrells, *Computational Complexity and Pragmatic Solutions for Flexible Tile Based DNA Self-Assembly* (2021)
41. J. Ellis-Monaghan, G. Pangborn, L. Beaudin, D. Miller, N. Bruno, A. Hashimoto, Minimal tile and bond-edge types for self-assembling DNA graphs, in *Discrete and Topological Models in Molecular Biology, Natural Computing Series* (2013)
42. M. Ferrari, A. Cook, A. Houlihan, R. Rouleau, N. Seeman, G. Pangborn, J. Ellis-Monaghan, Design formalism for dna self-assembly of polyhedral skeletons using rigid tiles. *J. Math. Chem.* **56**, 1365–1392 (2018). Funding Information: Acknowledgements The work of Joanna Ellis-Monaghan, Greta Pangborn, and Nadrian C. Seeman was supported by the National Science Foundation (NSF) under Grant DMS-1332411. Publisher Copyright: 2018, Springer International Publishing AG, part of Springer Nature
43. Ellis-Monaghan and Pangborn DNA Self-Assembly Website. <http://sites.google.com/site/nanoselfassembly>. Accessed 14 Aug 2017
44. E.D. Demaine, M.J. Patitz, R.T. Schweller, S.M. Summers, Self-assembly of arbitrary shapes using RNase enzymes: meeting the Kolmogorov bound with small scale factor (extended abstract), in *28th International Symposium on Theoretical Aspects of Computer Science*, vol. 9 of LIPIcs. Leibniz International Proceedings of Informatics, pp. 201–212 (Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2011)
45. S. Cannon, E. Demaine, M. Demaine, S. Eisenstat, M.J. Patitz, R. Schweller, S.M. Summers, A. Winslow, Two hands are better than one (up to constant factors): self-assembly in the 2HAM vs. aTAM, in *30th International Symposium on Theoretical Aspects of Computer Science*, vol. 20 of LIPIcs. Leibniz International Proceedings of Informatics, pp. 172–184 (Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2013)
46. M. Göös, T. Lempäinen, E. Czeizler, P. Orponen, Search methods for tile sets in patterned DNA self-assembly. *J. Comput. System Sci.* **80**(1), 297–319 (2014)
47. R. Schweller, A. Winslow, T. Wylie, “mplexities for high-temperature two-handed tile self-assembly, in *DNA Computing and Molecular Programming*, vol. 10467 of Lecture Notes in Computer Science, pp. 98–109 (Springer, Cham, 2017)
48. D. Doty, T.A. Rogers, D. Soloveichik, C. Thachuk, D. Woods, Thermodynamic binding networks, in *DNA Computing and Molecular Programming*, vol. 10467 of Lecture Notes in Computer Science, pp. 249–266 (Springer, Cham, 2017)
49. M. Göös, T. Lempäinen, E. Czeizler, P. Orponen, Search methods for tile sets in patterned DNA self-assembly. *J. Comput. Syst. Sci.* **80**(1), 297–319 (2014)
50. R. Veneziano, S. Ratanalert, K. Zhang, F. Zhang, H. Yan, W. Chiu, M. Bathe, et al., Designer nanoscale DNA assemblies programmed from the top down. *Science* (2016)
51. J. Ellis-Monaghan, A. McDowell, I. Moffatt, G. Pangborn, DNA origami and the complexity of Eulerian circuits with turning costs. *Nat. Comput.* **14**(3), 491–503 (2015)
52. N. Jonoska, M. Saito, *Boundary Components of Thickened Graphs, Revised Papers of 7th International Meeting on DNA Based Computers*, vol. 2340, pp. 70–81 (Springer LNCS, 2002)
53. W. Wang, S. Chen, B. An, et al., Complex wireframe DNA nanostructures from simple building blocks. *Nat. Commun.* **10**, 1067 (2019)
54. S.W. Bent, U. Manber, On nonintersecting Eulerian circuits. *Discrete Appl. Math.* **18**(1), 87–94 (1987)
55. N. Jonoska, N.C. Seeman, G. Wu, On existence of reporter strands in DNA-based graph structures. *Theoret. Comput. Sci.* **410**(15), 1448–1460 (2009)
56. M.N. Ellingham, J.A. Ellis-Monaghan, Edge-outer graph embedding and the complexity of the DNA reporter strand problem. *Theoret. Comput. Sci.* **785**, 117–127 (2019)
57. D. Archdeacon, Variations on a theme of kuratowski. *Discrete Math.* **302**, 22–31 (2005)
58. C. Thomassen, On the complexity of finding a minimum cycle cover of a graph. *Siam J. Comput.* **26**(3), 675–677 (1997)

59. P.W.K. Rothmund, Design of DNA origami, in *Proceedings of the 2005 IEEE/ACM International Conference on Computer-Aided Design*, pp. 471–478 (IEEE Computer Society, 2005)
60. J. Garrett, N. Jonoska, H. Kim, M. Saito, Algebraic systems motivated by DNA origami, in *Algebraic Informatics*, ed. by M. Ćirić, M. Droste, J.-É. Pin, pp. 164–176 (Springer International Publishing, 2019)
61. J. Garrett, N. Jonoska, H. Kim, M. Saito, DNA origami words, graphical structures and their rewriting systems. *Nat. Comput.*, pp. 1–15 (2021)
62. I. Dolinka, J. East, The idempotent-generated subsemigroup of the Kauffman monoid. *Glasgow Math. J.* **59**(3), 673–683 (2017)
63. W. Liu, *Design and Assembly of DNA Nano-Objects and 2D DNA Origami Arrays*. Ph.D. thesis (New York University, 2012)
64. N.C. Seeman, The design of single-stranded nucleic acid knots. *Mol. Eng.* **2**(3), 297–307 (1992)
65. J.L. Ramírez Alfonsín, Knots and links in spatial graphs: a survey. *Discrete Math.* **302**(1–3), 225–242 (2005)
66. N. Robertson, P. Seymour, R. Thomas, Sachs’ linkless embedding conjecture. *J. Combin. Theory Ser. B* **64**(2), 185–227 (1995)
67. J. Conway, C. Gordon, Knots and links in spatial graphs. *J. Graph Theor.* **7**, 445–453 (1983)
68. N. Goldberg, T. Mattman, R. Naimi, Many, many more intrinsically knotted graphs. *Algebr. Geom. Topol.* **14**, 1801–1823 (2014)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Origami Life



Cody Geary

Abstract Regardless of the scale it is built at, the geometric principles of origami make it an architecture both constrained by the act of folding and imbued with dynamics by folding. From space stations to molecules, origami solutions are inspiring a new type of design that considers its deployment as an integral part of the structure. In this essay, I will explain some surprising similarities between the ancient art of folding paper origami and the folding of RNA at the nanoscale.

The invention of folded paper is likely to have arrived very shortly after the invention of paper. The simple act of folding the sheet, something so seemingly obvious you might not think to give it a name, began to organically grow into the elegant art form we today call 'Origami'. Inspired by over a thousand years of tradition and innovation in paper folding, modern scientists and engineers have gained a new appreciation for origami design [1]—as folding solutions in paper seem to translate upward to the scale of buildings [2] and downward to the scale of molecules [3]. An origami begins by developing a pattern of creases in a sheet of paper; the folds allow the flat sheet of paper to be reconfigured into different geometric forms. The basic act of folding is one of the transformations: The mere placement of creases on a flat sheet allows for nearly any imaginable three-dimensional animal or plant or abstract form to be created—an act almost as magical as life springing out from a seed.

Folding and packing are the central aspects of paper origami, but are the rules of origami universal to all folded things? Much like origami, folding and packing are both extraordinarily important for the stabilization of structural RNA molecules, which play essential roles in regulation, chromosome maintenance and protein biosynthesis [4]. Are there lessons to be learned from origami that can help us to better navigate the problems of biopolymer folding in a tactile way?

Similar to how living animals develop from the confines of a single cell that divides, an origami model begins as creases within the bounds of a sheet of paper. Origami seems to be an ideal medium for mimicking the shapes and forms found

C. Geary (✉)

Interdisciplinary Nanoscience Center, Aarhus University, Aarhus, Denmark

e-mail: geary@inano.au.dk

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*, Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_12

207

in nature. Origami expert Jun Maekawa points out that in many traditional origami models, the colored top surface becomes the outside and the bottom of the sheet becomes the inside, and so as a consequence, the origami has the same topology as a blastula [5]. Modern origami techniques are now even able to mimic the stripes of zebras [6] and spots on beetles [7] by using both sides of the paper strategically. Nature perfected folding before humans invented origami, and examples can be observed all over nature; in the unfurling petals of flowers, the unfolding of insect wings, or the way that some plants dry out and coil up to spread their pollen or seeds. In all of these cases, the pattern of folds provides major evolutionary advantages to the organism. The ability to contract and expand (or deploy into a larger shape) is the mechanism of motion for all living things, even at a molecular level. From the way that our genes are densely packed into a fractal-like network of coiled coils, to how protein expression is dynamically turned on and off just when needed, deployable structures are part of the mechanism of life. The art of origami is indeed a beautiful representation of this concept.

Some of the earliest innovations in the art of paper folding originate from Japan. Traditionally folded from very high-quality paper, it was mostly used for ceremonial purposes, at weddings or as decorations on gifts [8]. It was not until around the early 1900s in Japan that books formalizing the art were published, and the name ‘Origami’ became associated with paper folding. The Japanese paper crane and variants, along with a handful of traditional models from other cultures, such as the paper balloon and pinwheel fold, represented the total knowledge of paper forms at the time. Then, in the 1950s, Akira Yoshizawa entered the scene and both revolutionized and popularized origami when he published ‘Atarashii Origami Geijutsu’ (New Origami Art), introducing many innovative new origami models and techniques for shaping origami, and perhaps even more importantly, standardized the diagrams used to depict the folding process [9]. The seeds were planted, and the ingenuity of the art form was accelerated from there. As the complexity of creasing patterns increased, the intricacy of the resulting forms began to gain origami international recognition as a new medium for sculpture. By the 1980s, there were thousands of published origami designs.

The formalization of axioms describing the geometry of origami [10] and contributions from innovators in origami design, such as Maekawa’s system for designing complex origami folds [5], inspired the creation of computer algorithms [11] that are able to automate much of the design of crease patterns for origami. The most notable of these algorithms, TreeMaker [12], written by Robert Lang in 1996, was the first to enable the design of origami with an arbitrary number of flaps of any length, which become the arms and legs of the origami (see Fig. 1a). TreeMaker led to a ‘Cambrian explosion’ of new complexity in origami folding among origami designers as they raced to generate astonishingly intricate models with ever more flaps with which to sculpt extra details like additional legs, claws, feathers and scales.

After many contributions from mathematicians, physicists and engineers [1], origami has become a highly multidisciplinary area of science. There are endless practical applications for folding, as origami can be found in many places where size and deployability are important, such as heart stents, airbags, or folding solar

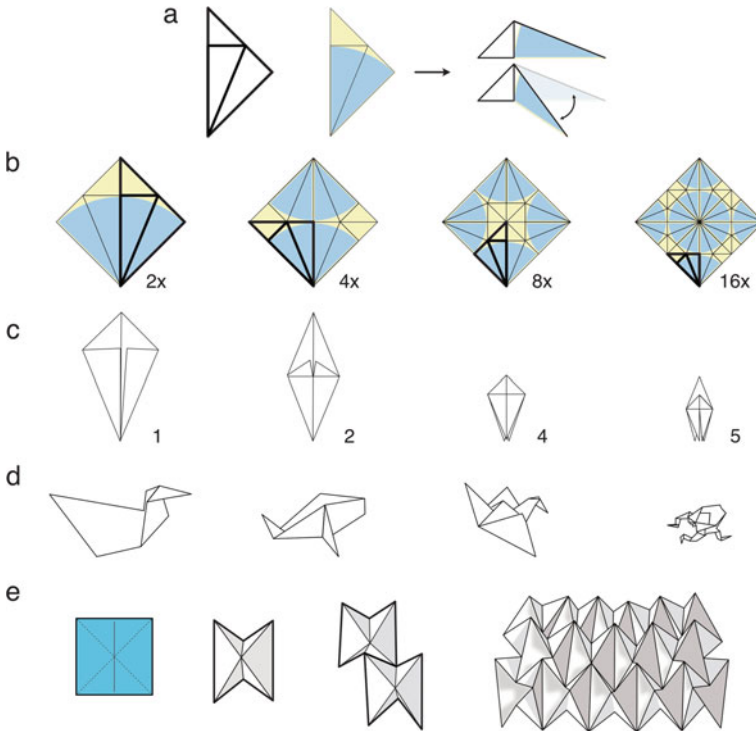


Fig. 1 Modularity in origami. **a** an origami ‘molecule’ fold based on the isosceles right triangle, one of the simplest folds that defines a flap. The lines show the position of folds, and the blue circle slice represents the area of the paper that contributes to the flap (right). **b** the four traditional bases of Japanese origami: the kite base, the fish base, the bird base and the frog base. Each of the bases tiles a different number of units of the isosceles triangle molecule shown in **a**, either 2, 4, 8 or 16. The blue areas show the part of the paper that will make a flap. **c** the folded form of each of the origami bases with 1, 2, 4 or 5 flaps, respectively. **d** an example origami folded from each base. Designs with greater number of flaps achieve much higher compaction in the final model. **e** an example of a modular folding pattern. The waterbomb base can be tiled such that each row is offset to the middle of its neighbors above and below. Figures **a–d** adapted from [13]

panels. Here, I will share a story about folding, how origami is a physical analogy for biological folding landscapes, and why multistability is both important and inherent in folded systems.

1 Origami Molecules

Origami is an art form that continually pushes the limits of our understanding of the mechanics of folding. As artists strive to pack more detail and information into the creases confined to the bounds of a sheet of paper, designs can become so intricate

that they take many hours to fold by hand. Origami models are often folded in two main stages: Initially an origami base is made, a large interconnected fold using the entire paper that creates a number of flaps (Fig. 1b, c). Next, the flaps that were created are sculpted into details such as legs or wings and other features (Fig. 1d). The base functions as a scaffold to organize flaps, and the flaps are used to create all the details that define the model.

Surprisingly, despite how diverse the variety of origami has become, most origami features the repeated usage of key folds, which are called ‘molecules’ in the Japanese tradition [13]. Molecule folds, of which there are many varieties, turn out to be especially useful because of their modularity within origami designs. One of the most recurrent molecule folds of origami, shown in Fig. 1a, is an isosceles triangle that is bisected by a crease at 22.5° , defining a single flap. Two copies of the molecule fold can be fit into a square (Fig. 1b, leftmost) to produce a kite base, the simplest of the four classic bases of origami: kite, fish, bird and frog. The incredible practicality of this fold is that, due to its shape, it can be scaled successively smaller to generate all of the traditional Japanese origami base folds (Fig. 1b), and this pattern can even be extended to larger designs [13].

Starting from the same molecule fold (Fig. 1a), each of the traditional origami bases produces different numbers of flaps (Fig. 1c), and the area of paper used for each flap is highlighted in blue on the crease pattern (Fig. 1b). The folded base can be further sculpted into familiar origami forms by adding additional detail folds to the flaps (Fig. 1d). The resulting origamis become smaller with respect to the paper as more flaps and details are added. To compensate for this fact, origami artists often use very large paper up to 50 cm square or larger, to produce the most complex models [14]. In the process of exploring how to subdivide the paper to maximize the number and size of flaps, artists discovered and documented many new rules for design: For example, common origami base folds, such as the waterbomb fold (Fig. 1e, left), can be tiled into larger and more complex folds (Fig. 1e, right). The concept of fold modularity in origami was pushed to new limits when several innovative origami designers at the time began devising new origami by combining molecule folds in new patterns [13]. The modularity of folding origami allows for common folds to be merged in surprising ways. John Montroll used this concept to elegantly create a bird base with a fifth extra flap [15], and Jun Maekawa created the elongated features of a crocodile [5] by merging two different traditional base folds.

Around the same time that the field of origami was having a breakthrough with respect to modular design principles, scientists around the world were just beginning to experiment with modular strands of DNA produced by new solid-phase synthesis technology—initiating the fields of DNA and RNA nanotechnology. The lab of Nadrian Seeman was investigating ways to program DNA strands to assemble into stable three-dimensional configurations, such as a cube [16]; and in this pursuit, they designed a new fold of DNA that would become the ‘origami molecule’ for nanotechnology—the double crossover junction [17]. Just as early progress in origami design was made by creating variations elaborated from one of a few different origami base

folds, the field of nanotechnology grew by building larger and more complex structures using the double crossover as a structural strut [18] and by folding genomes into repeating arrays of double crossovers that we call scaffolded DNA origami [3].

In parallel with developments in DNA nanotechnology, a theory of structural modularity for RNA molecules was being developed [19]. Westhof, Masquida and Jaeger hypothesized that RNA is modular and that new structures can be composed by rearranging and merging fragments of existing structures. This theory would lay the groundwork for developing a modular design method for constructing RNA structures [20]. RNA is the working component behind many biologically important molecular machines, responsible for protein synthesis and information processing in all living cells. Because the RNA machines of life have a modular construction that we can learn to use, there are many possibilities for the rational design of RNA by recombining structural modules.

Early work exploring the modularity of RNA tertiary interactions demonstrated that modular design could be applied to create programmable nanoscale assembly units of RNA [21]. Modular RNA structures self-assemble into defined shapes guided by the incorporation of different aptamers, junctions and programmable connectors embedded within their sequence. The full versatility of structural modularity for RNA initially proposed by Jaeger and colleagues [19, 20] was demonstrated by creating a building system of modular RNA assembly units that link up to form defined assemblies and repeating lattices (Fig. 2a) [22]. By merging different structural modules corresponding to bends, junctions and connectors, arbitrary and highly-detailed designs such as nano-hearts were produced (Fig. 2b). RNA architectures can be rationally mapped to a single-stranded path, encoded into a gene, and then expressed as self-folded shapes [22]. Although in RNA the folds are encoded into sequences of nucleotides, rather than formed by creases in origami paper, it is the same property of modularity of folds that makes it possible to fold paper into origami animals and compose RNA structures that self-assemble (or fold) into nanoscale shapes.

Drawing inspiration from the incredible structures of DNA origami [3] built using the double-crossover motif [17], I developed an RNA version of the motif that has a single-stranded topology [23]. In DNA origami, numerous short staple-strands help a long scaffold strand to fold into a compact form. Now, in RNA origami, programmable kissing loops (KLs) define one of the edges of the double-crossover motif and consequently enables designs based on this pattern to be produced cotranscriptionally (Fig. 2c). KL interactions span the space of a normal helix, but function to both connect and coaxially align distant helices. The single-stranded crossover fold, which I will call a KL crossover, is the equivalent of an origami ‘molecule’ fold for RNA and forms a compact modular unit that provides an underlying structure for numerous helices that can function as ‘flaps’ (Fig. 2c, d). Each unit of the KL crossover fold creates a geometric closure via the formation of a kissing loop [24], adding a growing element of stability to the origami, especially when cooperatively forming many KLs in alignment. As a scaffold material, RNA origami designs comprise a core of multiple KLs that are flanked by hairpins (Fig. 2c, e). These helix ‘flaps’ can be functionalized with a toolbox of programmable connectors, protein

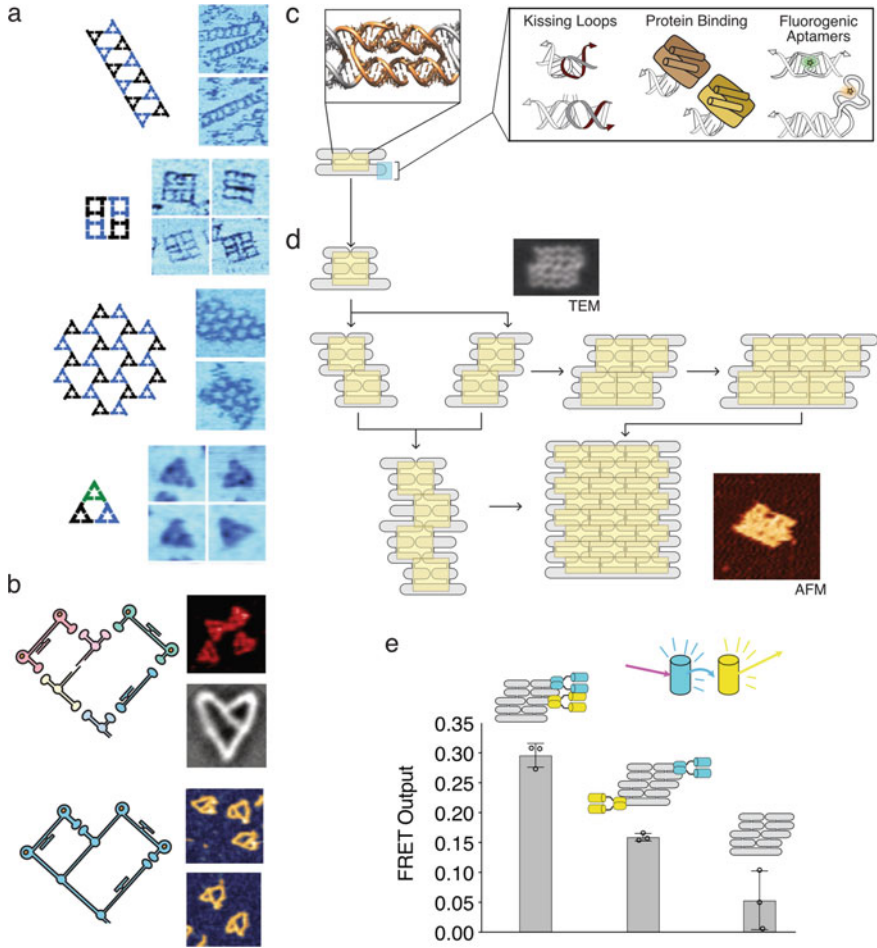


Fig. 2 Modularity of RNA origami. **a** programmable RNA lattices formed from square and triangle RNA units. **b** self-assembling RNA hearts from multiple units (top) or produced cotranscriptionally from a single strand (bottom). **c** the ‘origami molecule’ fold of RNA origami, a kissing loop interaction stabilizes the formation of a double-crossover section (inset and yellow box). On the edge of the structure are terminal hairpins, analogous to ‘flaps’ of origami, that can be functionalized with a variety of different RNA structural motifs such as programmable kissing loops, protein binding sites and fluorescent light-up aptamers such as Spinach and Mango. **d** larger and more complex RNA structures can be built by merging together copies of the fold in **a**. RNA origami structures range from 300 to 2300nts in scale. **e** functionalization of an RNA origami scaffold with binding sites for proteins fused to CFP and YFP. Adjacent binding sites bring the CFP and YFP proteins close enough to produce FRET. Figures a–b adapted from reference [22]. Figures c–e adapted from reference [25]

binding sites, or fluorescent light-up aptamers such as Spinach or Mango (Fig. 2c, inset).

RNA origami structures containing two, four, six or more copies of the KL crossover unit can be constructed because of the structural modularity, increasing design complexity through simple repetition [25] (Fig. 2d) similar to the molecule folds of paper origami (Fig. 1b, e). Analogous to the way that flaps are developed with detail folds in an origami model (Fig. 1d), the free arms of the RNA origami structure can be embedded with RNA motifs that ultimately decide the functionality of the structure. For demonstration, RNA origami was produced containing binding sites for two fluorescent protein complexes that can interact to produce FRET, a type of resonant energy transfer that can only occur when fluorophores are within a few nanometers of each other (Fig. 2e). Validating that origami scaffolds can colocalize and position proteins with precision, scaffolds created with adjacent binding sites produced a much brighter FRET response than scaffolds designed with far-apart binding sites or no binding sites at all [25] (Fig. 2e). In a paper origami, the folds of the base are all interconnected, while the folds of details are local and can be adjusted without compromising the base form. Similarly, RNA origami architecture uses a repeating modular pattern of interconnected local folds to form a stable core structure, out from which project helical arms can be sculpted with details from natural biological folds to guide protein binding, position aptamers, or other RNA active sites with precision.

2 Origami Design Algorithms

The culmination of years of exploration in the modularity of origami folds by numerous experts led to a mathematical formalization of paper origami and folding [10], systematic new approaches to origami design [5] and framework for creating arbitrarily complex folds [11]. These methods introduced a new school of thought to the origami world, designs became abstracted as tree-like stick figure diagrams, and the placement of origami flaps became distilled into a computational optimization problem. In particular, the TreeMaker [12] algorithm was very influential in the origami world because it enabled a new level of control over the number and relative size of every flap in a design. With this software, users can computationally generate creasing patterns for a wide variety of new origami.

Origami design using TreeMaker begins by studying the subject to represent in origami, in this example, a lizard (Fig. 3a). First, the lizard is abstracted as a weighted tree graph (Fig. 3b); each leaf on the tree will be folded into a separate flap, and the weights reflect the relative lengths of those body segments. Circles with radii weighted by the tree are fit, so their centers are within the bounds of the paper and then scaled so that their perimeters all touch but do not overlap (Fig. 3c)—each circle represents where a flap will be created. The distance between a center of a circle and that of another matches the weight of the path between the corresponding leaves on the edge-weighted tree. In this example, those weights happen to all be length one,

and so, for example, the distance between a front leg and a back leg is three (Fig. 3c). The blue ‘river’ area of the design that does not contain any circles functions like a flap that is connected at both ends and will be what the body of the design is folded from (Fig. 3c–e). Next, the algorithm fills in the polygons with molecule fold patterns (Fig. 3d). The ‘rabbit ear’ molecule is a general fold solution for a triangular space, and more complex molecule patterns exist that allow any polygon to be filled with creases [26]. During the folding compaction, the creases will function as hinges, and the polygons of paper between the creases are called ‘facets’. As the fold is collapsed, all of the facets move in concert (Fig. 3e), and the cross-section of the fully collapsed base fold has the same shape as the tree (Fig. 3b). Once the origami is folded, the addition of detail folds [13] and other well-established shaping techniques, such as wet folding [9], can be applied to further refine the origami so that it resembles a lizard (Fig. 3e).

While for paper origami, algorithms simplify the work to produce complicated models by automating the design of folds starting from a tree diagram; in RNA, algorithms are used instead to create and optimize a sequence that will fold into the desired tree structure. An RNA can be thought of as a kind of one-dimensional origami that is deployed by the act of transcription; the leaves and branches of the tree structure and its folding pathway are encoded into the sequence of the RNA strand. Be it creating a pattern of paper creases that define a shape [27] or a pattern of nucleotides that define a fold [28], design is difficult—and the time to compute a design increases rapidly with the complexity of the design. Once the design is completed, it can even take an origami master considerable time and patience to fold and collapse a complex crease pattern. In comparison, an RNA origami transcript that is designed well will fold all by itself! However, the key phrase is ‘designed well’, and because of the complexity of the inverse folding problem, design optimization by computer algorithms is absolutely essential.

Recently, I developed the RNA origami design software ROAD [25], which enables researchers to model, optimize and encode large and complex RNA folds into a single-stranded sequence (Fig. 2d), allowing RNA origami nanostructures that are kilobases in scale to be produced. ROAD, in a similar manner to TreeMaker, is an iterative optimization algorithm that attempts to design origami structures based on an inputted tree. However, rather than design crease patterns, ROAD designs a sequence of nucleotides that encodes folds in the RNA strand. The problem of inverse folding by sequence design is approached by guessing an initial sequence and then repeatedly revising that design to improve its score based on a number of heuristic tests. This approach relies on repeatedly evaluating guesses in an energy model that simulates RNA folding and benefits from fast algorithms such as ViennaRNA for reliably computing RNA folds [29]. However, pseudoknots, the interactions that form between distal single-stranded regions of a structure, are notoriously difficult to compute—spelling a potential problem for designs that require numerous KL connectors.

While many software already exist to design RNA, two features unique to ROAD make it ideal for producing RNA origami designs: First, it is able to rapidly assign origami KL sequences by systematically cross-checking all loops against each other,

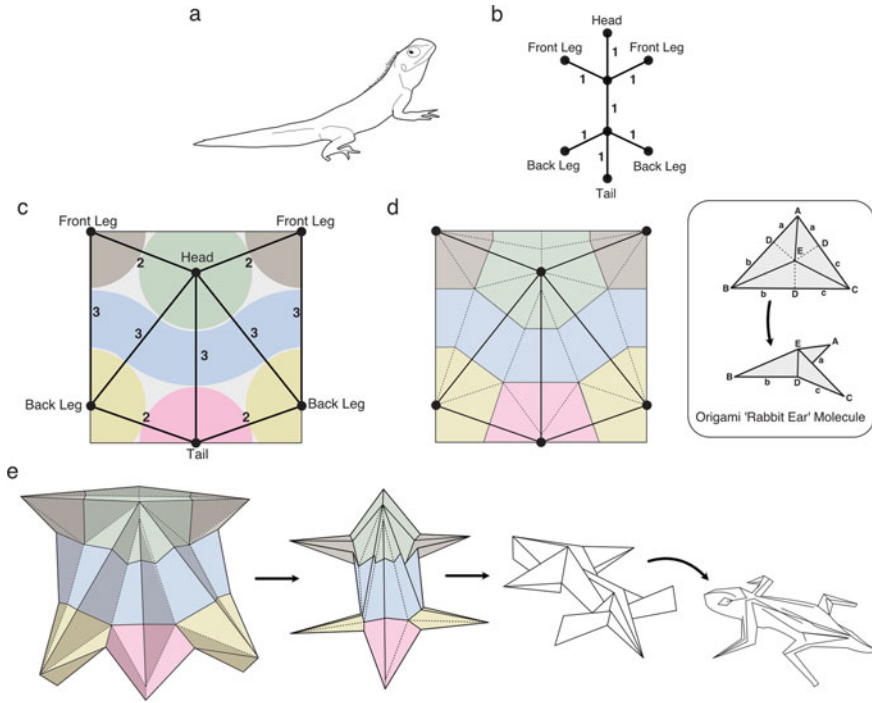


Fig. 3 Origami design with TreeMaker. **a** a lizard, as inspiration for design. **b** a tree diagram abstraction of the lizard, where each branch has a length of one. **c** to represent the tree in the bounds of the paper, each branch of the tree becomes abstracted as a circular area, and the body becomes the blue ‘river’ section, arranged to maximize the packing. Triangles on the graph connecting the nodes have lengths corresponding to the path length along the tree. **d** each polygon of the main crease pattern can be filled using an origami ‘molecule’ fold. Here, a general example for triangles is shown where creases bisecting each angle fold the triangle into three flaps. Each flap is colored according to the node it is assigned to in **c**. **e** sequence depicting the collapsing of the crease pattern into folded form with the same cross-section as the tree graph. With sculpting, the base fold can then be made to look like a lizard. Figure adapted from reference [12]

whereas other software that are able to design pseudoknots do not scale to such large designs. Second, it designs sequences with the constraints of cotranscriptional synthesis as a main consideration, avoiding specific sequences and patterns known to interfere with transcription or the experimental workflow. While early modular RNA designs were produced by hand-aligning structural modules to model RNA particles, for example, RNA squares created with exchangeable motifs for the corners [30] or tilings of small square and triangular assemblies (Fig. 2a) [22], the size and complexity of designs were ultimately very limited in scale compared to what can now be achieved with the computer-aided modeling and design offered by ROAD (Fig. 2d) [25].

The development of paper origami and DNA/RNA origami has both been greatly accelerated by computer-aided design algorithms. However, just as the crease patterns

generated by TreeMaker do not provide step-by-step folding instructions and do not guarantee that the compacted design will be an elegant one, RNA origami designs, even with computer-optimized sequences, still may not fold correctly if the order and staging of folding events are not planned well. Ambiguity of fold definitions, the topology of the strand/paper and the order of folding events are all factors determining the shape of the folding landscape for origami structures.

3 Origami Folding Pathways

Multistability occurs whenever a system has two or more stable equilibrium states or as a consequence of a folding pathway that gets stuck in non-equilibrium states. The presence of competing stable states is a common consequence of complex and dynamic systems, and origami is no exception. When folding a paper origami, certain combinations of mountain and valley folds produce bistability. A classic example of this is the waterbomb fold (Fig. 4a), which can transition between a triangular and square forms by inversion. The square form of the waterbomb is also commonly called the ‘Preliminary Fold’ [15] and is a starting fold for many origami models.

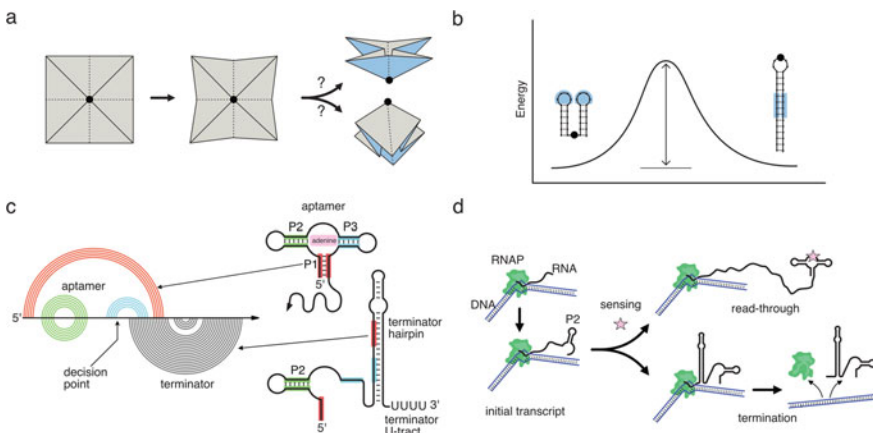


Fig. 4 Bistability in origami and RNA. **a** the waterbomb fold can collapse into two entirely different structures and is the most common example of bistability in origami. **b** a simple bistable RNA structure; it can alternate between either two short or one long hairpin. The energy barrier to transition between the structures can be as high as the unfolded state, indicated by arrow. Figure adapted from [31]. **c** the adenine riboswitch is a classic example of bistability. A color-coded arc representation of the competing secondary structures (left) shows how the terminator hairpin has bistability with stems P1 and P3. **d** schematic of adenosine sensing, depicting a window during transcription where the switch can activate based on ligand binding. If the aptamer is not stabilized in time, then the terminator stem forms and prevents the production of the rest of the RNA strand. Figures **c–d** adapted from reference [32]

The central vertex of the waterbomb fold is a ‘node’ that moves to actuate the inversion of the fold. As a node of the waterbomb is inverted, the fold becomes unstable at the transition point between the two states when the paper is fully flattened. This is because paper has non-zero thickness; creasing the paper sets a new preferred angle in the fibers of the paper along the fold lines, leading to the paper resisting being fully flattened. In the case of the waterbomb fold inversion, elastic deformation of the crease away from its preferred angle during the transition stores energy as the paper ‘pops’ from one fold to the other as you force the transition [33]. In other cases, such as for concentrically pleated folds, energy can be stored in the bending of non-triangular facets as the paper balances trying to be perfectly flat along the facets with the position and bend of the creases [34]. Bistability in origami is caused by competing forces between the creases wanting to bend and the facets of paper between the creases resisting bending.

In RNA, an analogous bistable fold to the waterbomb would be two short hairpins that can transition into a single longer hairpin (Fig. 4b). Just as bistable folds can be found all over origami, bistable sequences are naturally prevalent in RNA and furthermore are so simple to design that a bistable sequence can be produced for any pair of two structures [31]. The tendency for RNA sequences to have multiple folded states of similar energy can present a real challenge for producing well-folded RNA structures. And likewise just as for origami, multistability and also topology are both major factors in how RNA sequences fold. Multistability is a property of sequences that is often used in important ways by functional RNA molecules. Because RNA is synthesized directionally from 5’ to 3’, kinetic control of folding in RNA can be used to direct folding down a specific folding pathway. For example, the adenine riboswitch (Fig. 4c) is the classic example of bistability and can be controlled by tipping the stability balance with ligand binding at an early point during the transcription of the sequence [32]. Similarly, the more recently characterized ZTP riboswitch navigates its folding landscape in a complicated manner, but its behavior is ultimately determined by ligand binding during a narrow window of time early in the transcription [35]. In a linear folding pathway, the ability to freely transition between bistable alternatives occurs during the narrow window of time when the strand is actively folding.

In paper origami, the crease patterns will frequently have multistable elements with many bistable node points. In order for such a structure to be folded correctly, all of the nodes need to point the correct direction at the start of the collapse (Fig. 3e). Attempting an origami collapse with even a single misaligned node will lead to a conformationally trapped configuration that cannot flatten any further (Fig. 5a). This type of conformational trap is similar to the way that biopolymers can become kinetically trapped during their folding. Two basic properties in common between biopolymers and origami allow the problem of conformational trapping to arise: First, the paper or molecular strand cannot be stretched or compressed, only folded. Second, the paper or molecular strand cannot intersect itself in the folded form nor at any intermediate folding state [36].

To illustrate how conformational trapping can occur, consider the crease pattern for the paper crane base, one of the simplest variants of the waterbomb fold that

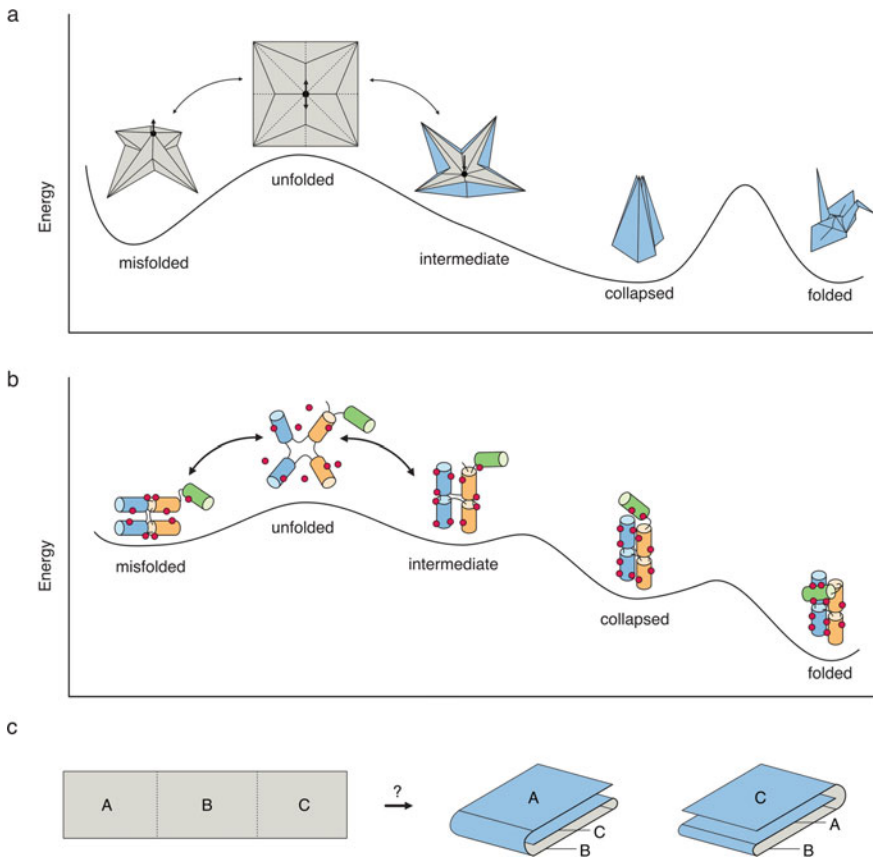


Fig. 5 Origami conformational traps. **a** a hypothetical energy landscape for a paper crane. Similar to the waterbomb fold, the bird base has a bistable landscape, although only one of the two folds can collapse fully and be made into a crane. Starting at the unfolded state, the central node can be displaced either up or down, leading to either the misfolded or collapsed state, respectively. The final folding of the model into a crane is represented by a large energy barrier. **b** a hypothetical energy landscape for the hairpin ribozyme junction, showing the magnesium-induced stabilization of the folded and collapsed states. Magnesium binding (red dots) overcomes electrostatic repulsion and allows the RNA to condense into tightly packed structure upon rearrangement. Alternative helix stacking order can produce partially stabilized structures that cannot fully compact. An interaction between the green and blue helices drives the final stabilization of the collapsed structure into the folded structure. **c** the order of folds can lead to bifurcation, for example, adjacent valley folds can lead to different stacking orders. Additionally, if either segment A or C is longer than B, then one of the two possible stacking orders becomes conformationally blocked. Figure c adapted from reference [36]

also has a trapped state (Fig. 5a). The energetic landscape of a waterbomb fold has been measured using mechanical models [33], and for the hypothetical crane fold I illustrate something similar but with one of the two states having a lower energy than the other (Fig. 5a, misfolded vs. collapsed). The highest energy point corresponds to the unfolded state of the paper, and this is because it takes energy to elastically deform the creases from their preferred angle. The central node point or ‘flap tip’ of the waterbomb fold pattern can be pushed either upward or downward, releasing energy as the structure begins to compact. If it starts to compact toward the misfolded state, it will eventually reach a point where the structure cannot compact any further (Fig. 5a, left). In order to reach the correct folded state from the misfolded state, the entire sheet must first go back through the unfolded state (Fig. 5a, right), a common obstacle in the compaction of complex crease patterns for paper origami [37]. Finally, to create the fully developed crane model from the collapsed crane base requires additional deformations, and is represented by a significant energy barrier (Fig. 5a, right).

Just like for paper origami, energy is required to unfold a misfolded RNA structure before refolding it into a correct state, and for RNA, the barrier to transition between alternative secondary structures is quite high due to how strong base pairs are (Fig. 4b). Thus, if an RNA falls into a kinetic trap while it is forming base pairs, it can take a long time to refold those misfolded regions depending on the height of the energy barrier between alternative structures. For this reason, it is important to consider not only the final folded structure, but also all of the possible intermediate structures, when evaluating if a particular RNA design is prone to be stuck in folding traps. Interestingly, it is possible at a theoretical level to intentionally design sequences with extremely long and winding low energy-barrier folding pathways [38]. This suggests that it might be possible to design RNAs that change shape over time or even perform complex computations by folding.

Folding via a sequential folding pathway, for example, as a riboswitch folds (Fig. 4d), can be advantageous for controlling the direction of bifurcated nodes. Traditional origami provides a set of precise folding steps to produce the paper crane (Fig. 5a) that entirely avoids any chance of misfolding. By contrast, crease patterns produced in computational origami can have hundreds of folds that would ideally need to collapse in a concerted manner; however, the origami folder is limited by their two hands. Computational patterns are not generated with any folding sequence and often contain large irreducible folds that are interlinked and need to be collapsed simultaneously, and so sequential folding may not even be possible for many patterns. It is an important point that collapsing folds can produce origami that could not otherwise be produced in a stepwise manner; this was first documented by Akira Yoshizawa in 1959 when he innovated his incredible Cicada origami model, which collapses in a single motion from a tessellation of eight copies of the bird base fold [9].

Even with all the proper creases in place, it can take a good measure of skill to collapse a computational origami pattern in this manner. Expert origami folders often approach this problem by pushing each flap tip vertex in the proper direction, bit by bit, over the entire pattern until it reaches a tipping point and is able

to compact [36]. Recent advances in the computer simulation of paper origami are able to estimate facet bending strain and model an idealized simultaneous collapse of all the creases in a design in a continuous motion [39]. The concerted collapse of a crease pattern echoes the annealing process that is often used to prepare DNA origami [3], whereby DNA strands are heated until they melt apart and then slowly cooled to let all interactions condense simultaneously. While DNA origami is typically folded from hundreds of short staple strands that work together to force a long scaffold strand to fold into a three-dimensional structure, there are also DNA and RNA origami structures that can collapse from a single long strand by annealing over a long temperature ramp [23, 40]. By contrast, the more biologically relevant cotranscriptional preparation of RNA origami is comparable to the stepwise folding instructions of traditional origami. Interestingly, the folding space for heat-annealed RNAs and cotranscriptionally folded RNAs does appear to be quite different [23], with annealing having the ability to produce some structures that kinetic folding was not able to, much like what was observed to be the case for paper origami folding by Yoshizawa [9].

In addition to multistability in base pairing, such as is utilized to channel the folding in riboswitches, RNA structures also frequently encounter multistability in the collapse of multi-helix junctions. This is because the problem of folding is not just specifying the helices that fold, but also how the helices should be oriented with respect to one another. The helix packing stage of RNA folding can take substantially longer than the initial folding [41], especially for larger structures with multiple junctions that need to rearrange. This type of structural rearrangement is depicted in Fig. 5b, showing the folding of the hairpin ribozyme [42]. In the hairpin ribozyme, the blue and orange helices can coaxially stack different ways, but only one conformation is further stabilized by an interaction with the green helix (Fig. 5b). Compared to multistability in base pairing (Fig. 4b), the energy barrier to move between alternate packing conformations is much lower (Fig. 5b), meaning that RNAs can explore this structural space more freely during folding. During the folding of RNA, magnesium counterions bridge and neutralize the negative charges between helix backbones, triggering a collapse into more compact structures [43]. Stabilization by binding metal ions has the effect of deepening the free energy curve for the fully folded structure over misfolded structures, in effect channeling folds to the correct structure given enough time [44].

Deepening the energy well for the desired fold can be used to rationally design RNA structures for which junctions fold rapidly and stably into the correct stacking conformation. The strategy of forming geometric closures through long-range interactions is widespread in biology [24] and is an effective way to stabilize multi-helix junctions. The hairpin ribozyme uses a long-range interaction to specify the antiparallel orientation of the junction [45], and likewise, other arrangements can be programmed based on the context of tertiary interactions. For example, embedding a loop–receptor interaction into two stems of a four-way junction can strongly favor a parallel-helix stacking orientation that aligns the loop and receptor [46]. In RNA origami structures, the same strategy is used, and both parallel and antiparallel

arrangements can be specified by choosing the proper length of the KL-crossover [23].

In paper origami, the order in which the folds are made can lead to different stacking orders of the paper layers (Fig. 5b). For a given crease pattern, determining a valid stacking order is computationally involved because of how interdependent the folds are [36]. Consider the simple placement of just two adjacent valley folds (Fig. 5c), if either segment A or C is longer than B, then one of the two possible stacking orders becomes conformationally blocked. The context dependence of folding makes it difficult to predict how facets in a crease pattern may interact, as what happens at the crease between segments B and C depends on how segment A was folded. In larger origami designs, these dependencies can be far more complex, often linking folds on opposite ends of a sheet [36].

Not surprisingly, context dependence is a major factor in RNA folding as well; just as the folding of paper can block later creases from folding, the folding of strands into pseudoknots during transcription can create folding barriers [47]. Locally, the energy of each base pair is dependent on its neighbors, and globally, the final compaction of helices into three-dimensional structure is driven by the stacking of helices at junctions and by the cooperativity of numerous weaker packing interactions [48]. The order in which the folds occur within an RNA origami can have a large influence on the outcome. As an RNA strand folds cotranscriptionally, the rate of folding into helices is many times faster than the rate of strand synthesis; long-range interactions such as KLs form comparatively slowly, giving helices and junctions time to stabilize before they lock into place. This hierarchy in rates makes it possible to design arbitrarily mazelike strand paths that arrange perfectly when KLs link up. Since each KL contributes roughly ~ 10 kcal/mol of binding energy in typical folding conditions, the total energetic contribution from many KLs can be substantial.

RNA origami is held together by numerous KLs working in concert, and the order in which the loops pair up in an RNA origami can lead to the formation of structural barriers. ROAD design software attempts to predict structural barriers by analyzing the position of KLs relative to the rest of the structure to see if any unpaired helices are nested within [25]. It is however up to the designer to refine their folding path based on the feedback. A recently developed program designed to optimize tree structures for cotranscriptional folding can help with this process by proposing structural variations that may have fewer barriers [49].

To illustrate the challenge of designing for sequential folding, consider a complex RNA tree (Fig. 6a) with branch lengths designed to fill the space of a rectangle (Fig. 6b). If we trace a path outlining the tree, it will make a representation of its secondary structure (Fig. 6c). Depending on where the 5' and 3' ends are placed on this circular path, different strand paths can be generated that represent the same fold—here, two examples are illustrated (Path1 and Path2, Fig. 6c). In the first example (Path1, Fig. 6d), the polymerase produces a long and winding structure that folds into a rectangular tile, condensing mostly at the end as the last KLs are put into place. At every intermediate position of the folding, there are no issues with the strand intersecting itself. In a second folding example (Path2, Fig. 6e), the location of the 5' end is moved such that several KL pairs are made very early in the fold.

When the KLs lock into place in this example, it creates several looped-out portions of single strand that still need to pair (Fig. 6e, orange strands). Since the RNA forms a helical turn every 11 base pairs, when the complementary portion of any of the longer orange sections is produced, it is going to encounter a structural barrier because the strands cannot intersect. As a result, some KLs will have to unpair before the red-colored portion can fold correctly (Fig. 6e, red strands). Cotranscriptional folding experiments comparing these two designs found that the Path1 design resulted in a considerably higher yield of correctly folded products compared to the Path2 design (Fig. 6de, right) [25]. While both Path1 and Path2 produced folded products, cotranscriptional folding via Path1 produced a much more homogeneous product at a higher yield. Furthermore, only Path1 produced TEM data of high enough quality to create an *ab initio* reconstruction of the RNA origami (Fig. 6d, right).

Misfolding appears to be a natural consequence of producing more complex folds. As more layers and more interactions between folds are added to a structure, the chance to create bifurcations during folding increases. Just as the wrong order of folds in a paper origami can lead to a conformation that blocks further folding (Fig. 5), the wrong order of synthesis through a single-stranded design can also lead to the formation of roadblocks to folding (Fig. 6e). The directional synthesis of RNA is one way to solve the problem of folding bifurcations, both with respect to the order of condensing helices, but also when alternative structures with equivalent or similar energies are present (Fig. 4b). Indeed, in the classic example of bistability, riboswitches take advantage of the slow and directional synthesis to control their own folding landscapes (Fig. 4d) [32]. Likewise, natural RNAs have evolved clever ways to cope with conformational multistability, as the hairpin ribozyme uses long-range interactions to lock the correct conformation of a junction into place [42]. Lastly, having a strand path that avoids getting tangled while it folds can result in higher yields and more homogeneous folded RNAs (Fig. 6d) [25].

4 Folded Origins

The word ‘Origami’ derives from two Japanese words: ‘ori’ meaning to fold and ‘kami’ meaning paper. Rather than describing the final folded product, the name origami refers to the active process by which it is produced. United by the idea of growth from folds, paper folding and biomolecular folding are both ultimately expressions of life. The two types of folding are governed by two limitations that are at the core of the concept of folding in general: The first is the property of non-intersection (injectivity) and the second is the property of non-stretching and non-compression of the material (isometry) [36]. Out of these two properties, almost all of the other features of folding emerge as a consequence.

Paper folding provides a tactile and intuitive way to understand the concept of biological folding landscapes, especially conceptually difficult ideas like folding traps. Paper is a versatile medium, it can be cut and glued, it has memory in the form of creases, and it can be dampened and reshaped [13]. It turns out that nucleic acids

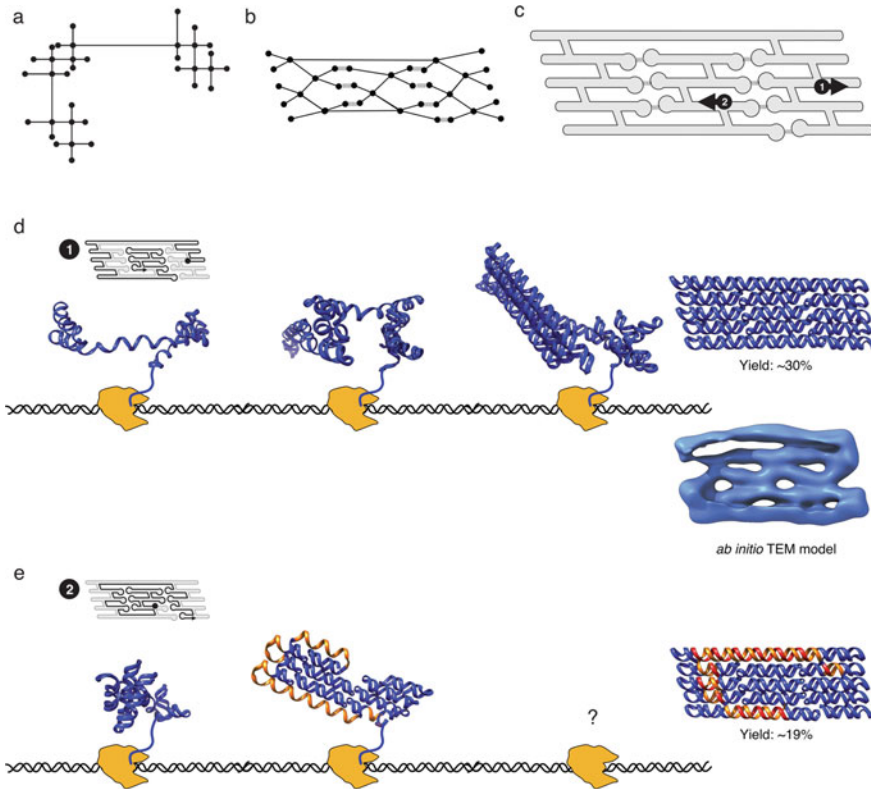


Fig. 6 RNA folding pathways. **a** a tree structure describing the secondary structure of an RNA origami. **b** the same tree, redrawn according to interactions between leaves on the tree. **c** the tree redrawn as a circular strand path, and aligning interacting helices. Two candidate positions (Path1 and Path2) place the 5'-end starting point at different points in the strand, arrow indicates strand direction. **d** Path1, beginning at the right edge, has a long circuitous path. An illustration of hypothetical folding intermediates is shown. A model of the folded structure is shown on the right, along with an *ab initio* structure produced by analyzing TEM micrographs. **e** Path2, starting in the middle of the design, forms a stable core structure with many long loops (highlighted in orange). As the structure folds, it can reach a midpoint where the continued folding is blocked (shown in red) because the growing chain needs to wrap around a trapped region (orange) to pair with it. Yields of Path2 were lower than for Path1, and the TEM data for Path2 were not high quality enough for an *ab initio* reconstruction. Figure adapted from reference [25]

have many of the same properties: although ribozymes and ligases do the cutting and gluing, base pairing and magnesium-induced stabilization provide the memory and glue—achieving effectively a similar result. In this way, DNA and RNA origamis become a natural extension of an art form that has thrived for over a thousand years.

In both paper origami and RNA origami, we encounter numerous types of bifurcations in folding pathways and solve them in similar ways. From multistable helices in riboswitches to the many conformers of helical junctions, the problem of folding

always requires consideration of its deployment when building structure. In the final vision of any architecture, there are often many unseen layers of logistics and organization that are critical to enable the construction. Perhaps, this is the most elegant aspect of it all, that there is a hidden beauty to folded architecture that is concealed by the dynamic process of its folding.

In spite of how well the analogy between origami and molecular folding works, there are also some interesting areas of contrast to mention: In origami, even ‘thick’ origami, the paper is usually much larger in dimensions compared to the thickness of the paper, so much so that the thickness of origami is easily idealized as zero-thickness in simulations. For RNA origami, a corresponding analogy might be to imagine the strand having nearly zero-thickness as well. However, when we zoom down to the scale of molecules, the thickness of the RNA helix and the length of the smallest helical features that can be designed are about the same, ~ 2 nm. As a result, RNA designs that have more structural features need to be made considerably larger than structures with fewer features—unlike paper which can be folded both more densely and smaller to achieve more features. Also, and particularly a contrast, when a biopolymer strand condenses into a misfolded state, it takes much longer to unfold than it does to misfold in the first place—with paper origami, it is quite the opposite and much more difficult to fold than it is to unfold!

The art of origami is an exploration of possibilities reached by folding and so is a reflection of life that is created through the act of folding. The RNA world hypothesis proposes that self-replicating biopolymers could have played a key role in the beginning of evolution [50]. In theory, life may have originated from the folding of a lifeless strand. Very recently, the Unrau lab made amazing progress toward producing a fabled ‘replicase’, a sequence of RNA that can copy itself [51]. The new ribozyme has a sense of ‘self’, using a bistable clamping mechanism that recognizes its own promoter sequence, demonstrating again how bistability is a fundamentally important property of living polymers. RNAs in nature navigate folding pathways to activate different functions in a context-dependent way. Today, we are only just beginning to unlock the secrets behind designing and folding RNA. An exciting new era for molecular structure that can deploy cotranscriptionally awaits!

Acknowledgements I acknowledge support from Novo Nordisk Foundation 0060694. I thank the reviewers Erik Benson, Shinnosuke Seki and Chris Thachuk for many useful suggestions to improve the manuscript. I also thank Robert J. Lang for helpful discussions about origami folding and energetics.

References

1. M. Meloni, et al., Engineering origami: a comprehensive review of recent applications, design methods, and tools. *Adv. Sci.*, 2000636 (2021)
2. R. Miranda, E. Babilio, N. Singh, F. Santos, F. Fraternali, Mechanics of smart origami sunscreens with energy harvesting ability. *Mech. Res. Commun.* **105**, 103503 (2020)

3. P.W.K. Rothmund, Folding DNA to create nanoscale shapes and patterns. *Nat.* **440**(7082), 297–302 (2006)
4. S.A. Strobel, J.A. Doudna, RNA seeing double: Close-packing of helices in RNA tertiary structure. *Trends Biochem. Sci.* **22**(7), 262–266 (1997)
5. J. Maekawa, Evolution of origami organisms. *Symmetry Cult. Sci.* **5**(2), 167–177 (1994)
6. J. Montroll, *Origami Inside-Out* (Dover Publications, 1993)
7. R.J. Lang, P. Doyle, *Origami Insects and Their Kin: Step-by-step instructions in over 1500 diagrams* (Dover Publications, 1995)
8. K. Hatori, in *History of origami in the east and the west before interfusion*. in *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education* (2016), pp. 3–11
9. A. Yoshizawa, K. Yoshizawa, T. Kikugawa, R.J. Lang, H. Ichiyama, K. Hamada, Akira Yoshizawa, Japan's greatest origami master : featuring over 60 models and 1000 diagrams by the master (2016)
10. T. Kawasaki, in *On the relation between mountain-creases and valley-creases of a flat origami*. Proceedings of the 1st International Meeting on Origami Science and Technology (1989), pp. 229–237
11. R.J. Lang, Mathematical algorithms for origami design. *Symmetry Cult. Sci.* **5**(2), 115–152 (1994)
12. R.J. Lang, in *Computational algorithm for origami design*. Proceedings of the Annual Symposium on Computational Geometry (1996), pp. 98–105
13. R.J. Lang, *Origami Design Secrets: Mathematical Methods for an Ancient Art* (2003)
14. R.J. Lang, Origami: complexity increasing. *Eng. Sci.* **52**(2), 16–23 (1989)
15. J. Montroll, *Animal rigami for the Enthusiast: Step-by-Step Instructions in Over 900 Diagrams* (Dover Publications, 1985)
16. J. Chen, N.C. Seeman, Synthesis from DNA of a molecule with the connectivity of a cube. *Nature* **350**(6319), 631–633 (1991)
17. S. Zhang, T.J. Fu, N.C. Seeman, Symmetric immobile DNA branched junctions. *Biochem.* **32**(32), 8062–8067 (1993)
18. W.M. Shih, J.D. Quispe, G.F. Joyce, A 1.7-kilobase single-stranded DNA that folds into a nanoscale octahedron. *Nat.* **427**(6975) (2004)
19. E. Westhof, B. Masquida, L. Jaeger, RNA tectonics: towards RNA design. *Fold. Des.* **1**(4), R78–R88 (1996)
20. L. Jaeger, A. Chworos, The architectonics of programmable RNA and DNA nanostructures. *Current Opinion Struct. Biol.* **16**(4), 531–543 (2006)
21. L. Jaeger, TectoRNA: modular assembly units for the construction of RNA nano-objects. *Nucleic Acids Res.* **29**(2), 455–463 (2001)
22. C. Geary, A. Chworos, E. Verzemnieks, N.R. Voss, L. Jaeger, Composing RNA nanostructures from a syntax of RNA structural modules. *Nano Lett.* **17**(11), 7095–7101 (2017)
23. C. Geary, P.W.K. Rothmund, E.S. Andersen, A single-stranded architecture for cotranscriptional folding of RNA nanostructures. *Science* **345**(6198), 799–804 (2014)
24. J.P. Bida, R. Das, Squaring theory with practice in RNA design. *Curr. Opin. Struct. Biol.* **22**(4), 457–466 (2012)
25. C. Geary, G. Grossi, E.K.S. McRae, P.W.K. Rothmund, E.S. Andersen, RNA origami design tools enable cotranscriptional folding of kilobase-sized nanoscaffolds. *Nat. Chem.* **13**(6), 549–558 (2021)
26. R.J. Lang, E.D. Demaine, Facet ordering and crease assignment in uniaxial bases. *Origami 4*, 201–218 (2011)
27. E.D. Demaine, S.P. Fekete, R.J. Lang, in *Circle packing for origami design is hard*. *Origami 5: Fifth International Meeting of Origami Science, Mathematics, and Education* (2016), pp. 609–626
28. É. Bonnet, P. Rzazewski, F. Sikora, Designing RNA secondary structures is hard. *J. Comput. Biol.* **27**(3), 302–316 (2020)
29. R. Lorenz, et al., ViennaRNA Package 2.0. *Algorithms Mol. Biol.* **6**(1), 26 (2011)

30. I. Severcan, C. Geary, E. Verzemnieks, A. Chworos, L. Jaeger, Square-shaped RNA particles from different RNA folds, *9*(3), 1270–1277 (2009)
31. C. Flamm, I.L. Hofacker, S. Maurer-Stroh, P.F. Stadler, M. Zehl, Design of multistable RNA molecules. *RNA* *7*(2), 254–265 (2001)
32. K.L. Frieda, S.M. Block, Direct observation of cotranscriptional folding in an adenine riboswitch. *Science* *338*(6105), 397–400 (2012)
33. B.H. Hanna, J.M. Lund, R.J. Lang, S.P. Magleby, L.L. Howell, Waterbomb base: a symmetric single-vertex bistable origami mechanism. *Smart Mater. Struct.* *23*(9) (2014)
34. E.D. Demaine, M.L. Demaine, V. Hart, G.N. Price, T. Tachi, (Non)existence of pleated folds: how paper folds between creases. *Graphs. Comb.* *27*(3), 377–397 (2011)
35. E.J. Strobel, L. Cheng, K.E. Berman, P.D. Carlson, J.B. Lucks, A ligand-gated strand displacement mechanism for ZTP riboswitch transcription control. *Nat. Chem. Biol.* *15*(11), 1067–1076 (2019)
36. R.J. Lang, *Twists, Tilings, and Tessellations: Mathematical Methods for Geometric Origami* (CRC Press, 2018)
37. T. Tachi, Introduction to structural origami. *J. Int. Assoc. Shell Spat. Struct.* *60*(1), 7–18 (2019)
38. A. Condon, B. Kirkpatrick, J. Mañuch, Design of nucleic acid strands with long low-barrier folding pathways. *Nat. Comput.* *16*(2), 261–284 (2017)
39. A. Ghassaei, E.D. Demaine, N. Gershenfeld, Fast, interactive origami simulation using GPU computation. *Proc. 7th Int. Meet. Origami Sci. Math. Educ. (OSME 2018)* *4*(2016), 1151–1166 (2018)
40. D. Han, et al., Single-stranded DNA and RNA origami. *Science* *358*(6369), eaao2648 (2017)
41. S.A. Woodson, Compact intermediates in RNA folding. *Ann. Rev. Biophys.* *39*(1), 61–77 (2010)
42. E. Tan, T.J. Wilson, M.K. Nahas, R.M. Clegg, D.M.J. Lilley, T. Ha, A four-way junction accelerates hairpin ribozyme folding via a discrete intermediate (2003)
43. S.A. Woodson, Metal ions and RNA folding: a highly charged topic with a dynamic future. *Current Opin. Chem. Biol.* *9*(2), 104–109 (2005)
44. R. Behrouzi, J.H. Roh, D. Kilburn, R.M. Briber, S.A. Woodson, Cooperative tertiary interaction network guides RNA folding. *Cell* *149*(2), 348–357 (2012)
45. D.M.J. Lilley, Structures of helical junctions in nucleic acids. *Quart. Rev. Biophys.* *33*(2), 109–159 (2000)
46. C. Geary, A. Chworos, L. Jaeger, Promoting RNA helical stacking via A-minor junctions. *Nucleic Acids Res.* *39*(3), 1066–1080 (2011)
47. A. Xayaphoummine, V. Vlasnoff, S. Harlepp, H. Isambert, Encoding folding paths of RNA switches. *Nucleic Acids Res.* *35*(2), 614–622 (2007)
48. J.A. Cruz, E. Westhof, The dynamic landscapes of RNA architecture. *Cell* *136*(4), 604–609 (2009)
49. A. Mohammed, P. Orponen, S. Pai, Algorithmic design of cotranscriptionally folding 2D RNA origami structures, pp. 159–172 (2018)
50. D.K. Treiber, M.S. Rook, P.P. Zarrinkar, J.R. Williamson, Kinetic intermediates trapped by native interactions in RNA folding. *Science* *279*(53580), 1943–1946 (1998)
51. R. Cojocaru, P.J. Unrau, Processive RNA polymerization and promoter recognition in an RNA world. *Science* *371*(6535), 1225–1232 (2021)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Ok: A Kinetic Model for Locally Reconfigurable Molecular Systems



Pierre Marcus, Nicolas Schabanel, and Shinnosuke Seki

Abstract Oritatami is a formal model of RNA co-transcriptional folding, in which an RNA sequence (transcript) folds upon itself while being synthesized (transcribed) out of its DNA template. This model is simple enough for further extension and also strong enough to study computational aspects of this phenomenon. Some of the structural motifs designed for Turing universal computations in oritatami have been demonstrated approximately in-vitro recently. This model has yet to take a significant aspect of co-transcriptional folding into full account, that is, reconfiguration of molecules. Here we propose a kinetic extension of this model called the oritatami kinetic (*Ok*) model, similar to what kinetic tile assembly model (*kTAM*) is to abstract tile assembly model (*aTAM*). In this extension, local rerouting of the transcript inside a randomly chosen area of parameterized radius competes with the transcription and the folding of the nascent beads (beads are abstract monomers which are the transcription units in oritatami). We compare this extension to a simulation of oritatami in the nubot model, another reconfiguration-based molecular folding model. We show that this new extension matches better a reconfiguration model and is also faster to simulate than passing through a nubot simulation.

1 Introduction

Transcription is a phenomenon in which a system encoded on a DNA sequence is copied sequentially out of ribonucleic acids by an RNA polymerase into an RNA transcript. The particularity of this process is that the transcript folds upon itself into an intricate structure while being synthesized, that is *co-transcriptionally*.

P. Marcus (✉)

École Normale Supérieure de Lyon (LIP, MC2), Lyon, France

e-mail: pierre.marcus@ens-lyon.fr

N. Schabanel (✉)

CNRS, École Normale Supérieure de Lyon (LIP, MC2), Lyon, France

e-mail: nicolas.schabanel@ens-lyon.fr

S. Seki (✉)

University of Electro-Communications, Tokyo, Japan

e-mail: s.seki@uec.ac.jp

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*, Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_13

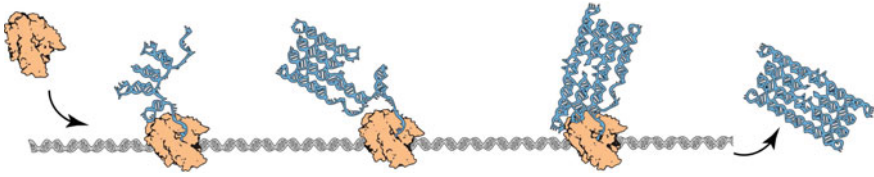


Fig. 1 Transcription and RNA origami [6]: An RNA polymerase (colored in orange) scans a DNA template (gray) and maps its sequence, nucleotide by nucleotide (A, T, C or G), into an RNA sequence (the *transcript*) according to the loss-less function $A \rightarrow U$, $C \rightarrow G$, $G \rightarrow C$, and $T \rightarrow A$. The transcript folds upon itself with high probability into a precise structure that can be programmed from the sequence of the DNA template

This phenomenon, called *co-transcriptional folding*, has proven to be programmable in-vitro. Indeed, in [6] Geary, Rothmund and Andersen demonstrated how to encode a rectangular tile-like structure in a transcript (actually, in its corresponding DNA template) so that following its folding pathway, the transcript folds co-transcriptionally into the target structure (Fig. 1). The design of such an *RNA origami* architecture has been highly automated by their software RNA Origami Automated Design (ROAD) [2]. ROAD “extends the scale and functional diversity of RNA scaffolds” so that they might be large and functional enough even to accommodate simple enough computation.

Besides serving as a scaffold for computation, co-transcriptional folding itself is capable of computing by encoding several folding pathways into a single transcript and letting an appropriate one be “called” depending on the environment [8, 9, 12]. The *oritatami* model was introduced in [3] to explore theoretically the computation capabilities allowed by co-transcriptional folding. It was first demonstrated to be capable of counting in binary [3, 7] and then of simulating arbitrary cyclic tag system [5]. As such, the oritatami model is efficiently Turing universal: it can simulate arbitrary Turing machines with a quadratic-time slow down only. Precisely, a cyclic tag system (CTS) is a binary word (over $\{0, 1\}$) rewriting system that consists of an initial tape word w^0 and a finite cyclic list of productions (binary words) and yields a sequence of nonempty tape words w^0, w^1, w^2, \dots ; at step $i \geq 1$, it rewrites the tape word w^{i-1} into w^i by (1) appending the current production at the end of w^{i-1} if and only if its leftmost letter w_0^{i-1} is 1 (and appending nothing otherwise), and then (2) deleting this first letter, and (3) rotating the cyclic list.

The oritatami CTS simulator [5] encodes the cyclic list of a given CTS in each period of its cyclic (periodic) transcript. The encoded list folds into a compact shape (called *switchbacks*) by default, unless a production must be appended at the end of the current tape word, in which case the current production inside that list folds in a self-supported expanded shape (called *glider*), extending the current tape word accordingly.

Multiple configurations in co-transcriptional molecular folding computing. As computation is achieved in tile assembly systems by gluing tiles together in different configurations in response to its surrounding, computation is achieved in co-

transcriptional folding model by having the transcript be folded in different shapes in response to its environment, i.e., its context. Some of the RNA structures created by ROAD [2] resemble structural motifs used for computing in oritatami, such as glider and switchbacks. That been said, the ability to encode two addressable shapes into one single sequence has not been demonstrated experimentally yet. Such compatibility between two foldable shapes, however, might be dispensable. Indeed, the Turing universal transcript in oritatami has been simplified significantly in [10]. In this later work, the periodic transcript folds co-transcriptionally into a space-time diagram of a 1D cellular automaton (CA). Each period of the transcript folds into a macrocell, an upscaled version of the corresponding simulated CA cell. The macrocell is divided into three parts: (1) the first detects the absence of a neighboring cell and build the missing boundary in that case, (2) the second builds the inner shell of the macrocell, and (3) the third part reads the input bits encoded on its NW and SW borders, and writes the output bits on its SE and NE borders. Only the first part requires a compatibility between two non-trivial patterns (actually, between glider and switchback). The inner shell is indeed hardcoded in the second part and uses only gliders and 60° - and 120° -turns. The third part consists of, first, reading gliders that get locally flatten when facing beads of specific types and, second, of a flat line encoding a transition table. The reading gliders get flat when passing by the parts of a neighboring macrocell border encoding a 1, which shifts forward the upcoming transition by an appropriate amount so as to expose only the output entry corresponding to the input. An interesting feature of this new I/O interface is its tolerance to misalignment of macrocells, a typical desired outcome in presence of molecular reconfigurations, even though the macrocells never get misaligned in the deterministic oritatami system.

The need for a co-transcriptional reconfiguration model. As opposed to tile assembly model, experimental evidence of computing using molecular co-transcriptional folding has not yet been proven, even though it has been shown theoretically possible thanks to the oritatami model. However, one main obstacle is the lack of a model which would take into account the probabilistic nature of experimental settings. This was solved for the tile assemblies by introducing the kinetic tile assembly model (*kTAM*) which provided useful hints, such as proofreading tiles [13], which allowed in turn to conduct successful experimental implementation of computing nanostructures [1, 14]. Co-transcriptional folding is significantly different from tile assembly as it is not composed of independent entities but requires that all the beads or monomer composing the resulting structure to be connected by a path. The *nubots* model introduced in [14] includes reconfigurations and allows to build structure with this kind of constraint, and even much more complex ones. Even if there is a way to simulate oritatami systems with nubots, this simulation is indirect and passes through unnatural intermediate states that should not be considered in a kinetic model (Sect. 2). This motivates the introduction of a new model, called *Oritatami kinetic (Ok)* model, which extends the oritatami model to include thermal reconfigurations of the folding structure. We hope this model to be powerful yet simple enough to design co-transcriptional folding scheme that will be robust enough to be imple-

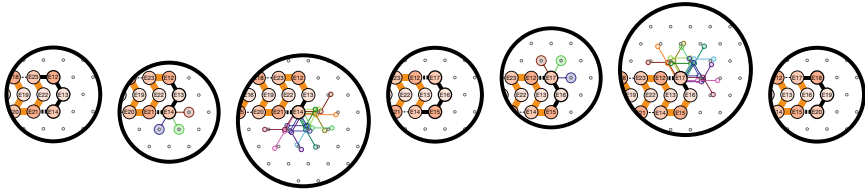


Fig. 2 Oritatami model: From left to right, the growth from bead E12 to bead E18 of a self-supported glider with delay $\delta = 3$, transcript $p = E12 \dots E23$ and rule $\{E12 \heartsuit E17, E14 \heartsuit E21, E18 \heartsuit E23, E20 \heartsuit E15\}$. At each step, the set of nascent paths and maximizing the number of bonds is shown. The nascent beads are highlighted in bold black. The nascent paths are drawn in bold black until the last bond made and ends in colors when their tail is free to move (i.e., is not bounded by any bond)

mented in-vitro. Furthermore one might hope that it may also open new way of design co-transcriptionally folding structures taking advantage of these new features (Fig. 2).

2 Molecular Reconfiguration: Oritatami and Nubots

Let us first compare the oritatami and nubots models.

Oritatami model. An oritatami system consists of a “molecule” (the *transcript*) consisting in a sequence of “beads” (monomers) that attract each other according to a given binary relation \heartsuit called the *rule*. The molecule grows in the triangular lattice, by one bead per step. At each step, the δ most recently produced *nascent* beads are free to move around to look for the position that maximizes the number of bonds they can make with each other or with beads placed already (hence the folding is co-transcriptional). Once that optimal position is found, the oldest nascent bead will adopt this position *forever*. Then, a new nascent bead is produced (according to the transcript sequence) and the process continues by optimizing the position of the new δ nascent beads. The transcript, i.e., the sequence of beads, is assumed to be finite or periodic. The parameter δ is called the *delay*. The folding starts from an initial configuration called the *seed*. A time- and space-efficient and easy-to-use oritatami model simulator is freely available at [11].

Nubots model. Nubots is a general purpose model capturing many (if not all) aspects of 2D molecular reconfigurability. Nubots are grown in the triangular lattice as well. They are composed of monomers that interact with their lattice neighbors in a non-deterministic manner. At each time step, a monomer can change its internal state, create a new neighbor monomer, or conversely disappear, change the nature of its bond with one of its neighbors (none, rigid or flexible), or move around one of its bonded neighbors taking along a part of its bond-connected component. Each of these

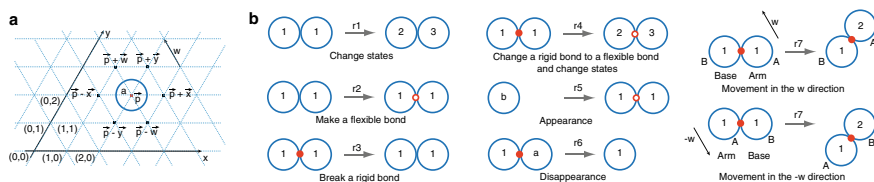


Fig. 3 Nubots model: (Figure extracted from [14]) **a** A monomer in state a at position \vec{p} in the triangular grid coordinate system $(\vec{x}, \vec{y}, \vec{w})$. **b** Examples of monomer interaction rules, written formally as follows: $r_1 = (1, 1, \text{null}, \vec{x}) \rightarrow (2, 3, \text{null}, \vec{x})$, $r_2 = (1, 1, \text{null}, \vec{x}) \rightarrow (1, 1, \text{flexible}, \vec{x})$, $r_3 = (1, 1, \text{rigid}, \vec{x}) \rightarrow (1, 1, \text{null}, \vec{x})$, $r_4 = (1, 1, \text{rigid}, \vec{x}) \rightarrow (2, 3, \text{flexible}, \vec{x})$, $r_5 = (b, \text{empty}, \text{null}, \vec{x}) \rightarrow (1, 1, \text{flexible}, \vec{x})$, $r_6 = (1, a, \text{rigid}, \vec{x}) \rightarrow (1, \text{empty}, \text{null}, \vec{x})$, and $r_7 = (1, 1, \text{rigid}, \vec{x}) \rightarrow (1, 2, \text{rigid}, \vec{y})$. For rule r_7 , the two potential symmetric movements are shown corresponding to two choices for arm and base, one of which is non-deterministically chosen

possibilities is described in a list of possible actions between any pair of neighboring monomers according to their respective internal states (Fig. 3). As for the oritatami model, the process starts from an initial configuration called the seed. This model takes advantage of both local and parallel reconfigurability to build large structure in a logarithmic number of parallel updates only.

Oritatami and nubots. One approach to introduce reconfigurability in the oritatami model is to implement it in the nubot model.

Theorem 1 *Any oritatami system can be implemented as a nubot.*

Proof Consider an oritatami system O with seed σ , periodic transcript p , rule \heartsuit and delay δ . The simulating nubot will consist of two kind of monomers: placed monomers and nascent monomers. Placed monomers correspond to the beads that have already been placed at their final location; their internal state will be the bead type of the corresponding bead according to the periodic transcript p . The nubots will grow and retract a chain of monomers linked by rigid bonds corresponding to the folded oritatami molecule. Nascent monomers correspond to the δ nascent beads of the simulated oritatami system; their internal state will encode not only the bead type of their corresponding bead but also some finite amount of information allowing to explore one by one all the possible paths, so as to compute the paths that maximize the number of bonds that the nascent beads can make in the simulated oritatami configuration. Note that the simulating nubot will conduct the path exploration by a simple depth first search where each nascent monomer spawn its child nascent monomer in every possible direction in a recursive manner. To conduct this exploration, each nascent monomer only needs to remember the currently explored direction, the current best number of bonds made by its descendants in some already explored directions, and its index in the transcript (to specify its bead type). As the total number of feasible bonds may not exceed $4\delta + 1$ (4 for the intermediate nascent monomer and 5 for the tailing one), the required number of internal states is $O(\log \delta + \log |p|)$, that is constant. Each parent nubot keeps the best number of

bonds among all of its children and adds its own number of bonds with the current environment, and then sends it to its parent before disappearing, until the process reaches back the path root nubot. This root will then be able to extend the oritatami path in the direction of the best move and will then restart the exploration from there.

This simulation of an oritatami system by a nubot is however unsatisfying as it requires the regular disappearance of monomers and thus introduces unnatural, and thus undesirable, intermediate states in the simulation. Other nubots simulation schemes exist at scale 1 that do not require the disappearance rule, but require instead to disconnect the nascent beads to rotate them around the environment, or to make room by pushing the environment around, but both of these intermediate states are equally undesirable.

Claim Nubots cannot mimic the nascent beads moves at scale 1 without the disappearance rule nor disconnecting the nascent beads.

Indeed, consider a seed configuration consisting of an Y-shaped tunnel with two arms of length δ where the oritatami system starts at the intersection. There are exactly two bead types, A and B , and the only attraction rule is $A \heartsuit B$. Its transcript is A^δ (A repeated δ times). The wall of the tunnels are made of A s but the bead placed at the end of the tunnel can either be A or B . In order to simulate faithfully the oritatami system, the nubot must reach both ends of the tunnel (otherwise one could exchange the two beads at the end of tunnel and invalidate the simulation). As no part of the grown nascent arm can be moved in any direction without being disconnected there are no other possibility than erasing the nubots grown when the wrong tunnel is explored first, which can be guaranteed by placing B in the second tunnel to be explored by the nubots.

3 The Ok model

The previous section showed that nubots are not well suited to model a dedicated kinetic co-transcriptional folding model because it would require passing through many unrealistic and time-costly intermediate states. Furthermore, in order to get closer to nature, we want, as in kTAM, our kinetic oritatami model to randomly reconfigure parts that have been folded already. As the connectivity of the path (made of covalent bonds) must be maintained, this would require lots of computation steps as well as lots of memory if modeled inside the nubots world.

3.1 Reconfiguration Events

In the *oritatami kinetic (Ok)* model, the dynamics changes. Instead of optimizing the nascent beads positions after each extension of the transcript by one bead, there are three kinds of reconfiguration events that are competing with each other:

Growth: A new bead is added at the end of the growing molecule. Its bead type is given by the transcript. It is placed at a uniformly and randomly chosen unoccupied location next to the current end of the path. If none of these locations is unoccupied, the growth fails and will be retried the next time the growth event is triggered.

Nascent beads reconfiguration: The path followed by the δ nascent beads (the δ most recently produced) is rerouted non-deterministically without overlaps, according to some probabilistic distribution to be discussed next.

Internal reconfiguration: An hexagon H of radius ρ was picked at random and all the subpaths inside this hexagon H are rerouted non-deterministically while keeping their extremities on the borders of H unchanged (see Fig. 4).

Figure 4 gives an example of a reconfiguration. Note that none of the reconfiguration events involves directly any bonding scheme optimisation process, as opposed to regular oritatami model. As for kTAM, this optimization will be induced by the rates at which these various reconfigurations are applied, as will be detailed in the following section.

3.2 Reconfiguration Distributions and Events Rates

Each of the reconfiguration events Growth, Nascent beads, and Internal reconfigurations will be triggered according to exponential random variable of respective rates r_G , r_N and r_I . The growth rate r_G only depends on the transcription speed of the polymerase (which may depends in turn on concentrations, temperature,...). The nascent beads and internal reconfiguration rates, r_N and r_I , do however depend on the local configuration where they are applied: the more bonds are made, the more stable the local configuration is and the less likely reconfiguration is made.

Local reconfiguration random distribution. In the *Ok* model, we assume that when a reconfiguration (nascent or internal) is applied, the new local configuration (of the nascent path or of the subpaths in the hexagon H) is drawn *uniformly at random* among all the valid local reconfigurations. Together with the upcoming definition of the rates, this ensures that the resulting distribution follows Boltzmann law.

Reconfiguration events rates. Following the steps of [13], we define the rate of each reconfiguration as a function of the number of bonds involved: a configuration involving b bonds will dissociate at a rate inversely proportional to some exponential of b . Precisely, as in [13] we define the *free energy of dissociation* of a single bond to be $\Delta G/RT = G_B$, and thus G_B contains a mix of entropic and enthalpic factors

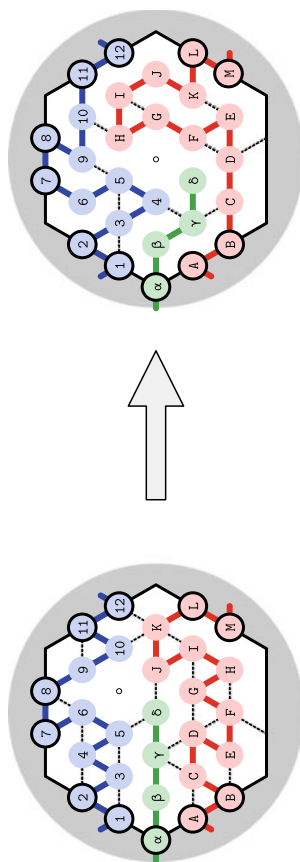


Fig. 4 Internal Reconfiguration inside an hexagon H of radius ρ (here $\rho = 3$): The reconfiguration involves four subpaths: three of them are clamped 2-3-4-5-6-7, 8-9-10-11 and B-C-D-E-F-G-H-I-J-K-L, and one has a free end, $\alpha - \beta - \gamma - \delta$. The reconfiguration step changes their routes inside the hexagon. Beads on the border of H (circled in bold) or outside do not move. The left and right configurations make 20 and 10 bonds (dashed lines), respectively, inside the ball

related to the formation of the bond between the two beads, measured in units of RT . The rate of the next internal reconfiguration event in a given hexagon H is then defined as:

$$r_I = k_f \exp(-bG_B), \quad (1)$$

where b is the number of bonds involving some bead strictly inside the selected hexagon H . k_f is a kinetic parameter, usually set to $k_f = 10^{-6}$.

To take into account the specificity of co-transcriptional folding where the nascent beads, close to the polymerase, fold at a much higher rate, we define similarly the *free energy of dissociation of a single nascent bond* (i.e., involving at least one nascent bead, and thus catalyzed by the proximity to the polymerase) to be $\Delta G/RT = G_N$. The rate of the next nascent reconfiguration event is then defined as:

$$r_N = k_f \exp(-bG_N), \quad (2)$$

where b is the number of nascent bonds, involving at least nascent bead.

In order to express the rate r_G of the growth of the transcript in the same terms, we introduce, as in [13], a fictitious energy G_G such that:

$$r_G = k_f \exp(-G_G). \quad (3)$$

Adjusting the rates. In order to match the oritatami dynamics, the nascent beads must explore up to 5^δ paths, so as to find the optimal nascent path, before the next nascent bead is produced. This exploration requires on average $\sim \delta \ln 5 \cdot 5^\delta$ uniform random nascent reconfiguration events, as collecting N coupons takes on average $N \ln N$ trials, and thus $\sim \delta \ln 5 \cdot 5^\delta / r_N$ time as each event occurs every $1/r_N$ on average. In order to match the co-transcriptional folding experimental observation that the growth occurs at a much lower rate than the optimal folding of the nascent beads, we must then have:

$$\frac{\delta 5^\delta \ln 5}{r_N} \ll \frac{1}{r_G} \quad \text{that is: } r_N \gg \delta 5^\delta \cdot r_G, \text{ i.e.. } bG_N \lesssim G_G - \delta \ln 5 \quad (4)$$

Note that this is consistent with the fact that the number of nascent bonds b is at most $4\delta + 1$: each nascent bond must account for a fixed amount of energy, that is: $G_G \gtrsim \delta(4G_N + \ln 5)$. In particular we should have: $G_G \gtrsim \delta \ln 5/4$ so that $G_N > 0$. This first rough estimation however needs to be confirmed by running effective simulation of the Ok model.

There are no explicit constraints between G_B and G_N besides the fact that $G_B > G_N$ as the nascent bonds should dissociate more easily than the (colder) bonds located away from the polymerase.

3.3 Implementing the Ok model

We do not have a running implementation of the Ok model yet. Here are a list of recommendations for its software implementation.

Ok model possible implementation. The Ok model is parametrized by $(\delta, \rho, G_G, G_B, G_N)$. The possible event are:

- **growth**: extending the molecule by one bead whose position is chosen uniformly at random among the unoccupied positions neighboring the current end of the molecule;
- **internal(x)**: rerouting uniformly at random the subpaths strictly inside the hexagon $H(x, \rho)$ centered on position x with radius ρ , without modifying their extremities on the boundary of $H(x, \rho)$;
- **nascent**: rerouting uniformly at random the path consisting of the δ nascent beads at the end of the molecule.

There are as many **internal(x)** events as there are hexagons $H(x, \rho)$ intersecting the molecule. Each event is associated with an occurrence time T picked at random according to their corresponding rate r_G, r_I or r_N . Note that the rates r_I and r_N depend on the current local configuration. As T is a memory-less exponential random variable of law $\text{Exp}(r)$, s.t. $\Pr\{T \geq t\} = e^{-rt}$, it can easily be picked using the formula: $T = -(\ln U)/r$ where U is a uniform real-valued random variable over $[0, 1]$. The events scheduling is classically implemented using a priority queue to extract the next upcoming event (with the lowest occurrence time). Thanks to the memory-less property, the occurrence times of all events impacted by an applied event are simply redrawn according to their recomputing rate: for instance, the occurrence times of the **internal(y)** events need to be updated for all $y \in H(x, \rho)$ after an **internal(x)** event occurred.

Performance optimization. The implementation of the **nascent** and **internal** will get computer intensive as soon as δ and ρ get larger than 10 and 5 respectively. Even for smaller values of ρ , we recommend to precompute and remember the set of possible subpaths for a given local configuration so as to speed up the uniform picking of the reconfiguration. As subpaths are clamped at both ends, this should not impact too much the memory. Note however that if a free path (whose only one end is clamped) belongs to the hexagon, it might however require too much computation time as there might be too many possible configurations to consider (as it could be as long as $\Theta(\rho^2)$ if it fills a constant fraction of the hexagon). In this situation, we recommend to allow only the δ last beads of the free path to be rerouted.

4 Conclusion

In this article, we have proposed a new model which includes randomly occurring local reconfigurations in the oritatami model. As demonstrated by the experimental implementation of the tile assembly model [12, 14], taking into account this natural phenomenon into model is a necessity to design successful in-vitro implementations. We hope to implement this model as an open source software soon, and are eager to explore whether the basic structures that made computation possible in the oritatami model, such as glider, switchback, folding meter and pocket, can be made tolerant to such thermal noise. One may also wonder if such reconfiguration could be exploited to discover new way of computing using co-transcriptional molecular folding.

References

1. C.G. Evans, *Crystals That Count! Physical Principles and Experimental Investigations of DNA Tile Self-Assembly*. Ph.D. thesis (California Institute of Technology, 2014)
2. C. Geary, G. Grossi, E.K.S. McRae, P.W.K. Rothmund, E.S. Andersen, RNA origami design tools enable cotranscriptional folding of kilobase-sized nanoscaffolds. *Nat. Chem.* **13**, 549–558 (2021)
3. C. Geary, P.E. Meunier, N. Schabanel, S. Seki, Oritatami: a computational model for molecular co-transcriptional folding. *Int. J. Mole. Sci.* **20**(9), 2259 (2016) (Its conference version was published in Proc. MFCS 2016)
4. C. Geary, P.-E. Meunier, N. Schabanel, S. Seki, Proving the Turing universality of oritatami cotranscriptional folding. in *Proceedings of ISAAC 2018, LIPIcs*, vol. 123, (2018), pp. 23:1–23:13
5. C. Geary, P.W.K. Rothmund, E.S. Andersen, A single-stranded architecture for cotranscriptional folding of RNA nanostructures. *Science* **345**, 799–804 (2014)
6. K. Maruyama, S. Seki, Counting infinitely by oritatami cotranscriptional folding. *Nat. Comput.* **20**(2), 329–340 (2021)
7. E.C. Merkhofer, P. Hu, T.L. Johnson, Introduction to cotranscriptional RNA splicing. *Methods Mole. Biol.* **1126**, 83–96 (2014)
8. R. Parales, D. Bentley, “co-transcriptionality”—the transcription elongation complex as a nexus for nuclear transactions. *Mole. Cell* **36**(2), 178–191 (2009)
9. D. Pchelina, N. Schabanel, S. Seki, Y. Ubukata, Simple intrinsic simulation of cellular automata in oritatami molecular folding model, in *Proceedings of LATIN2020, LNCS*, vol. 12118 (Springer, 2020), pp. 425–436
10. N. Schabanel, Simple OS simulator. <http://perso.ens-lyon.fr/nicolas.schabanel/OSsimulator/>
11. K.E. Watters, E.J. Strobel, A.M. Yu, J.T. Lis, J.B. Lucks, Cotranscriptional folding of a riboswitch at nucleotide resolution. *Nat. Struct. Mole. Biol.* **23**(12), 1124–1131 (2016)
12. E. Winfree, R. Bekbolatov, Proofreading tile sets: error correction for algorithmic self-assembly, in *DNA Computing, 9th International Workshop on DNA Based Computers, DNA9, Madison, WI, USA, June 1–3, 2003*, revised Papers, eds. by J. Chen, J.H. Reif. Lecture Notes in Computer Science, vol. 2943 (Springer, 2003), pp. 126–144. https://doi.org/10.1007/978-3-540-24628-2_13

13. D. Woods, H.L. Chen, S. Goodfriend, N. Dabby, E. Winfree, P. Yin, Active self-assembly of algorithmic shapes and patterns in polylogarithmic time, in *Proceedings of ITCS2013* (ACM, 2013), pp. 353–354
14. D. Woods, D. Doty, C. Myhrvold, J. Hui, F. Zhou, P. Yin, E. Winfree, Diverse and robust molecular algorithms using reprogrammable DNA self-assembly. *Nature* **567**, 366–372 (2019)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Implementing a Theoretician’s Toolkit for Self-Assembly with DNA Components



Matthew J. Patitz

Abstract A diverse array of theoretical models of DNA-based self-assembling systems have been proposed and studied. Beyond providing simplified abstractions in which to develop designs for molecular implementation, these models provide platforms to explore powers and limitations of self-assembling systems “in the limit” and to compare the relative strengths and weaknesses of systems and components of varying capabilities and constraints. As these models often intentionally overlook many types of errors encountered in physical implementations, the constructions can provide a road map for the possibilities of systems in which errors are controlled with ever greater precision. In this article, we discuss several such models, current work toward physical implementations, and potential future work that could help lead engineered systems further down the road to the full potential of self-assembling systems based on DNA nanotechnology.

1 Introduction

Beginning as a branch of mathematics, computer science is fundamentally focused on understanding the process of “computing” and how it can be embodied in physical systems. Mainstream studies largely focus on digital computers, composed of electronic circuits operating on Boolean logic. However, the underlying concepts of processing and transforming information via specified rules, or algorithms, can also be realized in many other formats (e.g., analog computing devices [1], quantum computers [2], etc.), including natural systems that provide continued inspiration for novel directions in engineering (such as information processing networks in cells [3]).

At the foundation of computational theory is the existence of “universal computers,” which are computing devices capable of running any possible program. These allow for the design of systems which can be given input in the form of an arbitrary program along with arbitrary data to be given to that program and which output the

M. J. Patitz (✉)
University of Arkansas, Fayetteville, AR, USA
e-mail: patitz@uark.edu

© The Author(s) 2023
N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*,
Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_14

241

result of running the input program on the input data. This is in contrast to having to design a specific computer for each program that one wants to run and provides for an immense diversity of programmable behavior for such systems.

Relatively quickly after the structure of DNA was revealed [4] and its basic properties began to be understood, computer scientists started to see its potential as a programmable substrate [5–7] which could yield additional directions in computing. Harnessing the combinatorial powers of DNA molecules opened the door for engineered computing at the nanoscale. While biology utilizes DNA as an information storage medium, scientists and engineers began to see its potential as a structural component (e.g., [8–11]) and a platform for computing (e.g., [12–14]). This has enabled the design of systems whose target behaviors and outputs include structure building and computing, and through a combination of those, interfacing with chemical and biological systems to process information about them and/or to generate output that becomes integrated within them [15–17].

Self-assembling systems are systems composed of relatively simple components which begin in a disorganized state and autonomously combine to form more complex structures. There are already many different approaches to realizing DNA-based self-assembling systems as platforms for structure building guided by computation, and one job of theoreticians has been to abstractly model them and to determine, in their mathematical limits, the strengths and weaknesses of each, especially as compared to each other. This includes an emphasis on those which may have more viable molecular implementations, and an additional aim has been to search for techniques which can be used to circumvent or at least minimize errors that occur in physical implementations (i.e., behaviors that deviate from those predicted by the high-level mathematical abstractions). In this paper, we seek to outline several of the key aspects of various abstract models of DNA-based self-assembling systems that have been studied and discuss the theoretical pros and cons they provide as well as the current, and potential future, development of DNA-based systems capable of implementing them. The continued growth of both the theoretical and experimental sides of DNA nanotechnology, with each relying upon the other for insights and direction, can lead to the implementation of more complex and diverse systems that more fully realize the theoretical potential of self-assembling systems.

To cover the wide diversity of models and techniques that have been explored, this paper is organized as follows. In Sect. 2, we cover some preliminary definitions, and in Sect. 3, we define some of the main metrics often used for comparison across models and systems. In Sect. 4, we discuss one spectrum across which systems may vary, that of how many times copies of each individual component type appear in a self-assembled structure, and demonstrate ways in which trade-offs in different metrics occur across that spectrum. In Sect. 5, we discuss a wide variety of methods that can be used to provide the input to a self-assembling system and direct its output and demonstrate ways in which those methods have been implemented and their corresponding trade-offs as well as current technical limitations. In Sect. 6, we cover a set of varying dynamical behaviors that systems and/or individual components of systems may have, and how those behaviors can influence the powers and limitations of the systems, as well as current experimental implementations and potential future

implementations that may more fully realize the powers provided those dynamical behaviors. Finally, in Sect. 7 we provide a brief summary and our optimism for the future of this area.

2 Definitions and Notation

In this section, we introduce a few basic definitions and some of the notation used throughout the following sections.

An individual building block of a self-assembling system will be referred to as a *tile*, or sometimes as a *monomer*. Although a tile actually represents a component made out of one or more strands of DNA, for most of the theoretical models we will be discussing they will be abstractly represented as two-dimensional squares (or three-dimensional cubes in a few cases). An example schematic depiction of a tile can be seen in Fig. 1.

Tiles are able to attach to each other via *glues* on their sides. A glue is an abstraction of a *binding domain*, which is typically a single-stranded portion of DNA that is free to bind to a strand containing the complementary sequence. Although the overall strength of binding of two complementary strands of DNA is highly dependent upon several factors (e.g., the specific sequences, their lengths, and the number of Gs and Cs as opposed to As and Ts), a common design goal is to create categories of binding domains which have very similar attachment strengths to all other binding domains within the same category. For this reason, it is possible in the abstract, mathematical models to instead refer to glues by their categories. Therefore, we will equate each strength category with a natural number and call a glue a “strength 1,” or “strength 2” glue, for instance. Strength 1 glues can be thought of as those whose binding strengths (with their complementary domains) are approximately equivalent to some system-specific basic, standard value. Strength 2 glues can be thought of as having approximately double that binding strength. As an additional abstraction, rather than referring to specific DNA sequences for glues, we will give each glue a text label. (e.g., in Fig. 1, the north glue has label a and strength 1.) A glue binds to its complement, and the complement of a glue label is represented with the *prime* character, e.g., the complement to a is a' .



Fig. 1 Schematic depiction of an example (square) tile. Each glue (a.k.a. binding domain) is represented by the text label and black rectangle(s) on one side. The number of black rectangles corresponds to the integer-valued strength of the glue. The north (i.e., top) side has a strength 1 glue of type a , the east (i.e., right) side has a strength 2 glue of type b , the south (i.e., bottom) side has no glue (or the *null* glue), and the west (i.e., left) side has a strength 2 glue of type c' . The entire tile has the label D

An entire tile may also be assigned a label (e.g., D for the tile in Fig. 1). Such a label is non-functional as far as the tile's binding to other tiles, but can be used to categorize the tile and potentially represent some sort of marking on, or functionalization of, the tile. For example, tiles labeled with a 1 could have an attached molecule that makes them distinguishable during imaging from tiles labeled with 0 that have no such attached molecule. In this way, a readable pattern may be formed in an assembly by tiles labeled 1 and 0.

Since we are focusing on systems which build structures, we will call the desired output of a system the *target structure* (or *target assembly*). Typically, there is a *target shape* that defines the two- or three-dimensional shape of the target structure.

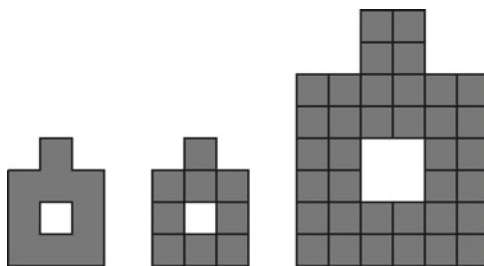
3 Metrics

If models, and systems within them, are to be measured against each other, there must be metrics on which to base the comparisons. There are several such metrics by which self-assembling systems can be measured. While accuracy in producing the desired output and robustness to environmental conditions are of great importance, several other metrics may influence important aspects, such as the feasibility of implementation.

We now define four metrics that we will focus on during our comparisons of models:

1. **Tile complexity:** the number of unique tile types required, i.e., the number of unique kinds of monomers that serve as building blocks in a particular system. This is a measure of the amount of monomer reuse that is achieved and is discussed at greater length in Sect. 4.
2. **Monomer complexity:** in general, the (maximum) size of individual monomer types. This is based upon the length and number of strands composing a single monomer type. Complexity may also more generally be derived from requirements for complex shapes, rigidity, or dynamic behavior, as well as the difficulty of fabrication (i.e., the number of experimental steps required for their production).
3. **Resolution:** the physical dimensions of a single coordinate location in the target shape (i.e., given the set of coordinates for the target shape's voxels, how large is the volume of each in the actual target structure). It is often the case that, somewhat counterintuitively, constructions in theoretical models can be shown to be more efficient (sometimes even achieving mathematically provably the greatest possible efficiency) in tile complexity by generating "scaled up" versions of 2D (resp. 3D) target shapes in which each pixel (resp. voxel) is replaced by a square (resp. cube) of potentially many tiles. (See Fig. 2 for an example.)
4. **Addressability:** the ability to uniquely address locations in the target structure with specific binding domains. For many applications, greater specificity in the ability to address unique locations of the target structure is desired, since such locations can be used to precisely place molecules linked to the DNA strands.

Fig. 2 (Left) Example shape S , (middle) S pixelated and at scale factor 1, (right) S at scale factor 2

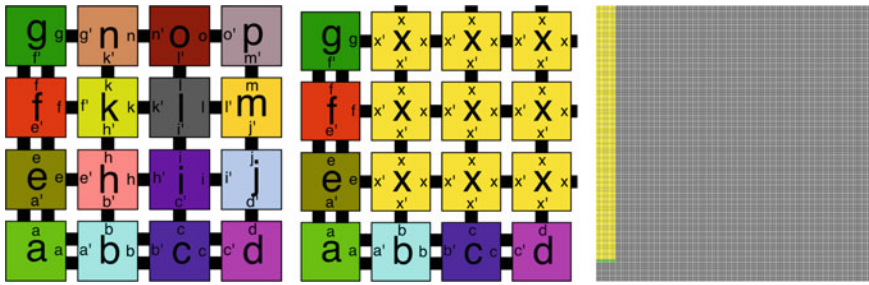


4 Monomer Reuse: Hard-Coded Versus Algorithmic

A major design consideration for self-assembling systems deals with monomer reuse. For instance, in a system of self-assembling tiles, given the set of tile types, how many times does a tile of each type appear in the target structure? In this section, we investigate theoretical models and experimental implementations that vary greatly in monomer reuse and discuss related trade-offs and potential future directions.

On one end of the spectrum, each type appears only one time in the target structure. We refer to such structures as *hard-coded*, a.k.a. *uniquely addressed*, and note that this paradigm has been successfully experimentally employed in both DNA origami [9] and DNA bricks [8]. (See Fig. 3a for an abstract example.) The advantages of such systems include excellent addressability as well as robustness to some of the types of errors seen with monomer reuse (see Sect. 6.1). Disadvantages include maximal tile complexity and that the size of a target assembly is limited by the number and size of unique monomer types which can be created and utilized. For instance, in DNA origami one long DNA strand, referred to as the *scaffold strand*, winds through the entire structure and is bent into the target shape by many short *staple strands* that each bind to approximately two or three short sections of the scaffold strand, bringing and holding these distant parts closer together to eventually form the final shape. With the size of a DNA origami structure determined by the length of its scaffold strand, the design of custom scaffold strands (as opposed to the original standard scaffold strand, the M13mp18 bacteriophage's genome) becomes important. Much progress is being made in this area [18–20], allowing for both smaller scaffolds, with only a few hundred nucleotides, and much larger scaffolds, with over 50,000. Continued improvements to allow greater diversity in scaffold sequences and lengths will further increase the range of structures producible via DNA origami.

One alternative to hard-coded structures is the generation of (theoretically) unbounded *periodic* structures. In this case, each monomer type is reused arbitrarily often in a repeating pattern (see [21, 22] for experimental examples and Fig. 4 for abstract examples). Advantages of such systems include theoretically unbounded sizes for target structures, potentially low tile complexity, and robustness to nucleation errors (since it is valid for growth to begin from any location, see Sect. 6.1) and



(a) Example with hard-coded (b) Example with tile reuse us- (c) Zoomed out example of a
(a.k.a. uniquely addressed) tiles. ing a trivial “filling” algorithm. 100 × 100 square using nearly
optimal tile reuse.

Fig. 3 Depiction of hard-coded assembly versus algorithmic assembly

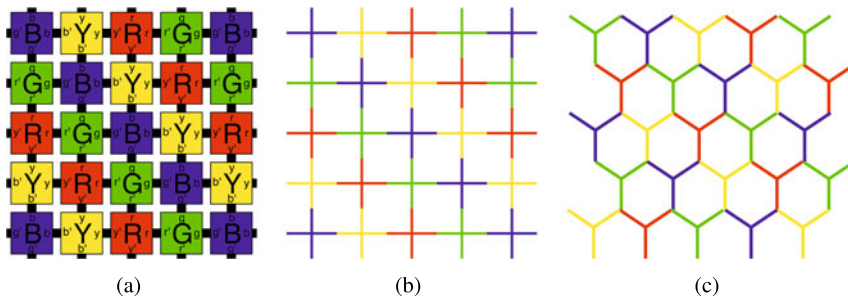


Fig. 4 Examples of periodic structures, including **a** a grid of square tiles and **b, c** lattices formed by abstract non-square monomers

growth errors (see Sect. 6.1). However, disadvantages include limited addressability (restricted to periodically repeating locations) and growth of uncontrolled numbers of copies relatively simple structures.

An additional alternative is *algorithmic* growth in which individual monomer types may be used arbitrarily often (even in a possibly aperiodic manner) in each target assembly, as the attachment of each implicitly follows the rules of a designated algorithm. (See Fig. 3b for a basic example and Fig. 5 for a more complex example.) Advantages include the ability to (theoretically) create assemblies of arbitrary but bounded size using mathematically optimal tile complexity. Using information theoretic and computational complexity arguments, it has been proven that algorithmic self-assembly systems in the aTAM are capable of universal computation [6], can achieve mathematically optimal tile complexities for systems constructing squares [23] and scaled versions of arbitrary finite shapes [24], and even include systems capable of universally simulating all other systems [25]. For instance, Fig. 3c shows a zoomed out image of a 100 × 100 square which self-assembled using an efficient algorithm for counting and filling. The green row is made up of 7 unique tile types, the yellow portion of only another 14, and the entire gray portion of only 6 unique

tile types. The algorithm used to create the system can take an arbitrary positive integer n as input and output a system that self-assembles an $n \times n$ square using $\log(n) + 14 + 6$ tile types, where the same 14 yellow and 6 gray tiles are used for all values of n , and an n -specific set of $\log(n)$ green tiles are used to encode a number related to n . The green tiles assemble into a row to which the 14 yellow tiles attach and execute the binary counting algorithm to grow to the necessary height, allowing the 6 gray “filler” tiles to fill in the rest of the square to precisely the dimensions $n \times n$. While this construction uses approximately $\log(n)$ tile types for arbitrarily large n , in [26] it was shown that this can be improved to $\Theta\left(\frac{\log(n)}{\log(\log(n))}\right)$, which was also shown to be the mathematical lower bound [23] for almost all values of n . Note that this is an *exponential* improvement over a hard-coded assembly, which even in the case of the 100×100 square, for example, would require 10,000 unique tile types, rather than the 27 used here.

Disadvantages of algorithmic self-assembly include restricted addressability and, quite importantly, the potential for various types of errors to occur. It has been shown that for algorithmic growth to occur, some form of *cooperation* (see Sect. 6.1) is required, and this can lead to errors like those discussed in Sect. 6.1. Furthermore, several theoretical results require large scaling of shapes (see Sect. 3 for the definition, and an example, of a scaled shape). In some constructions, the scale factor is quite large, so each point of the shape requires a relatively large volume and is filled by a large number of tiles, with that number often depending upon, and growing with the size of, the specific target shape. This can also require tile sets that are significantly larger than can be currently successfully implemented.

Developing large tile sets which self-assemble with few errors requires the design of large sets of orthogonal glue domains (i.e., sets of domains such that each has a strong affinity for its complementary domain, but very weak affinity with all others). The number of possible domains of any given length n (i.e., composed of n bases) is bounded by 4^n (since there are 4 bases to choose from in each location), but only a subset of those can be selected so that (1) they are orthogonal, and (2) the binding affinities of all complementary pairs are very nearly equivalent. The second condition is important so that all glues have similar behaviors and becomes even more important if glues in different theoretical strength categories are to be implemented (see Sect. 6.1). Additionally, as glue domains are forced to become longer to accommodate larger sets of glues, the potential for non-orthogonal interactions increases (i.e., glue domains may have positive binding affinities for domains other than their complements). Other commonly considered criteria include ensuring that sequences have minimal “self-structure” (i.e., they do not have a tendency for some subsequences to bind to other subsequences on the same strand), avoiding G-tetraplexes (i.e., sequences of 4 Gs in a row), etc. Work has been done to create models that predict domain interactions and software that can perform automated design and theoretical testing of glue sets based on mathematical models of DNA strand interactions [27–32]. However, additional enhancements and extensions to this so-called *sequence design* process for glue domains has the potential to greatly improve the size and quality of glue domain sets and therefore the sizes of tile sets which can be successfully implemented.

5 Inputs

In this section, we discuss various methods of providing input to self-assembling systems and theoretical models and experimental implementations that utilize them. The goal is to understand different ways of controlling self-assembling systems and how we can improve that control in the future.

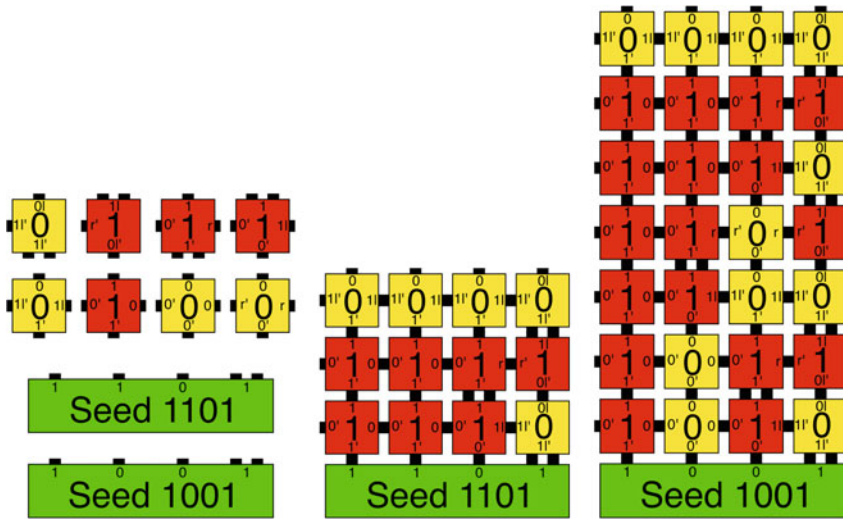
We can think of the monomers of the system as representing the instructions of the program to be executed. On the one hand, we can design deterministic systems which are *non-programmable*, in which we mean that (within a valid range of environmental parameters) the system is designed to always produce the same output, i.e., structure, regardless of any (reasonable) variations in the environment. That is, the environment does not provide meaningful input to the system and the same “program” is always “executed”. On the other hand, systems may be designed so that there is some environmental variable which can be tuned, with each setting yielding a distinct output. There are many techniques for providing input to self-assembling systems, and we now describe a few of them.

5.1 Seed Assemblies

A frequently utilized input technique is the use of a *seed assembly*. A seed assembly is a variable, preformed structure that is added to the system in addition to a constant set of monomer types. In theoretical models such as the abstract Tile Assembly Model (aTAM) [6], there are computationally universal systems each consisting of a single, constant set of “universal” tile types such that for every possible program and input data pair, a seed assembly encoding that program and input data can be added to a solution containing those universal tiles, causing the system to build a representation of the computation of that program on that input data. Furthermore, that computation can also determine the resulting shape of the self-assembled structure. In such a scenario, the seed assembly is incorporated into the target assembly and provides the input that determines its resultant shape and/or pattern. As an example, see Fig. 5 where an aTAM tile set is shown, as well as assemblies that self-assemble from two different seeds. In this aTAM example, the *binding threshold* (a.k.a. *temperature*) parameter is set to 2, meaning that tiles only attach to the seed, or the assembly containing the seed, if they can bind with at least one strength 2 glue, or two strength 1 glues. (See Sect. 6.1 for additional details.) Due to the binding threshold and the glue patterns, the assemblies that grow from each seed vary in size.

Current methods of experimentally implementing seed assemblies include using DNA origami as seeds with the tiles being either single-stranded or multi-stranded complexes that attach to, and grow away from, the origami seed [10, 28], or even just a single strand which serves as the seed for nucleation of growth [33].

Variants on the use of a seed assembly are also possible. For instance, a seed assembly could instead serve as a template to be filled in by the tiles (which later



(a) A generic tile set for counting in binary (yellow and red), and two example seed structures (green). (b) Example of the assembly that grows from the seed encoding 1101, with three rows of tiles before growth stops. (c) Example of the assembly that grows from the seed encoding 1001, with seven rows of tiles before growth stops.

Fig. 5 Example aTAM systems that use the same tile set but different seeds to provide inputs which yield differing assemblies

detach from it), or as a template of a shape to be replicated. However, using seed assemblies in such ways requires additional dynamics by which the target structures can be separated from the seed assemblies (see Sect. 6).

5.2 Tile Subsets

The variable input to a system could instead be the selection of only a specific subset of tile types from a larger set. In this case, a large set of tile types is designed and synthesized, and then for each particular target assembly, a specific subset is selected and added to the solution. This technique can be utilized to make target structures which are subsets of a larger structure, as in the DNA brick technique [8], or to select specific sequences of program instructions to be carried out from a generic set [28]. A disadvantage of this approach is the potential for large tile complexity, and an advantage is that simple selection and mixing of the necessary subset of types is sufficient to produce any of the potential systems.

5.3 *Monomer Concentrations*

Rather than strictly varying the monomers in a system in a binary way (i.e., they are either present or not), instead their relative concentrations can be varied. This technique, known as *concentration programming*, has been shown to allow for great tuning in a set of theoretical results [34, 35], so that by only varying tile concentrations, any finite shape can be targeted while using a constant tile set. However, not only does this require scaling of the target shape and thus a loss in resolution, a source of great difficulty with experimental implementations of this approach is the precision required for very fine-tuned relative concentrations of tiles. Modern equipment capable of mixing nanoliters of fluid is allowing finer control of concentrations, and it may be feasible in the future to realize more of the potential of concentration programming.

5.4 *Programmed Temperature Fluctuations*

In systems of tile-based self-assembly where each tile is composed of multiple strands of DNA, common laboratory protocol involves a single-pot system with annealing, where the individual strands are put into a solution that is first brought to a high enough temperature to ensure complete dissociation of strands. Then, the solution is cooled to a temperature where the bonds between the individual strands comprising each tile are strong enough to allow for the formation of the tile complexes, but the binding domains that would bind tiles to each other are not strong enough to form long-lasting bonds. After holding at that temperature for a period of time that ensures most strands will be incorporated in tiles, the temperature is further lowered to a point at which tiles can bind to each other.

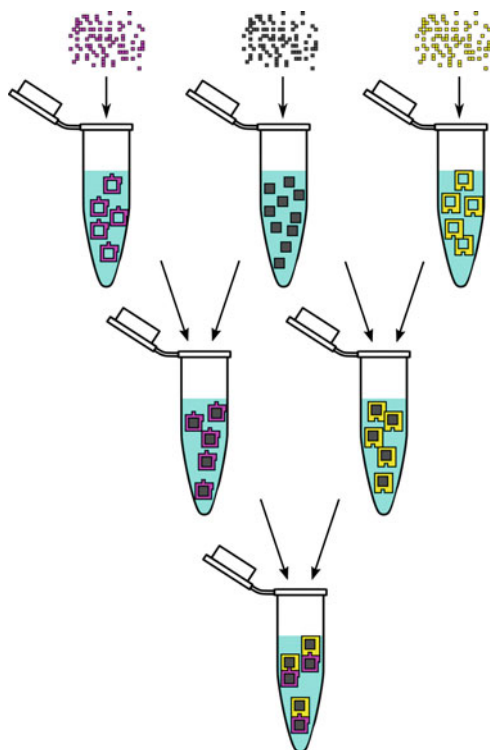
Another theoretically powerful method of supplying input to a self-assembling system is to vary the temperature of the system through a series of prescribed temperatures, both raising and lowering it multiple times. This can allow periods of growth followed by periods of melting, which may remove some tiles and create favorable locations for different types of tiles in the same locations during periods of lowered temperatures. Theoretical modeling of this procedure, known as *temperature programming* [36], has been shown to allow for a constant tile set that can be programmed, via only temperature change sequences, to form any finite shape [37]. Trade-offs can be made between the number of temperature changes required to direct growth of a target shape versus the resolution of the shape. However, physical implementations have not yet been achieved due to the difficulty of designing binding domains and systems with enough granularity to correctly bind and/or dissociate across a wide enough set of temperature levels.

Future advancements in sequence design that allow for the development of glue domains that exhibit fine enough granularity in binding strengths to support multiple discrete levels of binding and melting (possibly combined with novel tile designs) may allow temperature programming to become a useful tool.

5.5 Staged Assembly

The number of experimental steps required to implement a self-assembling system can vary greatly. Single-pot single-stage systems exist where a set of strands is added to solution, all in one test tube, which is first heated and then annealed, and the target structures completely form during that process. Alternatively, systems exist in which multiple products are made in different tubes, then the products of subsets of those tubes are combined together (with each mixing process and then the simultaneous self-assembly processes in the separate tubes considered a “stage”), and the number of stages can be quite large. (A simple example is shown in Fig. 6.) The simplicity of single-pot single-stage systems is beneficial from an experimental perspective, but in the theoretical Staged Tile Assembly Model [38, 39] it has been shown that tile complexity can be exchanged for stage complexity. That is, the number of tile types required to build shapes can be dramatically reduced (even down to a constant tile set for an infinite set of shapes) by increasing the number of stages. Experimental work which combines staged assembly and hierarchical growth (see Sect. 6.2) has also demonstrated the great power of this paradigm [40].

Fig. 6 Example staged assembly system. Individual tiles are added to the top tubes in the first stage, then assemblies from the top tubes are mixed for the next stage, etc.



Combined with methods that cause tile detachments as well (see Sect. 6.4), theoretical results in staged assembly have shown the possibility for new system behaviors such as the replication of input shapes or patterns [41–43] or the marking of input assemblies that match specific shapes [44].

The theoretical benefits of staged assembly come with a high cost on the experimental side. First, there is the additional work of carefully, uniquely mixing the inputs to each tube of each stage. Although this can be made much easier by automation, it then becomes difficult to first know how long the self-assembly at each stage should be allowed to proceed and second to extract only the correctly completed products of each tube of each stage for further mixing. Theoretical work has been done to study staged assembly systems in which the correctly completed products of each tube have maximal size difference from all others [45], but this technique (and those which effectively realize the tile complexity benefits of staged assembly) requires use of hierarchical assembly (see Sect. 6.2). Future work that realizes the full theoretical benefits of staged assembly would require improvements in purification techniques to make it easier to select only correctly completed products from each stage, as well as improved design and control of hierarchically self-assembling systems.

6 Dynamics

In this section, we investigate the consequences of a variety of changes to the dynamical behaviors allowed by different models and/or tile designs. By categorizing a variety of dynamical behaviors and demonstrating their powers in theoretical models, and also discussing experimental work that has begun to implement several, we hope to provide a road map for future work that can further realize the powers offered by these behaviors while effectively balancing the trade-offs.

6.1 Cooperativity

It was long speculated [23, 46] and recently proven [47] that algorithmic self-assembly cannot occur in the aTAM without behavior known as *cooperation*. Typically, we say a tile attaches to an assembly using cooperation when its initial binding to that assembly requires it to form bonds with more than one tile that is already part of that assembly. Note that biochemistry literature sometimes uses the term *avidity* to refer to the same concept. We can consider the binding domains which initially bind when a tile attaches as its “input” domains and the remaining domains (which may later serve to allow for the binding of additional tiles) as “output” domains. Intuitively, cooperation forces the attaching tile to “read” information from two separate tiles via their output domains, and careful design of the tile types of a system can ensure that the information encoded in the output domains of tiles represents specific logical transformations of the input information (e.g., the output domains

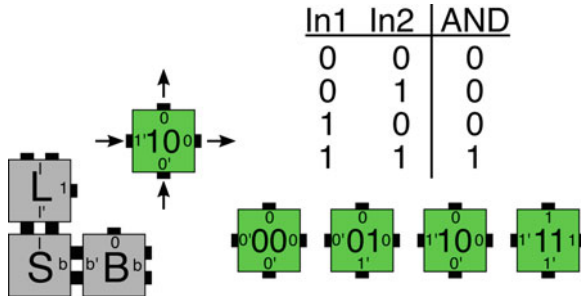


Fig. 7 Example of cooperative tile attachment: On the left is an assembly consisting of three tiles that make a “corner” which allows for cooperative attachment of a tile that matches the two glues labeled “1” and “0”. On the right, four tiles implementing AND logic (with input glues on the left and bottom and outputs on the top and right). Only the tile labeled “10” can cooperatively attach to the assembly

could encode the bit resulting from the logical AND operation performed on the bits represented by the input domains, as shown in Fig. 7). It has been shown that for an arbitrary program, it is possible to design a set of tile types such that they are forced, in the theoretical setting, to self-assemble in a pattern that follows the execution of that program [6].

In the aTAM [6], the requirement for cooperative binding is captured by a system parameter called the *temperature*, which is physically based upon factors such as the temperature of the system as well as the concentration of tile monomers. This temperature parameter is also commonly referred to as the *binding threshold*, and in the discrete formalization of the aTAM, it is commonly set as either 1 or 2. A value of 1 means that the binding of a single input domain is always sufficient to allow a tile to “permanently” attach to a growing assembly. A value of 2 means that either at least two input domains must correctly bind, or a single input domain of at least double strength (i.e., a *strength 2* glue) must bind for a tile to attach. The theoretical power of aTAM systems with the temperature parameter equal to 2 has been proven to be quite impressive, including algorithmic self-assembly capable of the natural simulation of any possible program, the self-assembly of structures using mathematically minimal tile complexity, etc.

Unfortunately, the physical reality (as it often does) differs from the theoretical model and sometimes experimental systems designed to behave as temperature 2 aTAM systems do not behave as such. In some cases, tile attachments occur in which tiles “temporarily” bind via a single strength 1 glue and the glue in the location intended to be the second input has an incorrect, “mismatching” glue domain which does not bind (or does so only partially, with low strength). (See Fig. 8 for an example.) Although such attachments are not expected to last long, with some nonzero probability a neighboring location may receive a tile which binds to the “erroneous” tile with one of its input domains, causing both tiles to be attached to the assembly and each other with enough binding strength to be “permanent”. We will call this

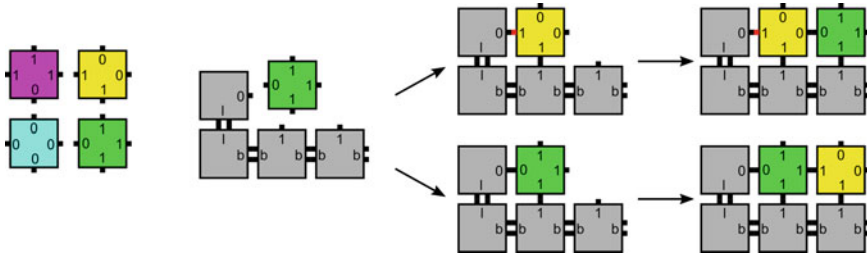


Fig. 8 Example of a tile binding with a glue mismatch, leading to an algorithmic error. On the left, a set of “computing” tiles. On the right, an assembly to which the green tile (but no other) can bind with strength 2 (which happens on the bottom path). On the top path, the yellow tile binds with strength 1 and a mismatched glue. Before it detaches (due to weak binding), another tile may bind and “lock in” the algorithmic error

a *growth error*, and in such a case, the erroneous tile may corrupt the computation being performed and cause the algorithmic growth to proceed incorrectly. This type of behavior is captured in the more physically realistic kinetic Tile Assembly Model [6], a.k.a. kTAM, and kTAM modeling has helped lead to several *proofreading* and error suppression techniques (where errors are considered to be tile attachments that differ from the expected tile attachments in the aTAM) that have been developed to reduce the prevalence of such errors [48–52]. It is notable that aTAM behavior can be approximated arbitrarily closely by the kTAM, and careful control of temperature and tile concentrations along with proofreading can help to the extent that the incidence of such errors in experimental systems has been decreased to around 0.017% [53] (or to 0.03% for larger tile sets [28]). However, even those seemingly excellent rates are still be too high for accurate algorithmic growth of even moderately complex (from a theoretical perspective) systems.

Cooperation has been shown to be necessary for algorithmic self-assembly, but it has also been proven that there are methods of cooperation other than the specific “glue cooperativity” already discussed. Other ways of causing the attachment of a tile to depend on two or more others, called *weak cooperativity* [54], have been shown in theoretical results using geometric hindrance [55, 56] and repulsive forces [57, 58]. To utilize geometric hindrance, theoretical systems with the parameter temperature set to 1 can be designed where tiles have shapes other than squares [55, 59, 60] (or in addition to squares [56]) so that the tiles that can correctly bind into a location are selected by matching a single glue for binding as well as having a complementary geometric shape (serving as the second input) that matches the second input location. To utilize repulsive forces, instead of relying on a complementary geometry as the second input, tile design can include the specific placement of a tile element which will experience a repulsive force when adjacent to another instance of that element on a neighboring tile. In this way, only a tile of a type which does not cause repulsive elements to align will be able to bind into a new location.

While the aforementioned methods of weak cooperation provide a perhaps stronger barrier to growth errors that can occur using glue cooperation, by actively

preventing incorrect tile binding rather than simply not favoring it, another problem arises in temperature 1 systems. This problem, known as *spurious nucleation*, stems from the fact that in a temperature 1 system, all glue bonds are individually enough to cause two tiles to bind. Algorithmic self-assembly requires growth to begin from a very particular state, usually from either a seed assembly (see Sect. 5.1) which is large enough to provide a location where one or more tiles can bind via two attachments to it, or a set of “hard-coded” input tiles that can bind to each other via sufficiently strong bonds to form an assembly that can then function like a larger seed assembly. From a carefully defined input and using cooperative attachments, tiles of relatively few different types can combine in complex algorithmic patterns with many copies of each tile type appearing throughout the growing assembly. However, in a temperature 1 system, growth can be initiated apart from any seed assembly with pairs of tiles “nucleating” growth that can then proceed to follow patterns corresponding to arbitrary subsets of algorithmic growth continuing from random inputs. This spuriously nucleated, unstructured algorithmic growth leads to the formation of “junk” structures and is therefore a fatal flaw for such systems. Great experimental work using cooperativity to control nucleation has been done in [61], and a schematic representation of their results (using both single-stranded DNA tiles and DNA origami-based tiles) is shown in Fig. 9. By using two planes in the third dimension, the “crisscross slat” tiles are able to extend further than square tiles and bind to a greater number of neighboring slats when attaching. Future work leveraging such expanded cooperative growth to prevent spurious nucleation, especially across wider temperature ranges, may improve the ability to control seeding in algorithmic systems.

Although “temperature 2” growth in experimental systems using glue cooperation helps restrict algorithmic growth to beginning from designated seeds, it requires careful design of glue domain strengths and careful control of actual system temperature and tile concentrations. Even so, these systems still suffer from growth errors, even after previously mentioned proofreading techniques are incorporated. Additionally, while weakly cooperative temperature 1 systems also allow for algorithmic growth and have the potential for reducing growth errors by more actively preventing attachments of incorrect tiles, they instead suffer more greatly from the problem of spurious nucleation. In order to realize the full potential of algorithmic self-assembly, systems designed to incorporate both types of cooperative behavior may be useful. For instance, if tile motifs could be designed such that geometric hindrance occurs in the case of glue mismatches while also providing glues to be used for glue cooperation enforced by temperature 2, future designs utilizing DNA origami-based tiles (e.g., [62]), or perhaps clever designs of smaller complexes, may have the potential to move forward the state of the art in algorithmic self-assembly. Additionally, experimental implementation of temperature 2 growth requires either “double-sized tiles” (such as in [10] where double-sized tiles are effectively two square tiles permanently bound together, allowing growth to extend outward from one row of growth into a new row) or the design of sets of glues with carefully separated groups representing strength 1 and strength 2 glues (as previously discussed in Sect. 5.1), and advances in sequence design would help in this effort. Yet another

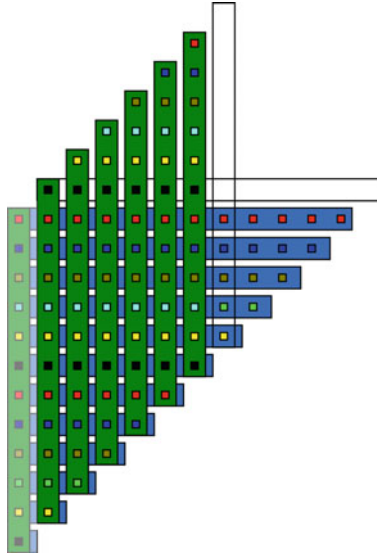


Fig. 9 Crisscross slat tiles (green and blue rectangles) assembling with a binding threshold of 6, which enforces high levels of cooperativity. Binding domains are shown as small colored squares (and although they are actually on the side of the green slats facing the blue slats, they are shown for clarity). Each attaching slat must bind to at least 6 others. The empty horizontal outline shows where the next slat attachment is possible, and the vertical outline shows where there will then be 6 domains allowing for the attachment of the following slat, leading to a new location for a horizontal slat, etc.

potential direction for advancement may come from further development of proof-reading techniques and error suppression mechanisms, which have already proven to be very useful.

6.2 *Single Tile or Hierarchical Growth*

The aTAM and models derived from it are based on dynamics of single-tile attachment, i.e., at each step of the assembly process, a single-tile monomer attaches to a growing assembly. An alternative to this allows assemblies of arbitrary size (i.e., composed of arbitrary numbers of individual tiles) to combine with each other. This is often modeled as *hierarchical assembly* in which a system begins self-assembly from a collection of individual “singleton” tiles which can combine with each other in pairs, and then those assemblies can combine with each other, etc., allowing up to a doubling of assembly size with each combination. A commonly studied theoretical model of this process is called the Two-Handed Assembly Model (2HAM) as it is based upon the intuition that one already produced assembly could be taken in each hand, and the pair could then be combined to form a new, larger assembly.

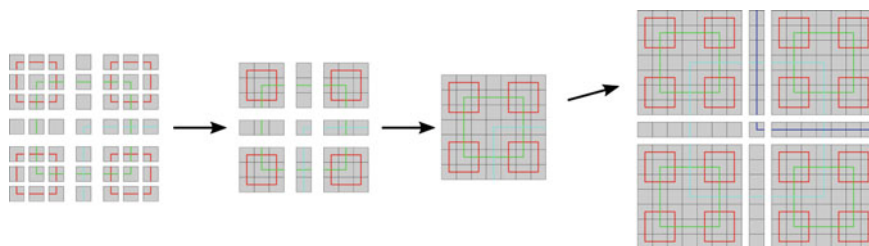


Fig. 10 Abstract example of hierarchical assembly (showing a portion of the infinite Robinson tiling pattern). From left to right, smaller assemblies combine to form larger assemblies, which then combine with each other to form the next hierarchical level

Hierarchical assembly occurs in biology (i.e., the constituent pieces of amino acids combine, then those amino acids are combined to form proteins, and the proteins then combine to form cellular structures) and has even been cleverly demonstrated in DNA-based experimental systems (e.g., [40, 63, 64]). Theoretical work has shown that, in general, 2HAM systems are capable of making a greater diversity of structures and utilizing lower tile complexity than systems in the aTAM [65]. However, somewhat counterintuitively, in [66] it was proven that (under physically realistic assumptions based on molecular counts applied to the abstract 2HAM) no asymptotic speedup is actually achievable over single-tile growth. Nonetheless, the 2HAM remains a very interesting model in which the dynamics allow for the theoretical designs of systems which efficiently (in terms of tile complexity) produce complex shapes. System design in these theoretical constructions tends to make heavy use of geometric hindrance, where the interfaces along which pairs of assemblies may bind have carefully designed patterns of “bumps” and “dents” that allow for great discrimination between which pairs of assemblies can bind to each other, while allowing the numbers of unique glue domains to remain very low (often a relatively small constant number across constructions capable of targeting any particular structure among an infinite collection). This has been demonstrated in theoretical results [67–69] as well as experimentally [62, 63, 70] (Fig. 10).

For future experimental work to implement additional theoretical constructions, a wide variety of improvements will most likely be necessary. To leverage the use of geometric hindrance, it will be necessary for assemblies to remain rigid, at least along binding surfaces, but in many constructions those interfaces may be quite long. Without sufficient rigidity, portions of the assembly which should block the attachment of incorrect assemblies may bend to allow those attachments. Prior experimental work with hierarchical assembly [40] showed a relatively sharp drop-off of the rates of correct completion of steps of the assembly process. This seriously restricts the potential complexity of designed systems and efforts to improve that would be valuable. For instance, as the previously mentioned theoretical work of [66] showed, a roadblock to the assembly of later steps can be the multitude of assemblies of earlier steps (complete and/or incomplete) that simultaneously exist in solution. As steps progress, the number of assembly types, or species, quickly grows since not all growth progresses

at the same rate, and this makes the likelihood of a pair of complementary assemblies of a later step encountering each other drop precipitously. Future improvements in the ability to relatively quickly, easily, and correctly purify the products of various steps may allow for a higher concentration of correctly matching assemblies from the same step and allow assembly to progress correctly at higher rates.

6.3 Activatable/Deactivatable Glues

In the aTAM and many similar models, the tiles are “static,” meaning they can be thought of as components whose properties do not change once they bind to an assembly or at any time afterward. However, many DNA-based nanotechnologies are based largely upon dynamic reactions such as strand displacement [71–73]. When strand displacement mechanics are incorporated into tile-based self-assembly, it is possible to make tiles whose binding domains turn “on” and “off.” This has been experimentally prototyped [74] and theoretically modeled [75–77], with tiles developed such that the binding of one glue on a tile can cause other glue domains on that tile to either become “active” (i.e., they were previously sequestered but then uncovered) or “inactive” (i.e., they go from either bound or able to bind to being sequestered such that they can no longer bind and any bond they previously formed with another tile is broken). See Fig. 11 for an example.

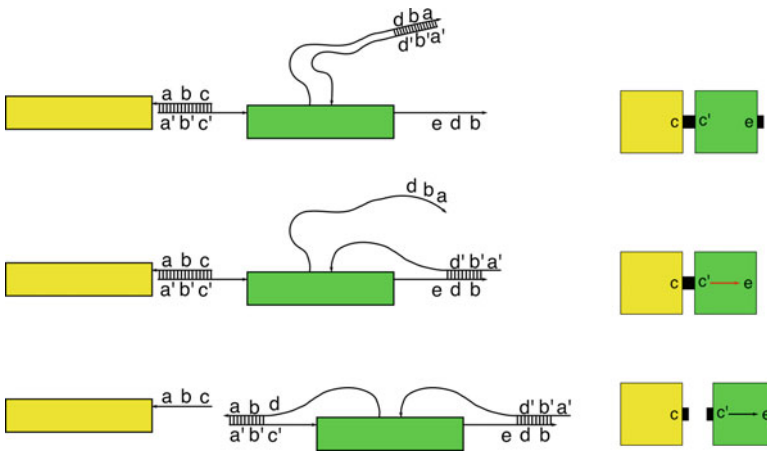


Fig. 11 Example of a “signal-passing” tile in two levels of abstraction. Assembly proceeds from the bottom to the top. On the left, the green tile initially has its left glue partially sequestered, but the unpaired c' domain provides an additional matching domain that allows it to bind more favorably with the glue on the yellow tile. That causes the previously bound strand to detach, which is then able to bind with the top strand on the right side, exposing the right-side glue. On the right, a further abstraction depicts the same process: The e glue of the green tile is initially “inactive”. When the c' glue binds, it causes the “firing” of the signal which eventually “activates” the e glue

Theoretical constructions with so-called *signal-passing tiles* have shown that not only can they self-assemble structures whose shapes are impossible to self-assemble with static tiles (e.g., the discrete self-similar fractal called the Sierpinski triangle cannot self-assemble in the aTAM [78], but it can self-assemble using signal-passing tiles [75]), but they can also perform universal computation without requiring the assembly to be as large as the product of the time and space requirements of the computation. That is, with static tiles, all steps of the computation must be permanently represented within the final assembly, so an assembly in which a computation occurs using n bits in each of m computation steps requires $n \cdot m$ tiles to be permanently attached. However, with signal-passing tiles it is possible for the glues of tiles to deactivate after they have participated in a step of a computation and thus for tiles to detach after facilitating a computational step and for assemblies to remain smaller while performing computations. The demonstrated theoretical power of systems of signal-passing tiles is in several ways greater than that of the aTAM, and although many constructions make use of tiles which have high signal complexity (i.e., many signal pathways across the same tile), theoretical work has also shown that by scaling up target shapes [79], signal complexity can often be brought down to only 2 signals, allowing for relatively simple tiles to exhibit the greatly enhanced power of signal passing.

Although some experimental work has been done with signal-passing tiles [74], in that work, only a single signal passed across each tile and glue deactivations were not used. In order to expand the use of signals, larger tile motifs will likely be required, but (small) DNA origami structures could potentially provide a good platform. The process of passing signals from one glue to another when the first binds could be implemented using techniques similar to those of “surface chemical reaction networks” [80–82] where strand displacement cascades are used to transmit the signals. Although the complexity of individual tiles implemented in this way would be much greater than simple single-stranded tile (SST) motifs (i.e., tile designs which use a single strand of DNA per tile), if even an additional fraction of the algorithmic control possible with signal-passing tiles could be realized that increased complexity has the potential to be justified. Furthermore, using DNA origami as the tile body also provides the potential for integrating geometric hindrance as a tool, adding even more control and error suppression to algorithmic growth.

6.4 *Tile Removal and Breaking of Assemblies*

The ability for tiles that previously joined together in an assembly to detach from each other at designated points allows for not only new dynamics but also for new categories of targeted behaviors. For instance, it becomes possible to develop theoretical systems which take as input a structure that already has the desired shape and then to produce assemblies having that same shape [41, 43], or to replicate patterns encoded into assemblies [42, 53]. It also becomes possible to design theoretical systems capable of attaching to the perimeters of input assemblies if and only if they

match a particular shape [44]. Experimental work has even succeeded in showing how the fracturing of assemblies can serve as the basis for the replication of patterns [53].

Theoretical models that allow for glue detachment include signal-passing tiles (see Sect. 6.3), the melting of subsets of weaker glue bonds via increased temperatures (see Sect. 5.4), and the dissolution of a subset of tiles within an assembly (for instance, in systems with tiles made of both DNA and RNA, the RNA-based tiles could be dissolved via an RNase enzyme [43]).

Development of systems leveraging the additional possibilities enabled by tile detachment and the breaking apart of assemblies will require overcoming the hurdles discussed for robustly implementing signal-passing tiles or temperature programming, or techniques such as incorporating RNA-based tiles into systems with DNA tiles and successfully dissolving them while leaving the DNA tiles intact. Also, for many of the theoretical constructions, greater control of hierarchical self-assembly will be required.

6.5 *Reconfiguration Via Flexibility*

When cellular machinery builds the wide variety of proteins encoded by genes, even though only a small number of amino acids are used as the building blocks, the diversity of protein shape and functionality that results are astonishing. Since we know the sequences of the genes and their mappings to the amino acid sequences, it may seem that it should be easy to predict those properties of proteins. However, as amino acids are attached one at a time, the forming chain folds upon itself in a complex three-dimensional pattern influenced by several types of molecular interactions. This process turns out to be computationally intractable to predict in general [83]. In contrast, DNA origami utilizes a rational design approach toward folding which starts with the desired shape to self-assemble and then develops a routing path for a scaffold strand that can then be folded into that path by staple strands.

Following nature's example, a cotranscriptional approach to utilizing folding with tiles based on RNA has been developed [84, 85]. A generalization of this process has been captured in the theoretical model called *oritatami* [86] (see Fig. 12 for an example), which has been shown to allow for universal computation [87] and have strong shape-building abilities [88]. While RNA seems to be the natural medium for such systems, perhaps some future DNA-based work could use related techniques.

A different approach has been taken by theoretical models [89, 90] which have been developed to at least partially mimic and capture similar folding behaviors, and unsurprisingly, it is intractable to compute most interesting properties of systems in these models, even despite their more discrete nature. For instance, tiles in the Flexible Tile Assembly Model (FTAM) [89] are considered to be rigid bodies, but they are allowed to have flexible bonds with their neighbors. The physical inspiration for the theoretical FTAM is the way that protein folding can allow chains of amino acids to rapidly explore possible configurations and adopt those that are (relatively) optimal.

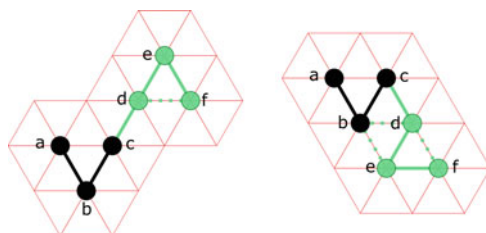


Fig. 12 Example of an oritatami system, where a chain of “beads” is transcribed, one at a time. (In this case, from *a* to *f*.) Each bead type can be designed to be able to form bonds with some subset of other bead types. As beads are transcribed, the chain folds (on the triangular grid, a portion of which is shown in red) to form a maximal number of bonds (shown here as dashed green lines). On the left, the chain has not folded to maximize bonds, but on the right it has



Fig. 13 Example of a reconfigurable assembly in the flexible tile assembly model. The glues between square tiles are flexible, similar to hinges, and allow the flat structure on the left to fold into the hollow box on the right

For reconfigurations that are not excessively large, it seems likely that this process of reconfiguration and exploration can proceed more quickly than bimolecular reactions which require the diffusion of new monomers for attachment, and that perhaps even displacement and reconfiguration of previously bound subassemblies may be possible to engineer, enabling shape-changing assemblies (Fig. 13).

A potential DNA-based implementation could achieve flexible bonds between tiles by including unpaired nucleotides on one or both sides of glues which have bound (with the bound portions forming rigid helices). This could allow for bound tiles or even subassemblies to change positions relative to other portions of an assembly, and thus, it may be possible to design algorithmic self-assembling systems which form reconfigurable assemblies that can be designed to first take one shape and can then reconfigure into a differently shaped assembly by the addition of just a few strands that displace targeted glue strands, or different environmental signals such as the concentration of a particular molecule (like MgCl_2 as was demonstrated experimentally in [91]) or pH (as shown experimentally in [92]) (Fig. 14).

6.6 Assembly Growth Controlled by CRNs

Chemical reaction networks (CRNs) are composed of sets of reactions, each of which has a set of reactant chemical species that react to produce a set of product chemical

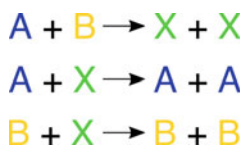


Fig. 14 Example CRN whose input is a set of A and B molecules. Each molecular species is denoted by a different letter. Each reaction is specified by a different line. For example, in the first reaction one copy of an A molecule combines with one copy of a B molecule to produce two copies of the X molecule. This CRN computes which input is in the majority. The system evolves to its output by increasing the count of the majority input while decreasing that of the minority input, since the reactant in the majority will be more likely to interact with both the X s and the other reactant

species. A set of such reactions which are chained together by having the outputs of one reaction act as the inputs of another can define a network capable of complex behaviors. Theoretical work has shown that arbitrary CRNs can be implemented as sets of DNA complexes [93] and that has led to an entire branch of DNA nanotechnology based upon the design of artificial CRNs, including programming languages that compile digital circuits into DNA complexes [94]. While the goal of such systems is typically centered around the integration of computing logic with chemical and/or biological systems rather than structure building, there has also been research which ties the two together. Although tile assembly can also be described by chemical reactions that model the combination of an assembly and a tile to form a larger assembly, the geometry of the forming structure helps define which tiles may attach. Also, tiles are neither transformed or consumed (at least in models such as the aTAM). More general CRNs do not consider geometry of structures and also allow for reactants to be consumed and/or converted into other species (while perhaps also consuming “fuel” species and creating “waste” species). The combination of DNA-based implementations of these more general CRNs with tile assembly systems (theoretically [95–97] and experimentally [98]) provides the ability to have the growth of assemblies controlled by computations performed by a set of general CRN reactions that can be based upon time delays, the presence or lack of specific inputs, or even feedback based on the growth of assemblies themselves by adjusting concentrations and/or counts of tiles used during the assembly process. The “signals” produced by a CRN in this case are global in nature, potentially influencing any or all of the assemblies growing in parallel, while the control provided by the signals of signal-passing tiles (see Sect. 6.3) is local in nature, impacting only the growth of the assembly on which a signal is initiated.

As the development of both DNA-based CRNs and tile-based self-assembly systems continues to mature, there is great potential for control of structure-forming systems by CRNs whose input can be delivered by a wide array of mechanisms, including (but not limited to) the presence of targeted molecules in the environment. Combined with reconfigurable assemblies (see Sect. 6.5), systems could be designed to release cargoes, expose previously sequestered functionalized surfaces, or perform other environmentally responsive behaviors.

7 Conclusion

We have summarized a wide variety of theoretical models of self-assembling systems that were primarily developed to provide high-level mathematical abstractions and give insights into the effects of varying aspects of components (e.g., sizes, shapes, rigidity, binding affinities, etc.) and model dynamics (e.g., methods of growth and/or breaking of assemblies, cooperativity, etc.). Some of these insights have already provided guidance to experimental designs, and we hope the models will continue to evolve and mature alongside the design and engineering techniques of DNA nanotechnology. Theoretical modeling can provide a framework that shows which properties of components and systems are needed for desired resultant behaviors and guide researchers in the right direction as they work to develop new molecular components and techniques. Additionally, it can serve as a foundation to categorize potential behaviors of newly developed components and dynamical behaviors made possible in the laboratory.

There is a symbiotic relationship between theory and experiment, and thus it also remains important that theory incorporates up-to-date knowledge of experimental roadblocks and challenges, which can then be used for the development of new models and theoretical studies. The rapid growth and great success of DNA nanotechnology have been achieved in part due to strong ties between theory and experiment, and conferences like the “International Conference on DNA Computing and Molecular Programming” [99] have been integral in building and maintaining this connection. We look forward to seeing where future developments will lead and are optimistic that many of the powers of self-assembling systems displayed within the theoretical domain will be realized in physical systems, and this theoreticians' toolkit for building self-assembly systems will come closer to reality.

Acknowledgements The “DNA community” is a truly amazing group of scientists, mentors, teachers, and friends. Introduced as a new Ph.D. student, I was immediately welcomed and patiently taught, and my (often outlandish) theoretical musings have been graciously tolerated and often cleverly refined and improved upon. The number of people in this community that have helped me and contributed to my work is immense, and although I do not have space to list them all, I wish to thank them. As guiding members, Natasha Jonoska and Erik Winfree have provided examples that I will continually strive to emulate, and I humbly thank them for the invitation to contribute to this volume. I am profoundly excited to see where this community will lead science and engineering in the next 40 years and to continue meeting and working with everyone in it.

References

1. O. Bournez, A. Pouly, A survey on analog models of computation, in *Handbook of Computability and Complexity in Analysis* (Springer, 2021), pp. 173–226
2. M. Savchuk, A. Fesenko, Quantum computing: survey and analysis. *Cybern. Syst. Anal.* **55**(1), 10–21 (2019)
3. S. Navlakha, Z. Bar-Joseph, Distributed information processing in biological and computational systems. *Commun. ACM* **58**(1), 94–102 (2014)

4. J.D. Watson, F.H. Crick, The structure of DNA, in *Cold Spring Harbor Symposia on Quantitative Biology*, vol. 18 (Cold Spring Harbor Laboratory Press, 1953), pp. 123–131
5. L.M. Adleman, Molecular computation of solutions to combinatorial problems. *Science* **266**, 1021–1024 (1994)
6. E. Winfree, *Algorithmic Self-Assembly of DNA*. Ph.D. thesis (California Institute of Technology, June 1998)
7. L. Kari, G. Păun, G. Rozenberg, A. Salomaa, S. Yu, DNA computing, sticker systems, and universality. *Acta Informatica* **35**(5), 401–420 (1998)
8. Y. Ke, L.L. Ong, W.M. Shih, P. Yin, Three-dimensional structures self-assembled from DNA bricks. *Science* **338**(6111), 1177–1183 (2012)
9. P.W.K. Rothmund, Folding DNA to create nanoscale shapes and patterns. *Nature* **440**, 297–302 (2006)
10. R.D. Barish, R. Schulman, P.W.K. Rothmund, E. Winfree, An information-bearing seed for nucleating algorithmic self-assembly. *Proc. Nat. Acad. Sci.* **106**, 6054–6059 (2009)
11. E.S. Andersen, M. Dong, M.M. Nielsen, K. Jahn, R. Subramani, W. Mamdouh, M.M. Golas, B. Sander, H. Stark, C.L.P. Oliveira, J.S. Pedersen, V. Birkedal, F. Besenbacher, K.V. Gothelf, J. Kjems, Self-assembly of a nanoscale DNA box with a controllable lid. *Nature* **459**, 73–76 (2009)
12. L. Qian, E. Winfree, Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**(6034), 1196–1201 (2011)
13. B. Wang, C. Thachuk, A.D. Ellington, E. Winfree, D. Soloveichik, Effective design principles for leakless strand displacement systems. *Proc. Nat. Acad. Sci.* **115**(52), E12182–E12191 (2018)
14. N.C. Seeman, C. Mao, T.H. LaBean, J.H. Reif, Logical computation using algorithmic self-assembly of DNA triple-crossover molecules. *Nature* **407**, 493–496 (2000)
15. Y.-J. Chen, B. Groves, R.A. Muscat, G. Seelig, DNA nanotechnology from the test tube to the cell. *Nat. Nanotechnol.* **10**(9), 748–760 (2015)
16. B. Groves, Y.-J. Chen, C. Zurla, S. Pochekailov, J.L. Kirschman, P.J. Santangelo, G. Seelig, Computing in mammalian cells with nucleic acid strand exchange. *Nat. Nanotechnol.* **11**(3), 287–294 (2016)
17. Y. Amir, E. Ben-Ishay, D. Levner, S. Ittah, A. Abu-Horowitz, I. Bachelet, Universal computing by DNA origami robots in a living animal. *Nat. Nanotechnol.* **9**(5), 353–357 (2014)
18. P.M. Nafisi, T. Aksel, S.M. Douglas, Construction of a novel phagemid to produce custom DNA origami scaffolds. *Synthetic Biol.* **3**, 08 (2018)
19. A.R. Chandrasekaran, M. Pushpanathan, K. Halvorsen, Evolution of DNA origami scaffolds. *Mater. Lett.* **170**, 221–224 (2016)
20. J. Bush, S. Singh, M. Vargas, E. Oktay, C.-H. Hu, R. Veneziano, Synthesis of DNA origami scaffolds: current and emerging strategies. *Molecules* **25**(15), 3386 (2020)
21. A.R. Chandrasekaran, R. Zhuo, A ‘tile’ tale: hierarchical self-assembly of DNA lattices. *Appl. Mater. Today* **2**, 7–16 (2016)
22. H. Yan, S.H. Park, G. Finkelstein, J.H. Reif, T.H. LaBean, DNA-templated self-assembly of protein arrays and highly conductive nanowires. *Science* **301**(5641), 1882–1884 (2003)
23. P.W.K. Rothmund, E. Winfree, The program-size complexity of self-assembled squares (extended abstract), in *STOC’00: Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing* (ACM, Portland, Oregon, United States, 2000), pp. 459–468
24. D. Soloveichik, E. Winfree, Complexity of self-assembled shapes. *SIAM J. Comput.* **36**(6), 1544–1569 (2007)
25. D. Doty, J.H. Lutz, M.J. Patitz, R.T. Schweller, S.M. Summers, D. Woods, The tile assembly model is intrinsically universal, in *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science* (FOCS 2012), pp. 302–310
26. L. Adleman, Q. Cheng, A. Goel, M.-D. Huang, Running time and program size for self-assembled squares, in *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, (Hersonissos, Greece, 2001), pp. 740–748

27. M. Arita, A. Nishikawa, M. Hagiya, K. Komiya, H. Gouzu, K. Sakamoto, Improving sequence design for DNA computing, in *Proceedings of the 2nd Annual Conference on Genetic and Evolutionary Computation*, 2000, pp. 875–882
28. D. Woods, D. Doty, C. Myhrvold, J. Hui, F. Zhou, P. Yin, E. Winfree, Diverse and robust molecular algorithms using reprogrammable DNA self-assembly. *Nature* **567**, 366–372 (2019)
29. C.G. Evans, E. Winfree, Physical principles for DNA tile self-assembly. *Chem. Soc. Rev.* **46**(12), 3808–3829 (2017)
30. C.G. Evans, E. Winfree, DNA sticky end design and assignment for robust algorithmic self-assembly, in *DNA Computing and Molecular Programming—19th International Conference, DNA 19, Tempe, AZ, USA, September 22–27, 2013. Proceedings*, eds. by D. Soloveichik, B. Yurke, vol. 8141. *Lecture Notes in Computer Science* (Springer, 2013), pp. 61–75
31. J.N. Zadeh, C.D. Steenberg, J.S. Bois, B.R. Wolfe, M.B. Pierce, A.R. Khan, R.M. Dirks, N.A. Pierce, NUPACK: analysis and design of nucleic acid systems. *J. Comput. Chem.* **32**(1), 170–173 (2011)
32. R. Lorenz, S.H. Bernhart, C.H. Zu Siederdissen, H. Tafer, C. Flamm, P.F. Stadler, I.L. Hofacker, ViennaRNA package 2.0, *Algorithms for Molecular Biology*, vol. 6, no. 1, 2011, pp. 1–14
33. Y. Zhang, A. Reinhardt, P. Wang, J. Song, Y. Ke, Programming the nucleation of DNA brick self-assembly with a seeding strand. *Angewandte Chemie Int. Edn.* **59**(22), 8594–8600 (2020)
34. M.-Y. Kao, R.T. Schweller, Randomized self-assembly for approximate shapes, in *ICALP (1)*, eds. by L. Aceto, I. Damgård, L.A. Goldberg, M.M. Halldórsson, A. Ingólfssdóttir, I. Walukiewicz, vol. 5125. *Lecture Notes in Computer Science* (Springer, 2008), pp. 370–384
35. D. Doty, Randomized self-assembly for exact shapes. *SIAM J. Comput.* **39**(8), 3521–3552 (2010)
36. M.-Y. Kao, R.T. Schweller, Reducing tile complexity for self-assembly through temperature programming, in *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2006)*, Miami, Florida, Jan 2006, pp. 571–580, 2007
37. S.M. Summers, Reducing tile complexity for the self-assembly of scaled shapes through temperature programming. *Algorithmica* **63**, 117–136 (2012)
38. E.D. Demaine, M.L. Demaine, S.P. Fekete, M. Ishaque, E. Rafalin, R.T. Schweller, D.L. Souvaine, Staged self-assembly: nanomanufacture of arbitrary shapes with $O(1)$ glues. *Nat. Comput.* **7**(3), 347–370 (2008)
39. C.T. Chalk, E. Martinez, R.T. Schweller, L. Vega, A. Winslow, T. Wylie, Optimal staged self-assembly of general shapes. *Algorithmica* **80**(4), 1383–1409 (2018)
40. G. Tikhomirov, P. Petersen, L. Qian, Fractal assembly of micrometre-scale DNA origami arrays with arbitrary patterns. *Nature* **552**(7683), 67–71 (2017)
41. J. Hendricks, M.J. Patitz, T.A. Rogers, Replication of arbitrary hole-free shapes via self-assembly with signal-passing tiles, in *Unconventional Computation and Natural Computation—14th International Conference, UCNC 2015, Auckland, New Zealand, Aug 30–Sept 3, 2015, Proceedings*, eds. by C.S. Calude, M.J. Dinneen, vol. 9252. *Lecture Notes in Computer Science* (Springer, 2015), pp. 202–214
42. A. Keenan, R. Schweller, X. Zhong, Exponential replication of patterns in the signal tile assembly model. *Nat. Comput.* **14**(2), 265–278 (2014)
43. Z. Abel, N. Benbernou, M. Damian, E.D. Demaine, M.L. Demaine, R. Flatland, S.D. Kominers, R.T. Schweller, Shape replication through self-assembly and RNase enzymes, in *SODA 2010: Proceedings of the Twenty-first Annual ACM-SIAM Symposium on Discrete Algorithms* (Society for Industrial and Applied Mathematics, Austin, Texas, 2010), pp. 1045–1064
44. M.J. Patitz, S.M. Summers, Identifying shapes using self-assembly. *Algorithmica* **64**(3), 481–510 (2012)
45. A. Winslow, Size-separable tile self-assembly: a tight bound for temperature-1 mismatch-free systems. *Nat. Comput.* **15**(1), 143–151 (2016)
46. D. Doty, M.J. Patitz, S.M. Summers, Limitations of self-assembly at temperature 1. *Theor. Comput. Sci.* **412**, 145–158 (2011)

47. P. Meunier, D. Regnault, D. Woods, The program-size complexity of self-assembled paths, in *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22–26, 2020*, eds. by K. Makarychev, Y. Makarychev, M. Tulsiani, G. Kamath, J. Chuzhoy (ACM, 2020), pp. 727–737
48. D. Soloveichik, M. Cook, E. Winfree, Combining self-healing and proofreading in self-assembly. *Nat. Comput.* **7**(2), 203–218 (2008)
49. D. Soloveichik, E. Winfree, Complexity of compact proofreading for self-assembled patterns, in *DNA Computing, 11th International Workshop on DNA Computing, DNA11, London, ON, Canada, June 6–9, 2005. Revised Selected Papers*, eds. by A. Carbone, N.A. Pierce, vol. 3892. Lecture Notes in Computer Science (Springer, 2005), pp. 305–324
50. E. Winfree, R. Bekbolatov, Proofreading tile sets: error correction for algorithmic self-assembly, in *DNA Computing, 9th International Workshop on DNA Based Computers, DNA9, Madison, WI, USA, June 1–3, 2003, Revised Papers*, eds. by J. Chen, J.H. Reif, vol. 2943. Lecture Notes in Computer Science (Springer, 2003), pp. 126–144
51. H.-L. Chen, A. Goel, Error free self-assembly using error prone tiles, in *10th International Workshop on DNA Computing, DNA10*, eds. by C. Ferretti, G. Mauri, C. Zandron, vol. 3384. LNCS (Springer Verlag, 2005), pp. 62–75
52. K. Fujibayashi, D.Y. Zhang, E. Winfree, S. Murata, Error suppression mechanisms for DNA tile self-assembly and their simulation. *Nat. Comput.* **8**(3), 589–612 (2009)
53. R. Schulman, B. Yurke, E. Winfree, Robust self-replication of combinatorial information via crystal growth and scission. *Proc. Nat. Acad. Sci.* **109**(17), 6405–10 (2012)
54. D. Hader, M.J. Patitz, Geometric tiles and powers and limitations of geometric hindrance in self-assembly, in *Unconventional Computation and Natural Computation—18th International Conference, UCNC 2019, Tokyo, Japan, June 3–7, 2019, Proceedings*, eds. by I. McQuillan, S. Seki, vol. 11493. Lecture Notes in Computer Science (Springer, 2019), pp. 191–204
55. B. Fu, M.J. Patitz, R.T. Schweller, R. Sheline, Self-assembly with geometric tiles, in *Automata, Languages, and Programming—39th International Colloquium, ICALP 2012, Warwick, UK, July 9–13, 2012, Proceedings, Part I*, eds. by A. Czumaj, K. Mehlhorn, A.M. Pitts, R. Wattenhofer, vol. 7391. LNCS (Springer, 2012), pp. 714–725
56. J. Hendricks, M.J. Patitz, T.A. Rogers, S.M. Summers, The power of duples (in self-assembly): It’s not so hip to be square. *Theor. Comput. Sci.* **743**, 148–166 (2018)
57. M.J. Patitz, R.T. Schweller, S.M. Summers, Exact shapes and Turing universality at temperature 1 with a single negative glue, in *DNA Computing and Molecular Programming—17th International Conference, DNA 17, Pasadena, CA, USA, September 19–23, 2011. Proceedings*, eds. by L. Cardelli, W.M. Shih, vol. 6937. Lecture Notes in Computer Science (Springer, 2011), pp. 175–189
58. D. Doty, L. Kari, B. Masson, Negative interactions in irreversible self-assembly. *Algorithmica* **66**(1), 153–172 (2013)
59. S.P. Fekete, J. Hendricks, M.J. Patitz, T.A. Rogers, R.T. Schweller, Universal computation with arbitrary polyomino tiles in non-cooperative self-assembly, in *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2015), San Diego, CA, USA, January 4–6, 2015*, pp. 148–167
60. O. Gilbert, J. Hendricks, M.J. Patitz, T.A. Rogers, Computing in continuous space with self-assembling polygonal tiles, in *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2016), Arlington, VA, USA, January 10–12, 2016*, pp. 937–956
61. D. Mineev, C.M. Wintersinger, A. Ershova, W.M. Shih, Robust nucleation control via crisscross polymerization of highly coordinated DNA slats. *Nat. Commun.* **12**(1), 1–9 (2021)
62. M. Endo, T. Sugita, Y. Katsuda, K. Hidaka, H. Sugiyama, Programmed-assembly system using DNA jigsaw pieces. *Chem. Euro. J.* 5362–5368 (2010)
63. T. Gerling, K.F. Wagenbauer, A.M. Neuner, H. Dietz, Dynamic DNA devices and assemblies formed by shape-complementary, non base-pairing 3D components. *Science* **347**(6229), 1446–1452 (2015)

64. C. Pistol, C. Dwyer, Scalable, low-cost, hierarchical assembly of programmable DNA nanostructures. *Nanotechnology* **18**(12), 125305 (2007)
65. S. Cannon, E.D. Demaine, M.L. Demaine, S. Eisenstat, M.J. Patitz, R.T. Schweller, S.M. Summers, A. Winslow, Two hands are better than one (up to constant factors): self-assembly in the 2HAM vs. aTAM, in *STACS*, eds. by N. Portier, T. Wilke, vol. 20. LIPIcs (Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2013), pp. 172–184
66. H.-L. Chen, D. Doty, Parallelism and time in hierarchical self-assembly, in *SODA 2012: Proceedings of the 23rd Annual ACM-SIAM Symposium on Discrete Algorithms* (SIAM, 2012), pp. 1163–1182
67. E.D. Demaine, M.J. Patitz, T.A. Rogers, R.T. Schweller, S.M. Summers, D. Woods, The two-handed tile assembly model is not intrinsically universal. *Algorithmica* **74**, 812–850 (2016)
68. J. Hendricks, J. Opseth, Self-assembly of 4-sided fractals in the two-handed tile assembly model, in *Proceedings of the 16th Annual Conference on Unconventional Computation and Natural Computation (UCNC 2017)*, Fayetteville, Arkansas, USA, June 5–9, 2017, pp. 113–128
69. D. Hader, M.J. Patitz, Geometric tiles and powers and limitations of geometric hindrance in self-assembly. *Nat. Comput.* **20**, 243–258 (2021)
70. S. Woo, P.W.K. Rothmund, Programmable molecular recognition based on the geometry of DNA nanostructures. *Nat. Chem.* **3**, 620–627 (2011)
71. L. Qian, E. Winfree, J. Bruck, Neural network computation with DNA strand displacement cascades. *Nature* **475**(7356), 368–372 (2011)
72. C. Thachuk, E. Winfree, D. Soloveichik, Leakless DNA strand displacement systems, in *International Workshop on DNA-Based Computers* (Springer, 2015), pp. 133–153
73. D.Y. Zhang, G. Seelig, Dynamic DNA nanotechnology using strand-displacement reactions. *Nat. Chem.* **3**(2), 103–113 (2011)
74. J.E. Padilla, R. Sha, M. Kristiansen, J. Chen, N. Jonoska, N.C. Seeman, A signal-passing DNA-strand-exchange mechanism for active self-assembly of DNA nanostructures. *Angewandte Chemie Int. Edn.* **54**, 5939–5942 (2015)
75. J.E. Padilla, M.J. Patitz, R.T. Schweller, N.C. Seeman, S.M. Summers, X. Zhong, Asynchronous signal passing for tile self-assembly: Fuel efficient computation and efficient assembly of shapes. *Int. J. Found. Comput. Sci.* **25**(4), 459–488 (2014)
76. N. Jonoska, D. Karpenko, Active tile self-assembly, part 1: universality at temperature 1. *Int. J. Found. Comput. Sci.* **25**(02), 141–163 (2014)
77. N. Jonoska, D. Karpenko, Active tile self-assembly, part 2: self-similar structures and structural recursion. *Int. J. Found. Comput. Sci.* **25**(02), 165–194 (2014)
78. J.I. Lathrop, J.H. Lutz, S.M. Summers, Strict self-assembly of discrete Sierpinski triangles. *Theor. Comput. Sci.* **410**, 384–405 (2009)
79. T. Fochtmann, J. Hendricks, J.E. Padilla, M.J. Patitz, T.A. Rogers, Signal transmission across tile assemblies: 3D static tiles simulate active self-assembly by 2D signal-passing tiles. *Nat. Comput.* **14**(2), 251–264 (2015)
80. S. Clamons, L. Qian, E. Winfree, Programming and simulating chemical reaction networks on a surface. *J. R. Soc. Interface* **17**(166), 20190790 (2020)
81. L. Qian, E. Winfree, Parallel and scalable computation and spatial dynamics with DNA-based chemical reaction networks on a surface, in *DNA Computing and Molecular Programming—20th International Conference, DNA 20, Kyoto, Japan, September 22–26, 2014. Proceedings*, eds. by S. Murata, S. Kobayashi, vol. 8727. Lecture Notes in Computer Science (Springer, 2014), pp. 114–131
82. H. Bui, S. Shah, R. Mokhtar, T. Song, S. Garg, J. Reif, Localized DNA hybridization chain reactions on DNA origami. *ACS Nano* **12**(2), 1146–1155 (2018)
83. A.S. Fraenkel, Complexity of protein folding. *Bullet. Math. Biol.* **55**(6), 1199–1210 (1993)
84. C. Geary, P.W.K. Rothmund, E.S. Andersen, A single-stranded architecture for cotranscriptional folding of RNA nanostructures. *Science* **345**(6198), 799–804 (2014)
85. C. Geary, G. Grossi, E.K. McRae, P.W. Rothmund, E.S. Andersen, RNA origami design tools enable cotranscriptional folding of kilobase-sized nanoscaffolds, in *Nature Chemistry*, 2021, pp. 1–10

86. C. Geary, P.-É. Meunier, N. Schabanel, S. Seki, Oritatami: a computational model for molecular co-transcriptional folding. *Int. J. Mole. Sci.* **20**(9), 2259 (2019)
87. C. Geary, P.-É. Meunier, N. Schabanel, S. Seki, Proving the Turing universality of Oritatami co-transcriptional folding, in *Proceedings of the 29th International Symposium on Algorithms and Computation, ISAAC 2018*, Jiaoxi, Yilan, Taiwan, December 16–19, 2018, pp. 23:1–23:13
88. E.D. Demaine, J. Hendricks, M. Olsen, M.J. Patitz, T.A. Rogers, N. Schabanel, S. Seki, H. Thomas, Know when to fold'em: self-assembly of shapes by folding in oritatami, in *DNA Computing and Molecular Programming—24th International Conference, DNA 24, Jinan, China, October 8–12, 2018, Proceedings*, eds. by D. Doty, H. Dietz, vol. 11145. LNCS (Springer, 2018), pp. 19–36
89. J. Durand-Lose, J. Hendricks, M.J. Patitz, I. Perkins, M. Sharp, Self-assembly of 3-D structures using 2-D folding tiles, in *DNA Computing and Molecular Programming—24th International Conference, DNA 24, Jinan, China, October 8–12, 2018, Proceedings*, eds. by D. Doty, H. Dietz, vol. 11145. Lecture Notes in Computer Science (Springer, 2018), pp. 105–121
90. N. Jonoska, G.L. McColm, Complexity classes for self-assembling flexible tiles. *Theor. Comput. Sci.* **410**, 332–346 (2009)
91. T. Gerling, K.F. Wagenbauer, A.M. Neuner, H. Dietz, Dynamic DNA devices and assemblies formed by shape-complementary, non-base pairing 3D components. *Science* **347**(6229), 1446–1452 (2015)
92. T. Liedl, F.C. Simmel, Switching the conformation of a DNA molecule with a chemical oscillator. *Nano Lett.* **5**(10), 1894–1898 (2005)
93. D. Soloveichik, G. Seelig, E. Winfree, DNA as a universal substrate for chemical kinetics. *Proc. Nat. Acad. Sci.* **107**(12), 5393–5398 (2010)
94. A. Phillips, L. Cardelli, A programming language for composable DNA circuits. *J. R. Soc. Interface* **6**(suppl_4), S419–S436 (2009)
95. D.Y. Zhang, R.F. Hariadi, H.M. Choi, E. Winfree, Integrating DNA strand-displacement circuitry with DNA tile self-assembly. *Nat. Commun.* **4**, 1–10 (2013)
96. N. Schiefer, E. Winfree, Universal computation and optimal construction in the chemical reaction network-controlled tile assembly model, in *DNA Computing and Molecular Programming—21st International Conference, DNA 21, Boston and Cambridge, MA, USA, August 17–21, 2015. Proceedings*, eds. by A. Phillips, P. Yin, vol. 9211. Lecture Notes in Computer Science (Springer, 2015), pp. 34–54
97. T.H. Klinge, J.I. Lathrop, S. Moreno, H.D. Potter, N.K. Raman, M.R. Riley, ALCH: an imperative language for chemical reaction network-controlled tile assembly, in *Natural Computing*, 2022, pp. 1–21
98. N. Schiefer, E. Winfree, Time complexity of computation and construction in the chemical reaction network-controlled tile assembly model, in *DNA Computing and Molecular Programming—22nd International Conference, DNA 22, Munich, Germany, September 4–8, 2016, Proceedings*, eds. by Y. Rondelez, D. Woods, vol. 9818. Lecture Notes in Computer Science (Springer, 2016), pp. 165–182
99. I. Kawamata, International conference on DNA computing and molecular programming. <http://www.dna-computing.org>, 2022 [online; accessed 22 Feb 2022]

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Reasoning As If



Jack H. Lutz and Robyn R. Lutz

Abstract It is occasionally useful to *reason as if* something were true, even when we know that it is almost certainly not true. We discuss two instances, one in distributed computing and one in tile self-assembly, and suggest directions for further investigation of this method.

1 Introduction

Great breakthroughs in science have a way of perforating traditional disciplinary boundaries. Ned Seeman's development of structural DNA nanotechnology [1, 2] is a case in point. Over the past forty years, researchers from most disciplines of science and engineering have been drawn into the thrilling creativity of this new field. This is most obviously motivated by opportunities for applying methods from many disciplines to the development of DNA nanotechnology. Often, however, benefits of such applications also accrue back to the disciplines from which they come, as the novel aspects of DNA technology force improvements of these methods and our understanding of them.

In this note, we discuss an example of this phenomenon, the use of a method from distributed computing theory [3] in the theory of DNA tile self-assembly [4–6], perhaps the earliest abstract theory to emerge from DNA nanotechnology. Both of these subjects deal with systems that evolve non-deterministically over time and are so complex that it is not feasible to explore all the individual trajectories along which they might evolve. In the case of distributed computing, the example of interest here consists of a large number of processors that asynchronously and non-deterministically pass messages to one another. In the case of tile self-assembly, the example of interest consists of a large two-dimensional structure that grows via the asynchronous, non-deterministic adsorption of various types of tiles (abstractions of

J. H. Lutz (✉) · R. R. Lutz (✉)
Iowa State University, Ames, IA 50011, USA
e-mail: lutz@iastate.edu

R. R. Lutz
e-mail: rlutz@iastate.edu

© The Author(s) 2023
N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*,
Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_15

DNA double-crossover molecules [7]) onto various locations along the boundary of the structure. In both examples, the population (number of processors or number of tiles) is assumed to be very large, and the number of trajectories of the system is far larger than its population.

The method that we discuss, which we call *reasoning as if*, has not been precisely formalized, and we will not succeed in formalizing it here. Our goal is simply to discuss two striking examples of the method, one in distributed computing and one in tile self-assembly, and to suggest directions for future research on the method.

The rest of this note is organized as follows. In Sect. 2 we review the *snapshot algorithm* of Chandy and Lamport [8], emphasizing the reasoning-as-if nature of this algorithm. In Sect. 3 we review the *local determinism* method of Soloveichik and Winfree [9], again emphasizing the reasoning-as-if nature of the method. In Sect. 4 we sketch research directions that we hope will lead to the further development of as-if reasoning as a useful method.

2 The Snapshot Algorithm

The snapshot algorithm was designed by Chandy and Lamport [8], and a colorful description of it, which we follow here, was provided by Dijkstra [10]. In it the algorithm assembles a “snapshot” of a possible but unlikely global system state in order to reason about the properties of the system. What makes this algorithm relevant to our focus here is that the system snapshot that is assembled is almost certainly not real. It is an imaginary state that probably never occurred in the system’s execution. Its power lies in its uncanny and provable mirroring of the global properties of the system *as if* the snapshot were real.

Chandy and Lamport use the analogy of photographers watching a sky filled with migrating birds to explain the algorithm [8]. The scene is vast, and the birds are in constant motion—no single photo suffices. Only a composite photo from snapshots taken by different photographers at different times and then thoughtfully pieced together can capture the whole scene.

Similarly, the snapshot algorithm uses *as-if reasoning* to enable us to determine properties of the global state of a distributed system. As the algorithm executes, it assembles a description of a snapshot state. When it terminates, we can query the snapshot state. Any stable predicate that holds in the assembled snapshot state holds for the system. An example of a stable predicate, from [11], is “the number of tokens traveling the network equals 7.” A stable predicate is one that, once it holds, holds forever.

A network is represented as a directed, finite, strongly connected graph in which each machine is a node and each edge is a first-in first-out buffer. A computation is specified by a sequence of atomic actions by individual machines. Each action updates the machine’s state, accepts at most one message on each of its input buffers, and sends at most one message on each of its output buffers. A global system state is determined by the state of each machine and the messages in each buffer.

The snapshot state is assembled from a snapshot of each machine, taken locally and at different times, and the assembled snapshot state is a fiction. However, any stable predicate that is true for the snapshot state is true for the system's final state.

Coloring. Initially, all nodes (machines) and edges (messages) are white. At termination, all machines and messages are red. Each machine turns red once, which Dijkstra calls “blushing,” sending white messages before turning red and red messages after turning red [11]. No red message is ever accepted by a white machine.

Assembling the Snapshot State. The snapshot state consists of the state of each machine as it turns red and the white messages accepted by the machine after it turns red. We designate one machine as the seed. It turns itself red and sends a special message, called a Marker, on each of its output buffers. It then records any white messages it receives while red. It knows that its local snapshot is complete when it has received a Marker on each of its input buffers. Each subsequent machine, when it receives a Marker on an input buffer, similarly turns red and sends a Marker on each of its output buffers. Since each machine is reachable, all machines turn red in finite time.

Rewriting History. The algorithm rewrites history—“by magic,” according to Dijkstra—so that the snapshot state is the point at which all machines turn red. To do this, whenever there is a red action followed by a white action, they must be from different machines, and it interchanges them, iterating until all white actions precede all red actions. The snapshot state is the cut in this rewritten history, that is, the point at which all subsequent actions are red and all prior actions are white. The snapshot algorithm thus uses *local snapshots* taken at each machine to construct an *imaginary but possible global state* that enables us to determine system properties *as if* the snapshot state were real.

Uses. In the nearly 30 years since Dijkstra saluted the snapshot algorithm as “beautiful,” it has been widely used to check global properties in distributed systems, e.g., a computation terminating, as well as for monitoring and debugging [8, 12]. For example, checkpointing and rollback recovery are key mechanisms of fault-tolerant distributed systems [13]. They enable systems that crash to recover and resume execution from a stored state rather than needing to restart from scratch. In checkpointing, each device in the system stores its local state, so that in rollback recovery after a failure, the node's state can be restored. Nakamura et al. point out that checkpoint-rollback recovery “inherently contains a snapshot algorithm to record the nodes' checkpoints in such a way that they compose a consistent global state” [14].

3 Local Determinism

The abstract Tile Assembly Model (aTAM) is an idealized model of molecular self-assembly on a two-dimensional surface. Winfree et al. [15] introduced the first form of the model, which was based on DNA double-crossover molecules [1] and was already Turing universal (i.e., able to simulate arbitrary computations). Winfree [4] developed the model further, and Rothmund and Winfree [5, 6] refined it to its

present-day form. The aTAM has been an active area of research in the molecular programming community ever since.

A *tile type* in the aTAM is a unit square that cannot be rotated, so that it has well-defined “north,” “west,” “south,” and “east” sides. Each of these sides is labeled with a *glue type* that we may take to be a non-negative integer. Each glue type has a *strength* that is 0, 1, or 2.

A *tile assembly system* (TAS) is an ordered pair $\mathcal{T} = (T, s)$, where T is a finite set of tile types and $s \in T$ is the *seed tile type*. An *assembly sequence* of \mathcal{T} is a finite or infinite sequence

$$\alpha = (\alpha_0, \alpha_1, \alpha_2, \dots)$$

of *assemblies* α_i satisfying the following conditions.

1. α_0 is the assembly consisting of a single tile of type s .
2. For each i , if the assembly sequence α is long enough for α_{i+1} to exist, then α_{i+1} is obtained by adding a single tile of some type $t \in T$ to the assembly α_i . Moreover, this tile is added by abutting it to one or more tiles of α_i in such a way that the sum of the strengths of the glues on the tile whose types match the glue types of α_i that they abut is at least 2.
3. If $\alpha = (\alpha_0, \dots, \alpha_i)$ is finite, then the assembly α_i is *terminal* in the sense that no tile of any type $t \in T$ can be added to α_i as in condition 2.
4. If $\alpha = (\alpha_0, \alpha_1, \alpha_2, \dots)$ is infinite, then it is (*strongly*) *fair* in the sense of distributed computing [16, 17], which means that, if a tile can be added to an assembly α_i at some location l , then there is some $j \geq i$ such that α_{j+1} is obtained from α_j by adding a tile at location l .

Several things should be noted about the above definition. First, in any assembly sequence $\alpha = (\alpha_0, \alpha_1, \alpha_2, \dots)$, each assembly α_i consists of exactly $i + 1$ tiles. Second, condition 3 requires an assembly sequence α to “go on for as long as it can.” Third, once tiles are added to an assembly, they do not move. This ensures that every assembly sequence α has a well-defined *result*, which is an assembly denoted $\text{res}(\alpha)$. If $\alpha = (\alpha_0, \dots, \alpha_i)$ is finite, then $\text{res}(\alpha)$ is the finite assembly α_i . If $\alpha = (\alpha_0, \dots, \alpha_i)$ is infinite, then $\text{res}(\alpha)$ is the minimal infinite assembly that has each α_i as a subassembly in the obvious sense. Fourth, the fairness condition 4 ensures that the result of an infinite assembly sequence must, like the result of a finite assembly sequence, be terminal in the sense defined in condition 3.

Many aTAM investigations involve *design problems*. In such a problem, there is a target assembly α^* , and the task is to design a tile assembly system $\mathcal{T} = (T, s)$ such that *every* assembly sequence α of \mathcal{T} has α^* as its result. Moreover, α^* is often very large, and it is typically desirable to (i) have the number $|T|$ of tile types in \mathcal{T} be much smaller than the number of tiles in α^* and (ii) exploit the inherent parallelism of chemistry by having it be typical for an assembly α_{i+1} in an assembly sequence α for α^* to be just one of many assemblies that could have been obtained by adding a tile to α_i . These two things together imply that a solution $\mathcal{T} = (T, s)$ for the design problem of causing α^* to reliably self-assemble typically spawns an exceedingly large number of assembly sequences α , all with result α^* .

In fact, anyone who has designed non-trivial tile assembly systems knows all too well how easy it is to design a tile assembly system \mathcal{T} in which the target assembly α^* results from many or most, but not all assembly sequences α of \mathcal{T} . When erroneous designs of this type occur, it is typically because the designer envisions a particular assembly sequence α with result α^* , but some alternative assembly sequence α' with a different result can occur (often because of a “race condition” in which growth in α' can block envisioned growth in α).

The local determinism method of Soloveichik and Winfree [9] is a verification method that is also a beautiful solution to the above design problem. The key to this method is a definition of what it means for an assembly sequence α of a tile assembly system \mathcal{T} to be *locally deterministic*. The details of this definition need not concern us here except to note its key properties. The definition is subtle, but it is simple enough that it is typically routine to verify that a given assembly sequence α is locally deterministic (if it is). The definition does *not* require the underlying tile assembly system \mathcal{T} to be deterministic: A locally deterministic assembly sequence α is typically just one of a huge number of assembly sequences that can occur in \mathcal{T} .

The remarkable main theorem about local determinism is that, if α is a locally deterministic assembly sequence of a tile assembly system \mathcal{T} , then *every* assembly sequence α' of \mathcal{T} has the same result, i.e., satisfies $\text{res}(\alpha') = \text{res}(\alpha)$ [9].

This local determinism theorem says that a designer of a tile assembly system \mathcal{T} for a target assembly α^* who envisions a particular tile assembly sequence α of \mathcal{T} with result α^* may safely *reason as if* α is the assembly sequence that occurs, even if the likelihood of α occurring is vanishingly small, provided only that the designer also verifies that α is locally deterministic. Since it is natural for designers to think in terms of an envisioned assembly sequence (“First this substructure will assemble, then this one . . .”), local determinism is a very designer-friendly method. Moreover, this friendliness arises directly from the fact that the method entitles the designer to *reason as if* a convenient, envisioned, and unlikely assembly sequence occurs.

To put the associated verification problem fancifully, if one wants to prove that all roads (assembly sequences) lead to Rome (the target assembly), it suffices to exhibit a fancy (locally deterministic) road and show that this leads to Rome.

But even more is true. Soloveichik and Winfree’s proof also shows that, if there is a fancy road, then all roads are fancy. That is, if a tile assembly system \mathcal{T} has a locally deterministic sequence α , then *all* assembly sequences of \mathcal{T} are locally deterministic. The method is thus even more designer friendly than indicated above, since *any* assembly sequence that the designer chooses will be locally deterministic (if some assembly sequence is).

4 The Future of As If

Structural DNA nanotechnology opened up the fascinating field of molecular programming. The challenges of this new world have pushed participating computer scientists to develop new techniques for dealing with complex interactions of geometry

and non-determinism at scales surpassing prior computing experience and rivaling those of the coming Internet of Things. One of these techniques, local determinism, is an instance of *reasoning as if*, a method rarely used in science and still not well understood. Here, we have reviewed local determinism together with the snapshot algorithm, an earlier instance of as-if reasoning, in the hope of promoting investigation of this method.

We briefly suggest future directions for research on as-if reasoning in molecular programming. First, can local determinism be usefully adapted from the abstract Tile Assembly Model to the other models of molecular programming? Obvious candidates here include other models of tile self-assembly such as the kinetic Tile Assembly Model (kTAM) [4] and the Two-Handed Assembly Model (2HAM) [18]. More ambitious possibilities include chemical reaction networks (CRNs) [19], the CRN-directed Tile Assembly Model (CRN-TAM) [20, 21], and thermodynamic binding networks [22].

More general questions also present themselves. Without delving too deeply into philosophy [23], can we more clearly formalize as-if reasoning? A concrete start would be to formalize what it is that the snapshot algorithm and local determinism have in common.

Finally, local determinism is, roughly speaking, a condition that allows us to reason from the *possibility* of a target structure self-assembling to the *necessity* of that target structure self-assembling. Modal logic, originally developed to reason about possibility and necessity [24], has become a powerful tool in distributed computing [25, 26]. Could it provide a means of formalizing—and strengthening—as-if reasoning?

In his book, Ned graciously wrote of molecular programming researchers that “more than any other, this community has contributed valuable ideas and workers to the field of structural DNA nanotechnology,” and that “this parallel field is one of the key drivers of structural DNA nanotechnology” [2]. We are delighted to work in this parallel field, which would not exist without Ned, but we know that he is in the driver’s seat, and that is the way we want it.

Acknowledgements We thank Matt Cook, Robbie Schweller, and an anonymous reviewer for useful and stimulating remarks. This work was supported in part by National Science Foundation grants 1545028 and 1900716.

References

1. N.C. Seeman, Nucleic acid junctions and lattices. *J. Theor. Biol.* **99**(2), 237–247 (1982)
2. N.C. Seeman, *Structural DNA Nanotechnology* (Cambridge University Press, 2015)
3. A.S. Tanenbaum, M. Van Steen, *Distributed Systems: Principles and Paradigms* (Prentice-Hall, 2007)
4. E. Winfree, *Algorithmic Self-Assembly of DNA*. Ph.D. thesis (California Institute of Technology, 1998)

5. P.W.K. Rothmund, E. Winfree, The program-size complexity of self-assembled squares (extended abstract), in *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, May 21–23, 2000, Portland, OR, USA*, eds. by F.F. Yao, E.M. Luks (ACM, 2000), pp. 459–468
6. P.W.K. Rothmund, *Theory and Experiments in Algorithmic Self-assembly*. Ph.D. thesis (University of Southern California, 2001)
7. T.J. Fu, N.C. Seeman, DNA double-crossover molecules. *Biochemistry* **32**(13), 3211–3220 (1993)
8. K.M. Chandy, L. Lamport, Distributed snapshots: determining global states of distributed systems. *ACM Trans. Comput. Syst.* **3**(1), 63–75 (1985)
9. D. Soloveichik, E. Winfree, Complexity of self-assembled shapes. *SIAM J. Comput.* **36**(6), 1544–1569 (2007)
10. E.W. Dijkstra, *The Distributed Snapshot of K.M. Chandy and L. Lamport* (E.W. Dijkstra Archive, 1983)
11. E.W. Dijkstra, *The Distributed Snapshot of Chandy/Lamport/Misra* (E.W. Dijkstra Archive, 1983)
12. F. Mattern, Efficient algorithms for distributed snapshots and global virtual time approximation. *J. Parallel Distrib. Comput.* **18**(4), 423–434 (1993)
13. A.D. Kshemkalyani, M. Singhal, *Distributed Computing: Principles, Algorithms, and Systems* (Cambridge University Press, 2011)
14. J. Nakamura, Y. Kim, Y. Katayama, T. Masuzawa, A cooperative partial snapshot algorithm for checkpoint-rollback recovery of large-scale and dynamic distributed systems and experimental evaluations. *Concurr. Comput. Pract. Experience* **33**(12), e5647 (2021)
15. E. Winfree, X. Yang, N.C. Seeman, Universal computation via self-assembly of DNA: some theory and experiments, in *Proceedings of the 2nd DIMACS Workshop on DNA Based Computers*, vol. 44 (AMS, 1998), pp. 191–213
16. M.Z. Kwiatkowska, Survey of fairness notions. *Inf. Softw. Technol.* **31**(7), 371–386 (1989)
17. C. Baier, J.-P. Katoen, *Principles of Model Checking* (MIT Press, 2008)
18. S. Cannon, E.D. Demaine, M.L. Demaine, S. Eisenstat, M.J. Patitz, R.T. Schweller, S.M. Summers, A. Winslow, Two hands are better than one (up to constant factors): self-assembly in the 2HAM vs. aTAM, in *30th International Symposium on Theoretical Aspects of Computer Science*, 2013, p. 172
19. M. Feinberg, *Foundations of Chemical Reaction Network Theory* (Springer, 2019)
20. N. Schiefer, E. Winfree, Universal computation and optimal construction in the chemical reaction network-controlled tile assembly model, in *DNA Computing and Molecular Programming*, eds. by A. Phillips, P. Yin (Springer International Publishing, Cham, 2015), pp. 34–54
21. N. Schiefer, E. Winfree, Time complexity of computation and construction in the chemical reaction network-controlled tile assembly model, in *Proceedings of the 22nd International Conference on DNA Computing and Molecular Programming* (Springer International Publishing, 2016), pp. 165–182
22. D. Doty, T.A. Rogers, D. Soloveichik, C. Thachuk, D. Woods, Thermodynamic binding networks, in *DNA Computing and Molecular Programming*, eds. by R. Brijder, L. Qian (Springer International Publishing, Cham, 2017), pp. 249–266
23. H. Vaihinger, *The Philosophy of As If* (Routledge, 1924)
24. E. Lemmon, D. Scott, *An Introduction to Modal Logic* (Blackwell, 1977)
25. R. Fagin, J.Y. Halpern, Y. Moses, M.Y. Vardi, *Reasoning About Knowledge* (MIT Press, 1995)
26. Y. Shoham, K. Leyton-Brown, *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations* (Cambridge University Press, 2008)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Biochemical Circuits

Scaling Up DNA Computing with Array-Based Synthesis and High-Throughput Sequencing



Yuan-Jyue Chen and Georg Seelig

Abstract It was 40 years ago today, when Ned taught DNA to play [32]. When Ned Seeman began laying the theoretical foundations of what is now DNA nanotechnology, he likely did not imagine the entire diversity and scale of molecular structures, machines, and computing devices that would be enabled by his work. While there are many reasons for the success of the field, not least the creativity shown by Ned and the community he helped build, such progress would not have been possible without breakthroughs in DNA synthesis and molecular analysis technology. Here, we argue that the technologies that will enable the next generation of DNA nanotechnology have already arrived but that we have not yet fully taken advantage of them. Specifically, we believe that it will become possible, in the near future, to dramatically scale up DNA nanotechnology through the use of array-synthesized DNA and high-throughput DNA sequencing. In this article, we provide an example of how DNA logic gates and circuits can be produced through enzymatic processing of array-synthesized DNA and can be read out by sequencing in a massively parallel format. We experimentally demonstrate processing and readout of 380 molecular gates in a single reaction. We further speculate that in the longer term, very large-scale DNA computing will find applications in the context of molecular diagnostics and, in particular, DNA data storage.

1 Introduction

Over the last four decades, structural DNA nanotechnology became intertwined with and provided a foundation for a range of other fields, most notably dynamic DNA nanotechnology and DNA computing. Adleman's original demonstration of DNA computing took advantage of the inherent parallelism of chemical reactions and the

Y.-J. Chen (✉)

Microsoft, Redmond, WA, USA
e-mail: yuanjc@microsoft.com

G. Seelig (✉)

University of Washington, Seattle, WA, USA
e-mail: gseelig@uw.edu

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*, Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_16

281

predictability of DNA base pairing to solve an instance of the traveling salesman problem [1]. However, it soon became apparent that DNA computing would not be able to compete with electronic computers because of the exponentially large amounts of DNA required for solving problems out of reach of conventional computers [11]. Researchers subsequently explored alternative computing paradigms, from tile-assembly systems, an area that Seeman made foundational contributions to [3, 29, 34, 43], to Boolean logic gates [25, 31, 37, 38], neural networks [13, 20, 27, 42, 45] or chemical reaction networks [6, 7, 33]. Importantly, the goal of this more recent work is not to compete with electronics but to efficiently process information that is already available in molecular form or to use embedded computation to organize molecules into shapes and patterns.

1.1 Scaling up DNA Computing for Molecular Diagnostics

The potential of DNA computing to analyze information encoded in biological nucleic acids, in particular single-stranded RNA, was recognized early on [5, 31, 37]. Benenson et al. developed a molecular automaton that could perform a computation where the outcome (the release of an antisense drug mimic) was dependent on the absence or presence of specific inputs (ssDNA with sequence analogous to diagnostically relevant mRNA) [5]. In our own recent work, we constructed a DNA-based molecular classifier that assigns samples to a class (bacterial or viral infection) based on the levels of seven distinct mRNA mimics [21]. Zhang et al. extended this work by including an amplification step that made it possible to work with actual patient samples rather than synthetic or *in vitro* transcribed RNA thus bringing molecular computation for diagnostics closer to a practical application [44]. However, classification accuracy could be further improved by considering more input features (i.e., distinct mRNA). In fact, an optimal *in silico* classifier designed to solve the same problem as our molecular classifier takes into account the levels of over two hundred distinct transcripts [41]. Moreover, *in silico* classifiers designed to solve more complex computational problems such as identifying a cancer tissue of origin routinely take thousands of genes as inputs [22, 28]. It is easy to imagine that future diagnostic applications of molecular computation could benefit from technologies that would allow the construction and analysis of circuits with thousands of gates.

1.2 Scaling up DNA Computing for DNA Data Storage

Digital data will soon outgrow available storage capacity and novel approaches for data storage are required. DNA is a promising material for digital data storage because of its high information density and durability (reviewed in Ref. [9]). Documented suggestions for the use of DNA for digital data storage go as far back as the 1960s [23], but only recent improvements in DNA synthesis and sequencing have started making

DNA storage practical [14, 17, 18, 24]. The amount of data stored in DNA increased from 0.7 MB in 2012 to 200 MB in 2018 and will keep growing at an accelerating pace given the interest in this technology from both industry and academia. However, having large volumes of data stored in DNA creates new challenges, namely the need to access and search these data or perform other forms of information processing. Sequencing all stored data and converting it back to electronic form for processing are impractical and will not scale. A molecular computation approach for performing data analysis directly in the chemical realm, i.e., “near data,” before converting results to electronic form, is a more promising alternative [8]. However, although some problems such as random access [24], image similarity search [4, 36], image preview [40] and some types of Boolean search [2] can be mapped to hybridization reactions and thus be implemented at relatively large scale already, other tasks such as image classification will likely require more complex computational primitives and orders of magnitude more computational elements than are currently feasible [13].

1.3 Limitations of Current Approaches to DNA Computing

Almost all DNA circuitry built to date has been assembled from individually column-synthesized oligonucleotides, because this technology provides a low synthesis error rate and control over the concentration of individual strands. Having a low rate of deletions, insertions or substitutions are desirable because such errors can result in side reactions (i.e., “leaks”) once the strand is incorporated into a gate [12, 35, 46]. Control over concentration is necessary because most DNA gates are assembled from multiple strands with a defined stoichiometry. However, at a cost of \$1,000 or so for 96 oligos synthesized in a 96-well plate format, and the need to manipulate oligos individually, any approach based on ordering column-synthesized oligos cannot scale.

A second, related challenge for scaling up DNA circuits is the time-consuming step of gate assembly and purification. Currently, each gate complex needs to be assembled separately; otherwise signal strand supposed to be bound to an upstream gate may bind to downstream gate instead. Moreover, gate complexes often need to be gel purified to remove excess strands or partial complexes, which can cause undesired leak reactions and computation error.

A third limitation to scaling up DNA circuit computation comes from the use of fluorescence-based reporters for reading out the results of a computation. Typical commercial fluorometers typically read up to four distinct fluorescence channels and a similar number of samples in parallel. Plate readers allow monitoring of up to 96 reactions in parallel but are typically limited to reading no more than two fluorescence channels simultaneously. These limitations mean that only a small number of variables can be monitored in a given computation (i.e., as many as there are independent fluorescence channels), and only, a limited number of computations can be performed in parallel (i.e., as many as there are separate reaction chambers).

Other limitations such as low reaction speed will be briefly discussed in the discussion, but addressing them is not the focus of the work presented here.

2 A Vision for the Future

Pools of array-synthesized DNA (reviewed in [19]) provide a promising alternative to individually column-synthesized DNA because cost per oligo is orders of magnitude lower and pools with over a million distinct sequences are now available commercially. Such very large-scale and (relatively) low-cost pools provided the foundation for breakthroughs in DNA data storage [14, 17, 24] but so far, array-synthesized oligos have not been used for molecular programming applications because of the lower synthesis quality, variation in concentration between oligos, and the low yield of each individual oligo in the pool.

Similarly, next-generation DNA sequencing can be used to read out hundreds of millions of DNA sequences in parallel but so far has not been used to read out DNA computations because current gate architectures are not compatible with read out by sequencing. Specifically, whether an output strand was released from a gate or is still hybridized to it is unlikely to impact whether it is detected in a sequencing reaction and thus naive sequencing of a DNA circuit reaction mix cannot reveal the result of the computation.

Building on these technologies is a necessity if we are to scale up the complexity and size of DNA nanostructures and circuits. However, doing so will require novel gate and structural architectures that are compatible with low abundance and low quality DNA. We expect that enzymatic processing will be important in order to selectively amplify the array-synthesized DNA and process it into functional gates or other devices.

We note that the idea to use array-synthesized DNA for DNA computing is at least a decade old. Qian and Winfree outlined how a Seesaw gate could be made through enzymatic processing of a single DNA hairpin synthesized on an array [26]. The proposed approach ensures correct stoichiometry of the two components strands in the final gate and also provides some degree of sequence proof-reading due to the enzymatic cleavage step. However, gate concentration is bounded by that of intact oligonucleotides in the pool and performance is likely limited by leak reactions involving gates derived from truncated or otherwise imperfect oligonucleotides.

Here, we introduce an alternative approach for making general purpose DNA logic gates from array-derived DNA. Our approach uses PCR amplification to selectively amplify full length but low abundance DNA. We further show how DNA sequencing can be used to read out the result of a computation thus providing a path toward reading out a very large number of gate operations in parallel (Fig. 1).

3 Results

In this section, we first introduce our logic gate design and reaction mechanism and then explain how gates can be processed from single-stranded DNA through PCR

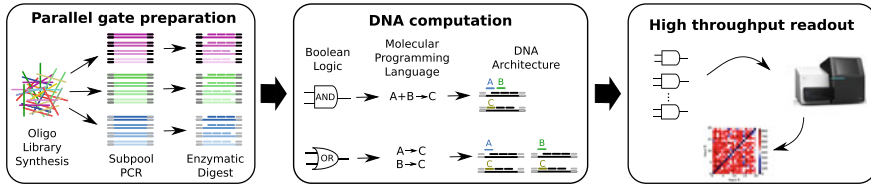


Fig. 1 Overview of circuit construction, computation, and read out. Left: DNA gates are derived from large pools of oligonucleotides through PCR amplification and enzymatic processing. Middle: computation occurs in a pooled reaction upon addition of inputs. Right: end point results of the Boolean logic computation are read out in parallel using next-generation sequencing (NGS)

amplification and nicking enzyme digestion. We then provide experimental results for gate operation and demonstrate readout of up to 380 gates in a single reaction using DNA sequencing.

3.1 Nicked Double-Stranded DNA Gates Reaction Mechanism

We use a modified version of the nicked double-stranded DNA (ndsDNA) gate architecture proposed by Cardelli [7]. The reaction mechanism for a single-input, single-output gate, a logical repeater or sequence translator, is detailed in Fig. 2. The input and output have the same domain architecture with a short toehold followed by a longer recognition domain. Each complete logic gate consists of a join and a fork gate. Several auxiliary species, referred to as helper strands, are also required for the reaction. Inputs bind to the join gate and trigger a sequence of strand displacement reactions that result in the release of an intermediate output strand. This intermediate output then reacts with the fork gate, triggering a second sequence of reactions that finally result in the release of the output. The two-component join-fork architecture and use of helper strands ensure that the input and output sequence are completely unrelated and have the same domain architecture. The gate architecture is modular, and gates with multiple inputs and outputs can easily be designed.

3.2 Gate Design

A key advantage of the ndsDNA gate architecture is that each gate can be derived from fully double-stranded DNA through enzymatic processing. While in our earlier work we used plasmid DNA as a starting material in order to minimize error rate, we here propose to generate the double-stranded DNA by PCR amplification of single-stranded DNA synthesized on an array. We will make two modifications to the architecture introduced in Ref. [12]. First, we will ensure that all toeholds

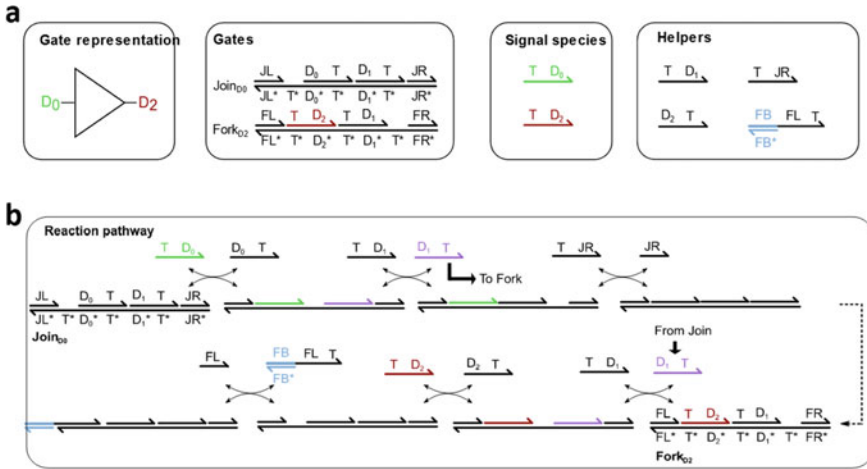
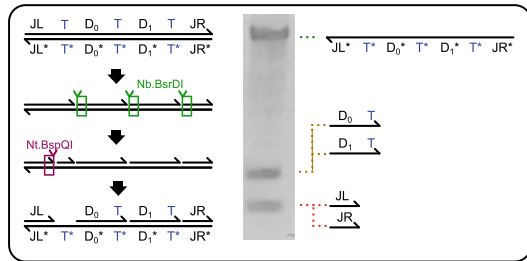


Fig. 2 DNA components (a) and reaction mechanism (b) for a one-input, one-output gate. The repeater gate consists of a join and fork component and a set of helper strands. Inputs and outputs are referred to as signal species. Here, we propose to derive join and fork gates from DNA pools through enzymatic processing while inputs and helpers are synthetic DNA. The gate architecture is similar to that described in Ref. [12] with two important exceptions: first, all toeholds are internal, meaning that they are flanked by double-stranded domains on both sides. This change was made to ensure that all toeholds have similar binding energies. Second, double-stranded domains (light and dark blue) appended to some of the helper strands are used for selective PCR and sequencing of reacted gate species as detailed in the text

Fig. 3 Enzymatic processing steps used for gate preparation. Functional domains and nicking enzyme binding sites are indicated. The denaturing gel data show that bands corresponding to DNA oligonucleotides of the expected length for a correctly processed gate



have the same sequence and are “internal,” i.e., they are flanked by double-stranded domains on both sides (see the leftmost toehold in Fig. 3 for an example of an internal toehold). As a result, all toeholds should have very similar binding energies and all individual strand displacement reactions should occur at similar rates. Second, and more importantly, we modify the gate architecture to enable read out of gate activation by sequencing. Specifically, we add a double-stranded domain (shown in blue or light blue in Fig. 2) to the final helper strand for both the join and fork gate. These domains do not participate in the strand displacement reactions but are appended to the gate complex when the reaction has completed. These domains can be detected by sequencing and can be used to distinguish reacted from unreacted gates.

A third important point is that we operate in a different regime than that described in Ref. [12]. Specifically, we assume that inputs are in excess of the gates such that the gates are either fully triggered or completely off. Thus, while these gates were initially designed to approximate a desired analog (chemical kinetics) behavior, we here treat them as digital logic gates.

3.3 *Making ndsDNA Gates from Array-Synthesized DNA*

The workflow of gate generation is detailed in Fig. 3. An initial gate strand is chemically synthesized including flanking sequences that can be used for PCR amplification. Using matching primers, this strand is amplified to generate double-stranded DNA. The resulting duplex DNA is then processed with nicking enzymes that generate breaks in the top strand of the gate and also generate an initiating toehold.

By using PCR and nicking enzymes to generate each join and fork gate from a single strand, we overcome two major issues. First, PCR amplification is only successful if both primer sequences are present. Therefore, the PCR reaction will preferentially amplify full-length molecules over a background of fractional synthesis products. Moreover, most multi-stranded gate architectures require careful control over stoichiometry, which is currently impossible to achieve with array-synthesized DNA. Using only a single strand to generate the entire gate overcomes this limitation. Moreover, all strands that are synthesized with the same primer sequences will be amplified together. By using a common set of primers for all gates that belong to the same functional module, we can amplify and then process all components of a circuit of interest in a one-pot reaction.

Figure 3 shows an example of join gate processing. In this case, the initial duplex was generated by PCR of a single column-synthesized oligo. Enzymatic digestion was performed as outlined on the left of the figure. Denaturing gel electrophoresis was used to show that the digestion results in digestion products of the expected length.

3.4 *Characterizing Gate Kinetics*

Before turning to a readout by sequencing, we will characterize the behavior of individual gates and small circuits using traditional fluorescence-based assays. For an initial test, we generated an AND logic gate (both join and fork gates) starting from column-synthesized oligos. Upon PCR amplification and enzymatic digestion, we first characterized gate behavior using non-denaturing electrophoresis (Fig. 4, left). We then triggered the gate using either both or only one input and measured the rate of output release using a fluorescent reporter, following the protocol detailed in Ref. [12]. The kinetics data show correct AND logic behavior (Fig. 4, right).

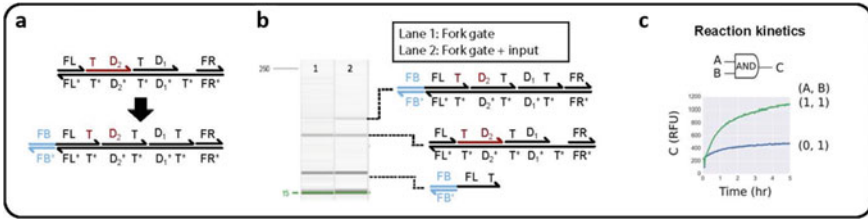


Fig. 4 DNA AND logic gate. **a** Fork gate configuration before and after reaction. **b** Native gel assay shows the expected DNA species for the fork gate before and after reaction. **c** Fluorescence kinetics data for the complete AND gate (fork + join + helpers) confirm that signal is high only if both inputs are present as expected for AND logic

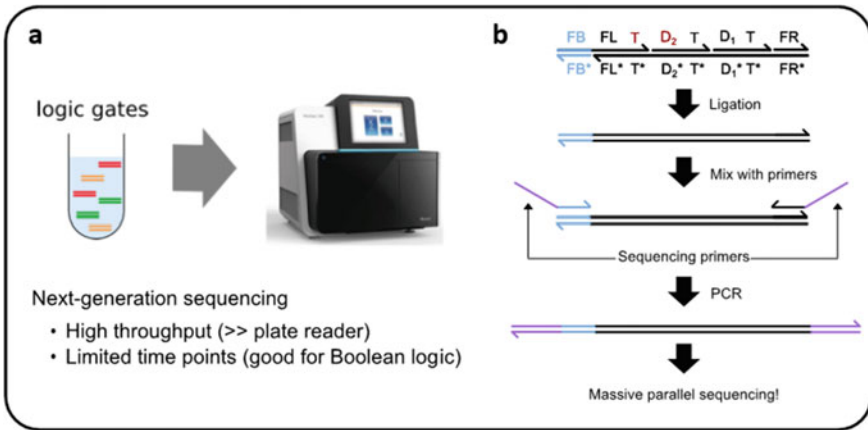


Fig. 5 Reading out DNA logic gates with next-generation sequencing. **a** Using a sequencing-based readout rather than a fluorescence-based readout in principle makes it possible to read out millions of reactions in parallel. However, unlike in a fluorescence-based assay, only reaction end points can be assessed because of a lack of time resolution, making a sequencing-based readout ideal for Boolean logic. **b** After completion of the computation, ligation is used to seal the nicks in the gate complexes. Because only reacted gates contain the domain shown in light blue, they can be selectively amplified with PCR. After that, additional DNA domains necessary for sequencing (purple) are appended to the reacted gates using standard Illumina ligation protocol

3.5 Reading Out DNA Computation with Next-Generation DNA Sequencing

Our approach relies on high-throughput (Illumina) sequencing to detect whether a gate was activated or not. High-throughput sequencing is routinely used to read out hundreds of millions of sequences in parallel, and a key advantage of our approach is that the state of all gates in a reaction mix can be read out in a single experiment. In our work, each sequencing read will correspond to a single gate molecule and will provide both information about the identity of the gate and its activation status.

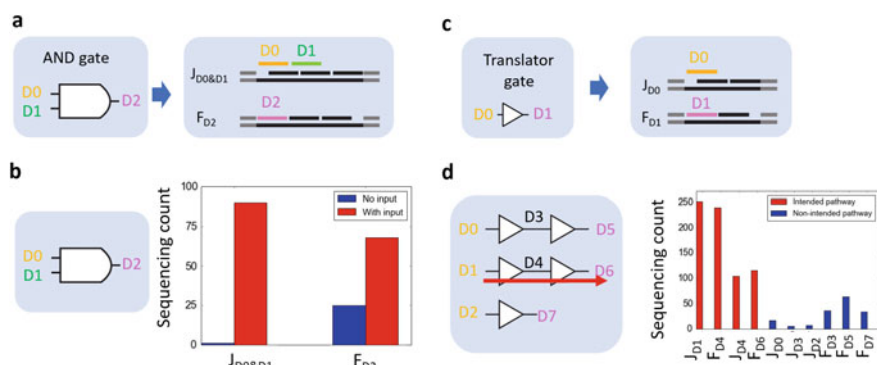


Fig. 6 Preliminary data confirm that gate operation can be read out using next-generation sequencing. **a** A logic AND gate and a simplified representation of the corresponding DNA gates. **b** Sequencing read counts for the join and fork gate that together make up the AND gate. **c** A translator gate and a simplified representation of the corresponding DNA gates. **d** Sequencing read counts for two-stage reaction cascades. Each one-input, one-output logic gate consists of a join and a fork gate. Blue bars indicate counts in the absence of inputs, and red bars show counts if inputs are present. The results show that two gates can be cascaded and can be read out by sequencing reacted gates

As explained above, reacted gates can be distinguished from unreacted gates because an additional double-stranded domain (shown in blue in Fig. 5) is appended to the reacted gate. In the sequencing reaction, the entire gate sequence including the appended auxiliary domain will be detected. Presence (or absence) of the auxiliary domain indicates whether a specific gate participated in a reaction. Moreover, the sequence of the gate uniquely identifies the gate (i.e., the sequence shows which inputs are accepted and outputs released). The sequencing reaction thus effectively amounts to counting the number of reacted (and unreacted) gates of each type.

To prepare gates for sequencing, we first use ligation to seal all nicks, followed by PCR to enrich full-length products. Next, we use ligation to add the sequencing adaptors required for Illumina sequencing. These adaptors are used to bind the product to the flow cell and also serve as binding sites for the sequencing primers.

To test whether gate operation could be read out by sequencing, we again turned to gates derived from individual ultramers (Fig. 6). We tested both a single AND gate and two distinct cascades of repeater gates. In all instances, we found the gates behaved broadly as expected, confirming that sequencing could be used to read out the performance of ndsDNA gates.

3.6 Reading Pools of Array-Derived Gates

Next, we turned to the challenge of scaling up circuit construction and analysis. We asked whether DNA gates could be synthesized on an array and whether gate operation could be read out with sequencing. To that end, we designed a set of

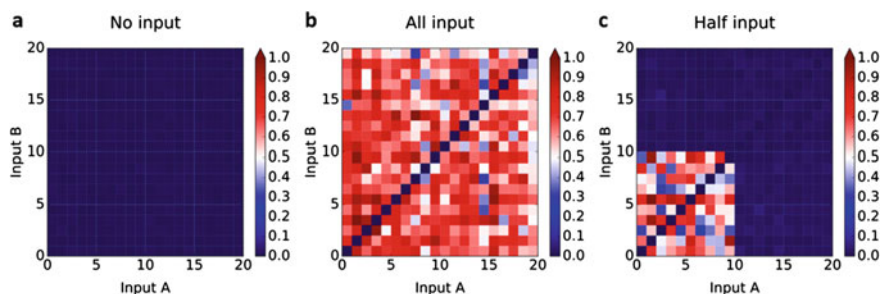


Fig. 7 Preliminary data confirm that gate operation can be read out using next-generation sequencing in a highly multiplexed format. In this experiment, 380 join gates were tested simultaneously. Each join gate accepts two inputs and gates were designed to accommodate all possible combinations of 20 distinct inputs (no gates were designed that accept the same input twice). Although input order should not matter for the required logic operation, inputs react with join gates sequentially, and two different gate designs are triggered by each input combination. Each entry in the heat map corresponds to read counts for one join gate. **a** In the absence of any inputs, read counts are low. **b** When all inputs are present, read counts are high for all gates. **c** When only a subset of inputs are added to the mix, read counts are high only for the triggered gates

380 join gates that could be triggered with all possible combination of 20 distinct inputs (see Fig. 7). Gates that respond to two copies of the same input, i.e., inputs (A, A), were not synthesized but we did synthesize gates for both input pairs (A, B) and (B, A). Although these are logically identical, the DNA domains are ordered differently, potentially resulting in performance variation. In the current iteration of our approach, all single-stranded inputs and auxiliary strands are individually column synthesized. In principle, it would be possible to also obtain at least some of these strands through enzymatic processing. However, we here instead chose to limit the number of strands that need to be ordered by reusing sequences between different modules that are tested independently. Although we observed some variation in levels of triggering, all gates behaved as designed, providing support for the work proposed here. Next, we will extend this approach to testing full AND gates, characterizing gate kinetics and increase the size and complexity of the circuit.

4 Discussion

In this manuscript, we provided a detailed example of how high-throughput DNA synthesis and sequencing technologies can be used to scale up DNA computing. We experimentally demonstrated AND logic gates as well as testing a pool of 380 array.

A potential challenge to scaling up circuit size is that the concentration of each gate in a pool is inversely proportional to the size of a pool [26]. That is, if PCR is used to amplify a pool with 10 gates to a total concentration of $1\ \mu\text{M}$, each individual gate will be at an operating concentration of 100 nM. For a circuit with 10,000 gates,

the operating concentration of each gate would only be 0.1 nM, likely resulting in a very slow computation (likely taking hours or even days) for multi-layered circuits. However, we believe that there are multiple avenues for overcoming this limitation. For example, recent work by our own group and others [10, 15, 16, 30, 39] showed that spatial colocalization of circuit elements using DNA origami scaffolds can dramatically accelerate DNA circuit computation. Similar colocalization approaches based on DNA nanostructures, beads, or other scaffolds could also be developed for array-derived logic gates.

Another potential drawback of our approach is that it only provides end point data rather than full reaction kinetics. While this is not a concern in the context of Boolean logic circuits for which we only care whether an output is true or false once the computation has completed, it could prove limiting for other applications. However, we believe that approaches based on nanopore sequencing which can provide real-time readout of reaction progress can overcome this limitation.

Although this work is still preliminary, we were able to demonstrate parallel operation and readout of 380 Join gates in a single reaction. We believe that this work will provide a foundation for large-scale DNA computing with applications in disease diagnostics and DNA data storage.

References

1. L. Adleman, Molecular computation of solutions to combinatorial problems. *Science* **266**(5187), 1021 (1994)
2. J.L. Banal, T.R. Shepherd, J. Berleant, H. Huang, M. Reyes, C.M. Ackerman, P.C. Blainey, M. Bathe, Random access DNA memory using Boolean search in an archival file storage system, in *Nature Materials*, 2021, pp. 1–9
3. R.D. Barish, P.W. Rothmund, E. Winfree, Two computational primitives for algorithmic self-assembly: copying and counting. *Nano Lett.* **5**(12), 2586–2592 (2005)
4. C. Bee, Y.-J. Chen, D. Ward, X. Liu, G. Seelig, K. Strauss, L. Ceze, in *Content-Based Similarity Search in Large-Scale DNA Data Storage Systems*. *bioRxiv*, 2020
5. Y. Benenson, B. Gil, U. Ben-Dor, R. Adar, E. Shapiro, An autonomous molecular computer for logical control of gene expression. *Nature* **429**(6990), 423 (2004)
6. L. Cardelli, Strand algebras for DNA computing, *DNA Computing and Molecular Programming*, 2009, pp. 12–24
7. L. Cardelli, Two-domain DNA strand displacement. *Math. Struct. Comput. Sci.* **23**(02), 247–271 (2013)
8. D. Carmean, L. Ceze, G. Seelig, K. Stewart, K. Strauss, M. Willsey, DNA data storage and hybrid molecular-electronic computing. *Proc. IEEE* **107**(1), 63–72 (2018)
9. L. Ceze, J. Nivala, K. Strauss, Molecular digital data storage using DNA, in *Nature Reviews Genetics*, 2019, p. 1
10. G. Chatterjee, N. Dalchau, R.A. Muscat, A. Phillips, G. Seelig, A spatially localized architecture for fast and modular DNA computing. *Nat. Nanotechnol.* **12**(9), 920 (2017)
11. X. Chen, A. Ellington, Shaping up nucleic acid computation. *Curr. Opin. Biotechnol.* **21**(4), 392–400 (2010)
12. Y.-J. Chen, N. Dalchau, N. Srinivas, A. Phillips, L. Cardelli, D. Soloveichik, G. Seelig, Programmable chemical controllers made from DNA, in *Nature Nanotechnology*, 2013
13. K.M. Cherry, L. Qian, Scaling up molecular pattern recognition with DNA-based winner-take-all neural networks. *Nature* **559**(7714), 370 (2018)

14. G.M. Church, Y. Gao, S. Kosuri, Next-generation digital information storage in DNA. *Science* **337**(6102), 1628 (2012)
15. K.E. Dunn, M.A. Trefzer, S. Johnson, A.M. Tyrrell, Investigating the dynamics of surface-immobilized DNA nanomachines. *Sci. Rep.* **6**, 29581 (2016)
16. Y.V. Gerasimova, D.M. Kolpashchikov, Towards a dna nanoprocessor: reusable tile-integrated DNA circuits. *Angewandte Chemie* **128**(35), 10400–10403 (2016)
17. N. Goldman, P. Bertone, S. Chen, C. Dessimoz, E.M. LeProust, B. Sipos, E. Birney, Towards practical, high-capacity, low-maintenance information storage in synthesized DNA. *Nature* **494**(7435), 77 (2013)
18. R.N. Grass, R. Heckel, M. Puddu, D. Paunescu, W.J. Stark, Robust chemical preservation of digital information on DNA in silica with error-correcting codes. *Angewandte Chemie Int. Edn.* **54**(8), 2552–2555 (2015)
19. S. Kosuri, G.M. Church, Large-scale de novo DNA synthesis: technologies and applications. *Nature methods* **11**(5), 499 (2014)
20. M.R. Lakin, D. Stefanovic, Supervised learning in adaptive DNA strand displacement networks. *ACS synthetic biology* **5**(8), 885–897 (2016)
21. R. Lopez, R. Wang, G. Seelig, A molecular multi-gene classifier for disease diagnostics. *Nat. Chem.* **10**(7), 746 (2018)
22. J. Lu, G. Getz, E.A. Miska, E. Alvarez-Saavedra, J. Lamb, D. Peck, A. Sweet-Cordero, B.L. Ebert, R.H. Mak, A.A. Ferrando, J.R. Downing, T. Jacks, R.H. Horvitz, T.R. Golub, MicroRNA expression profiles classify human cancers. *Nature* **435**(7043), 834–838 (2005)
23. M. Neiman, On the molecular memory systems and the directed mutations. *Radiotekhnika* **6**, 1–8 (1965)
24. L. Organick, S.D. Ang, Y.-J. Chen, R. Lopez, S. Yekhanin, K. Makarychev, M.Z. Racz, G. Kamath, P. Gopalan, B. Nguyen et al., Random access in large-scale DNA data storage. *Nature biotechnology* **36**(3), 242 (2018)
25. L. Qian, E. Winfree, Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**(6034), 1196 (2011)
26. L. Qian, E. Winfree, A simple DNA gate motif for synthesizing large-scale circuits. *J. R. Soc. Interface* **8**(62), 1281–1297 (2011)
27. L. Qian, E. Winfree, J. Bruck, Neural network computation with DNA strand displacement cascades. *Nature* **475**(7356), 368–372 (2011)
28. N. Rosenfeld, R. Aharonov, E. Meiri, S. Rosenwald, Y. Spector, M. Zepeniuk, H. Benjamin, N. Shabes, S. Tabak, A. Levy, D. Lebanony, Y. Goren, E. Silberschein, N. Targan, A. Ben-Ari, S. Gilad, N. Sion-Vardy, A. Tobar, M. Feinmesser, O. Kharenko, O. Nativ, D. Nassa, M. Perelman, A. Yosepovich, B. Shalmon, S. Polak-Charcon, E. Fridman, A. Avniel, I. Bentwich, Z. Bentwich, D. Cohen, A. Chajut, I. Barshack, MicroRNAs accurately identify cancer tissue origin. *Nat. Biotechnol.* **26**(4), 462–469 (2008)
29. P.W. Rothmund, N. Papadakis, E. Winfree, Algorithmic self-assembly of DNA sierpinski triangles. *PLoS Biol.* **2**(12), e424 (2004)
30. I.M. Ruiz, J.-M. Arbona, A. Lad, O. Mendoza, J.-P. Aimé, J. Elezgaray, Connecting localized DNA strand displacement reactions. *Nanoscale* **7**(30), 12970–12978 (2015)
31. G. Seelig, D. Soloveichik, D. Zhang, E. Winfree, Enzyme-free nucleic acid logic circuits. *Science* **314**(5805), 1585 (2006)
32. N.C. Seeman, Nucleic acid junctions and lattices. *J. Theor. Biol.* **99**(2), 237–247 (1982)
33. D. Soloveichik, G. Seelig, E. Winfree, DNA as a universal substrate for chemical kinetics. *Proc. Nat. Acad. Sci.* **107**(12), 5393 (2010)
34. D. Soloveichik, E. Winfree, Complexity of self-assembled shapes. *SIAM J. Comput.* **36**(6), 1544–1569 (2007)
35. N. Srinivas, J. Parkin, G. Seelig, E. Winfree, D. Soloveichik, Enzyme-free nucleic acid dynamical systems. *Science* **358**(6369), eaal2052 (2017)
36. K. Stewart, Y.-J. Chen, D. Ward, X. Liu, G. Seelig, K. Strauss, L. Ceze, A content-addressable DNA database with learned sequence encodings, in *International Conference on DNA Computing and Molecular Programming* (Springer, 2018), pp. 55–70

37. M.N. Stojanovic, T.E. Mitchell, D. Stefanovic, Deoxyribozyme-based logic gates. *J. Am. Chem. Soc.* **124**(14), 3555–3561 (2002)
38. K. Takahashi, S. Yaegashi, A. Kameda, M. Hagiya, Chain reaction systems based on loop dissociation of DNA, in *DNA Computing: 11th International Workshop on DNA Computing, DNA11, Lecture Notes in Computer Science*, vol. 3892, 2006, pp. 347–358
39. M. Teichmann, E. Kopperger, F.C. Simmel, Robustness of localized DNA strand displacement cascades. *ACS Nano* **8**(8), 8487–8496 (2014)
40. K.J. Tomek, K. Volkel, E.W. Indermaur, J.M. Tuck, A.J. Keung, Promiscuous molecules for smarter file operations in DNA-based data storage. *Nat. Commun.* **12**(1), 1–10 (2021)
41. E.L. Tsalik, R. Henao, M. Nichols, T. Burke, E.R. Ko, M.T. McClain, L.L. Hudson, A. Mazur, D.H. Freeman, T. Veldman et al., Host gene expression classifiers diagnose acute respiratory illness etiology. *Sci. Transl. Med.* **8**(322), 322ra11 (2016)
42. M. Vasic, C. Chalk, S. Khurshid, D. Soloveichik, Deep molecular programming: a natural implementation of binary-weight relu neural networks, in *International Conference on Machine Learning* (PMLR, 2020), pp. 9701–9711
43. E. Winfree, F. Liu, L.A. Wenzler, N.C. Seeman, Design and self-assembly of two-dimensional DNA crystals. *Nature* **394**(6693), 539 (1998)
44. C. Zhang, Y. Zhao, X. Xu, R. Xu, H. Li, X. Teng, Y. Du, Y. Miao, H.-C. Lin, D. Han, Cancer diagnosis with DNA molecular computation. *Nat. Nanotechnol.* **15**(8), 709–715 (2020)
45. D.Y. Zhang, G. Seelig, DNA-based fixed gain amplifiers and linear classifier circuits, in *International Workshop on DNA-Based Computers* (Springer, 2010), pp. 176–186
46. D.Y. Zhang, E. Winfree, Robustness and modularity properties of a non-covalent DNA catalytic reaction. *Nucl. Acids Res.* **38**(12), 4182–4197 (2010)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Sequenceable Event Recorders



Luca Cardelli

Abstract With recent high-throughput technology, we can synthesize large heterogeneous collections of DNA structures and also read them all out precisely in a single procedure. Can we use these tools, not only to do things faster, but also to devise new techniques and algorithms? In this paper, we examine some DNA algorithms that assume high-throughput synthesis and sequencing. We record the order in which N events occur, using N^2 redundant detectors but only N distinct DNA domains, and (after sequencing) reconstruct the order by transitive reduction.

1 Introduction

With recent high-throughput technology, we can synthesize large heterogeneous collections of DNA structures [7, 13] and also read them all out precisely in a single procedure [10]. This contrasts with the older practice of assembling structures one at a time and of reading them out individually (e.g., by fluorescence), or reading them together ambiguously (e.g., by gel electrophoresis). Can we take advantage of these high-throughput and high-precision technologies, not only to do things faster but also to devise new techniques and algorithms? In this paper, we examine some DNA algorithms that assume both high-throughput synthesis and high-throughput sequencing: they would not be very practical otherwise.

A sequence ‘s’ of DNA nucleotides *hybridizes* (forms a double strand) with its reverse Watson-Crick complement denoted ‘s*’; we write the resulting double strand as ‘s’. Subsequences of ‘s’ are called *domains* provided they are *independent* of each other, that is, provided that differently identified domains do not hybridize with each other, or with significantly long parts of each other [16]. Under normal laboratory conditions, a domain ‘a’ is called *short* if it hybridizes reversibly with ‘a*’, and *long* if it hybridizes irreversibly with it.

A short single-stranded domain ‘t’, called a toehold, followed in the same sequence by a long single-stranded domain ‘a’ can initiate *strand displacement*. This is the

L. Cardelli (✉)
University of Oxford, Oxford, UK
e-mail: luca.a.cardelli@gmail.com

process (later detailed in Fig. 3) where a single-stranded sequence ‘ta’ hybridizes to a double strand composed of ‘t*’ attached to the bottom strand of a double-stranded ‘a’. The invading ‘ta’ can displace and possibly replace the existing ‘a’ domain of the double strand through a random-walk competition between the two ‘a’ domains hybridizing to the same ‘a*’.

A *nick* is an interruption in one of the two strands of a double strand, at the boundary between two domains. By cascading short and long domains, occasionally separated by nicks, we can achieve multi-step strand displacements where the whole sequence of displacements can itself be reversible or irreversible. This way we can emulate reversible and irreversible chemical reactions [14] and other computational abstractions [9].

The readout of the outcome of such sequences of displacements is often done by fluorescence. Fluorophore/quencher pairs are attached to some domains that participate in the reactions, those in particular whose displacement indicates that a significant event has occurred. The displacement separates the fluorophore from the quencher and hence induces visible fluorescence. This provides a real-time account of the computation, but the readout capability is restricted: a limited number of separate events can be detected by using different fluorescence colors. This is analogous to debugging a program by inserting a limited number of print statements at a time, each one printing a single letter.

Another way of achieving a readout is via gel electrophoresis, to distinguish the sequences in a solution by their length at the end of the experiment (or at predetermined time points). Many different sequences can be identified provided they have different lengths (masses) and provided that we know their length ahead of time. Unexpected lengths can be hard to identify. This is analogous to debugging a program by using control flow counters to tell us how many times each routine is invoked, or each structure is accessed, without any insight about the order of events.

Finally, and especially with more recent high-throughput technology, we can obtain a readout by ligating the nicks and sequencing all the strands in the solution at the end of the experiment (or at predetermined time points). With high-throughput sequencing, we can inspect potentially the entire composition of the solution. The debugging analogy now is that of taking a core dump: analyzing in complete detail the entire state of a computation, but only infrequently or at the end, and again without any obvious insight on the order of events that occurred.

The order of events is usually of great interest: for example, multiple laborious gene knockout experiments are frequently carried out to determine the order of gene activations. What if we could instead take a single core dump that tells us the order of all the events of interest? To that end, we should record the order of events within the state of the system, so that we can inspect such recording at the end as part of the core dump. Assuming high-throughput sequencing, we can embed a large amount of information within the solution. We are going to assume that we can embed N^2 pieces of information, where N is the number of events of interest. This seems achievable for reasonably small N while providing a lot of information, encoding for each event

whether it happened before, together, or after any other event. Each one of the N^2 event order detectors is a structure that accepts inputs but does not produce outputs: when it detects certain conditions, it locks down in a stable state and waits to be sequenced later.

Our strategy is therefore to embed a *preorder* (a reflexive and transitive relation) of events within the solution. This is a *pre-order* because we may not be able to detect the precise order of two events if they happen very close to each other, in which case both directions are recorded. With N^2 detectors, we can determine the order of any pair of events without needing to coordinate the detectors with each other or with a central structure; hence, each detector can be relatively simple. An alternative is to use only N detectors that sequentially add records to a central *tape*, but this requires a way of guaranteeing atomic access to the tape [9]. Still, event recorders of the tape variety, readable by sequencing, have been nicely demonstrated using natural DNA and protein mechanisms [12, 15].

A preorder is not the entire history of a computation. We are considering the preorder of *first-occurrence* of events: any subsequent occurrences for the same signal are not recorded. This limited information can still provide support for causality: if an event always precedes another event over a number of runs, then this supports the first event causing the second, or having a common cause.

In the rest of this paper, we aim to describe the architecture of such a preorder recorder, using DNA strand displacement technology, slowly building up from simpler problems. A property of all the designs in this paper is that (apart from the single-stranded input signals) all DNA structures are nicked double strands with no additional modifications or secondary structure. Therefore, the required and potentially large numbers of components can be fabricated by bacterial cloning as a single or a few long DNA double strands, followed by enzymatic cutting and nicking [4] (see Appendix). Other technologies for high-throughput synthesis of large heterogeneous libraries exist [7, 13]. Thus, we rely on both high-throughput synthesis for producing the N^2 detectors and on high-throughput sequencing to read them out.

2 Occurrence Recorder

We begin by investigating the simplest event recorder: recording the occurrence of events at any time during an experiment. By an ‘event’ here, we mean the appearance of a whole population of identical molecules and in fact a specific structure of molecules that can be uniformly identified. Any event that does not fit that description must first be transduced into one of these uniform molecular structures. By a *signal*, we mean a population of one such molecular species over time, and by an *event*, we mean the appearance of a signal population (we do not detect the disappearance of a population).

In discussions, we summarize DNA structures by a textual notation. In addition to lowercase letters like ‘a’ for single-stranded long domains, and underlined letters like ‘a’ for the corresponding double-stranded long domains, a single short domain

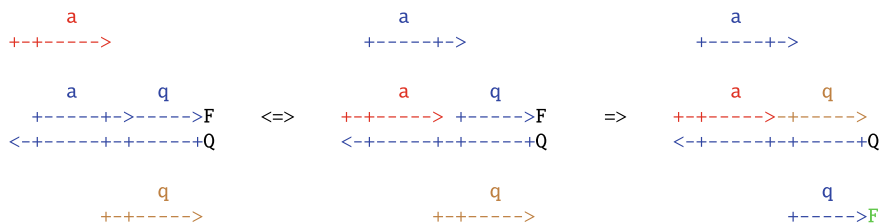


Fig. 1 Yes gate for detection by fluorescence

is used for all toeholds: ‘ $-$ ’ is an open (i.e., un-hybridized) toehold on a single strand or on the upper strand of a double strand, ‘ $_$ ’ is an open toehold on the lower strand of a double strand, and ‘ $\underline{-}$ ’ is a covered (double-stranded) toehold. A sequence of domains on a double strand with an initial open toehold and an intermediate covered toehold looks like ‘ $\underline{a-b}$ ’. This summary notation omits information about nicks, which are instead detailed in corresponding figures. Note that, before sequencing, all the open domains should be complemented, and all the nicks should be ligated.

Figures instead depict the corresponding single and double strands graphically (e.g., Fig. 1). A domain is a short or long sequence of dashes ‘ $-$ ’ with domain delimiters ‘ $>$ ’ and ‘ $<$ ’ pointing in the 5’-to-3’ direction to indicate either a nick (an interruption in the strand) or the 3’ end, and ‘ $+$ ’ to indicate the 5’ end or the logical boundary of a domain (not a nick). The name of a domain is a lower-case letter placed on top of the upper strand, with implicitly the reverse complement domain on the lower strand. All toeholds are the same sequence: they have a blank name. Reversible reactions are ‘ \Leftrightarrow ’ and irreversible reactions are ‘ \Rightarrow ’.

2.1 Yes Gate

The events that we want to detect are represented by single-strands ‘ $-a$ ’ each consisting of a (short) toehold ‘ $-$ ’ attached to a (long) domain ‘ a ’. If ‘ $-a$ ’ is ever present, we want to know about it: this is the purpose of the Yes gate for ‘ a ’.

First (Fig. 1) let us consider the traditional way of detecting ‘ $-a$ ’. A double-stranded structure ‘ $\underline{a-q}$ ’ with an open toehold ‘ $_$ ’ accepts the single-strand ‘ $-a$ ’ (reversibly) and opens up another toehold, yielding ‘ $\underline{-a q}$ ’. That structure then locks down (irreversibly) by combining with an auxiliary single-strand ‘ $-q$ ’ to produce the fully hybridized ‘ $\underline{-a-q}$ ’ and the toehold-free ‘ q ’.

If we attach a fluorophore (F) and quencher (Q) pair at the right end of ‘ $\underline{a-q}$ ’ (and not at the end of ‘ $-q$ ’), we can detect the occurrence of ‘ $-a$ ’ because it separates F from Q and causes visible fluorescence. However, if we were to (ligate and) sequence the solution, it would be difficult or impossible to tell the difference between the initial and final state, because they differ only by open toeholds and by the positions of nicks that are erased by ligation.

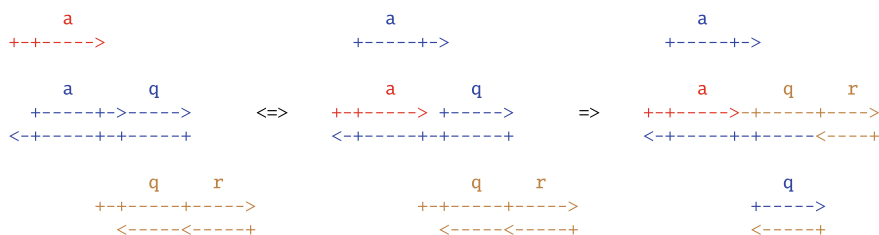


Fig. 2 Yes gate for detection by sequencing



Fig. 3 3-way and 4-way displacements in the first and last steps of Fig. 2

Let us now consider (Fig. 2) an additional domain ‘r’ that will help us tag the desired outcome. The ‘-q’ single-strand is replaced by a ‘-qr’ double strand, but with a nick on the bottom between ‘q’ and ‘r’.¹ The first reaction is the same as before, but the second reaction is now a 4-way strand displacement² (Fig. 3 right). This detector is non-catalytic: it captures some of the ‘-a’ strands and releases ‘a-’ strands (which are usually harmless).

If this gate is triggered, then the main outcome is ‘-a-qr’, which is a nicked but fully complemented double strand: it is ready for ligation and sequencing. If the gate is not triggered, then the outcome is the initial ‘a-q’ which is distinguishable after sequencing.³

For a catalytic version (one that does not sequester the input), consider the design in Fig. 4: we add two more structures to Fig. 2 that absorb the ‘a-’ that was left over and convert it back to a free ‘-a’. Such catalytic irreversible gates avoid sequestering

¹ The ‘q’ domain can be single-stranded, interacting by a simpler 3-way displacement, but that would rule out producing the structure directly by cloning [4]. If the whole ‘-qr’ were single stranded, a polymerase could not attach to the final structure to complement the ‘r’ domain, as required for sequencing and positive detection, but a PCR step could be used instead.

² 4-way strand displacement is slower than 3-way [8] Chap. 5 (although potentially more robust [6]). This may degrade the ability of our algorithms to separate events in time, but otherwise it does not affect their logic, which includes the possibility of coincidence of events. The 4-way displacement is of the unusual ‘open’ kind [8], that is, initiated by a single toehold binding instead of two.

³ ‘a-q’ needs to be fully complemented, ligated, and sequenced. To that end, we can add an additional double-stranded domain on the left of the initial toehold, as in [5]. This allows a polymerase to proceed in the 3’-5’ direction of the bottom strand and fully complement the top strand. For this presentation, we omit these domains because they have no other function and do not participate in the described reactions. Moreover, even if sequencing misread the initial state, we would still get our answer by the presence or absence of the final state ‘-a-qr’ + ‘q’.

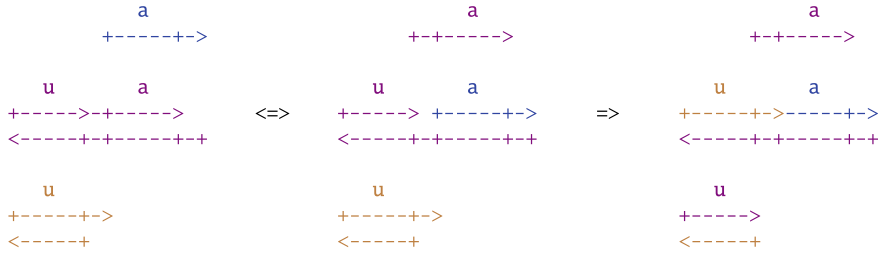


Fig. 4 Catalytic Yes gate, additional reactions

weak signals, while being fully activated by weak signals, leading to robust detection (if the signals are not drained too quickly by downstream processing).

2.2 Occurrence Recorder Algorithm

We can use Yes gates to detect a collection of signals in an experiment via a single high-throughput readout: we prepare a Yes gate detector for each signal, we mix them in at the beginning, and we sequence the entire solution at the end, revealing any detectors that have fired.

3 Coincidence Recorder

We now move to a more interesting task: detecting the simultaneous presence of signals. The idea enabling the sequencing-based readout of gates, and in particular the novel use of 4-way displacement, is due to Chen and Seelig [5] (the Yes gate of Fig. 2 is also a special case of this). Their design was originally meant as an AND gate made of a sequenceable Join part accepting inputs, and a sequenceable Fork part producing outputs. We are going to use just a sequenceable Join half to detect the simultaneous occurrence of any pair of signals in a given set of signals, relying on high-throughput sequencing to inspect all possible combinations.

3.1 Join Gate

The design in Fig. 5 is rooted in a fluorophore-oriented Join gate, along the lines of Fig. 1, which ultimately comes from [11] and [2]. However, again here we want to find a sequencing-friendly version, where the initial structures ‘a-b-q’ and ‘-qr’ with input signals ‘-a’ and ‘-b’ are sequencing-distinguishable from the final structure ‘-a-b-qr’, which indicates that both signals were present at the same time. The gate

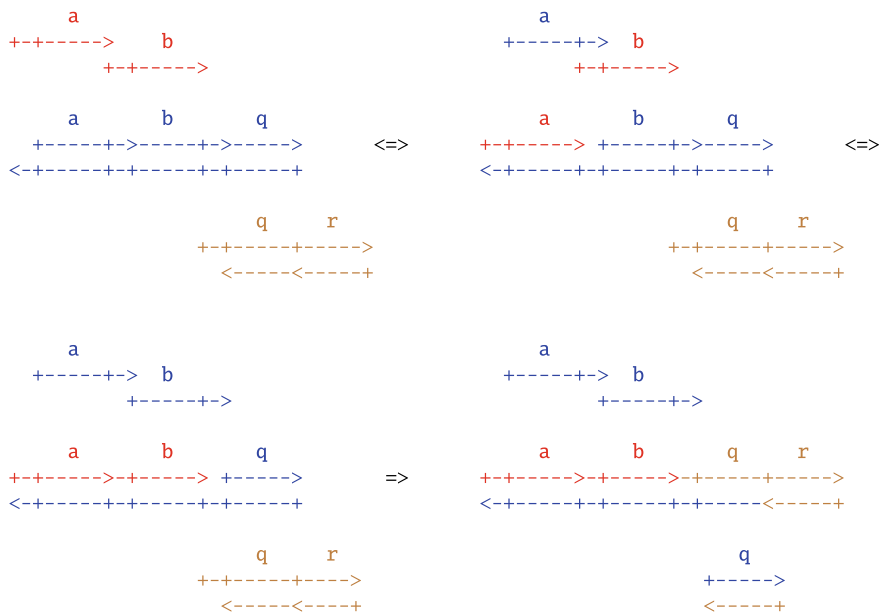


Fig. 5 Join gate for detection by sequencing

locks down when the two signals are received in turn. If one signal appears first and persists until the second arrives, this gives the same result as both signal appearing together. If one signal is removed before the other one appears, the gate reverts and the result indicates no co-occurrence. The gate can be made more kinetically symmetrical by mixing $\text{Join}(a,b)$ with $\text{Join}(b,a)$.

As in the previous case, we can add structures to this gate that convert it to a catalytic gate. But we need to handle the two signals together in the additional structures, because 'a-' must be able to revert to '-a' when '-b' is not present. Hence, we use the binary structure in Fig. 6 for distinct a,b. This structure cannot coexist with a catalytic $\text{Yes}(a)$ as it would lock down $\text{Join}(a,b)$ on the first input: a later non-coincident '-b' would give a false positive. $\text{Join}(a,a)$ must not have the additional catalytic structures for the same reason: it is best to replace it with a non-catalytic $\text{Yes}(a)$.

3.2 Coincidence Recorder Algorithm

We can use Join gates to detect the simultaneous occurrence of any pair of distinct signals in a collection: we prepare a Join gate detector for each such pair, we mix them in at the beginning, and we sequence the entire solution at the end, revealing any detectors that have fired. If we detect $\text{Join}(a,b)$ and $\text{Join}(b,c)$, we can deduce the

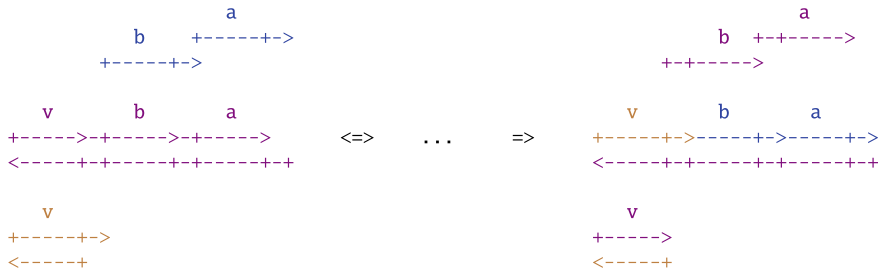


Fig. 6 Catalytic Join gate, additional reactions

coincidence of a and c , and we should also detect $\text{Join}(a,c)$: that redundancy serves as a crosscheck. We could use fewer Join gates, but if we did not include the transitive $\text{Join}(a,c)$ and b never came, we would not detect the coincidence of a and c .

4 Preorder Recorder

We now aim to build a device to record the order of occurrence of events in an experiment. The question is: given a set of events a, b, c, d, \dots that occur in some order, in what order did they *first* occur? If some events can occur together (up to experimental uncertainty), the relationship is a *preorder*: a reflexive and transitive relation. We want to reconstruct the temporal preorder of events from a single observation at the end of a run, with a single mass sequencing.

Such a *preorder recorder* would be useful for monitoring a process over time without sampling the system at multiple time points. Our recorder does not record timing and does not record sequence, but it records the first-occurrence preorder, storing it within the system itself. Recording the order, rather than the full timing of events, means that we need not use energy during periods of inactivity, and we need not worry about how often we should sample the system. The energy expenditure is all preloaded: no additional resources are needed no matter how long or complex the events history becomes, and there can be no ‘memory overflow’ of the recording. Repeated preorder experiments can build up evidence for causality, by observing which events always happen in the same order, independently of timing and other conditions.

The algorithm below uses a number of gates that is quadratic in the number of signals N but is independent of the observation time. After the initial setup, it requires no further energy because it reacts to signals and does not actively inspect the environment for their presence. More subtly, the algorithm uses a number of distinct domains that is just $N + 4$ (+1 for toeholds). This is important to avoid crosstalk among domains, which becomes more difficult to avoid when we have more domains. The situation would be much worse if we needed N^2 distinct domains in addition to N^2 gates.

The coincidence recorder in the previous section was obtained by iterating a Join gate. For the preorder recorder, we iterate a choice gate, which we describe next. Instead of presenting directly the domain structures (for which there are multiple possibilities), we first describe abstractly how the choice gate behaves, and how the algorithm uses it. The DNA implementation is described later.

4.1 Choice Gate Specification

A choice gate is a two-input gate denoted $a?b$ between input events a and b . As an abstract operator it is symmetric: $a?b = b?a$. Its desired behavior is as follows:

- If a arrives no later than b , then $a?b$ produces a distinct result that we indicate $a \leq b$ or equivalently $b \geq a$.
- If b arrives no later than a , then $a?b$ produces a distinct result that we indicate $b \leq a$ or equivalently $a \geq b$.
- If a and b arrive together, then $a?b$ produces a result that we indicate $a \sim b$ or equivalently $b \sim a$. (This is in practice an equal mixture of $a \leq b$ and $b \leq a$, or an unequal mixture if they arrive slightly offset.)
- As a special case, if a ever arrives, then $a?a$ produces a result $a \sim a$.

The three results between different a and b are assumed to be distinct and distinguishable by sequencing. Our algorithm requires only that there are three detectable final configurations: $a \leq b$ and $b \leq a$ depending on which of two inputs arrives first, and a mixture of the two, $a \sim b$, if they arrive together. We may further analyze the results quantitatively: a 100%/0% mixture of $a \leq b$ and $b \leq a$ indicates that enough of a arrived to exhaust the gate population before any of b (if any) arrived. Other mixtures may indicate how much events overlapped in time, their relative strength, or some confusion between those. Weak signals may appear to have arrived together.

There are many ways to achieve this specification, and we will discuss at least two. But first we describe the algorithm that uses these gates.

4.2 Preorder Recorder Algorithm

Suppose we have a (moderately large) set of events a, b, c, d, e, f, \dots , like the occurrence of some mRNAs in a cell-free extract. They will activate in some order like $b.cd.ae.d$ (b first, then cd together, then ae together, then d). We want to store that order as the events arise, and read it back at the end.

For N signals, we need N^2 distinct DNA structures: all the possible combinations of two signals, including all the $x?x$ cases. We are not going to distinguish event sequences with repetitions and oscillations: we only look at the first occurrence of a signal. For example, the sequence $b.b.b$ is the same as b for us, and $a.b.a$ is the same

as $a.b$ (we can still tell that the first a arrived before the first b : the second a does not confound it).

We do not provide any external timing: there are no clocks needed to sample these signals over time, and there is no predetermined sampling frequency. We just need to assume that the sequence of events is slow enough. If it is not slow enough then $a.b$ will look just like ab (in practice, the closeness of two signals will be reflected in the relative proportions of $a \leq b$ and $b \leq a$, so we can still get some more information). The time resolution is thus determined by the speed of the DNA reactions. If they happen to be fast enough for the intended observed system, then sampling over longer time periods does not require any more gates or any more energy: the gates just naturally sit waiting for the signals to arrive.

The input to our algorithm is a preorder of signals, like $a.bc.def.g$ that is occurring in real time in our experiment. We initially add to the solution all the choice gates $x?y$ such that x and y range over all those signals (including $x = y$). At the end, we sequence all the leftover structures (e.g., $x \leq y$) and we reconstruct the preorder from them. The process of reconstructing the preorder graph from what is essentially its reachability matrix is called *transitive reduction* and has the same complexity as transitive closure and matrix multiplication [1].

4.3 Crosstalking Choice Gate

We now describe a DNA implementation of the choice gate $a?b$. We discuss below how the gates crosstalk, and what are the consequences of crosstalking. But in summary, for our application, this implementation is sufficient, and it is also considerably more economical than a ‘proper’ non-crosstalking implementation.

The inputs are the usual two-domain signals with toehold on the left. For each abstract choice operator $a?b$, we use two pairs of double strands abbreviated as group $[a?b]$ and group $[b?a]$, with $a?b = [a?b] + [b?a]$. They are symmetric but different because $[a?b]$ reacts to a ‘ $-b$ ’ strand, while $[b?a]$ reacts to an ‘ $-a$ ’ strand. Conversely, $[a?b]$ reacts also to an ‘ $a-$ ’ strand and $[b?a]$ reacts also to a ‘ $b-$ ’ strand, through the same toehold but in opposite directions.

In Fig. 7, each of the primary structures (top) eventually binds to one and only one of the two end caps (bottom): we arbitrarily associate one end cap with $[a?b]$ and the other with $[b?a]$ (the square bracket indicates the side the end cap is with), so in fact $a?b = [a?b] + [b?a] = [b?a] + [a?b] = b?a$. The central portions with the ‘ a ’ and ‘ b ’ domains are surrounded by four fixed domains ‘ s ’, ‘ p ’, ‘ q ’, ‘ r ’: these are the same sequences for all the choice gates, regardless of variations in ‘ a ’ and ‘ b ’.⁴ The nameless toehold is the same sequence everywhere.

⁴ In fact, all four ‘ s ’, ‘ p ’, ‘ q ’, ‘ r ’ domains can be the same sequence without ambiguity in the outcomes of Fig. 8. Still, we keep them distinct in light of other possible constraints, such as in Fig. 10.

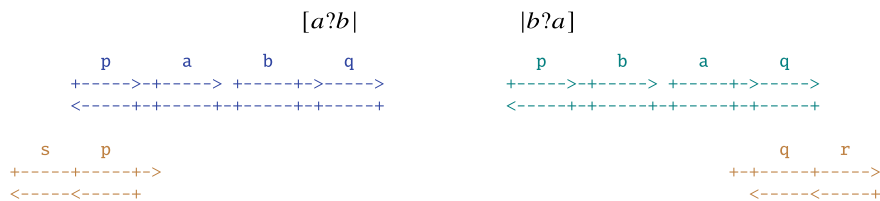


Fig. 7 Crosstalking choice gate

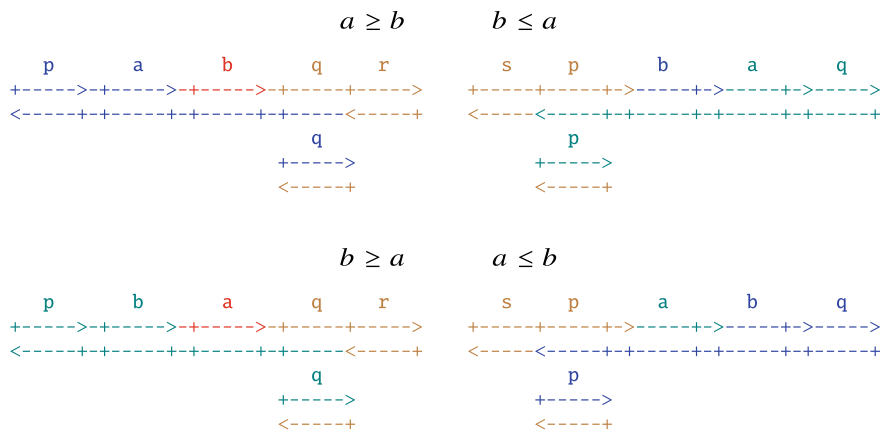


Fig. 8 Crosstalking choice gate outcomes for $a?b$. Top: for input ‘-b’ (red), which also releases back ‘-b’ (teal, not shown). Bottom: for input ‘-a’ (red), which also releases back ‘-a’ (blue, not shown)

If a signal ‘-b’ (with toehold on the left) binds to $[a?b]$, it blocks the toehold and displaces to the right. It also releases ‘-b’ (with toehold to the right), which goes to $[b?a]$, again blocks the toehold there, and displaces to the left, catalytically releasing a copy of the original ‘-b’. The end caps can bind to the remaining open toeholds and lock down the configuration. If ‘-a’ arrives later, it finds all the toeholds blocked and cannot bind to the remaining structures. Thus ‘-b’ arriving first prevents ‘-a’ from binding later. If ‘-a’ arrives first, the situation is symmetric, with the end caps binding to the opposite structures than in the ‘-b’-first case.

In more detail, the initial binding of signals opens up new toeholds for the double-stranded ‘sp’, ‘qr’ end caps: they cause 4-way strand displacements and stabilize the outcomes in a way that is distinguishable by sequencing. For a ‘-b’ input the final structures are ‘p-a-b-qr’ + ‘q’, which is the result we earlier called $a \geq b$, and ‘sp-b-a-q’ + ‘p’, which is the result we earlier called $b \leq a$ (Fig. 8, top). The opposite happens if ‘-a’ arrives first (Fig. 8, bottom). If ‘-a’ and ‘-b’ arrive together, then both results are produced because the released ‘-a’ and ‘-b’ bind concurrently to as yet untouched copies of the gates.

These activations are irreversible and catalytic: ‘-a’ and ‘-b’ are released back without requiring additional structures. This is going to help kinetically and is also

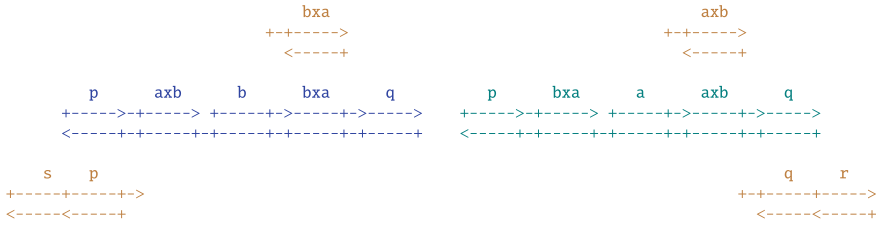


Fig. 9 Non-crosstalking choice gate

less likely to perturb the system we are observing. Reflexive gates $a?a$ work as expected: we need them to signify that a signal ‘ $-a$ ’ has arrived at some time. We produce the $a?a$ structures by the general recipe, meaning as $[a?a] + |a?a|$, hence with *twice* the concentration of the main structure. This is in fact what we need to keep the kinetics balanced with respect to non-reflexive gates.

A single choice gate works as described, but we need to consider the situation where there are multiple choice gates together. In a gate with $[a?b]$, the input ‘ $-b$ ’ releases ‘ $b-$ ’, which goes on to bind to $|b?a|$, but also to any other $|b?x|$: crosstalk! Normally this would be incorrect, but here we want to activate $|b?x|$ as well, since it tells us that ‘ $-b$ ’ arrived before ‘ $-x$ ’. If there is a $|b?x|$, then there is also an $[x?b]$, which driven by ‘ $-b$ ’ activates $|b?x|$ anyway. So the crosstalk between gates does not hurt in this particular instance. The most interesting consequence is that, as we noted, although we have N^2 gates, we only have to encode N distinct domains (plus the 4 auxiliary ones). This greatly reduces the potential interference between domains that would be an obstacle to scaling up the number of signals. As an added benefit, these crosstalking gates are automatically catalytic (*cf.* Fig. 9).

As an example, for 3 signals a, b, c , we use the following 9 choice gates (first column) and corresponding initial structures (second column):

gates	structures	after ‘ $-c$ ’	after ‘ $-b$ ’
$a?a$	$[a?a] a?a $	$[a?a] a?a $	$[a?a] a?a $
$b?b$	$[b?b] b?b $	$[b?b] b?b $	$b \geq b \ b \leq b$
$c?c$	$[c?c] c?c $	$c \geq c \ c \leq c$	$c \geq c \ c \leq c$
$a?b$	$[a?b] b?a $	$[a?b] b?a $	$a \geq b \ b \leq a$
$a?c$	$[a?c] c?a $	$a \geq c \ c \leq a$	$a \geq c \ c \leq a$
$b?c$	$[b?c] c?b $	$b \geq c \ c \leq b$	$b \geq c \ c \leq b$

If a signal ‘ $-c$ ’ arrives, it initially activates 3 structures, the ones of the form $[x?c]$, producing outcomes $x \geq c$. Soon after, the signal ‘ $c-$ ’ that is released by those activations crosstalks with the structures of the form $|c?y|$, producing outcomes $c \leq y$ (third column). If a signal ‘ $-b$ ’ arrives next, it further activates some gates, but not the ones that have been used up by ‘ $-c$ ’ (fourth column). If we sequence the structures at this point, we can conclude (with multiple redundancies) that:

$$c \leq b \leq a$$

That's a definite $c < b$, because we observe $c \leq b$ but not $b \leq c$. Moreover, we do not observe $a \leq a$ which means that a never arrived. If we were to observe $c \leq b$ and $b \leq c$, then we would deduce that c, b arrived together, up to our time resolution.

Detection of the preorder should be robust because of the redundancies. Background noise and bad gates can be tolerated, because we just need to detect which of $a \leq b$ vs. $b \leq a$ is strongest. Moreover, our set of observed structures must be transitively closed: if the input is the sequence a.b.c then we should observe $a \leq b$ (and not $b \leq a$) and $b \leq c$ (and not $c \leq b$), and transitively also $a \leq c$ (and not $c \leq a$). The transitive closures can act as consistency checks.

4.4 A “Proper” Choice Gate

If we want to use a choice gate in some general and modular way within some bigger design, then we need a gate that respects all the conventions, and in particular that does not crosstalk with unrelated gates. In the design in Fig. 9 the domains called ‘axb’ and ‘bxa’ are uniquely determined by ‘a’ and ‘b’ to avoid crosstalk with other gates. Here, a ‘b-’ input does not release a ‘-b’ signal that connects with other gates, but rather a ‘-bxa’ signal that binds uniquely to the other half of that choice gate. In our preorder recorder application, where we use N^2 gates, we would now need $N + N^2$ distinct signal domains. Other than that, this choice gate could replace the crosstalking one. A catalytic version can be obtained as in Fig. 4.

5 Conclusions

We have described a class of DNA algorithms designed to take advantage of high-throughput sequencing and also relying on high-throughput synthesis. A combinatorial number of different structures are activated on demand without any timing or synchronization, operating by natural parallelism. The outcome is produced not as an output but as the final state of the system to be read by sequencing.

Acknowledgements Thanks to Matthew Lakin, Georg Seelig, and David Soloveichik, for helpful comments, and to Yuan-Jyue Chen and Georg Seelig for initial discussions that lead to this paper.

Appendix: Restriction Enzymes

Naturally occurring *restriction enzymes* bind to double-stranded DNA and make a double-strand cut. Some natural enzymes and some engineered versions of restriction enzymes cut only one of the strands: these are called *nicking enzymes* [3]. Although there are many such enzymes used for cutting natural and synthetic DNA, their properties are severely restricted.⁵ In nature, there is also a Cas9 protein that is essentially a programmable restriction enzyme: it can cut DNA at almost any desired location determined by a separate RNA strand.

Let us imagine that we are able to design our own restriction enzymes. We show that it is then possible to cut and nick the crosstalking choice gates in Fig. 7 out of a longer DNA double strand. Such a strand is ideally obtained by bacterial cloning, which can produce large quantities of very long high-quality strands, enabling the mass production of DNA gates by cutting them out of long cloned strands [4].

There are many alternatives to the hypothetical choices of restriction enzymes show below, depending on where the DNA cuts are located with respect to the binding sites of an enzyme. But let's assume the following possibilities:

- 'B . . #' is a blunt-cutting enzyme that binds to a recognition sequence 'B' and makes a blunt double cut (i.e., at the same location on both strands) indicated by #, at some non-critical distance toward 5'.
- '^ . . L' is a nicking enzyme that binds to a recognition sequence 'L', and makes a nick indicated by ^ toward 3', on the opposite strand, at a distance corresponding to a toehold length.
- 'R . . ^' is a nicking enzyme that binds to a recognition sequence 'R', and makes a nick indicated by ^ toward 5', on the opposite strand, at a distance corresponding to a toehold length.

The embedding of the 'B' binding sequence is straightforward because it can be placed outside the gates, in the surrounding DNA. The main gate structures however have several internal nicks; hence, the enzyme binding sites must be placed inside the signal domains (we cannot assume they can cut precisely at a very long distance from their binding site). Since these domains occur twice with different surrounding nicks, the placement of the binding sequences is non-trivial. However, the following scheme is adequate: each domain used for encoding signals has 'L' and 'R' enzyme binding sequences embedded as in Fig. 10 top left: they produce nicks at a toehold length just outside of the domain. For the staggered cutting of the end caps, though, it is non obvious that 'L' and 'R' would work together to produce a staggered double strand cut as indicated. Alternatively, two separate staggered-double-strand-cut enzymes need to be used there, with the stagger being the length of a toehold.

This scheme came from a discussion with Yuan-Jyue Chen, after he pointed out that the placement of restriction binding sequences was problematic.

⁵ <https://www.aatbio.com/data-sets/restriction-enzymes-cut-sites-reference-table>.

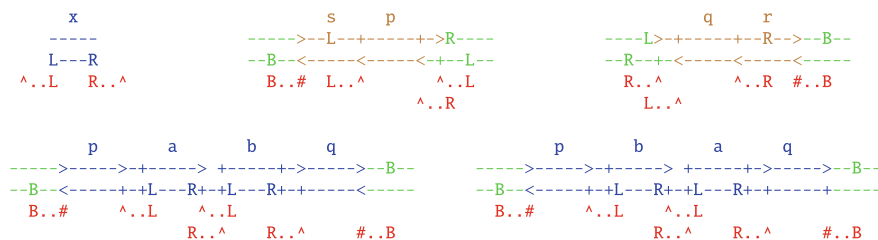


Fig. 10 Restriction enzyme scheme for the crosstalking choice gate. Enzymes are shown below the DNA structures, aligned to their binding sequences, with cutting points indicated by ‘^’ (opposite strand nick) and ‘#’ (blunt double cut). Top left: common pattern for all signal domains, with the bottom strand pointing left. Top middle and right: patterns for the end caps in context. Bottom: patterns for the main structures in context; note how the top left pattern leads to opening up the central toehold. Note also that ‘L’ on a left-pointing strand cuts to the left, but on a right-pointing strand cuts to the right; similarly for ‘R’ and ‘B’

References

1. A.V. Aho, M.R. Garey, J.D. Ullman, The transitive reduction of a directed graph. *SIAM J. Comput.* **1**(2), 131–137 (1972). <https://doi.org/10.1137/0201008>
2. L. Cardelli, Two-domain DNA strand displacement. *Math. Struct. Comput. Sci.* **23**(2), 247–271 (2013). <https://doi.org/10.1017/S0960129512000102>
3. S.H. Chan, B.L. Stoddard, S.Y. Xu, Natural and engineered nicking endonucleases—from cleavage mechanism to engineering of strand-specificity. *Nucleic Acids Res.* **39**(1), 1–18 (2010). <https://doi.org/10.1093/nar/gkq742>
4. Y.J. Chen, N. Dalchau, N. Srinivas, A. Phillips, L. Cardelli, D. Soloveichik, G. Seelig, Programmable chemical controllers made from DNA. *Nature* **8**(10), 755–762 (2013). <https://doi.org/10.1038/nnano.2013.189>
5. Chen, Y.J., Seelig, G.: *Scaling Up DNA Computing with Array-Based Synthesis and High-Throughput Sequencing*. In this volume (2021)
6. D.Y. Duose, R.M. Schweller, J. Zimak, A.R. Rogers, W.N. Hittelman, M.R. Diehl, Configuring robust DNA strand displacement reactions for in situ molecular analyses. *Nucleic Acids Res.* **40**(7), 3289–3298 (2012). <https://doi.org/10.1093/nar/gkr1209>
7. R.A. Hughes, A.D. Ellington, Synthetic DNA synthesis and assembly: putting the synthetic in synthetic biology. *Cold Spring Harb. Perspect. Biol.* **9**(1), a023812 (2017). <https://doi.org/10.1101/cshperspect.a023812>
8. D. Nadine, *Synthetic Molecular Machines for Active Self-assembly: Prototype Algorithms, Designs, and Experimental Study*. Dissertation (Ph.D.) (2013). <https://doi.org/10.7907/T0ZG-PA07>
9. L. Qian, D. Soloveichik, E. Winfree, Efficient turing-universal computation with DNA polymers, in *DNA Comput. Mol. Program.* ed. by Y. Sakakibara, Y. Mi (Springer, Berlin Heidelberg, Berlin, Heidelberg, 2011), pp.123–140
10. J.A. Reuter, D.V. Spacek, M.P. Snyder, High-throughput sequencing technologies. *Mol. Cell* **58**(4), 586–97 (2015). <https://doi.org/10.1016/j.molcel.2015.05.004>
11. G. Seelig, D. Soloveichik, D.Y. Zhang, E. Winfree, Enzyme-free nucleic acid logic circuits. *Science* **314**(5805), 1585–1588 (2006). <https://doi.org/10.1126/science.1132493>
12. S.L. Shipman, J. Nivala, J.D. Macklis, G.M. Church, CRISPR-Cas encoding of a digital movie into the genomes of a population of living bacteria. *Nature* **7663**, 345–349 (2017). <https://doi.org/10.1038/nature23017>

13. A.N. Sinyakov, V.A. Ryabinin, E. Kostina, Application of array-based oligonucleotides for synthesis of genetic designs. *Mol. Biol.* **55**, 487–500 (2021). <https://doi.org/10.1134/S0026893321030109>
14. D. Soloveichik, G. Seelig, E. Winfree, DNA as a universal substrate for chemical kinetics. *Proc. Natl. Acad. Sci.* **107**(12), 5393–5398 (2010). <https://doi.org/10.1073/pnas.0909380107>
15. T. Tanna, F. Schmidt, M.Y. Cherepkova, M. Okoniewski, R.J. Platt, Recording transcriptional histories using Record-seq. *Nat. Protoc.* **15**, 513–539 (2020). <https://doi.org/10.1038/s41596-019-0253-4>
16. D.Y. Zhang, Towards domain-based sequence design for DNA strand displacement reactions, in *DNA Comput. Mol. Program.* ed. by Y. Sakakibara, Y. Mi (Springer, Berlin Heidelberg, Berlin, Heidelberg, 2011), pp.162–175

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Computational Design of Nucleic Acid Circuits: Past, Present, and Future



Matthew R. Lakin, Carlo Spaccasassi, and Andrew Phillips

Abstract Over the past 40 years, significant progress has been made on the design and implementation of nucleic acid circuits, which represent the computational core of dynamic DNA nanotechnology. This progress has been enabled primarily by substantial advances in experimental techniques, but also by parallel advances in computational methods for nucleic acid circuit design. In this perspective, we look back at the evolution of these computational design methods through the lens of the Visual DSD system, which has been developed over the past decade for the design and analysis of nucleic acid circuits. We trace the evolution of Visual DSD over time in relation to computational design methods more broadly, and outline how these computational design methods have tried to keep pace with rapid progress in experimental techniques. Along the way, we summarize the key theoretical concepts from computer science and mathematics that underpin these design methods, weaving them together using a common running example of a simple Join circuit. On the occasion of the 40th anniversary of DNA nanotechnology, we also offer some thoughts on possible future directions for the computational design of nucleic acid circuits and how this may influence, and be influenced by, experimental developments.

1 Past

1.1 Visual DSD Origins

The first paper on what would subsequently become the Visual DSD system was published in 2009 and presented a simple programming language for nucleic acid circuit design [1]. The language was developed by observing the current state-of-the-

M. R. Lakin

Department of Computer Science, Department of Chemical & Biological Engineering, Center for Biomedical Engineering, University of New Mexico, Albuquerque, NM, USA
e-mail: mlakin@cs.unm.edu

C. Spaccasassi · A. Phillips (✉)

Microsoft Research, Cambridge, UK
e-mail: andrew.phillips@microsoft.com

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*, Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_18

311

art in nucleic acid circuits, which at the time relied primarily on toehold-mediated DNA strand displacement [2, 3], a powerful technique for implementing enzyme-free molecular computation programmed by sequence-specific DNA hybridization. This involves an invading single strand of DNA displacing an incumbent strand hybridized to a template strand, and is mediated by a short, single stranded region of DNA referred to as a *toehold*. The Visual DSD system was implemented using the functional programming language F# [4, 5], which facilitated the translation of its theoretical underpinnings to program code, and was released soon after as a web application [6], which simplified user adoption. DNA strand displacement has since been used to implement a broad range of computational circuits in DNA including digital logic [7], artificial neural networks [8, 9], and distributed algorithms [10], among many others [2, 3].

A key aspect of the origins and subsequent development of the Visual DSD system was the application of fundamental concepts from computer science in general, and programming language theory in particular, to formally represent corresponding concepts from dynamic DNA nanotechnology. We contend that dynamic DNA nanotechnology is therefore an embodiment of computer science in the truest sense, and illustrate this via examples of computer science methods that underpin the design and analysis of dynamic DNA nanotechnology systems.

1.1.1 Formal Syntax and Operational Semantics

Our work on the Visual DSD language was inspired by previous work on the use of *process calculi* to model biological systems. At the time it was already recognized that theoretical approaches originally developed to model concurrent computer systems could also be applied to model biological systems, given their inherent parallelism. A promising approach was the use of process calculi such as the *pi-calculus* [11], originally developed to model mobile computing systems such as telecommunications networks. Pi-calculus processes can create fresh names, send and receive them over *channels*, and spawn new *processes*. The first work on biological modeling using the stochastic pi-calculus [12] was carried out by Regev et al. [13] and subsequent work developed an *operational semantics* for a stochastic pi-calculus programming language and its corresponding implementation [14, 15]. Operational semantics formally defines the meaning of programs by specifying what happens when programs are executed, typically using *reduction* rules that determine a set of transitions from one program to another. The set of valid programs is defined by a formal *syntax*.

In the case of the pi-calculus, a simplified syntax can be defined as follows, where the options on the right, separated by short vertical bars, represent valid instances of the syntactic category on the left:

$$\begin{aligned}
 P &::= \pi.P \mid (P_1 \mid P_2) \mid \nu x P \\
 \pi &::= \bar{x}(y) \mid x(z)
 \end{aligned}$$

The definition states that a process P can be an action $\pi.P$, which runs the prefix π followed by P ; a parallel composition $(P_1 \mid P_2)$, which runs P_1 in parallel with P_2 ; or the creation of a fresh channel $\nu x P$, which creates a fresh channel x visible only to P . An action π can be a sender $\bar{x}(y)$, which sends the message y over channel x , or a receiver $x(z)$, which receives a message z on channel x . As is standard for syntax definitions, the syntax of a process is recursive, meaning that it can be unfolded to represent processes of arbitrary size.

A simplified operational semantics can be defined using a reduction rule (1) to specify the conditions under which two parallel processes can communicate; the term can then be *reduced* to the result of the computation step. In this rule, if a sender process $\bar{x}(y).P$ runs in parallel with a receiver process $x(z).Q$, then the sender and receiver can communicate on channel x , after which P and Q continue running in parallel, with the message y assigned to variable z in Q , written $Q_{\{y/z\}}$. To allow multiple communicating processes to run concurrently, a rule (2) states that if P can reduce to P' then the same reduction can still be applied when P runs in parallel with another process Q :

$$\bar{x}(y).P \mid x(z).Q \longrightarrow P \mid Q_{\{y/z\}} \quad (1)$$

$$\text{if } P \longrightarrow P' \text{ then } P \mid Q \longrightarrow P' \mid Q \quad (2)$$

Full definitions of pi-calculus syntax and semantics are provided in [11, 15, 16].

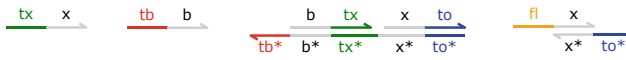
The Visual DSD language was inspired by the pi-calculus syntax and operational semantics. This includes formally defining dynamic DNA nanotechnology species as processes that can be composed in parallel and can interact via shared complementary domains, similar to how processes in the pi-calculus can send and receive messages on channels. To this end, the Visual DSD language used a simple syntax to represent a class of linear heteropolymer structures that mapped to a large fraction of the nucleic acid circuits being designed at the time, both theoretically and experimentally. This included *seesaw gates*, proposed by Qian and Winfree [17], which fit this paradigm and were used in two seminal papers in 2011 to implement large-scale digital logic circuits [7] and artificial neural networks [8]. In addition, Soloveichik et al. showed that this class of structures could be used to encode arbitrary chemical reaction networks as DNA circuits [18]. The syntax for this class of structures [1, 15, 19] can be summarized as follows:

$P ::= A \mid C \mid (P_1 \mid P_2)$	Strand, Complex or Parallel composition
$A ::= \langle S \rangle \mid \{S\}$	Upper or Lower strand
$C ::= \{S_L\}\langle S'_L \rangle[S]\langle S'_R \rangle\{S_R\}$	Segment with left and right overhangs
$\mid C_1 : C_2$	Complexes joined on Lower strand
$\mid C_1 :: C_2$	Complexes joined on Upper strand
$S ::= D_1 \dots D_N$	Sequence of Domains
$D ::= X \mid X^\wedge \mid X^* \mid X^{\sim}$	Long or short Domain, or its complement

The definition states that a Process P can either be a Strand A , a Complex C , or a parallel compositions of Processes $(P_1 \mid P_2)$. Strands are either Upper $\langle S \rangle$ or Lower

{S} and contain a Sequence $D_1 \dots D_N$ of Domains, where a Domain D may be a long domain (X), a short *toehold* domain (X^\wedge), or a *complement* of one of these, written X^* and X^{*^*} , respectively. Complexes are comprised of double stranded segments [S] which may have Upper and Lower strand overhangs on either side and can be concatenated along their Upper or Lower strands. This syntax allows complexes to be written compositionally, resulting in a close correspondence between syntax and structure that facilitates reading and writing of programs by the user. Note that here we use Upper and Lower to refer to the position of strands in a 2D graphical representation, which is common practice and aids in the visualization of complexes.

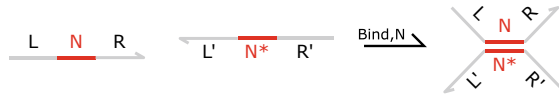
Using this syntax, we can program a simple *Join* circuit, which takes two inputs and produces one output as shown below, where the graphical representation is equivalent to the program code:



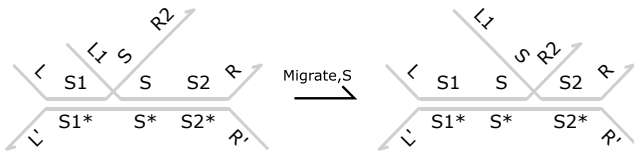
$(\langle tb^\wedge b \rangle \mid \langle tx^\wedge x \rangle \mid \{tb^{*\wedge}\}[b\ tx^\wedge]:[x\ to^\wedge] \mid \langle fl^\wedge \rangle[x]\{to^{*\wedge}\})$

This circuit will be used as a running example throughout the remainder of the paper. It illustrates the syntax of strands and complexes, which are the two main types of DNA *species*, and their parallel composition. As stated in the above formal definition, a strand is represented as a sequence of domains enclosed in angle brackets, where the 3' end of the strand is assumed to be on the right, represented graphically by an arrowhead, and a toehold domain is represented by appending the (\wedge) character to the domain name. For example, $\langle tb^\wedge b \rangle$ represents a strand consisting of the toehold domain tb^\wedge followed by the domain b . Note that a DNA strand can also be represented as a sequence of domains enclosed in curly brackets, where the 3' end of the strand is instead assumed to be on the left. For example, the strand $\langle tb^\wedge b \rangle$ can also be written as $\{b\ tb^\wedge\}$. This is because strands are identical up to *rotation symmetry*, such that we can write the same strand either from left to right or from right to left. A complex is represented as a sequence of *segments*, where each segment is a double stranded duplex with overhanging upper or lower strands to the left or right. Complementary domains are represented by appending the ($*$) character to the domain name. For example, the code $\{tb^{*\wedge}\}[b\ tx^\wedge]:[x\ to^\wedge]$ represents a complex consisting of two segments. The first segment $\{tb^{*\wedge}\}[b\ tx^\wedge]$ represents a double stranded duplex $[b\ tx^\wedge]$ consisting of the strand $\langle b\ tx^\wedge \rangle$ bound to its complementary strand $\{b^*\ tx^{*\wedge}\}$, with an additional single lower strand overhang $\{tb^{*\wedge}\}$ to the left of the duplex. The second segment $[x\ to^\wedge]$ represents a double stranded duplex consisting of the strand $\langle x\ to^\wedge \rangle$ bound to its complementary strand $\{x^*\ to^{*\wedge}\}$. These two segments are joined together along the lower strand by the operator ($:$). Although the textual representation of complexes is defined as a connection of segments, in reality the connection results in a single continuous lower strand, as can be seen by the graphical representation. More generally, this is also the case for complexes joined along the upper strand, with the potential for multiple disconnected bottom strands.

We also defined an operational semantics for the language, which formalizes how DNA species can interact. Using the syntax introduced above, we present two simplified rules: a *Bind* rule for toehold-mediated binding of strands, and a *Migrate* rule for rightward branch migration of an invading strand displacing an incumbent:



$$\langle L N^{\wedge} R \rangle | \{ L' N^{*} R' \} \xrightarrow{\text{Bind}, N} \{ L' \} \langle L \rangle [N] \langle R \rangle \{ R' \}$$



$$\{ L' \} \langle L \rangle [S_1] \langle S \rangle R_2 : \langle L_1 \rangle [S \ S_2] \langle R \rangle \{ R' \} \xrightarrow{\text{Migrate}, S} \{ L' \} \langle L \rangle [S_1 \ S] \langle R_2 \rangle : \langle L_1 \ S \rangle [S_2] \langle R \rangle \{ R' \}$$

In the rules above, the symbols L, L', L_1 and R, R', R_2 match a (potentially empty) sequence of domains. In the *Bind* rule, the upper strand $\langle L N^{\wedge} R \rangle$ binds to the lower strand $\{ L' N^{*} R' \}$ via the complementary toehold domains N^{\wedge} and N^{*} , producing a single complex $\{ L' \} \langle L \rangle [N] \langle R \rangle \{ R' \}$. A corresponding *Unbind* rule (not shown) would be similar, except that the direction of the reaction would be reversed. In the *Migrate* rule, the invading domains S that are unbound on the left-hand side of the rule become bound on the right-hand side of the rule, displacing the incumbent domains S that were previously bound. Given that the bound S_2 domains are also present, the strand itself remains attached to the complex. A corresponding *Migrate Left* rules is also needed for migration in the other direction (not shown). Full definitions of the Visual DSD syntax and semantics are provided in [1, 15, 19], including additional rules that allow reductions to take place inside joined complexes and parallel compositions.

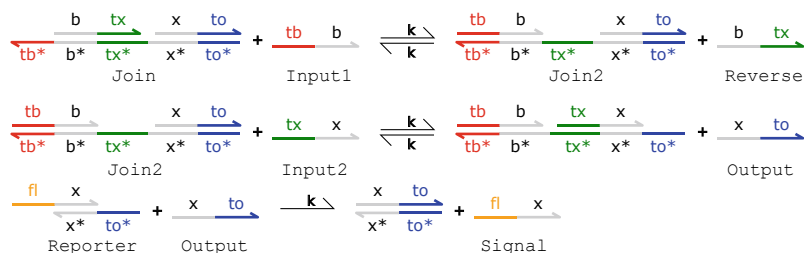
1.1.2 Chemical Reaction Networks

An additional contribution of the Visual DSD system was to formalize the *compilation* of a program, representing an initial set of DNA species, into a computational model describing how these species can interact with each other. We use the term *compilation* from computer science, where it denotes the translation of a higher-level language into a lower-level one – typically executable machine code. In the case of Visual DSD, the output of compilation is an executable kinetic model, formalized as a *chemical reaction network* (CRN) [20], which is defined as a set of *reactions*. Each reaction consists of a multiset of *reactant* species, a *reaction rate* and a multiset of

product species. CRNs bear many similarities to Petri nets, a graph-based formalism for distributed systems that takes the form of a bigraph containing places (analogous to chemical species) and transitions (analogous to chemical reactions), which has also been used to model biological systems [21, 22]. Here, the species of the CRN are the species of the Visual DSD program, and the compilation rules are defined by the operational semantics of the Visual DSD language. This allows compilation of arbitrary programs drawn from an infinite set. In this way, Visual DSD mirrors the *edit-compile-run* cycle of traditional computer programming. Essentially, the programmer uses their mental model of how DNA species interact to carefully design a program consisting of an initial set of species with an intended behavior. Compilation applies the rules of the Visual DSD language to the program to generate a CRN representing all possible interactions between species. The CRN is then executed using a chosen simulation algorithm for a particular set of initial conditions, resulting in an execution trace of the behavior of the species over time. The programmer can then revise the Visual DSD program if the observed behavior differs from the intended behavior.

Compilation of DNA species to CRNs is achieved by applying the reduction rules in a recursive loop to enumerate all possible reactions. Briefly, the compilation algorithm [1, 19] works by starting with an empty set of processed species and a set of unprocessed species corresponding to the species initially present in the Visual DSD program. At each step of the loop, the algorithm removes a species from the unprocessed set and computes all possible unimolecular reactions together with all possible bimolecular reactions involving the existing processed species. The resulting reactions can in turn generate new species, which are added to the unprocessed set. This loop is repeated until the set of unprocessed species is empty. In this way, the algorithm enumerates all species and reactions that can be generated from the initial species. We also generalized this approach to compile and simulate programs expressed in different languages [15], including the stochastic pi-calculus. In general, this approach allows for potentially unbounded numbers of species, meaning that compilation of Visual DSD programs may not terminate. As with most programming languages, checking for termination is not possible in general so it is up to the programmer to develop programs that terminate. To help mitigate this, Visual DSD also provides a just-in-time mode that interleaves compilation with simulation (see Sect. 1.2.1).

The compiled CRN for our running Join circuit example is as follows:



The first reaction in the forward direction is derived from the *Bind* rule, with $N = tb$, $R = b$, and $L, L', R' = \emptyset$. This is followed by an application of the *Migrate* rule, which displaces the b domain, with $S_1 = tb^{\wedge}$, $S = b$, $S_2 = tx^{\wedge}$, and $L, L', L_1, R_2, R, R' = \emptyset$. This is followed by a reaction to unbind the distal tx^{\wedge} toehold, which is an application of the *Unbind* rule with $N = tx$, $L = b$ and $L', R, R' = \emptyset$. In the figure above, these three reactions are merged into a single forward reaction, and similarly for the corresponding reverse reaction, resulting in the first reversible reaction shown above. The merge assumes that displacement and toehold unbinding reactions are much faster than toehold binding reactions, as defined by an *Infinite* DSD semantics [19], which we discuss further below. The other two reactions are derived similarly, where the final reaction is irreversible.

The behavior of the CRN can be summarized as follows, where strands and complexes are named for convenience. In the first reversible reaction, the Input1 strand $\langle tb^{\wedge} b \rangle$ binds to the Join complex $\{tb^{\wedge}*\} [b tx^{\wedge}] : [x to^{\wedge}]$ to produce a Reverse strand $\langle b tx^{\wedge} \rangle$ and an intermediate Join2 complex. In the second reversible reaction, the Input2 strand $\langle tx^{\wedge} x \rangle$ binds to the Join2 to complex produce an Output strand $\langle x to^{\wedge} \rangle$. In the third reaction, the Output strand binds to the Reporter complex $\langle fl^{\wedge} \rangle [x] \{to^{\wedge}*\}$ to produce a Signal strand $\langle fl^{\wedge} x \rangle$, whose fluorophore emits light that can be measured. Note that the Output is only produced if both inputs are present, which corresponds to the desired Join circuit behavior. This CRN can then be simulated and analyzed using a range of computational methods that facilitate nucleic acid circuit design, as outlined below.

1.2 Visual DSD Evolution

Building on our initial work, over time we sought to generalize the set of nucleic acid circuits that could be designed and analyzed with the Visual DSD system. This was motivated by corresponding experimental advances that implemented new types of nucleic acid structures and interactions not yet supported by Visual DSD. In addition, even when the Visual DSD system was first created there were a number of published experimental systems that were not supported, including hairpin assembly systems [23], hybridization chain reaction systems [24], and the original DNA strand displacement tweezer system of Yurke et al. [25]. Furthermore, these limitations in syntax meant that certain structures, such as branching structures, that could potentially be generated even when starting from the subset of supported linear structures, could not be represented. This further motivated the subsequent evolution and generalization of the Visual DSD syntax and semantics, which we outline below.

1.2.1 Polymer Structures (2011)

Some of our earliest work on expanding the set of nucleic acid structures supported by Visual DSD was to include unbounded *polymer* structures created by connecting multiple nucleic acid complexes. This was achieved by adding new semantic rules to enable multi-stranded complexes to bind via complementary overhanging sequences on the ends of the complexes (but not partway along, which would result in the formation of tree-like structures). Such polymer systems had previously been shown to enable the direct representation of a Turing-complete stack machine in nucleic acids [26].

A practical issue with these types of structures is the possibility of generating an infinite CRN, since polymers can grow without bound during reaction enumeration. To address this, we introduced a *just-in-time* (JIT) enumeration algorithm for stochastic simulation [15], which compiled reactions on-the-fly as they became possible during simulation. By interleaving reaction enumeration and simulation steps, the algorithm only enumerates the finite set of reactions that occur in a *single* stochastic simulation, rather than attempting to enumerate the infinite set of possible reactions.

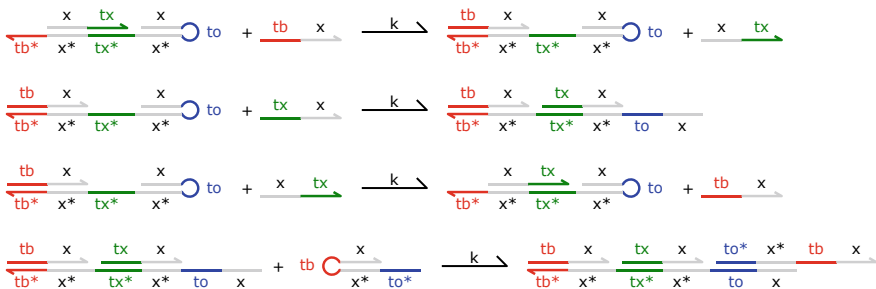
This approach draws on the notion of *just-in-time compilation* from computer science. In general, computer programs written in high-level languages are typically compiled into a lower-level language such as machine code, before they can be executed. This can be done ahead of time, such as for the C programming language, or at runtime while the program is being executed, such as for the Java programming language. This is called just-in-time compilation, where Virtual Machine bytecode such as Java bytecode is compiled into machine code at runtime, and it inspired our algorithm for stochastic simulation of potentially infinite CRNs [15].

We illustrate our approach by modifying our running Join circuit example as shown below, including replacing the double stranded to^{\wedge} duplex with a single stranded hairpin:



$$(\langle tb^{\wedge} x \rangle \mid \langle tx^{\wedge} x \rangle \mid \{tb^{\wedge}*\}[x tx^{\wedge}]:[x]\{to^{\wedge}\} \mid \langle tb^{\wedge}\rangle[x]\{to^{\wedge}*\})$$

This results in a hairpin-based Join circuit capable of forming unbounded polymers. Below are some of the initial reactions from the resulting CRN, as enumerated by Visual DSD using JIT compilation:



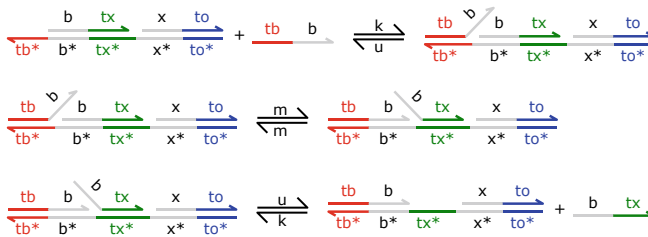
The last reaction demonstrates the binding of two complexes to open up the $\tau b^{\wedge} x$ hairpin, resulting in an exposed $\langle \tau b^{\wedge} x \rangle$ strand that can in turn interact with the Join complex to continue growing the polymer indefinitely. Note that here we omit the hairpin closing reactions, however these can be enabled as a type of *leak* reaction (see Sect. 1.2.3 and the Visual DSD manual for details).

Using this approach, we were able to simulate and analyze a Turing-complete stack machine in Visual DSD, based on an efficient encoding of stacks as DNA nanostructures [27].

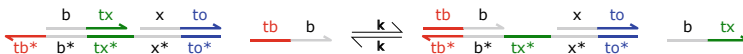
1.2.2 Semantic Abstractions (2012)

We also sought to enhance the flexibility of Visual DSD by encoding a number of distinct assumptions about DNA strand displacement kinetics as semantics rules. This produced an *abstraction hierarchy* of modeling assumptions [19], ranging from a *Detailed* semantics that explicitly models all toehold binding, branch migration, and toehold unbinding steps as distinct reactions, to an *Infinite* semantics in which branch migration and toehold unbinding are assumed to be instantaneous. These assumptions were implemented as options to the reaction enumeration algorithm. Infinite mode tends to be a good approximation at low concentrations, where unbinding and branch migration are substantially faster than binding and can be effectively modeled as infinitely fast. Detailed mode tends to be a better approximation at higher concentrations, though this comes with a higher computational cost.

An example of a strand displacement step compiled in *Detailed* mode is as follows, based on our running Join circuit example:



Here, binding, unbinding, and migration reactions are assumed to have finite rates k , u , and m , respectively. In contrast, when the circuit is compiled in *Infinite* mode the above 3 reversible reactions are merged into a single reversible reaction:



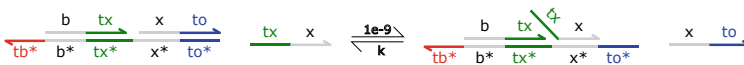
Here, the concentration of species is assumed to be sufficiently low that the rates of unbinding and migration reactions are infinite compared to the rates of binding reactions. As a result, strand displacement is assumed to take place in a single step that merges binding, migration, and unbinding. Since branch migration is infinite, complexes are also considered equal up to branch migration. More generally, this

approach allows a circuit to be analyzed at varying levels of detail without needing to modify the Visual DSD program, facilitating a trade-off between the accuracy of the model and the computational cost of the analysis [19].

1.2.3 Leaks (2012)

Experimental results were increasingly demonstrating that DNA strand displacements circuits often did not function as intended, in many cases due to the presence of *leak* reactions in which an invader strand displaces an incumbent at a low rate, in the absence of a complementary toehold domain. Such leaks are typically unintended because they do not follow a toehold-mediated reaction pathway and can be a major source of unwanted signal in experimental implementations of DNA circuits. In response to the growing body of experimental data demonstrating the occurrence of leaks, we extended the Visual DSD system to model these types of reactions [19]. We achieved this by extending the Visual DSD semantics with additional leak rules, which essentially correspond to versions of the branch migration rules in which no toehold is present and the invading strand is not part of the same complex as the incumbent.

A leak reaction occurs when the nucleotides at one extremity of a bound strand spontaneously unbind, creating a short toehold that facilitates a strand displacement reaction. In our running Join circuit example, the Input2 strand can displace a bound Output strand from the Join complex, even in absence of the Input1 strand. This happens when one or two nucleotides at the 5' end of the bound Output strand spontaneously unbind, creating a short toehold that allows the x domain of the Input2 strand to displace the Output, as follows:



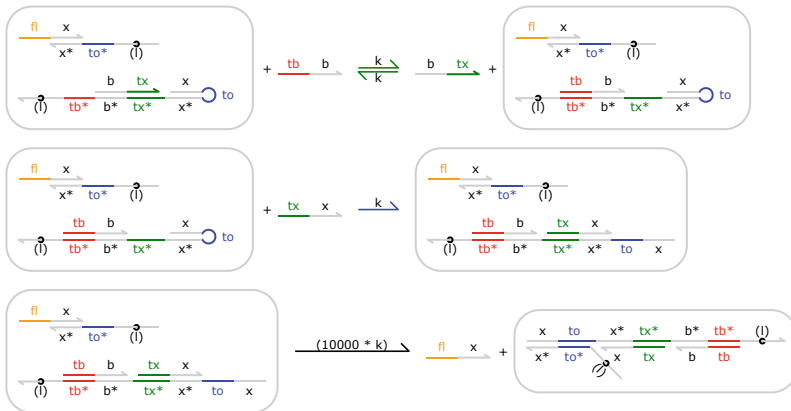
While the leak rate ($10^{-9} \text{ nM}^{-1} \text{ s}^{-1}$) is typically several orders of magnitude slower than the toehold-mediated strand displacement rate (k), over time it can still result in the accumulation of unwanted Signal strand even when only one of the two input strands is present, which is not the intended functionality of the Join circuit. More generally, this work highlights the potential of Visual DSD to model and predict experimental interactions that are not specifically intended by the circuit designer.

1.2.4 Localized Components (2014)

Another important development in the field of dynamic DNA nanotechnology was the design of spatially localized circuits on DNA origami tiles, first computationally [28, 29] and later experimentally [30, 31]. This approach combines aspects of dynamic and structural DNA nanotechnology to improve both the speed and scalability of circuit designs. To incorporate this new development, we generalized the Visual DSD

system by introducing syntax to represent nanostructures tethered to a tile surface [32]. Our initial approach was to *tag* tethered components with labels that indicate which components are tethered close enough to interact, and to model the effects of locality on reaction kinetics by associating a *local concentration* with each tag. This concentration was either estimated by the user or inferred from experimental data. We separately showed that SMT-based constraint solving can be used to determine satisfiability of the geometric constraints inherent in tethered molecular structures [33]. More recently, we used simple biophysical models to computationally estimate the rate constants for such localized reactions [34].

We illustrate our approach with a modified version of our running Join circuit example:

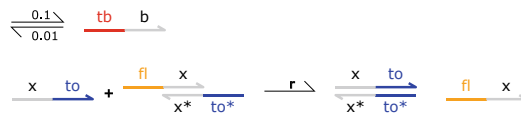


The `Join` species is modified by replacing the double stranded to^{\wedge} duplex with a single stranded to^{\wedge} hairpin and by adding a tether with location tag `l` on the 3' end of the tb^{\wedge} overhang. In addition, the `Reporter` is modified so that it contains a tether with the same location tag `l` on the 5' end of the to^{\wedge} overhang. This models the assumption that the two species are tethered close to each other, such that their effective concentration is given by `l`, here assumed to be 10,000 nM. In practice, these local concentrations can be estimated from data using parameter inference methods [30]. In the resulting CRN, the freely diffusing `Input1` strand $\langle tb^{\wedge} b \rangle$ binds to the `Join` complex tethered to the tile and displaces the $\langle b tx^{\wedge} \rangle$ strand. The freely diffusing `Input2` strand $\langle tx^{\wedge} x \rangle$ then opens the hairpin, exposing the to^{\wedge} toehold. This then binds to the exposed to^{\wedge} toehold of the tethered `Reporter` complex and displaces the `Signal` strand $\langle fl tx^{\wedge} \rangle$. The interaction is scaled by the local concentration `l`, since the `Reporter` and `Join` complex are tethered in close proximity to each other. Importantly, the resulting scaled rate $10000 * k$ of this reaction is unimolecular with units s^{-1} , since it involves two complexes tethered to the same origami at fixed locations, and therefore the interaction between these two tethered complexes is not affected by the concentration of strands in solution. This approach was used to model the kinetics of localized logic circuits, by inferring local concentrations from experimental data [30].

1.2.5 Custom Reactions (2014)

Throughout this period, a range of architectures for dynamic DNA nanotechnology were continually being developed, including some that made use of not only DNA strands but also DNA or RNA enzymes. Two prominent examples include the *PEN DNA toolbox*, developed in the Rondelez lab [35, 36], and the *Genelet* system, developed in the Winfree lab [37] and subsequently refined in the Schulman lab [38]. Other, related enzyme-driven architectures have also been developed [39, 40]. These architectures rely on DNA or RNA enzymes such as polymerases, exonucleases, and nickases to implement computational circuits. The PEN DNA toolbox, in particular, has a suite of multiple software tools for circuit design and analysis [41–43]. Since the enzymatic components of these systems were not supported in the Visual DSD system, which solely modeled nucleic acid interactions, we developed an extension to enable the insertion of custom, user-specific chemical reactions into the compiled CRN model.

A simple example of a custom reaction is shown below, by modifying our running Join circuit example.



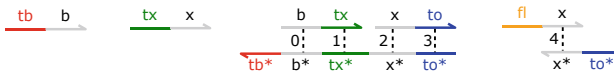
This models the assumption that the `Input1` strand `<tb^b>` is produced at rate 0.1 and consumed at rate 0.01 by enzymatic synthesis and degradation, respectively. Note that production is modeled here with a constant rate, hence no reactant is specified. Since Visual DSD does not support an explicit representation of enzymes, their effects are modeled at an abstract level using custom reactions. Furthermore, in the Join circuit example the default rate of the strand displacement reaction is equal to the rate k associated with the toehold `to^` that mediates the reaction. However, in practice we may wish to allow different reactions mediated by the same toehold to take place at different rates, in this case to model the fact that the presence of the fluorophore on the reporter complex gives rise to a different strand displacement rate r . We used custom reactions such as these to demonstrate the modeling of feedback control circuits in Visual DSD with three different architectures: strand displacement reactions, Genelets, and the PEN DNA toolbox [44].

1.2.6 Complex Topologies (2016)

The above versions of Visual DSD were all based on the original underlying syntax of linear polymers. However, by this time experimental techniques using complex branching structures were becoming increasingly common, but were not supported by this syntax. To keep pace with these experimental techniques, we developed a new version of Visual DSD that supported arbitrary graph structures, based on the notion of *strand graphs* [45]. Graphs consist of nodes connected by edges and have found application in many areas of computer science, such as modeling the structure of the Internet. We note that other work had previously used graphs to model

DNA nanostructures [46–48]. *Site graphs* are a generalization of graphs in which each node has multiple named *sites* and each edge connects two sites. Site graphs were already being used to model complex protein structures and their interactions, for instance by the Kappa language [49]. We proposed *strand graphs* [45] as a variant of site graphs in which the sites are ordered, where each node represents a strand and each site represents a domain, the order of the sites corresponds to the order of the domains in the strand, and an edge represents a bond between two complementary domains. The strand graph data structure is highly general, enabling arbitrary DNA nanostructures to be represented, including pseudoknots (structures with non-nested bonding patterns). We defined semantic rules that generalized the interactions between DNA strands to support these complex topologies, while also preserving all of the rules from previous versions of Visual DSD based on linear polymers [45]. For the remainder of this paper we refer to the Visual DSD syntax based on linear polymers as the *classic* syntax.

The complexes from our running Join circuit example can be expressed in strand graph syntax as follows, where the graphical representation above corresponds to the textual representation below:



```
( <tb^ b> | <tx^ x>
 | [ <b!0 tx^!1> | <x!2 to^!3>
 | <to^*!3 x*!2 tx^*!1 b*!0 tb^*> ]
 | [ <fl^ x!4> | <to^* x*!4> ] )
```

The program consists of a multiset of strands, separated by the parallel composition operator. As with the classic syntax there are two types of species, single strands and complexes. The single stranded Input1 species $\langle tb^ b \rangle$ and Input2 species $\langle tx^ x \rangle$ have the same textual representation as in the classic syntax, while the Join complex consists of two shorter strands $\langle x!2 to^!1 \rangle$ and $\langle b!4 tx^!3 \rangle$ bound to a longer strand $\langle to^*!1 x*!2 tx^*!3 b*!4 tb^* \rangle$, and the Reporter complex consists of a Signal strand $fl^ x!4$ bound to a Quencher strand $\langle to^* x*!4 \rangle$. Complexes are enclosed in square brackets, assuming that the strands within the square brackets form a *connected component*, meaning they are all connected to each other via named bonds, with autogenerated numbers used for new bonds generated during reaction enumeration. Complexes are considered equal up to renaming of bonds, meaning that individual bond names do not matter provided they are distinct. Note that the textual syntax assumes that all strands are written from the 5' end on the left to the 3' end on the right, whereas in the graphical representation the longer strand is rotated to align with the two shorter strands. This reflects the fact that complementary strands bind anti-parallel to each other, where the 3' end of one strand aligns with the 5' end of the other. In practice the bonds can be omitted from the graphical representation for conciseness and improved

readability, since they are only used to indicate connectivity. While the Join example shown can also be represented in the classic syntax, the strand graph syntax can be used to represent much more complex structures, including branching structures and pseudoknots. Side-conditions on the corresponding inference rules serve to limit applications of the rules that could generate physically implausible structures, such as binding within tight hairpin loops [45]. To illustrate this, we used the strand graph version of Visual DSD to analyze nucleic acid circuits with branching topologies that were previously implemented experimentally. We refer the reader to Petersen et al. [45] for details of the strand graph semantics that supports arbitrary structures, and to Spaccasassi et al. [50] for a more general semantics in terms of logic programming, which we discuss in Sect. 2.1.

1.3 Visual DSD Analysis

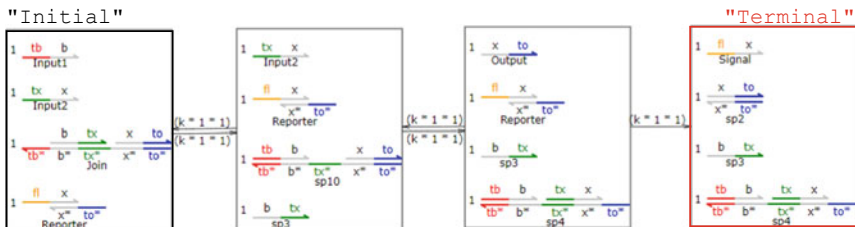
Over time, a number of model analysis capabilities were added to the Visual DSD system. This was facilitated by the formal underpinnings of Visual DSD including a well-established formal semantics, which allowed a range of computer-aided verification and analysis techniques to be brought to bear [51]. In this section we briefly outline some of these methods and illustrate their application using the running Join circuit example.

1.3.1 Probabilistic Model Checking (2012)

Model checkers can automatically verify whether a state-transition system satisfies a specification expressed in a temporal logic such as computation tree logic (CTL), which is a branching time logic. For example, the following CTL formula

$$A [F \text{ "terminal" }]$$

states that *all possible paths* (A) through the state space will *finally reach* (F) a “terminal” state. We can illustrate the application of this formula to the running Join circuit example, by analyzing the state space that is generated from an initial state containing one copy of each species. This state space can be represented graphically as follows:

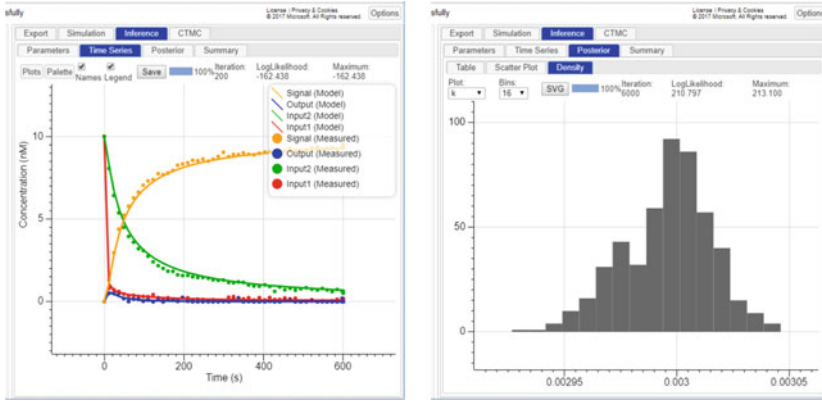


Each state contains a multiset of species, and each transition between states corresponds to the execution of a single reaction, where the rate of the transition is given by the rate of the reaction multiplied by the number of copies of each species involved in the reaction. Here the state space is relatively simple since only one copy of each species is present initially, however the number of states can grow exponentially with the number of copies. The “initial” and “terminal” states are labeled and outlined in bold using black and red colors, respectively, where the terminal state is defined as a state with no outbound edges, meaning that no reactions are possible from this state. Here we can see by inspection that the system satisfies the above formula, since it always converges to the "terminal" state.

For more complex state spaces we used the PRISM probabilistic model checker in combination with Visual DSD to verify a range of correctness properties expressed in a probabilistic temporal logic [52]. Using this approach, we showed how probabilistic model checking can identify design flaws resulting from circuit crosstalk, validate garbage collection schemes that clean up strands with exposed toe-holds, and compute the probability of reaching a consensus state in an approximate majority voting circuit [52]. This approach is tractable for relatively small numbers of molecules, where stochastic effects are important, such as spatially localized molecules. For large numbers of molecules in solution, system behavior can often be viewed as deterministic, in which case an ODE is generally a suitable approximation.

1.3.2 Model Simulation and Parameter Inference (2013)

Visual DSD was originally developed with built-in methods to simulate the behavior of a CRN over time and dynamically visualize the simulation output [6]. This immediate visual feedback of circuit behavior helped to accelerate the design of nucleic acid circuits. A standard stochastic simulator was implemented first, followed by a just-in-time stochastic simulator that supported unbounded CRNs [15]. A deterministic solver was also added, which generated ordinary differential equation (ODE) models from the CRN according to standard mass action kinetics. This deterministic solver was used to incorporate Bayesian parameter inference using Markov Chain Monte Carlo (MCMC), allowing model parameters to be inferred from experimental data. The plots below show fitting of the model parameters to measured datapoints (left) and examples of marginal posterior distributions for parameter values obtained via Bayesian parameter inference (right). The rate $k = 0.003/\text{nM/s}$ was inferred from the data points shown using simulations for initial concentrations of 10 nM `Input1` and `Input2`, 100 nM `Join`, and `Reporter`.



This parameter inference method was used to help design an experimental implementation of a molecular consensus algorithm using two-domain DNA strand displacement reactions [10], and a spatially localized architecture for fast and modular DNA computing [30].

1.3.3 Satisfiability Modulo Theories Solving (2013)

Satisfiability modulo theories (SMT) solvers generalize Boolean Satisfiability (SAT) solvers by incorporating additional theories, such as theories for integer-valued or real-valued arithmetic. This approach can be used to verify functional properties of nucleic acid circuits, including to guarantee properties of the terminal states of a circuit as a function of its initials state. One key advantage of using the SMT approach is that the analysis can scale to millions of copies of a circuit in parallel, though this comes at the cost of not being able to analyze temporal or probabilistic properties of the circuit. Another key advantage is that circuits can be verified for arbitrary input conditions, as opposed to a single combination of inputs.

We integrated the Z3 SMT solver [53] with Visual DSD and used this to prove the functional correctness of large-scale DNA strand displacement circuits [54]. We illustrate the method using the running Join circuit example, by verifying that this circuit satisfies the predicate $P_{\text{AND}}(q_0, q)$ defined in [54]. The predicate specifies that the system running from initial state q_0 to final state q produces a final quantity of output $q(O)$ that is the smaller of the two initial input quantities, $q_0(I_1)$ and $q_0(I_2)$:

$$P_{\text{AND}}(q_0, q) \iff q(O) = \min(q_0(I_1), q_0(I_2))$$

The method also identifies the precise constraints on the initial conditions that are required for the specification to be satisfied, in this case that the initial number of Reporter and Join complexes must be greater than the initial number of Input1 and Input2 strands. This allows us to prove the correctness of the circuit for arbitrary inputs, provided the constraints are satisfied. The logical behavior of

the AND circuit is formalized using a threshold θ , where an input or output strand represents a logical True if and only if the number of copies of the strand is greater than θ . We verify that the system satisfies the following formula, which specifies that the final output value $q(O)$ being above a threshold θ is equivalent to *both* initial input values $q_0(I_1)$ and $q_0(I_2)$ being above that threshold:

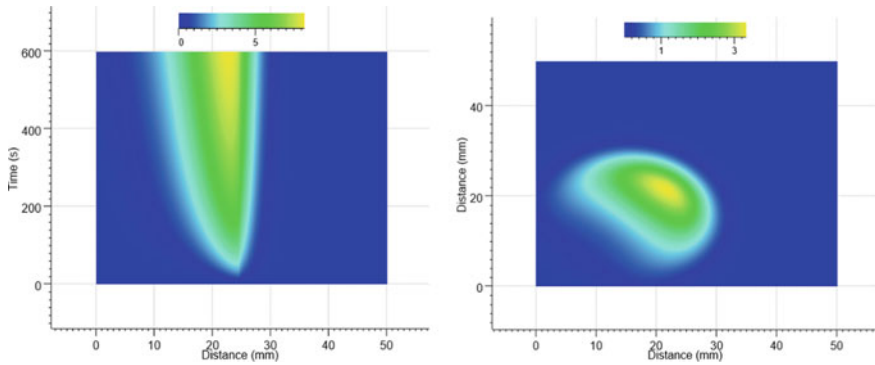
$$[q(O) > \theta] \iff [q_0(I_1) > \theta] \wedge [q_0(I_2) > \theta]$$

Together these formulae specify that the Join circuit will function as an AND logic gate on its two inputs. Using this approach, we verified the behavior of a 4-bit square root circuit, together with the components used for its construction, even when millions of copies of the circuits are interacting with each other in parallel. This method is also applicable to the analysis of chemical reaction networks with large species counts, to provide guarantees for arbitrary initial conditions, though without taking into account kinetic rates. This complements alternative methods such as interactive theorem proving, which has been used to verify coupled phase transitions in population protocols [55]. We have also used SMT solving to check the geometric constraints inherent in localized molecular circuit interactions [33].

1.3.4 Spatial Simulation with Partial Differential Equations (2014)

To keep pace with the increasing number of nucleic acid circuits being implemented in a spatial context at the time [56, 57], we added a partial differential equation (PDE) solver to Visual DSD to model diffusive aspects of spatially heterogeneous circuit designs. The general form of PDEs that Visual DSD can solve are those that can be described by the equation $\frac{\partial c}{\partial t} = f(c) + D\nabla^2 c$.

We illustrate this spatial simulation method using the running Join circuit example, by specifying the initial placement of the two input species. The `Input1` species is initialized with a value of 10 nM in a circular region of width 0.4 located at coordinates (0.3, 0.3), while the `Input2` species is initialized in a similar fashion at coordinates (0.7,0.7), where the coordinates are expressed as fractions of the dimensions of the 2D surface. As in the well-mixed system, we assume 100 nM `Join` and `Reporter` complexes in solution throughout. We also include additional spatial directives, which specify zero flux boundary conditions, and that the two input species diffuse at rate 0.5. The 2D spatial domain is a square of edge 50 mm, with a grid resolution of 100 divisions in each dimension and a simulation time step of 1s. The heat maps below shows the quantity of `Signal` using the 1-dimensional (left) and 2-dimensional (right) PDE solver.

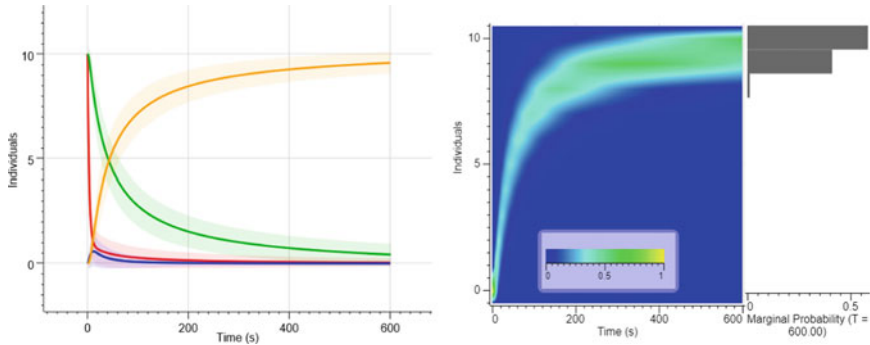


In both simulations we observe signal in the region of space where both signals overlap due to diffusion, consistent with the desired behavior of a spatial Join circuit. This approach was used to model a number of DSD systems with non-trivial spatial dynamics, including an autocatalytic network, a predator-prey system, and a spatial molecular consensus algorithm [58]. It was later used to assist with the design of spatial DNA-based communication in populations of synthetic protocells [59].

1.3.5 Probabilistic Model Checking with the Chemical Master Equation (2015)

While our previous probabilistic model checking approach was highly efficient for analyzing a system at a single point in time [52], we sought to substantially improve the efficiency of analysis over multiple time points. We achieved this through numerical integration of the Chemical Master Equation (CME), which is a system of ordinary differential equations whose solution yields the probability that the system is in a given state over time. For circuits with large numbers of molecules, a standard deterministic simulator based on mass action kinetics can typically be used. However, for circuits with small numbers of molecules, such as most localized circuits, methods such as CME integration are significantly more accurate.

We illustrate our method using the running Join circuit example. The plots below show the results of the analysis assuming 10 copies of the Input1 and Input2 strands and 100 copies of the Join and Reporter complexes. The plots show a timecourse (left) of the mean (solid line) and standard deviation (shaded region) of the Input1 (red), Input2 (green), Output (blue) and Signal (yellow), and a heatmap (right) of the full probability distribution for the Signal strand over time, together with a histogram of the probability distribution at the final time point.



This method is particularly well-suited to analyzing localized molecular circuits, since each location typically contains a small number of molecules. In the example above, even though only 10 copies of each input are present initially, the state space consists of 286 states, including a single terminal state, and 1100 transitions between these states. We analyzed a localized circuit for computing the square root of a four-bit number, proved its correctness with respect to its functional specification and analyzed the extent to which the localized design improves both speed and scalability in comparison to well-mixed circuits [60].

1.3.6 Verification of CRN Equivalence (2015)

A key goal in computer-aided design is to formally verify the correctness of systems in a modular way, such that verified components can be combined to build more complex systems that are correct by construction. We have illustrated how nucleic acid circuits can be modeled as CRNs that can be simulated and analyzed. However, CRNs are also a powerful means of *specifying* the intended behavior of nucleic acid circuits, where a high-level CRN specification is compiled to its low-level nucleic acid implementation [18]. An important challenge in the field has been to verify that a nucleic acid circuit is a correct implementation of a CRN specification.

We developed a technique for proving correctness of CRN implementations [61] based on the concept of *serializability* from database theory, which requires that interleaved concurrent updates to a database must be equivalent to some serial schedule of those updates. While this proof technique has not yet been implemented in the Visual DSD system, we can demonstrate its application using our running Join circuit example, by analyzing the state space diagram of the circuit from Sect. 1.3.1. We can prove that the Join circuit is a correct implementation of the CRN specification $\text{Input1} + \text{Input2} \rightarrow \text{Signal}$, where the species in this CRN denote the *formal species* of the Join circuit, and the `Join` and `Reporter` species denote the *fuels* of the circuit, defined as any species that must be present initially in order for the chemical reactions in the circuit to run to completion. The remaining species of the Join circuit are defined as either *intermediates* or *waste*. The proof relies on demonstrating that any trace generated by the circuit can be rewritten to produce a

serial trace which corresponds to a valid execution of the CRN specification. For this simple example we can show that the state space from Sect. 1.3.1 satisfies the required conditions for correctness from the theorems proved in [61], including that the terminal state is universally reachable and that every trace from the initial state has a *commit* reaction, which is an irreversible step where all formal reactants have been consumed before any formal products are produced. In this case the rightmost transition denotes a commit reaction. Thus, the Join circuit is a correct implementation of the CRN specification. Furthermore, the correctness is preserved in arbitrary contexts provided certain constraints are satisfied, including that the only shared species are formal species, waste, and certain intermediates where the gate design can tolerate the presence of additional copies of that species. This obviates the need to check the correctness of the circuit under different initial conditions or when it is composed with other circuits, which is a key advantage of our approach. We used this method to verify two different nucleic acid implementations of a distributed consensus algorithm, specified as a CRN with multiple reactions, where each reaction implementation was verified separately and composed in a modular fashion.

More recently, the Winfree group has reported a number of proof techniques based on pathway decomposition [62] and bisimulation-based approaches [63], which have been integrated into the Nuskell CRN-to-DSD compiler [64]. This provides an integrated system for designing and verifying CRN translation schemes. We refer the reader to Sect. 2.2 for further discussion of this work.

2 Present

2.1 Logic Programming Framework

As nucleic acid circuits continued to increase in complexity, we sought to develop a unifying framework that could support not only complex nucleic acid topologies but also both DNA and RNA enzymes. In addition, we sought to develop a system that was flexible enough to encode a range of alternative modeling hypotheses and to readily incorporate new dynamic DNA nanotechnology implementation strategies developed in the future. This led us to rebuild the Visual DSD system based on a logic programming framework [50], which we refer to as *Logic DSD*. Importantly, Logic DSD subsumes and unifies all previous syntactic and semantic extensions to the Visual DSD system, by implementing a single rule-based abstraction. The challenge in this work was to develop a modeling language that not only embodied a specific semantics but was also sufficiently expressive to allow new user-specified semantic rules to be defined within the language itself.

Logic programming is a powerful computational paradigm that originally found favor in the fields of knowledge representation and artificial intelligence. It allows the programmer to implement arbitrary rule sets in a framework that encapsulates the assumptions underlying their particular experimental system. Logic programs

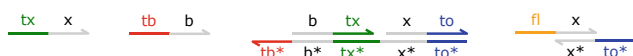
consist of collections of *facts* and *clauses*, which can be used in a proof search procedure to determine whether user-provided *queries* are satisfiable given those facts and clauses. As an example, the logic program

```
human("Socrates").      mortal(X) :- human(X).
```

specifies that Socrates is human (a fact) and that if X is human then X is also mortal (a clause). These can be used by a *resolution* engine to prove that Socrates is therefore mortal. In Logic DSD we use logic programming over strand graph representations of nucleic acid nanostructures, by generalizing the strand graph syntax developed previously [45]. This enables models to be compiled by proof search on semantic rules written in a logic programming language, which encodes specific user assumptions about how reactions and their rates are generated [50].

In Logic DSD, instead of the compilation rules being hard-coded within the language itself, the language includes a very general set of rules for proof search over nucleic acid structures [50]. This required the development of an *equational theory* of strands, which is a notion of equality between species that is required so that the system can determine the equivalence of candidate structures and those represented in the rules. This means that the rules that specify the desired semantics are provided as part of the program, alongside the circuit design. We define a default set of rules that corresponds to the semantics of previous versions of the Visual DSD system, however these can also be replaced by custom rules written by the user. This allows the user to potentially use different sets of rules for different types of circuits, depending on the modeling assumptions and implementation strategies, such as whether specific enzymes are present.

We illustrate Logic DSD using the running Join circuit example. The program code is the same as the strand graph code from Sect. 1.2.6, copied below for convenience:



```
( <tb^ b> | <tx^ x> | [ <fl^ x!4> | <to^* x*!4> ] |
  [<b!0 tx^!1> | <x!2 to^!3> | <to^*!3 x*!2 tx^*!1 b*!0 tb^*>])
```

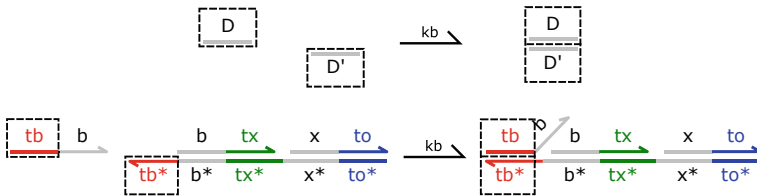
In addition, this code now needs to be accompanied by logical rules that define its semantics. The rules make use of *contexts* that are matched to the program in order for the rules to be applied. A given process can be matched to a *context* with N holes, written $C[]_1 \dots []_N$, where each hole represents a part of the structure that is matched to a rule and can be subsequently modified when the rule is applied.

Each hole in a rule is filled by a *pattern*, which can be one of the following: a strand $\langle S \rangle$ containing a sequence S of domains or logical variables that match a domain; the 5' end of a strand $\langle S$; the 3' end of a strand $S \rangle$; a sequence S of domains or logical variables; or a *nick* $S1 \rangle | \langle S2$ between two strands, which denotes a break. When a context is matched to a process, the patterns in the holes are matched to the corresponding parts of the process and the variable C is matched

to the rest of the process. This combination of contexts and patterns allows strand graph rewriting rules to be expressed. In addition, the use of predicates allows highly expressive conditions to be defined, which need to be satisfied in order for the rule to be applied.

We illustrate two of the main rules and their application to the Join circuit example below, and refer the reader to [50] for complete definitions. The following *bind* rule defines the semantics of binding, and is accompanied by a corresponding graphical representation of the rule and its application to the Join circuit example:

```
bind(P1, P2, Q, D!i) :-
    P1 = C1 [D], P2 = C2[D'], compl(D, D'),
    Q = C1 [D!i] | C2 [D'!i], freshBond(D!i, P1|P2).
reaction([P1;P2], "kb",Q) :- bind(P1,P2,Q,_).
```



When this rule is applied to the Join circuit example we have the following matches, indicated by boxes with dashed lines, which allows binding to take place on the toehold tb^{\wedge} :

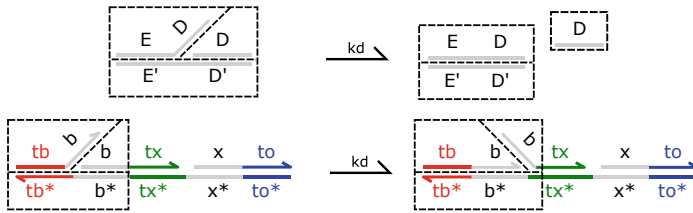
```
P1=C1[tb^{\wedge}], P2=C2[tb^{\wedge}*], Q=C1[tb^{\wedge}!5] | C2[tb^{\wedge}*!5]
```

The rule specifies that two processes $P1$ and $P2$ can bind provided $P1$ contains domain D and $P2$ contains domain D' such that D and D' are complementary, written $compl(D, D')$. If so, the resulting process Q replaces the matched domains with corresponding bound versions $D!i$ and $D'!i$, respectively, and places the two processes in parallel, where i does not occur anywhere else in the process, written $freshBond(D!i, P1|P2)$. The built-in *reaction* predicate is used as input to the built-in reaction enumeration algorithm, which generates a CRN in a similar fashion to the previous algorithm. Here the *reaction* predicate indicates that the reactants are processes $P1$ and $P2$, the product is Q and the rate of the reaction is kb . Note that in this case the rate of binding is fixed, however the language also allows arbitrary functions to be used to compute the rates, including associating the rate to the toehold and its surrounding context, such as whether it is at the 3' or 5' end of a strand [50].

Similarly, the following *displace* rule defines the semantics of strand displacement:

```
displace(P, Q, E!j, D!i) :-
    P = C [E!j D] [D!i] [D'!i E'!j],
```

```
Q = C [E!j D!1] [D] [D'!i E'!j].
reaction([P], "kd", Q) :- displace(P, Q, _, _).
```



Where the ordering of the holes in the graphical representation is from left to right then top to bottom. When this rule is applied to the Join circuit example we have the following matches:

```
P = C [tb^!5 b] [b!4] [b*!4 tb^*!5]
Q = C [tb^!5 b!4] [b] [b*!4 tb^*!5]
```

This approach essentially replaces the operational semantics described in Sect. 1.1.1 with an executable logic programming encoding that can be directly edited by the user and saved as part of the Visual DSD program.

More generally, we used this method to encode nucleic acid implementation strategies that make use of polymerase, exonoclease, and nickase enzymes. Our approach greatly simplifies the encoding of enzyme-based systems, by avoiding the need to manually encode each enzyme operation for all possible species. We also encoded implementation strategies that rely on localization to a substrate and complex nucleic acid topologies including branches and pseudoknots [50].

The set of compilation rules that can be expressed in this framework is extremely broad, as logic programming systems of this kind are known to be Turing-complete. This means that future experimental developments in dynamic DNA nanotechnology can be accommodated without rewriting the core of the Logic DSD system, simply by writing a new set of rules. In practice, however, expensive computations such as detailed biophysical simulations will likely prove impractical if used in rule definitions. Furthermore, proof search in logic programming is highly parallelizable, making it conceptually straightforward to harness the power of massively parallel computational hardware or cloud services to scale up proof search. In this way, the current version of the Visual DSD system could serve as a foundation for the design of future dynamic DNA nanotechnology systems based on experimental techniques that have yet to be conceived.

2.2 Related Work

Throughout the time that the Visual DSD system was under development, a number of groups were developing computational design tools for DNA circuits. These

approaches, and their temporal relationships to each other and to our own work, are summarized in Fig. 1. For a more detailed comparison of domain-specific languages for DNA circuit design we refer the reader to a review article on the topic [51]. Perhaps most notably, the Winfree group developed a range of tools over a number of years that have been integrated into automated pipelines for computational nucleic acid circuit design. For example, the Piperine design pipeline [68] enables the high-level design of CRN-based circuit architectures [72] and their subsequent compilation into lower-level DNA strand displacement implementations and ultimately into nucleotide sequences suitable for experimental testing.

The Nuskell system [64] is a compiler that converts abstract CRN designs into strand displacement implementations using CRN compilation schemes that can be specified by the user using a built-in domain-specific language for this task. The Nuskell system also incorporates verification capabilities, using approaches based on pathway decomposition [62] or bisimulation [63] to formally check that the generated strand displacement network is a correct implementation of the original input CRN in a rate-independent sense. This required the definition of a semantics for the domain-level representation so that a correspondence between domain-level species and the abstract species of the input CRN can be rigorously stated and verified. Nuskell uses the Peppercorn software [71] to enumerate non-pseudoknotted structures at the domain level. Peppercorn uses a set of rules similar in spirit to those of the Visual DSD system but expresses them in a pattern-based notation similar to “DU+” used by NUPACK [73] for concise specification of secondary structures. Of particular interest in Peppercorn is its explicit support for condensing reaction networks into simpler versions, in which multiple microstates can be combined into a single “resting state” connected only by relatively fast reactions. This is a more flexible approach to reaction merging than the system that we initially developed as part of the hierarchy of abstractions used in previous versions of the Visual DSD system [19].

Other related tools include the Multistrand stochastic simulator that simulates DNA nanostructures at the single-base level [67, 74] and the KinDA [70] system that predicts kinetics and thermodynamics of domain-level designs after specific nucleotide sequences have been assigned to those domains. The Seesaw compiler developed by the Qian group [69] enables digital logic circuits to be compiled into seesaw gate strand displacement circuits [17] and enables circuits to be implemented using even unpurified oligonucleotides. The DyNAMiC Workbench system [66] was developed as a web-based tool to target the “port-based” design abstraction developed for hairpin assembly systems by Yin et al. [23]. Finally, a range of design tools have also been developed specifically for the enzyme-driven PEN toolbox system [41], including several that automate various aspects of network design [42, 43].

Taken together, these examples illustrate the evolution of computational tools, including those based on the Visual DSD system, over this period of rapid growth in the capabilities and complexity of molecular computing circuits that could be implemented experimentally using dynamic DNA nanostructures.

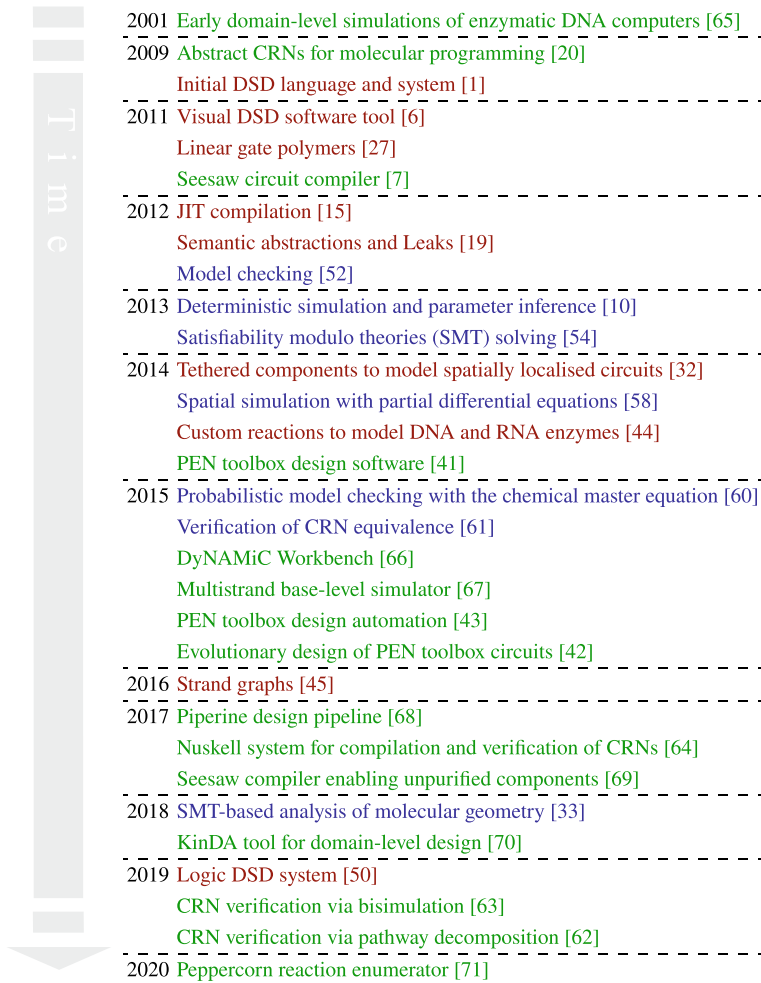


Fig. 1 Timeline of developments of the Visual DSD system and related work for high-level design and analysis of nucleic acid circuits. Red: Evolution of DSD syntax and semantics. Blue: Analysis techniques applied to DSD models. Green: Related work

3 Future

We conclude with some thoughts on what the future may hold for the computational design of nucleic acid circuits, looking ahead to the next 40 years of dynamic DNA nanotechnology at the intersection of computational tools, laboratory experiments, and biomedical applications.

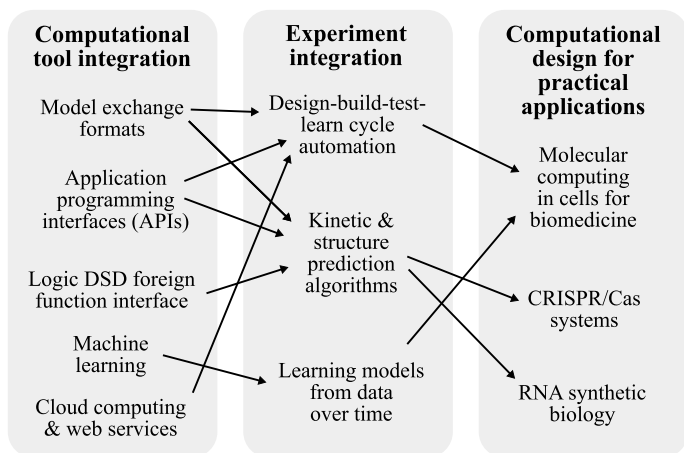


Fig. 2 Potential future developments for computational modeling in dynamic DNA nanotechnology, with some key dependencies highlighted as arrows

3.1 Computational Tool Integration

Enhancing the integration of computational tools has the potential to substantially improve the design of dynamic DNA nanotechnology systems. For instance, the Winfree group and others have integrated multiple tools that span a range of activities required to design and implement an experimental system, via the development of pipelines such as Nuskell [64] and Piperine [68]. More generally, this highlights the power of approaches in which a number of distinct categories of tools, such as nanostructure design tools [75], coarse-grained simulation tools [76], sequence-level design and analysis tools [70, 73, 77–79], CRN compilers [64, 68], reaction enumerators [6, 50, 71], and verification tools [62, 63] can be used in conjunction to analyze different aspects of system design.

Over time, tool integration will likely move from an ecosystem on a local machine to a cloud-based framework with interoperable services. The Visual DSD system adopted a web-based approach, with the most recent versions taking advantage of JavaScript compilation to allow execution directly in a browser on the client machine, eliminating the need for server compute infrastructure. While convenient for deployment, this limits the user to the power of their equipment. To facilitate remote execution on a cluster, a command-line executable of Visual DSD was also developed. While this facilitated increased computational performance, it came at the cost of usability, where the web-based graphical user interface was replaced with a command-line text interface, and integration with computational infrastructure needed to be manually configured depending on the infrastructure being used. To enable both usability and scalability, we envisage a graphical user interface with a built-in option to either run locally or run on the cloud through a simple checkbox, unleashing the power of the cloud on demand without compromising ease of use.

As more tools continue to move to the cloud, software integration will likely take place via web services that scale according to the computational resources required (Fig. 2).

To enable tools from multiple groups to interoperate seamlessly, whether through local interfaces or cloud-based web services, common data interchange formats will also be needed. For example, the Synthetic Biology Markup Language (SBOL) [80] has been developed for the synthetic biology community to share genetic circuit designs and is part of a family of interchange formats for systems and synthetic biology [81]. The Systems Biology Markup Language (SBML) [82] provides a common interchange format for kinetic models expressed primarily as chemical reaction networks. To date, there are limited interchange formats for dynamic DNA nanotechnology. In future, as DNA nanotechnology and synthetic biology continue to converge, there may be an opportunity to extend the SBOL standard, or to create new standards based on existing formats such as the *dotparen* notation for secondary structure or *Cadnano* JSON files for structural DNA nanotechnology designs. In addition to ongoing grassroots efforts, existing organizations such as the International Society for Nanoscale Science, Computation, and Engineering (ISNSCE) could play a coordinating role.

There is also an opportunity to facilitate the development of computational tools for DNA nanotechnology through better programming language integration. Recent work on the NUPACK system included an application programming interface (API) for the Python programming language [77], thereby enabling programmatic access to tool features using a general purpose programming language. We have previously argued [51] that domain-specific programming languages offer powerful tools for the development of design systems for particular application domains, such as dynamic DNA nanotechnology. This view is not inconsistent with the desire for an integrated toolchain glued together by a general purpose scripting language such as Python. In Visual DSD, improved integration could be achieved by adding the ability to call out to external functions, for instance to determine rate constants and to enumerate reactions based on the output of external solvers such as the Z3 SMT solver [53], nucleic acid design tools such as NUPACK [78, 79], or coarse-grained biophysical modeling tools such as oxDNA [76]. From a language implementation perspective, this would require a *foreign function interface* (FFI) to retrieve the results of calls to external tools: this could build on previous work on functional logic programming systems [83] that integrate logic programming-style proof search with function calls. From a computational feasibility perspective, oxDNA simulations are often time consuming and can sometimes take several days, so relationships would most likely need to be determined first by oxDNA or experimentally and then encoded in the logic programming framework.

Finally, the potential for machine learning to be applied to biodesign has been widely acknowledged [84], including for example the opportunity to use machine learning tools to predict DNA interactions from sequences [85, 86], using both feature-based and recurrent neural network-based approaches, and incorporating domain knowledge into the models. The significant computational power of modern GPUs could also be harnessed for computational design and simulation of DNA

nanotechnology systems, as demonstrated by recent work applying differentiable programming to design gene regulatory networks [87]. As mentioned above, cloud computing resources could be harnessed for computational design by exploiting the highly parallelizable nature of proof search in logic programming systems such as Logic DSD.

3.2 *Experiment Integration*

Experimental techniques in DNA nanotechnology have continued to rapidly advance in the decade since Qian and Winfree reported their seminal “four-bit square root” strand displacement circuit [7]. As things stand, sequence design tools such as NUPACK [78, 79] are almost indispensable for designing such circuits. As circuit complexity continues to increase, computational design tools at the domain level will become increasingly valuable for debugging failure modes and tuning system dynamics. When it comes to analyzing error modes of these systems, such as unintended *leak* interactions between DNA strands, already we are reaching the limits of what is possible computationally, even for very simple circuits. For instance, leak analysis in Visual DSD can generate thousands of unintended reactions even for very simple circuits, due to the combinatorial explosion of variant structures produced via leaks, which can themselves undergo further leak reactions, recursively. Recent promising developments have included the design of circuits that resist leak [88], which could facilitate the implementation of substantially larger circuits. More generally, the need for higher-level computational design tools will likely only grow as circuit complexity increases due to further improvements in experimental techniques.

Furthermore, recent developments in areas such as enzyme-driven DNA circuits [40] hint at the future convergence of DNA nanotechnology with synthetic biology, which will likely open new areas for modeling the interactions of DNA-based molecular devices with biological and biochemical components, including in living cells. In terms of keeping pace with such developments, one goal of the newest Logic DSD system [50] is to provide headroom by adopting a Turing-powerful language in which to express the semantic rules used for compilation of structural models into kinetic ones. In principle, end-users can reprogram the behavior of this DSD compiler to implement the specific assumptions of their experimental system, which is embodied by the user-supplied rules rather than hard-coded into the semantics of the DSD compiler itself.

In the longer term, there is a need to not only integrate the different modeling methodologies but also to track the evolution of knowledge over time, in the form of computational models. Previous work has investigated training models of nucleic acid circuits through parameter inference methodologies [10, 30]. In future we envisage an approach where computational models are at the center of the *Design-Build-Test-Learn* (DBTL) cycle, and continually updated over time as new experiments are performed.

We also note the substantial potential for the integration of new and future experimental techniques for high-throughput testing and readout of DNA-based molecular circuits. Most notably, high-throughput sequencing methods and nanopore sequencing may enable large-scale interrogation of the dynamics of DNA circuits, which has previously been limited to the observation of a relatively small number of signals due to spectral overlap issues inherent in the use of fluorescent reporters. Similar techniques could be used for larger-scale data collection, enabling more data to be gathered on the internal states of DNA circuits during execution.

Future directions in experimental DNA nanotechnology could facilitate system design by improving the correspondence between computational modeling and experimental reality. For instance, a key issue in computational modeling of experimental systems is ensuring that the initial conditions of the experiment match the initial conditions of the simulation. In previous work, this has been achieved in a number of ways, including explicitly modeling some fraction of *inactive* gates to account for incorrectly assembled gates [10] or by adjusting the initial concentrations of components in the experimental system to make the fraction of *active* gates match that required by the circuit design [69]. Perhaps a future research challenge in experimental DNA computing could be to aim toward “Ångström resolution” in *dynamic* DNA nanotechnology, similar to the goal set for the field of *structural* DNA nanotechnology [89]. This might be a more challenging task as it would involve not just precision in placement of atoms in the *initial* state of the system but also in the programming of the subsequent pathways taken when the system computes over time. A possible initial goal would be to reliably construct just the initial state of a system, in terms of filtering out incorrectly synthesized or misfolded structures at the single-molecule level. At present, this is largely done by annealing followed by manual purification via polyacrylamide gel electrophoresis (PAGE). However, this does not guarantee that every gate in the sample will necessarily be correctly formed, opening up the possibility of errors. Future advances in single-molecule analysis, such as imaging or nanopore sequencing, could enable structures the size of currently-used DNA strand displacement gates to be analyzed, and perhaps even sorted, to produce highly pure samples for use in molecular computing reactions that can more faithfully reproduce the starting conditions assumed in the models used for computational design. Alternatively, high-fidelity biological synthesis of both DNA [10] and RNA [90] strand displacement components could improve the purity of experimental systems.

3.3 Computational Design for Practical Applications

A key challenge for molecular computing as a field is to successfully demonstrate practical applications for the technologies that have been developed over recent years. One possibility here is the use of dynamic DNA nanotechnology within living organisms to sense and control biological networks [91, 92]. These could be used for cell labeling [93], for diagnostic applications [94], for therapeutic effects (e.g.,

by silencing specific genes) [95], or even to build theranostic (therapeutic and diagnostic) systems that autonomously detect and treat diseases. Such applications in nanomedicine would exploit the innate biocompatibility of DNA nanotechnology to carry out computation within living cells, where traditional silicon-based microprocessors cannot operate.

Previous work on implementing strand displacement reactions in living cells [92, 96, 97], including more recently using heterochiral DNA nanotechnology [98, 99], brings this possibility closer. However, even with chemical modifications added to the DNA, the interactions of DNA-based molecular computing systems with living cells are complex. This highlights a potentially fruitful avenue of future work in the modeling and design of computational nucleic acid systems: to interface nucleic acid circuit models with whole-cell models of cellular processes [100], including predicting, and designing against, degradation of circuit components by nuclease enzymes and other forms of interference.

Finally, over the next 40 years we anticipate that the fields of DNA nanotechnology and synthetic biology will continue to converge. In particular, CRISPR/Cas systems for RNA-guided gene editing [101] and transcriptional regulation [102] have recently been integrated with strand displacement reactions for programmable control of CRISPR targeting [103–106]. Similar approaches have used strand displacement reactions to regulate translation via RNA-based “toehold switches” [107, 108] and transcription via small transcription-activating RNAs (STARs) [109]. More generally, the control of biological systems via “output interfaces” such as the CRISPR/Cas system is likely to be a substantial growth area for applications of dynamic DNA (and RNA) nanotechnology. We anticipate further opportunities to apply rule-based modeling tools such as Visual DSD to hybrid systems that span both fields. We also anticipate that integrating computational design tools for DNA circuits with tools specific to the application domain will be critical, paving the way for new and exciting applications of DNA nanotechnology over the next 40 years.

Acknowledgements We would like to sincerely thank Erik Winfree, Dominic Scalise, Stefan Badelt and an anonymous reviewer for their detailed and valuable comments on this manuscript.

References

1. A. Phillips, L. Cardelli, A programming language for composable DNA circuits. *J. R. Soc. Interface* **6**, S419–S436 (2009)
2. D.Y. Zhang, G. Seelig, Dynamic DNA nanotechnology using strand-displacement reactions. *Nat. Chem.* **3**, 103–113 (2011)
3. F.C. Simmel, B. Yurke, H.R. Singh, Principles and applications of nucleic acid strand displacement reactions. *Chem. Rev.* **119**(10), 6326–6369 (2019)
4. M.R. Lakin, A. Phillips, Compiling DNA strand displacement reactions using a functional programming language, in *Proceedings of Practical Aspects of Declarative Languages 2014*, ed. by M. Flatt, H.-F. Guo, vol. 8324 of Lecture Notes in Computer Science, pp. 81–86 (Springer International Publishing Switzerland, 2014)

5. D. Syme, The early history of $\#$. Proc. ACM Program. Lang **4**(HOPL), 75 (2020)
6. M.R. Lakin, S. Youssef, F. Polo, S. Emmott, A. Phillips, Visual DSD: a design and analysis tool for DNA strand displacement systems. Bioinformatics **27**(22), 3211–3213 (2011)
7. L. Qian, E. Winfree, Scaling up digital circuit computation with DNA strand displacement cascades. Science **332**, 1196–1201 (2011)
8. L. Qian, E. Winfree, J. Bruck, Neural network computation with DNA strand displacement cascades. Nature **475**, 368–372 (2011)
9. K.M. Cherry, L. Qian, Scaling up molecular pattern recognition with DNA-based winner-take-all neural networks. Nature **559**, 370–376 (2018)
10. Y.-J. Chen, N. Dalchau, N. Srinivas, A. Phillips, L. Cardelli, D. Soloveichik, G. Seelig, Programmable chemical controllers made from DNA. Nat. Nanotechnol. **8**, 755–762 (2013)
11. R. Milner, J. Parrow, D. Walker, A calculus of mobile processes. I. Inf. Comput. **100**(1), 1–40 (1992)
12. C. Priami, Stochastic π -calculus. Comput. J. **38**(7), 1995 (1995)
13. A. Regev, W. Silverman, E. Shapiro, Representation and simulation of biochemical processes using the pi-calculus process algebra, in *Proceedings of the Pacific Symposium on Biocomputing*, vol. 2001, pp. 459–470 (2001)
14. A. Phillips, L. Cardelli, Efficient, correct simulation of biological processes in the stochastic pi-calculus, in *Computational Methods in Systems Biology*, vol. 4695 of Lecture Notes in Computer Science, pp. 184–199 (Springer, 2007)
15. M.R. Lakin, L. Paulevé, A. Phillips, Stochastic simulation of multiple process calculi for biology. Theoret. Comput. Sci. **431**, 181–206 (2012)
16. D. Sangiorgi, D. Walker, *The Pi-Calculus—A Theory of Mobile Processes* (Cambridge University Press, 2001)
17. L. Qian, E. Winfree, A simple DNA gate motif for synthesizing large-scale circuits. J. R. Soc. Interface **8**(62), 1281–1297 (2011)
18. D. Soloveichik, G. Seelig, E. Winfree, DNA as a universal substrate for chemical kinetics. Proc. Natl. Acad. Sci. USA **107**(12), 5393–5398 (2010)
19. M.R. Lakin, S. Youssef, L. Cardelli, A. Phillips, Abstractions for DNA circuit design. J. R. Soc. Interface **9**(68), 460–486 (2012)
20. M. Cook, D. Soloveichik, E. Winfree, J. Bruck, Programmability of chemical reaction networks, in *Algorithmic Bioprocesses*, ed. by A. Condon, D. Harel, J.N. Kok, A. Salomaa, E. Winfree, pp. 543–584 (Springer, 2009)
21. P.J.E. Goss, J. Peccoud, Quantitative modeling of stochastic systems in molecular biology by using stochastic Petri nets. Proc. Natl. Acad. Sci. USA **95**(12), 6750–6755 (1998)
22. M. Heiner, D. Gilbert, R. Donaldson, Petri nets for systems and synthetic biology, in *SFM 2008*, ed. by M. Bernardo, P. Degano, G. Zavattaro, vol. 5016 of Lecture Notes in Computer Science, pp. 215–264 (Springer, 2008)
23. P. Yin, H.M.T. Choi, C.R. Calvert, N.A. Pierce, Programming biomolecular self-assembly pathways. Nature **451**, 318–322 (2008)
24. R.M. Dirks, N. Pierce, Triggered amplification by hybridization chain reaction. Proc. Natl. Acad. Sci. USA **101**(43), 15275–15278 (2004)
25. B. Yurke, A.J. Turberfield, A.P. Mills Jr., F.C. Simmel, J.L. Neumann, A DNA-fuelled molecular machine made of DNA. Nature **406**, 605–608 (2000)
26. L. Qian, D. Soloveichik, E. Winfree, Efficient turing-universal computation with DNA polymers, in *Proceedings of the 16th International Conference on DNA Computing and Molecular Programming*, ed. by Y. Sakakibara, Y. Mi, vol. 6518 of Lecture Notes in Computer Science, pp. 123–140 (Springer, 2011)
27. M.R. Lakin, A. Phillips, Modelling, simulating and verifying Turing-powerful strand displacement systems, in *Proceedings of the 17th International Conference on DNA Computing and Molecular Programming*, ed. by L. Cardelli, W. Shih, vol. 6937 of Lecture Notes in Computer Science, pp. 130–144 (Springer, 2011)

28. H. Chandran, N. Gopalkrishnan, A. Phillips, J. Reif, Localized hybridization circuits, in *Proceedings of the 17th International Conference on DNA Computing and Molecular Programming*, ed. by L. Cardelli, W. Shih, vol. 6937 of Lecture Notes in Computer Science, pp. 64–83 (Springer, 2011)
29. R.A. Muscat, K. Strauss, L. Ceze, G. Seelig, DNA-based molecular architecture with spatially localized components, in *ISCA '13: Proceedings of the 40th Annual International Symposium on Computer Architecture*, pp. 177–188 (2013)
30. G. Chatterjee, N. Dalchau, R.A. Muscat, A. Phillips, G. Seelig, A spatially localized architecture for fast and modular DNA computing. *Nat. Nanotechnol.* **12**, 920–927 (2017)
31. H. Bui, S. Shah, R. Mokhtar, T. Song, S. Garg, J. Reif, Localized DNA hybridization chain reactions on DNA origami. *ACS Nano* **12**(2), 1146–1155 (2018)
32. M.R. Lakin, R. Petersen, K.E. Gray, A. Phillips, Abstract modelling of tethered DNA circuits, in *Proceedings of the 20th International Conference on DNA Computing and Molecular Programming*, ed. by S. Murata, S. Kobayashi, vol. 8727 of Lecture Notes in Computer Science, pp. 132–147 (Springer International Publishing, 2014)
33. M.R. Lakin, A. Phillips, Automated analysis of tethered DNA nanostructures using constraint solving. *Natural Comput.* **17**(4), 709–722 (2018)
34. S. Kumar, J.M. Weisburd, M.R. Lakin, Structure sampling for computational estimation of localized DNA interaction rates. *Sci. Rep.* **11**, 12730 (2021)
35. K. Montagne, R. Plasson, Y. Sakai, T. Fujii, Y. Rondelez, Programming an in vitro DNA oscillator using a molecular networking strategy. *Mol. Syst. Biol.* **7**, 466 (2011)
36. A. Baccouche, K. Montagne, A. Padirac, T. Fujii, Y. Rondelez, Dynamic DNA-toolbox reaction circuits: a walkthrough. *Methods* **67**(2), 234–249 (2014)
37. J. Kim, E. Winfree, Synthetic in vitro transcriptional oscillators. *Mol. Syst. Biol.* **7**, 465 (2011)
38. S.W. Schaffter, R. Schulman, Building in vitro transcriptional regulatory networks by successively integrating multiple functional circuit modules. *Nat. Chem.* **11**, 829–838 (2019)
39. J.Y. Kishi, T.E. Schaus, N. Gopalkrishnan, F. Xuan, P. Yin, Programmable autonomous synthesis of single-stranded DNA. *Nat. Chem.* **10**, 155–164 (2018)
40. T. Song, A. Eshra, S. Shah, H. Bui, D. Fu, M. Yang, R. Mokhtar, J. Reif, Fast and compact DNA logic circuits based on single-stranded gates using strand-displacing polymerase. *Nat. Nanotechnol.* **14**, 1075–1081 (2019)
41. N. Aubert, C. Mosca, T. Fujii, M. Hagiya, Y. Rondelez, Computer-assisted design for scaling up systems based on DNA reaction networks. *J. R. Soc Interface* **11**, 20131167 (2014)
42. H.Q. Dinh, N. Aubert, N. Noman, T. Fujii, Y. Rondelez, H. Iba, An effective method for evolving reaction networks in synthetic biochemical systems. *IEEE Trans. Evol. Comput.* **19**(3), 374–386 (2015)
43. H.W.H. van Roekel, L.H.H. Meijer, S. Masroor, Z.C.F. Garza, A. Estévez-Torres, Y. Rondelez, A. Zagaris, M.A. Peletier, P.A.J. Hilbers, T.F.A. de Greef, Automated design of programmable enzyme-driven DNA circuits. *ACS Synth. Biol.* **4**(6), 735–745 (2015)
44. B. Yordanov, J. Kim, R.L. Petersen, A. Shudy, V.V. Kulkarni, A. Phillips, Computational design of nucleic acid feedback control circuits. *ACS Synth. Biol.* **3**(8), 600–616 (2014)
45. R.L. Petersen, M.R. Lakin, A. Phillips, A strand graph semantics for DNA-based computation. *Theoret. Comput. Sci.* **632**, 43–73 (2016)
46. S. Sahu, B. Wang, J.H. Reif, A framework for modeling DNA based molecular systems. *J. Comput. Theoret. Nanosci.* **5**(11), 2124–2134 (2008)
47. I. Kawamata, N. Aubert, M. Hamano, M. Hagiya, Abstraction of graph-based models of biomolecular reaction systems for efficient simulation, in *Computational Methods in Systems Biology: CMSB 2012*, ed. by D. Gilbert, M. Heiner, vol. 7605 of Lecture Notes in Computer Science, pp. 187–206 (Springer, 2012)
48. R. Mokhtar, S. Garg, H. Chandran, H. Bui, T. Song, J. Reif, Modeling DNA nanodevices using graph rewrite systems, in *Advances in Unconventional Computing Volume 2: Prototypes, Models and Algorithms*, ed. by A. Adamatzky, vol. 23 of Emergence, Complexity, and Computation, Chap. 15, pp. 347–395 (Springer International Publishing, 2017)
49. V. Danos, C. Laneve, Formal molecular biology. *Theoret. Comput. Sci.* **325**, 69–110 (2004)

50. C. Spaccasassi, M.R. Lakin, A. Phillips, A logic programming language for computational nucleic acid devices. *ACS Synth. Biol.* **8**(7), 1530–1547 (2019)
51. M.R. Lakin, A. Phillips, Domain-specific programming languages for computational nucleic acid systems. *ACS Synth. Biol.* **9**(7), 1499–1513 (2020)
52. M.R. Lakin, D. Parker, L. Cardelli, M. Kwiatkowska, A. Phillips, Design and analysis of DNA strand displacement devices using probabilistic model checking. *J. R. Soc. Interface* **9**(72), 1470–1485 (2012)
53. L. de Moura, N. Bjørner, Z3: An efficient SMT solver, in *Proceedings of TACAS 2008*, vol. 4963 of LNCS, pp. 337–340 (Springer, 2008)
54. B. Yordanov, C.M. Wintersteiger, Y. Hamadi, A. Phillips, H. Kugler, Functional analysis of large-scale DNA strand displacement circuits, in *Proceedings of the 19th International Conference on DNA Computing and Molecular Programming*, ed. by D. Soloveichik, B. Yurke, vol. 8141 of Lecture Notes in Computer Science, pp. 189–203 (Springer International Publishing, 2013)
55. J.I. Lathrop, J.H. Lutz, R.R. Lutz, H.D. Potter, M.R. Riley, Population-induced phase transitions and the verification of chemical reaction networks, in *26th International Conference on DNA Computing and Molecular Programming (DNA 26)*, ed. by C. Geary, M.J. Patitz, vol. 174 of Leibniz International Proceedings in Informatics (LIPIcs), Dagstuhl, Germany, pp. 5:1–5:17 (Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020)
56. A. Padirac, T. Fujii, A. Estévez-Torres, Y. Rondelez, Spatial waves in synthetic biochemical networks. *J. Am. Chem. Soc.* **135**(39), 14586–14592 (2013)
57. D. Scalise, R. Schulman, Designing modular reaction-diffusion programs for complex pattern formation. *Technology* **2**(1), 55 (2014)
58. N. Dalchau, G. Seelig, and A. Phillips, Computational design of reaction-diffusion patterns using DNA-based chemical reaction networks, in *Proceedings of the 20th International Conference on DNA Computing and Molecular Programming*, ed. by S. Murata, S. Kobayashi, vol. 8727 of Lecture Notes in Computer Science, pp. 84–99 (Springer International Publishing, 2014)
59. A. Joesaar, S. Yang, B. Bögels, A. van der Linden, P. Pieters, B.V.V.S.P. Kumar, N. Dalchau, A. Phillips, S. Mann, T.F.A. de Greef, DNA-based communication in populations of synthetic protocells. *Nat. Nanotechnol.* **14**, 369–378 (2019)
60. N. Dalchau, H. Chandran, N. Gopalkrishnan, A. Phillips, J. Reif, Probabilistic analysis of localized DNA hybridization circuits. *ACS Synth. Biol.* **4**(8), 898–913 (2015)
61. M.R. Lakin, D. Stefanovic, A. Phillips, Modular verification of chemical reaction network encodings via serializability analysis. *Theoret. Comput. Sci.* **632**, 21–42 (2016)
62. S.W. Shin, C. Thachuk, E. Winfree, Verifying chemical reaction network implementations: A pathway decomposition approach. *Theoret. Comput. Sci.* **765**, 67–96 (2019)
63. R.F. Johnson, Q. Dong, E. Winfree, Verifying chemical reaction network implementations: a bisimulation approach. *Theoret. Comput. Sci.* **765**, 3–46 (2019)
64. S. Badelt, S. W. Shin, R. F. Johnson, Q. Dong, C. Thachuk, and E. Winfree, A general-purpose CRN-to-DSD compiler with formal verification, optimization, and simulation capabilities, in *Proceedings of the 23rd International Conference on DNA Computing and Molecular Programming*, ed. by R. Brijder, L. Qian, vol. 10467 of Lecture Notes in Computer Science, pp. 232–248 (2017)
65. A. Nishikawa, M. Yamamura, M. Hagiya, DNA computation simulator based on abstract bases. *Soft Comput.* **5**, 25–38 (2001)
66. C. Grun, J. Werfel, D.Y. Zhang, P. Yin, DyNAMiC Workbench: an integrated development environment for dynamic DNA nanotechnology. *J. R. Soc. Interface* **12**, 20150580 (2015)
67. J. M. Schaeffer, C. Thachuk, and E. Winfree, Stochastic simulation of the kinetics of multiple interacting nucleic acid strands, in *Proceedings of the 21st International Conference on DNA Computing and Molecular Programming*, ed. by A. Phillips, P. Yin, vol. 9211 of Lecture Notes in Computer Science, pp. 194–211 (Springer International Publishing, 2015)
68. N. Srinivas, J. Parkin, G. Seelig, E. Winfree, D. Soloveichik, Enzyme-free nucleic acid dynamical systems. *Science* **358**(6369), eaal2052 (2017)

69. A.J. Thubagere, C. Thachuk, J. Berleant, R.F. Johnson, D.A. Ardelean, K.M. Cherry, L. Qian, Compiler-aided systematic construction of large-scale DNA strand displacement circuits using unpurified components. *Nat. Commun.* **8**, 14373 (2017)
70. J. Berleant, C. Berlind, S. Badelt, F. Dannenberg, J. Schaeffer, E. Winfree, Automated sequence-level analysis of kinetics and thermodynamics for domain-level DNA strand-displacement systems. *J. R. Soc. Interface* **15**, 20180107 (2018)
71. S. Badelt, C. Grun, K.V. Sarma, B. Wolfe, S.W. Shin, E. Winfree, A domain-level DNA strand displacement reaction enumerator allowing arbitrary non-pseudoknotted secondary structures. *J. R. Soc. Interface* **17**, 20190866 (2020)
72. D. Soloveichik, M. Cook, E. Winfree, J. Bruck, Computation with finite stochastic chemical reaction networks. *Nat. Comput.* **7**, 615–633 (2008)
73. J.N. Zadeh, C.D. Steenberg, J.S. Bois, B.R. Wolfe, M.B. Pierce, A.R. Khan, R.M. Dirks, N.A. Pierce, NUPACK: analysis and design of nucleic acid systems. *J. Comput. Chem.* **32**(1), 170–173 (2011)
74. N. Srinivas, T.E. Ouldridge, P. Šulc, J.M. Schaeffer, B. Yurke, A.A. Louis, J.P.K. Doye, E. Winfree, On the biophysics and kinetics of toehold-mediated DNA strand displacement. *Nucleic Acids Res.* **41**(22), 10641–10658 (2013)
75. S.M. Douglas, A.H. Marblestone, S. Teerapittayanon, A. Vazquez, G.M. Church, W.M. Shih, Rapid prototyping of 3D DNA-origami shapes with caDNAno. *Nucleic Acids Res.* **37**, 5001–5006 (2009)
76. J.P.K. Doye, T.E. Ouldridge, A.A. Louis, F. Romano, P. Šulc, C. Matek, B.E.K. Snodin, L. Rovigatti, J.S. Schreck, R.M. Harrison, W.P.J. Smith, Coarse-graining DNA for simulations of DNA nanotechnology. *Phys. Chem. Chem. Phys.* **15**, 20395–20414 (2013)
77. M.E. Fornace, N.J. Porubsky, N.A. Pierce, A unified dynamic programming framework for the analysis of interacting nucleic acid strands: Enhanced models, scalability, and speed. *ACS Synth. Biol.* **9**(10), 2665–2678 (2020)
78. J.N. Zadeh, B.R. Wolfe, N.A. Pierce, Nucleic acid sequence design via efficient ensemble defect optimization. *J. Comput. Chem.* **32**, 439–452 (2011)
79. B.R. Wolfe, N.J. Porubsky, J.N. Zadeh, R.M. Dirks, N.A. Pierce, Constrained multistate sequence design for nucleic acid reaction pathway engineering. *J. Am. Chem. Soc.* **139**(8), 3134–3144 (2017)
80. N. Roehner, J. Beal, K. Clancy, B. Bartley, G. Misirli, R. Grünberg, E. Oberortner, M. Pocock, M. Bissell, C. Madsen, T. Nguyen, M. Zhang, Z. Zhang, Z. Zundel, D. Densmore, J.H. Gennari, A. Wipat, H.M. Sauro, C.J. Myers, Sharing structure and function in biological design with SBOL 2.0. *ACS Synth. Biol.* **5**(6), 498–506 (2016)
81. F. Schreiber, B. Sommer, T. Czauderna, M. Golebiewski, T. E. Gorochoowski, M. Hucka, S. M. Keating, M. König, C. Myers, D. Nickerson, D. Waltemath, Specifications of standards in systems and synthetic biology: status and developments in 2020. *J. Integr. Bioinf.* **17**(2–3) (2020)
82. M. Hucka, A. Finney, H.M. Sauro, H. Bolouri, J.C. Doyle, H. Kitano, and the rest of the SBML Forum: A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish-Bowden, A.A. Cuellar, S. Dronov, E.D. Gilles, M. Ginkel, V. Gor, I. Goryanin, W.J. Hedley, T.C. Hodgman, J.-H. Hofmeyr, P.J. Hunter, N.S. Juty, J. L. Kasberger, A. Kremling, U. Kummer, N. Le Novère, L.M. Loew, D. Lucio, P. Mendes, E. Minch, E.D. Mjolsness, Y. Nakayama, M.R. Nelson, P.F. Nielsen, T. Sakurada, J.C. Schaff, B.E. Shapiro, T.S. Shimizu, H.D. Spence, J. Stelling, K. Takahashi, M. Tomita, J. Wagner, J. Wang, The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* **19**(4), 524–531 (2003)
83. M. Hanus, The integration of functions into logic programming: from theory to practice. *J. Logic Program.* **19–20**(Supplement 1), 583–628 (1994)
84. D.M. Camacho, K.M. Collins, R.K. Powers, J.C. Costello, J.J. Collins, Next-generation machine learning for biological networks. *Cell* **173**(7), P1581–P1592 (2018)
85. J.X. Zhang, J.Z. Fang, W. Duan, L.R. Wu, A.W. Zhang, N. Dalchau, B. Yordanov, R. Petersen, A. Phillips, D.Y. Zhang, Predicting DNA hybridization kinetics from sequence. *Nat. Chem.* **10**, 91–98 (2017)

86. J.X. Zhang, B. Yordanov, A. Gaunt, M.X. Wang, P. Dai, Y.-J. Chen, K. Zhang, J.Z. Fang, N. Dalchau, J. Li, A. Phillips, D.Y. Zhang, A deep learning model for predicting next-generation sequencing depth from DNA sequence. *Nat. Commun.* **12**, 4387 (2021)
87. T.W. Hiscock, Adapting machine-learning algorithms to design gene circuits. *BMC Bioinf.* **20**, 214 (2019)
88. B. Wang, C. Thachuk, A.D. Ellington, E. Winfree, D. Soloveichik, Effective design principles for leakless strand displacement systems. *Pro. Natl. Acad. Sci. USA* **115**(52), E12182–E12191 (2018)
89. P.S. Weiss, A conversation with Prof. Ned Seeman: founder of DNA nanotechnology. *ACS Nano* **2**(6), 1089–1096 (2008)
90. W. Bae, G.-B.V. Stan, T.E. Ouldrige, In situ generation of RNA complexes for synthetic molecular strand-displacement circuits in autonomous systems. *Nano Lett.* **21**(1), 265–271 (2021)
91. Y.-J. Chen, B. Groves, R.A. Muscat, G. Seelig, DNA nanotechnology from the test tube to the cell. *Nat. Nanotechnol.* **10**, 748–760 (2015)
92. B. Groves, Y.-J. Chen, C. Zurla, S. Pohekailov, J.L. Kirschman, P.J. Santangelo, G. Seelig, Computing in mammalian cells with nucleic acid strand exchange. *Nat. Nanotechnol.* **11**, 287–294 (2016)
93. M. Rudchenko, S. Taylor, P. Pallavi, A. Dechkovskaia, S. Khan, V.P. Butler Jr., S. Rudchenko, M.N. Stojanovic, Autonomous molecular cascades for evaluation of cell surfaces. *Nat. Nanotechnol.* **8**, 580–586 (2013)
94. C. Zhang, Y. Zhao, X. Xu, R. Xu, H. Li, X. Teng, Y. Du, Y. Miao, H. Chu Lin, D. Han, Cancer diagnosis with DNA molecular computation. *Nat. Nanotechnol.* **15**, 709–715 (2020)
95. L.M. Hochrein, M. Schwarzkopf, M. Shahgholi, P. Yin, N.A. Pierce, Conditional dicer substrate formation via shape and sequence transduction with small conditional RNAs. *J. Am. Chem. Soc.* **135**(46), 17322–17330 (2013)
96. J. Hemphill, A. Deiters, DNA computation in mammalian cells: MicroRNA logic operations. *J. Am. Chem. Soc.* **135**(28), 10512–10518 (2013)
97. G. Chatterjee, Y.-J. Chen, G. Seelig, Nucleic acid strand displacement with synthetic mRNA inputs in living mammalian cells. *ACS Synth. Biol.* **7**(12), 2737–2741 (2018)
98. W. Zhong, J.T. Szczepanski, Direct comparison of d-DNA and l-DNA strand-displacement reactions in living mammalian cells. *ACS Synth. Biol.* **10**(1), 209–212 (2021)
99. T.L. Mallette, M.N. Stojanovic, D. Stefanovic, M.R. Lakin, Robust heterochiral strand displacement using leakless translators. *ACS Synth. Biol.* **9**(7), 1907–1910 (2020)
100. A.P. Goldberg, B. Szigeti, Y.H. Chew, J.A.P. Sekar, Y.D. Roth, J.R. Karr, Emerging whole-cell modeling principles and methods. *Curr. Opin. Biotechnol.* **51**, 97–102 (2018)
101. F. Jiang, J.A. Doudna, CRISPR-Cas9 structures and mechanisms. *Annu. Rev. Biophys.* **46**, 505–529 (2017)
102. L.S. Qi, M.H. Larson, L.A. Gilbert, J.A. Doudna, J.S. Weissman, A.P. Arkin, W.A. Lim, Repurposing CRISPR as an RNA-guided platform for sequence-specific control of gene expression. *Cell* **152**, 1173–1183 (2013)
103. K.-H. Siu, W. Chen, Riboregulated toehold-gated gRNA for programmable CRISPR-Cas9 function. *Nat. Chem. Biol.* **15**, 217–220 (2019)
104. L. Oesinghaus, F.C. Simmel, Switching the activity of Cas12a using guide RNA strand displacement circuits. *Nat. Commun.* **10**, 2092 (2019)
105. M.H. Hanewich-Hollatz, Z. Chen, L.M. Hochrein, J. Huang, N.A. Pierce, Conditional guide RNAs: Programmable conditional regulation of CRISPR/Cas function in bacterial and mammalian cells via dynamic RNA nanotechnology. *ACS Central Sci.* **5**(7), 1241–1249 (2019)
106. M. Jin, N.G. de Loubresse, Y. Kim, J. Kim, P. Yin, Programmable CRISPR-Cas repression, activation, and computation with sequence-independent targets and triggers. *ACS Synth. Biol.* **8**(7), 1583–1589 (2019)
107. A.A. Green, P.A. Silver, J.J. Collins, P. Yin, Toehold switches: De-novo-designed regulators of gene expression. *Cell* **159**(4), 925–939 (2014)

108. Green AA, Kim J, Ma D, Silver PA, Collins JJ, Yin P (2017) Complex cellular logic computation using ribocomputing devices. *Nature* 548(7665):117–121
109. J. Chappell, A. Westbrook, M. Verosloff, J.B. Lucks, Computational design of small transcription activating RNAs for versatile and dynamic gene regulation. *Nat. Commun.* 8, 1051 (2017)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Spatial Systems

Parallel Computations with DNA-Encoded Chemical Reaction Networks



Guillaume Gines, Anthony J. Genot, and Yannick Rondelez

Abstract Molecular programs use chemical reactions as primitives to process information. An interesting property of many of these amorphous systems is their scale-invariant property: They can be split into sub-parts without affecting their function. In combination with emerging techniques to compartmentalize and manipulate extremely small volumes of liquid, this opens a route to parallel molecular computations involving possibly millions to billions of individual processors. In this short perspective, we use selected examples from the DNA-based molecular programming literature to discuss some of the technical aspects associated with distributing chemical computations in spatially defined microscopic sub-units. We also present some future directions to leverage the potential of parallel molecular networks in applications.

1 Harnessing Parallelization in Chemical Reaction Networks

1.1 *D(R)NA-Based Deterministic Chemical Reaction Networks*

D(R)NA-based artificial reaction networks are man-made, rationally designed biochemical systems targeting specific dynamical or computational behaviors. They use DNA or RNA as a substrate, as nucleic acids have proved incredibly versatile for storing information and executing chemical computations. Nucleic acids can be commercially synthesized with user-designed sequences, they can be tracked and

G. Gines · Y. Rondelez (✉)

Laboratoire Gulliver, UMR 7083, CNRS, ESPCI Paris, PSL Research University, 10 Rue Vauquelin, 75005 Paris, France

e-mail: yannick.rondelez@espci.fr

A. J. Genot

LIMMS, CNRS-Institute of Industrial Science, UMI 2820, University of Tokyo, Tokyo 153-8505, Japan

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*, Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_19

349

characterized with great details (with fluorescence, sequencing, mass spectrometry, electrophoresis...) and can be conjugated to a large variety of other molecules.

These systems are based on modules (nodes or gates) that one can connect at will to create more-or-less arbitrary circuit architectures. This approach leverages the canonical programmability of DNA/DNA interactions—Watson–Crick rules—and the existence of generic reactions, which are agnostic to sequence. D(R)NA-based molecular programs take molecular signals as *input*, for example, the presence/absence of particular species or the concentration of these species, and process through a cascade of reactions—in many cases including feedbacks and multiple intermediates—down to the production of particular *output* species. The concentration of these output species, which can be monitored via physicochemical signals such as fluorescence, is then interpreted as the result of a computation. The computed function is encoded by the details of the reaction circuit, including its topology and the kinetic laws and parameters of the constituting reactions. A number of experimental toolboxes have been developed to experimentally assemble these architectures [1–6]. In principle, using these frameworks, one can basically set up a molecular system performing any computation of interest [7].

The fundamental building block of all reaction networking schemes is catalysis [8–10]. Catalysis allows one compound of the network to act on others, without being itself included in the transformation. For nucleic acid (NA)-based systems, catalysis is either related to mechanisms that can accelerate the slow reaction of strand exchange [4, 8, 9, 11] or to enzymatic manipulation of NA synthesis and degradation (e.g., polymerases, ligases and nucleases) [1, 2, 6].

In the case of enzymatic networking schemes, for example, *genelets* [1] use an RNA polymerase and one or two RNases, while the PEN toolbox [3, 6, 12] uses a DNA polymerase, a nickase and a DNA exonuclease. These enzymes can be seen as the *hardware* of a machine whose behavior is controlled by a DNA *software* consisting of synthetic templates. Although less explored than DNA-only systems, enzymatic approaches provide benefits in terms of compacity, ease of preparation, amplification folds and control of leaks. Importantly for the topic discussed here, some of them provide both a well-controlled kinetic bottleneck on the fuel consumption and an efficient chemical sink and are thus able to maintain an out-of-equilibrium state of long periods in a closed reactor. Demonstrated out-of-equilibrium DNA-encoded enzymatic networks include oscillators [3, 13, 14], inverters, memories and toggle switches [1, 12, 15]. Note that other NA-based molecular programming paradigms have been described and also experimentally validated [1, 2, 4, 7–9].

Chemical reaction networks (CRNs) share many formal similarities with other computational frameworks, such as electronic or mechanical circuits. They are modular, and computations are built by linking together simple computing modules (e.g., transistors in the case of electronic circuits and chemical reactions in the case of chemical reaction networks). However, contrary to these more classic approaches, CRNs run in a homogenous, liquid medium. In the general case, CRN-encoding compounds freely diffuse within this medium. Because of this, individual network connections within a CRN are not defined by spatial addresses, but rather by chemical recognition rules. A link between A and B in the network means a wire between

the location of a single copy of A and a single copy of B for an electronic circuit, but, for a CRN, it means a specific (strong) chemical interaction between the molecules of type A and molecules of type B, both present in multiple, independent copies.

The standard DNA program runs in a test tube, that is, a typical volume of 10 s–100 s of microliters. One tube performs one—possibly complex—computation (a computation being defined here as a series of chemical reactions producing output species that depend on the input species). The computation is generally done only once, as in most cases resetting is not possible (i.e., regenerating the inputs if they have been consumed, erasing the outputs and restoring the intermediates species that drove the reaction). CRNs operate on molecular concentrations, in the sense that data flows are instantiated by changes in the *concentration* of some species over time, and the output is typically represented by the concentration of one or more specific species. The concentration of most compounds participating in a CRN needs to be in the nanomolar range or above to get appreciable kinetics, because most programmable reactions, including DNA hybridization [16], are limited by diffusion below this value. These typical volumes and concentrations imply that molecular counts are extremely high, typically 10^9 – 10^{15} copies of each specific molecular type. The fact that computation by a bulk CRN involves the production of so many identical molecules naturally leads to consider the potential for parallel implementations.

1.2 CRNs Run on Inherently Parallel Processes

CRNs controlled by mass-action kinetics (see below for stochastic effects in very small volumes) are scale-free. Their behavior does not depend on the size of the system. This is because concentrations are intensive thermodynamic variables and leads to an intriguing property: the dynamical or information-processing functions of CRNs are independent of their size. For example, if the experimenter partitions a circuit from one large to two smaller containers, he will simply obtain two systems that are dynamically and computationally identical to the parent reaction. In other words, the function has not changed, but the number of threads has been doubled. Let us assume as an example that the CRN was running an addressable bistable system [12, 17], in order to maintain one bit of memory storage. The splitting in that case will simply lead to having two independent copies of the same memory element. Post splitting, one of them can in principle be reset to a different state. Altogether, one now disposes of two bits of memory. Given enough compartments, this operation can be repeated over and over.

Using microcompartmentalization techniques, where individual reactors' volumes range from femtoliters to nanoliters, and using the above-mentioned test tube as starting material, it is possible to generate millions to billions of compartments. For example, the splitting of a bulk DNA-encoded enzymatic predator–prey oscillator into ~100 picoliter microdroplets, using a microfluidic approach, immediately generated many millions of independent networks, all displaying the same oscillatory behavior [18, 19].

This scale-freeness of CRNs is only true down to the stochastic limit, where volumes get so small that molecular counts are close to unity (Fig. 1). In this range, the continuous approximation breaks for concentrations and kinetics, which cannot be seen as deterministic anymore. Rather, they become discrete and stochastic: molecular counts are randomly updated by integer increments—which fundamentally alters the behaviors of CRN. For instance, in the deterministic regime and with conventional chemistry, it is not possible for a chemical species to reach its null-state in finite time (*i.e.*, its concentration cannot get to 0). This implies that an autocatalytic system can always “regenerate” itself after dilution, because there are always some molecules around to seed the reaction. Yet in the stochastic regime, a species can permanently “crash” to zero with a non-null probability. Keeping the nanomolar concentration as a reference, the stochastic limit occurs around femtoliter volumes (the volume of a typical bacterium). Therefore, depending on compartment size, the splitting of molecular programs can either provide parallel circuits with classical behaviors or enable new, non-deterministic operations [20, 21].

To truly leverage the potential of CRNs for parallelism, one ideally wants to perform different sub-computations at each position of the distributed system. The way these sub-computations interact with each other through space (by free diffusion, through restricted diffusion, or not at all) will then define the various options for parallel molecular programming.

In this short and non-exhaustive contribution, we review experimental works that have explored the parallelization of DNA-encoded mass-action CRNs, focusing in particular on microcompartmentalization and high-throughput strategies. Based on these selected experimental approaches, we also suggest and discuss possible applications and future directions for parallelized molecular computing.

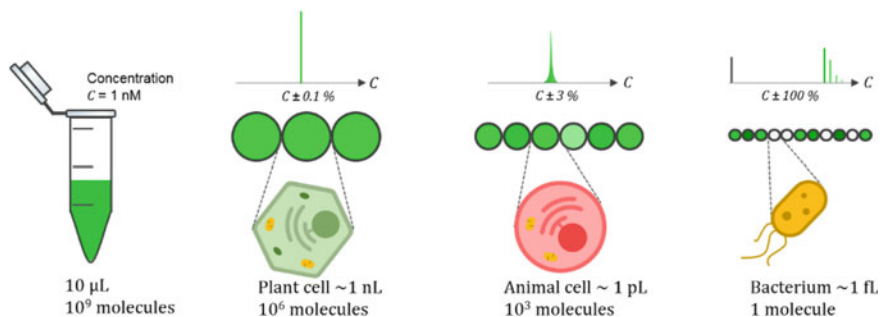


Fig. 1 Splitting a sample containing a molecule at concentration $C = 1 \text{ nM}$ produces multiple compartments with the same average concentration and a coefficient of variation $CV = \frac{\sqrt{\lambda}}{\lambda}$, where λ is the average number of molecules per compartment. When the compartment volume reaches down to the femtoliter $\lambda \sim CV \sim 1$, concentrations take discrete values and stochastic partitioning leads to strong inhomogeneity between compartments

2 Creating Sub-Computations

A variety of approaches are suitable for the parallelization of molecular programs. Below, we group them into approaches that use impermeable physical boundaries (e.g., water-in-oil droplets), permeable boundaries or no boundaries at all (Fig. 2). Important characteristics associated with these technical choices are the size of the individual elements and the associated throughput (the number of compartments one can generate or manipulate in a standard experiment).

2.1 No-Diffusion (Leak-Tight) Compartments

A number of tight microcompartmentalization approaches have been explored. The absence of transport across the boundaries ensures that each microcompartment behaves as a fully independent computing element. On the other hand, it makes the delivery of specific inputs to each individual reactor more problematic. In the case of dynamical systems (out-of-equilibrium CRNs), it also mandates that the encapsulated chemistry be autonomous. For example, although many chemical (non-DNA) oscillators are known, most require continuous influx and outflux of reactants in open reactors, and only a handful are able to display (pseudo-) limit cycles in batch mode. Fortunately, several autonomous DNA circuits with non-trivial dynamics have been reported over the years and provide useful test cases for compartmentalization approaches (Fig. 3).

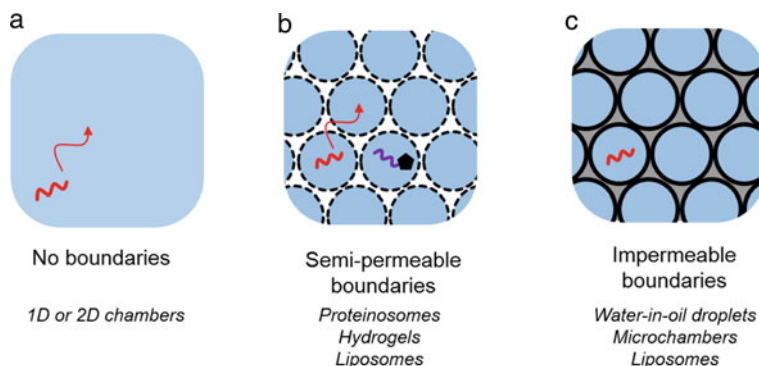


Fig. 2 Parallel molecular computations can be classified according to the diffusion rules between sub-units. **a** in the absence of boundaries, the behavior is governed by reaction–diffusion processes. Competition between reaction and diffusion determines the elementary length scale. **b** semi-permeable compartments, where the diffusion of molecular program components between partitions is permitted or not depending on the permeability rules (for example, size cutoff), can allow the emergence of new collective functions from simple sub-units. **c** impermeable compartments, where no diffusion is allowed between compartments, open the way to pure parallel processing

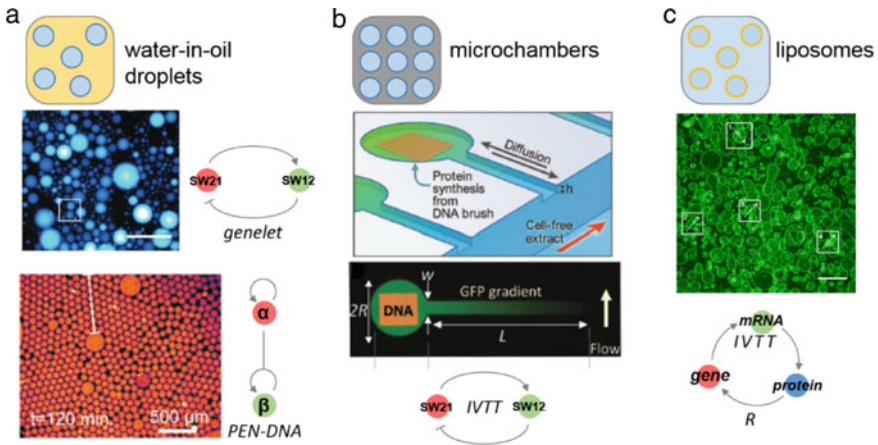


Fig. 3 Experimental approaches for independent compartments. **a** water-in-oil emulsions containing various DNA-encoded oscillatory networks, including genelets [22] and PEN-DNA circuits [18]. **b** microfabricated silicon chambers encapsulate IVTT-based functions, including oscillators [23]. **c** liposomes, used in this example as vehicles for IVTT-replication (IVTTR) gene amplification system [24]

2.1.1 Emulsions

Emulsions are made of a continuous and organic phase that contains many small water-in-oil droplets that can be considered as chemically independent (see [25, 26] for discussion of droplet-to-droplet exchange in some specific cases). Micro-partitioning a master mix using emulsions can be as simple as combining the aqueous reaction mix with oil and surfactant, using for example vortexing. Simmel, Winfree and colleagues used this approach to parallelize a *genelet* oscillator and reported the observation of microscopic droplets showing individual oscillatory behavior [22] (Fig. 3a). While many other emulsification techniques are possible, including the use of filters, mechanical mixers, bead shakers, sonication, etc., it is not yet clear how the harsh conditions they entail impact the macromolecules of the molecular programs. In addition, although they are very fast—with millions or billions of compartments created in seconds—these techniques do not provide precise control over the size distribution.

An alternative is droplet microfluidics [27]—in which reagents are controllably mixed and encapsulated at the microscale—which offers a better control on the size distribution and a gentler emulsification process. Although the compartments are produced in a sequential manner, generation rates reaching 10 s or 100 s of kHz can be achieved, and parallel designs [28] with tens of thousands of nozzles have been reported [29]. The use of microfluidic emulsions is well developed in the field of high-throughput directed evolution, combinatorial chemistry or droplet digital Polymerase Chain Reaction (ddPCR) [30]. In the latter case, a number of commercial solutions are available.

2.1.2 Liposomes

Liposomes more closely mimic the architecture of biological cells. While droplets are made of an aqueous compartment dispersed in oil, liposomes are aqueous compartments surrounded by a lipid membrane (a phospholipid bilayer) and dispersed in an aqueous phase. There are thus two aqueous phases: one inside each liposome and one outside the liposome. Since unmodified lipid bilayers are typically impermeant to DNA strands or large proteins, liposomes present in principle an attractive alternative to droplets for microcompartmentalization of molecular programs. Their generation is however more technical and delicate than droplets, especially when the size distribution of liposomes must be controlled [31]. This may explain why liposomes have been less explored for computational CRN [32], although there are some examples of liposome-compartmentalized complex biomolecular mixtures, generally designed as minimal artificial cells [33]. The group of Vincent Noireaux, among others, encapsulated *In Vitro* Transcription and Translation systems (IVTT) based on cell extract to recapitulate protein expression [34]. Danelon et al. went a step further by showing that genetic replication, supported by *in situ* expression of genetically encoded enzymatic activities, can be installed within IVTT-containing liposomes [24] (Fig. 3c). In principle, such minimal cells reproduce the basic functions necessary to enter a process of Darwinian evolution.

2.1.3 Microchambers

Microfabrication offers a top-down way to create microcompartments with precisely defined geometries, within which very small volumes of liquid can be enclosed [35]. This can be used for example to create partially confined reactors, able to support localized behaviors while still diffusively exchanging with a feeding tank. Bar-Ziv and colleagues leveraged this approach to carve microchambers and connecting channels inside a silicon wafer [23] (Fig. 3b). The chambers are patterned at their bottom with DNA brushes, which are dense monolayers of DNA strands' coding for a gene expression CRN with positive or negative feedbacks. Each chamber is connected to a central feed channel that runs through the chip, acting as a chemical source and sink. The feed channel bathes the chambers in a cell-free extract, which contains the molecular machinery to transcribe and translate the gene encoded by the DNA brush. This reaction is predominantly local, but species diffuse and escape over time to compartments nearby—creating a spatial network of connected compartments. The authors showed that the geometry of the channels and chamber controls the interplay between reaction and diffusion. For instance, a positive feedback loop network was unable to kick off when the channel connecting the chamber to the feed was too short. This was expected because the removal of signaling molecules dominated over their autocatalytic production. When the connecting channel was made longer, thereby decreasing the effective dilution rate, the positive feedback loop ignited and the chamber glowed green. The author extended this approach to create small spatial networks of microreactors, where each microreactor could contain a different

set of encoding DNA modules (here, gene-encoding templates) [36]. In principle, this approach could allow the design of a variety of exotic dynamical behaviors, by controlling the relative position of each DNA node [37].

Note that some hybrid techniques, using both top-down micro-patterning and oil-water interfaces, can be used to create regular arrays containing millions of extremely small and monodisperse compartments (down to the femtoliter), compatible with biomolecular reactions [38].

2.2 *Compartment-Free Approaches*

Spatially resolved CRN behaviors can also be observed without any physical boundaries. This is well known in the community of nonlinear chemistry, which has studied a number of beautiful reaction–diffusion systems [39, 40]. In another field, chemical engineers are also interested by the emergence of localized behaviors, which in some cases may have deleterious consequences. In the absence of physical boundaries, whether or not local behaviors are possibly depends on the relative strength of chemical reactions and diffusion. This simple statement is captured by the so-called Damköhler number [41] which compares transport and reactive rates:

$$D_a = \text{reaction rate/transport rate.}$$

Localized behaviors are possible when this dimensionless number is greater than 1, i.e., when reactions are faster than mixing. For diffusion-limited reactions involving short oligos, the hybridization rate (which is mostly independent of sequence, length and temperature) is in the order of $10^6 \text{ M}^{-1} \text{ s}^{-1}$, whereas diffusion is around $D \sim 10\text{--}100 \mu\text{m}^2 \text{ s}^{-1}$ (single- or double-stranded 10–100 mers) [42]. Knowing the concentrations of gate or node molecules, one can then assess the spatial scale of localized behaviors, in the absence of any mixing. For example, if the reaction rate is around 1 min^{-1} , this length scale is $10 \mu\text{m}$; smaller reactors can be considered spontaneously well-mixed, whereas larger ones may generate reaction–diffusion systems (Fig. 4).

Similar to the case of well-mixed systems, the unfolding of a reaction–diffusion process can be interpreted as a computation, whose input will typically include geometrical clues. Biology provides a number of examples, such as the creation of concentration patterns during embryo morphogenesis via clock-and-wavefront, French-flag or Turing mechanisms, which will later on trigger cellular differentiation. At a simpler scale, the bacterial Min system performs an accurate determination of the geometric middle of a rod-shaped container using reaction–diffusion processes [43]. For molecular programming in reaction–diffusion settings (i.e., high Damköhler numbers), one needs to control, in addition to topology laws and rates, the diffusion characteristic of each species, as well as the boundary conditions.

For example, Padirac et al. spatialized a predator–prey oscillator by enclosing the corresponding DNA-encoded circuit in a flat chamber $200 \mu\text{m}$ thick and one centimeter wide, installed on a microscope stage. This chamber was locally seeded

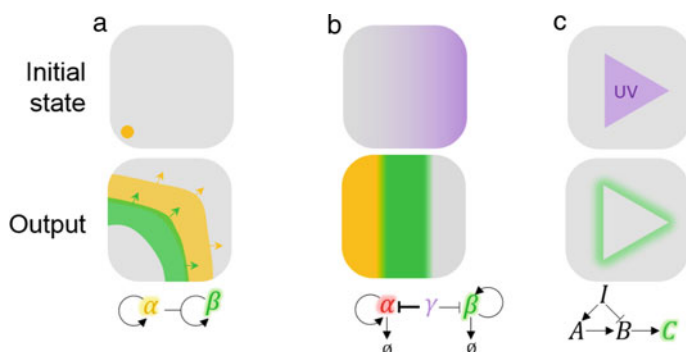


Fig. 4 Boundary-free spatial computations (reaction–diffusion systems). An initial state, taken as input, is transformed into dynamic or stationary patterns. **a** a spot of trigger molecules (orange spot) initiates the propagation of successive waves of species α (prey) and β (predator) in a predator–prey molecular system [44]. **b** a morphogen gradient (purple) is interpreted by a tristable system to create a three-stripe pattern, recapitulating the “French-Flag” embryogenesis [45]. **c** an incoherent feedforward loop coupled to differential diffusion coefficients allowed the transformation of an input pattern injected through UV-sensitive DNA modifications [46]

with prey, which created inhomogeneous initial conditions, and fluorescent reporters were used to observe the concentrations of both dynamic species. Incubation revealed traveling waves of prey, chased by waves of predators—reminiscent of their ecological counterparts (Fig. 4a). The waves traveled at a velocity of hundreds of micrometers per minute. These chemical waves result from the interplay between reaction and diffusion, and they transport information (e.g., the concentration of chemical species) over large distances (in the same way as light or electrical impulses could). Their velocity can be roughly predicted from simple scaling arguments $v = \sqrt{(D/\tau)} \approx 100 \mu\text{m}\cdot\text{min}^{-1}$.

Similar PEN-toolbox traveling waves and fronts provide a useful primitive to build more complex spatial operations. Triggered at the entrance of a millimetric maze built of PDMS, a wavefront followed the walls of the channels, split into two at a junction and died when it reached a cul-de-sac [47]. The shape of the wavefront was sensitive to geometric clues such as the curvature of the channels. When the wavefront entered a turn with a short curvature radius, it showed a transient dispersion where the front on the outer end of the turn progressed faster than the front on the inner end. This dispersion was not observed in turns with larger radii.

Biological embryo development provides inspirational examples of how complex geometrical shapes can unfold from chemically encoded programs. The synthetic version, artificial morphogenesis thus represents an exciting frontier for molecular programming. The challenge is ambitious, because, in addition to topology, rates and laws, artificial morphogenesis requires the control of diffusion rates in a species-specific way. For example, Turing patterns generally require a marked difference between the diffusion rate of activating and inhibiting compounds; in the field of small-molecule CRN, a strategy to slow down the diffusion rate of inhibitory

compounds has been instrumental to the engineering of the first synthetic Turing motifs [39]. For DNA-based approaches, the group of Estevez-Torres and Galas in Sorbonne Université/CNRS introduced some strategies for this purpose, based on immobilized binding (complementary) partners of the active DNA species [48]. They also studied the interaction between traveling fronts or confined them within guiding gels [49]. These efforts, reviewed in [50], culminated in an artificial reconstitution of the famous “French-Flag” embryo patterning algorithm, a system that takes a shallow 1D chemical gradient as input and outputs three sharply demarcated spatial regions (Fig. 4b) [45].

The diffusional coupling between reacting voxels (i.e., elementary reaction volumes) in reaction–diffusion systems can also be leveraged to compute fine spatial transformations (i.e., the transformation of a spatial pattern of concentrations into another pattern). Ellington and colleagues created a strand-displacement network that takes a 2D illumination pattern as input and outputs a species that delineates the edges of the pattern. This system is built around an incoherent feedforward loop (Fig. 4c) [46]; light both activates and inhibits the creation of a signal species. In area without illumination, nothing happens. The same is true for areas of homogenous illumination, where chemical activation and inhibition cancel each other. But, on the edges of an illuminated area, species that were activated by light diffuse to nearby dark area and escape photoinhibition, thus revealing edges. In principle, this approach could be generalized to more complex image processing and be connected to other chemical (like polymerization) or biological (like cell culture and differentiation) processes. This could enable wholly new ways to orchestrate complex chemistry or biology using spatially structured illumination to coordinate and manage complex processes.

2.3 Intermediate Cases: Some Species Diffuse, Some Do Not

In this section, we group approaches that are inbetween the two extreme cases presented above: completely tight compartments on the one hand and unrestricted diffusion on the other hand. Indeed, it is possible to imagine that the experimental design does not restrict diffusion of all species (Fig. 5). Although this situation may bring some issues (compartment content leaks out), it can also offer a number of advantages. As seen in the silicon microchambers example above, partial enclosing provides a strategy to deliver fuel molecules and implement a physical first-order decay process. This is very useful for systems that cannot sustain dynamical behavior on the long run in closed containers and is a strategy that has also been extensively used in the field of small-molecule nonlinear chemistry [39]. In addition, controlled leakage enables compartment-to-compartment communication and coupling, as well as the delivery of input signals.

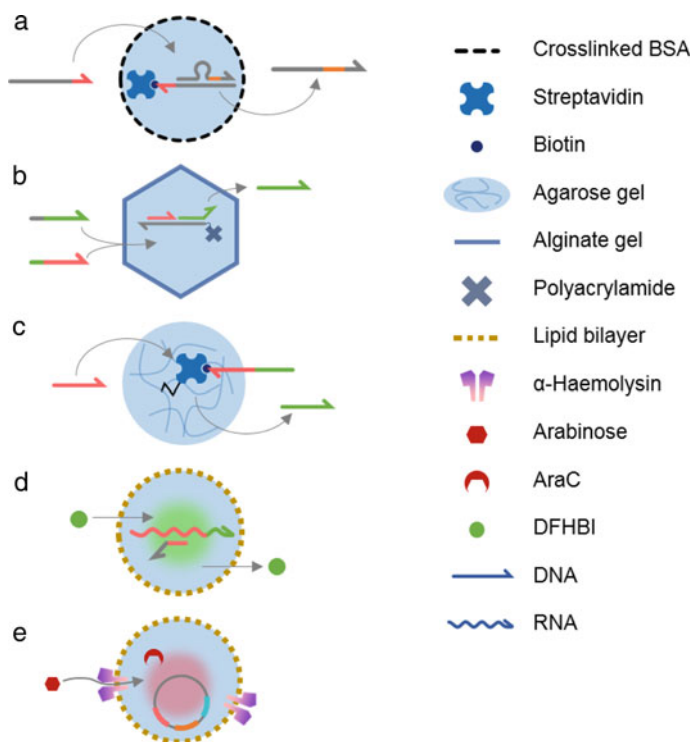


Fig. 5 Strategies to create compartments with selective boundaries. **a** proteinosomes form compartments with size-selective boundaries. While streptavidin-tethered DNA species cannot cross the membrane, other DNA strands freely diffuse between compartments [51]. **b** a similar approach builds *on* microchambers with alginate walls, impermeable to polyacrylamide-grafted DNA strands [52]. **c** the attachment of DNA species directly to an immobile support (here, gel beads) prevents their diffusion, while input and output species can freely diffuse between the beads [41]. **d** lipid bilayers allow the passive diffusion of hydrophobic compounds while confining biomolecules [26]. **e** protein pores (e.g., α -hemolysin) can be inserted in the membrane allowing for the selective diffusion of molecules (e.g., arabinose) [26]

2.3.1 Proteinosomes

One approach uses semipermeable membranes that filter molecular species based on their sizes. Proteinosomes, for example, are artificial semipermeable microenclosure based on a layer of crosslinked proteins. They can be generated in a highly monodisperse way, and their membrane cutoff can be adjusted through the cross-link density. De Greef et al. created spatial arrays of proteinosomes containing non-enzymatic DNA networks (Fig. 5a) [51]. The compartmentalized circuit modules were trapped within the proteinosome, using affinity tagging to bulky partners, but small DNA signals could diffuse across the boundary, thus opening a communication channel.

2.3.2 Hydrogel Barriers

Hosoya et al. reported a different strategy to selectively permit signal transmission across different compartments, which uses the differential diffusion coefficients of DNA molecules across a gel matrix (Fig. 5b) [52]. Using an aluminum mold, they manufactured an array of millimetric-sized compartments separated by 500- μm thick alginate gel walls. While input/output single-stranded DNA can freely diffuse across the walls, DNA gates are grafted to polyacrylamide anchors that prevent their diffusion from compartment to compartment. By filling different compartments with different molecules (input, gates or catalyst), the authors created a primitive cellular automaton—termed *gellular automaton*—that forms specific patterns according to the position and type of the initialized cells. The work also showed that, in absence of an active signal amplification system (i.e., a positive feedback loop), diffusion is a poor mechanism to transmit information over large distances.

2.3.3 Liposomes and Lipid Bilayers

Although liposomes' boundaries are impermeable to many hydrophilic or charged compounds, it is possible to engineer pores with soluble proteins that spontaneously insert in the bilayer—many of these proteins are toxins that bacteria use to vampirize target cells. Protein pores have an inner channel whose dimensions are well-defined and can be extremely selective in the size and type of molecules that they allow through the membrane. Some pores are controllable through external signals such as light [53]. Insertion of α -hemolysin pores in liposomes creates semi-opened liposomal compartments that exchange fuel and waste with the environment [54], extending the lifetime of the enclosed molecular network (Fig. 5e). In another study, the same pore was used to create a self-selection feedback for directed evolution [55]. However, α -hemolysin's small size limits its use to transport DNA signals.

When brought in close contact, phospholipid-stabilized droplets form bilayers, known as Droplet Interface Bilayer (DIB), in which membrane pores can be inserted. Dupin et al. followed this approach to spatially array dozens of microdroplets connected again by α -hemolysin [26]. Since the inner channel of α -hemolysin is narrow, the study relied on small chemicals rather than DNA as signal molecules. The authors thus created interfacing modules to connect these signals to IVTT- or strand displacement-based circuits. In the case of IVTT, they showed that the protein pore itself can be one of the outputs of the circuit, allowing feedbacks that build on the reinforcement of the communication between two compartments. Pore insertion is a random phenomenon with a potential for strong signal amplification (given that an important chemical gradient exists across the membrane), and this led to the observation of the stochastic differentiation of initially identical microcompartments.

2.3.4 On-Support Immobilization of DNA Species

Since DNA-based networks always involve multicomponent reactions, an alternative to building (semi) permeable enclosures is to attach one (or more, but not all) of the reactive species to an immobile support [56]. This is an attractive option because many strategies and commercial modifications are available to graft DNA to various supports, and these chemical steps have been intensively studied for applications such as microarray and, more recently, Next-Generation Sequencing (NGS). In addition, a number of DNA amplification schemes or DNA reactions compatible with a solid support are available, including bridge PCR, primer walking reactions [56], primer grafted Recombinase Polymerase Amplification (RPA) or other enzymatic reactions [57].

This option was explored in [41], where porous microbeads carrying DNA instructions encoding autocatalytic CRNs were dispersed in a continuous layer of PEN-toolbox mixture (i.e., the dNTP and enzymes necessary to run the DNA instructions, Fig. 5c). In the simplest case, sharp and sustained gradients of concentrations were established around the active microbeads. In addition, various bead types, carrying different instruction sets, could be combined to create large scale collective behaviors. Precise mathematical formulas were derived to rationalize the observation of localized PEN out-of-equilibrium behaviors (depending on whether the geometry was 2D, 2.5D or 3D). A key insight from this study was that the kinetics of the reactions involved in the CRN impose a critical length scale for the emergence of localized behaviors in the absence of physical boundaries. In particular, DNA instructions-carrying beads have to be large enough to be able to maintain an individuality; if the bead is too small or if the chemistry is too slow compared to diffusion, then all beads perform identically, and the potential for parallelism is lost.

3 Discussion and Applications

Mass-action kinetic systems can be split without altering their dynamical properties, and this property extends to computational chemical networks. The insensitivity to scale is generally true down to microscopic length scales, offering the possibility for massive parallelism using the same amount of material as used for a single “classic” experiment (e.g., micro- to milliliters).

This property of CRNs is relatively unique in computational systems. It is alien to standard silicon-based hardware and in general from all computation devices based on physical processes (e.g., mechanical computers). However, being split-resilient is a property observed in many biological systems, not only unicellular, but also higher organisms including many plants, fungi and some animals: for these organisms, a splitting event creates two viable individuals from one, with properties identical to each other and to the parent. In many cases, the two parts will grow back to the size and shape (intensive variables) of their parent or at least have the potential to do so,

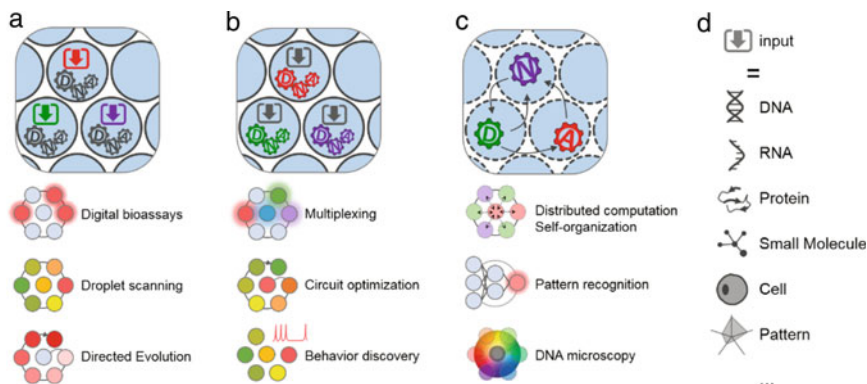


Fig. 6 Applications of parallelized molecular computations. **a** independent compartments containing an identical circuit but receiving different inputs. **b** independent compartments containing different circuits (e.g., from a parametric family) all working on the same inputs. **c** cross-talking compartments collaborating to compute a global response. **d** examples of input types

if the conditions are favorable. In the *animalia* kingdom, the hydra is the classical example.

Beside the biological analogy, we believe this unique asset ought to be better explored and exploited in the future of artificial molecular computations. Depending on whether the sub-compartments are independent or not, we see three interesting directions to harness parallel CRNs (Fig. 6).

- i. Independent compartments containing an identical circuit but receiving different inputs
- ii. Independent compartments containing different circuits, all working on the same inputs
- iii. Cross-talking compartments collaborating to compute a global response.

3.1 Independent Compartments Containing an Identical Circuit but Receiving Different Inputs

In the simplest case, split resilience can be used to repeatedly perform the same operation on different inputs. Bacteria, for example, can be seen as vehicles of their genomes, each testing the reproductive success of a slightly different genome variant or testing the same variant under different external forcings.

The same idea can be used for microcompartmentalized artificial molecular programs. This is because microcompartmentalization itself is a way to diversify inputs. For a given reactor of volume V , as soon as input concentration gets close to the critical value $C_c = (VN)^{-1}$ (where N is Avogadro's number), stochastic partitioning ensures that each compartment receives a different count of inputs. In the case where $[inputs] \ll C_c$, only two types of compartments are created, with or without

one molecule of input. This latter case is of particular interest if input molecules come from a library: a collection of similar, yet distinct molecules—for example, a partially randomized DNA strand. Indeed, Poissonian partitioning ensures that all members of the library will be processed in an identical way and independently of all others.

Building on this idea, one of us reported a Positive Feedback Loop (PFL) network that tests the presence of a specific enzymatic activity and accordingly enables (or not) genetic amplification. Distributed in microdroplets together with a genetic library, this system implements an *in vitro* evolution algorithm, poised to select for a pre-defined catalytic activity. Over rounds of selection and diversification, this molecular algorithm will spontaneously ascend the local fitness landscape [58]. Although Directed Evolution experiments are not usually described in terms of an algorithmic search process, it is likely that the ability to perform precise computations on molecular signals, at a molecular level, and with a molecular outcome—i.e., molecular programming—can bring strong benefits to this field.

This molecular evolution approach could be extended to optimize DNA-based molecular programs themselves, since their behavior is also genetically encoded. An array of compartmentalized molecular programs could clone and differentiate, each new compartment computing a function in a slightly different manner. By applying a selection pressure, one could then optimize a given molecular program or discover new functional architectures. This idea has been applied *in silico* [59], but presents intimidating technical difficulties *in vitro*, since it requires one to engineer a feedback loop that affects the architecture of the molecular program itself.

In principle, different inputs can also be deterministically injected in the various sub-compartments. When working with very small volumes, it can be experimentally tricky to address each compartment individually, although various microfluidic or robotic platforms exist [60–62]. Approaches to create bead-displayed diversity, for example, using Split and Pool [63], could be explored in this respect. Given that the dominant application of molecular programming is to process input signals that are molecular—in an attempt to maintain the full sensing–processing–actuation chain at the molecular level—it becomes important to develop interfacing strategies for various types of molecules, including for example proteins [64, 65] and small molecules [66].

3.2 Independent Compartments Containing Different Circuits, All Working on the Same Inputs

The second option to harness the parallelism of molecular circuits is to create many compartmentalized reactors containing different circuits. This basically allows one to apply a family of functions to a common input condition—an approach that resembles the paradigm of Single Instruction Multiple Data in parallel computing.

Genot et al. demonstrated this idea by introducing a microfluidic droplet chip that scans up to three component concentrations simultaneously [21]. With this device, millions of droplets, each containing a different combination of the three compounds of interest, and collectively mapping any desired parameter subspace, were generated in less than one hour. Fluorescent inert dropcodes encoded the concentration of the parametric compounds and allowed one to recover the experimental condition for each droplet, even after they have been mixed and manipulated at will.

In the process of building a DNA-based reaction network, one must decide the concentration of each functional DNA strand, enzymes, additives, etc. Since all these parameters are described by continuous variables, this complexity leads to painstaking rounds of trial and error, before a functional region can be found in parameter space. The microfluidic platform was then used to map the experimental bifurcation diagram of a predator–prey oscillator, scanning simultaneously its principal control parameters: the prey reproduction rate, the global growth rate and the species lifetime (through the concentration of an exonuclease). The finesse of scanning exposed a tiny region of parameters with an exotic behavior: there, droplets would oscillate for some time, then stay completely flat for a number of periods, then resume strong regular oscillations. The observation of this mode, known as “hard excitable”, remains to date the only example in DNA-based circuits.

More generally, this droplet-scanning platform addresses a common issue of multicomponent systems which require the titration of many concentrations. In principle, parallelization using microcompartments could be used to expedite the optimization process, by testing different conditions in each microreactor. Note that this assumes that the compound of interest does not cross the compartment boundaries [25, 26].

3.3 Cross-Talking Compartments Collaborating to Compute a Global Response

Notwithstanding the examples presented above, it is still difficult to distribute a molecular program in many sub-compartments and simultaneously control the communication channels between these units. Pioneering works based on top-down microfluidics and microfabrication [23] involve a dozen of compartments at most, but it is likely that thousands—or even millions—of compartments would be needed for a powerful parallel computation to emerge.

On the bottom-up (self-assembly) side, the trafficking of molecules across vesicles or droplet interfaces is still challenging. Molecular pores, which spontaneously insert in lipid bilayers, provide an interesting option, but DNA strands are large macromolecules, and biological channel proteins do not naturally allow DNA traffic. For instance, the pores used in nanopore sequencing evolved to traffic small ions and organic molecules. It took a lot of protein engineering to thread DNA through them, and the process is not spontaneous: it requires a voltage difference as driving force

and a polymerase for regulation [67]. The engineering of DNA-based membranes pores with larger apertures could provide some interesting options.

Here again, Nature could provide some inspiration, as cells do not use protein pores to traffic DNA, but instead the fusion and splitting of small vesicles—of which exosomes provide a prominent example. Exosomes are nanometer-sized (~40–200 nm) extracellular compartments that are secreted by some cells, enclosing biomolecules like RNA, DNA or proteins, and which are captured by other cells at remote locations. Although the exact mechanism regulating the production and absorption (endocytosis) is still being elucidated, exosomes seem to work as a “postal service” by which cells can send molecular messages to each other [68].

Finally, although we have focused here on the spatial parallelization of mass-action CRNs—that is, systems governed by concentrations and Ordinary Differential Equations (ODEs)—we stress that other approaches exploit the parallelism of computational chemical operations to manipulate complex libraries [69, 70]. One possibility is to perform the computation directly within a single (supra-) molecule in solution, such as a single DNA nanostructure [71–73]. Another approach would be to encode the computation in successive transformation of a single DNA sequence. For this purpose, the developing toolkit of solid-phase DNA manipulations [57] could enable sequential molecular transformations at tremendous throughputs.

What could be the future applications of parallel molecular computations? There are many tantalizing areas where it could solve open problems. In directed evolution, it could steer the evolution of DNA, RNA or proteins toward highly functional forms. Rather than screening for the absence or presence of a function in a molecule (single objective evolution), a molecular circuit could weigh multiple objectives (e.g., for an enzyme, it could assess both its thermodynamic constants, measured by its Michaelis–Menten constant K_m , and its kinetic constant, measured by its turnover rate V) and assign an integrated fitness score to each compartment. After each round, a compartment would get a chance to be replicated according to its fitness, thus steering the population toward a region of desirable characteristics. Parallel molecular computations could also empower the nascent field of DNA data storage, for instance, to find the number of occurrences of a pattern in a database. A DNA database could be split and compartmentalized, and the same computation could be applied simultaneously to each compartment by a molecular program. For very large database (say in the Petabytes or Exabytes range or over), molecular computations may beat electronic computations in terms of energy-cost or speed, simply because their scaling is much better.

Acknowledgements The authors wish to acknowledge the support of the ERC (CoG ProFF 647275 to Y.R. and StG MoP-MiP 949493 to G.G.), the PEPR MolecularArXiv and the insightful remarks of Tom de Greef, Fritz Simmel and an anonymous reviewer during the review process.

References

1. J. Kim, K.S. White, E. Winfree, Construction of an in vitro bistable circuit from synthetic transcriptional switches. *Molecular Syst. Biol.* **2**, 68 (2006)
2. S. Ayukawa, M. Takinoue, D. Kiga, RTRACS: a modularized RNA-dependent RNA transcription system with high programmability. *Accounts Chem. Res.* **44**, 1369–1379 (2011)
3. K. Montagne, R. Plasson, Y. Sakai, T. Fujii, Y. Rondelez, Programming an in vitro DNA oscillator using a molecular networking strategy. *Molecular Syst. Biol.* **7** (2011)
4. G. Seelig, D. Soloveichik, D.Y. Zhang, E. Winfree, Enzyme-free nucleic acid logic circuits. *Science* **314**, 1585–1588 (2006)
5. A. Phillips, L. Cardelli, A programming language for composable DNA circuits. *J Roy Soc Interface* **6**, S419–S436 (2009)
6. A. Baccouche, K. Montagne, A. Padirac, T. Fujii, Y. Rondelez, Dynamic DNA-toolbox reaction circuits: a walkthrough. *Methods* **67**, 234–249 (2014)
7. D. Soloveichik, G. Seelig, E. Winfree, DNA as a universal substrate for chemical kinetics. *Proc. National Acad. Sci.* **107**, 5393–5398 (2010)
8. D.Y. Zhang, E. Winfree, Dynamic allosteric control of noncovalent DNA catalysis reactions. *JACS* (2008)
9. D.Y. Zhang, A.J. Turberfield, B. Yurke, E. Winfree, Engineering entropy-driven reactions and networks catalyzed by DNA. *Science* **318**, 1121–1125 (2007)
10. A. Padirac, T. Fujii, Y. Rondelez, Nucleic acids for the rational design of reaction circuits. *Curr. Opin. Biotech.* **24**, 575–580 (2013)
11. L. Qian, E. Winfree, Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**, 1196–1201 (2011)
12. A. Padirac, T. Fujii, Y. Rondelez, Bottom-up construction of in vitro switchable memories. *Proc. National Acad. Sci.* **109**, E3212–E3220 (2012)
13. T. Fujii, Y. Rondelez, Predator-prey molecular ecosystems. *ACS Nano.* **7**, 27–34 (2013)
14. J. Kim, E. Winfree, Synthetic in vitro transcriptional oscillators. *Mol. Syst. Biol.* **7**, 465–465 (2011)
15. K. Montagne, G. Gines, T. Fujii, Y. Rondelez, Boosting functionality of synthetic DNA circuits with tailored deactivation. *Nat. Commun.* **7**, 13474 (2016)
16. D.Y. Zhang, E. Winfree, Control of DNA strand displacement kinetics using toehold exchange. *J. Am. Chem. Soc.* **131**, 17303–17314 (2009)
17. S.W. Schaffter, R. Schulman, Building in vitro transcriptional regulatory networks by successively integrating multiple functional circuit modules. *Nat. Chem.* **11**, 829–838 (2019)
18. K. Hasatani et al., High-throughput and long-term observation of compartmentalized biochemical oscillators. *Chem. Commu.* **49**, 8090–8092 (2013)
19. A. Baccouche et al., Massively parallel and multiparameter titration of biochemical assays with droplet microfluidics. *Nat. Protoc.* **12**, 1912–1932 (2017)
20. M. Weitz et al., Dynamical diversity of a compartmentalized programmable biochemical oscillator. *Proc. Natl. Acad. Sci.* **104**, 20167–20172 (2007)
21. A.J. Genot et al., High-resolution mapping of bifurcations in nonlinear biochemical circuits. *Nat. Chem.* **8**, 760–767 (2016)
22. M. Weitz et al., Diversity in the dynamical behaviour of a compartmentalized programmable biochemical oscillator. *Nat. Chem.* **6**, 295–302 (2014)
23. E. Karzbrun, A.M. Tayar, V. Noireaux, R.H. Bar-Ziv, Programmable on-chip DNA compartments as artificial cells. *Science* **345**, 829–832 (2014)
24. P. van Nies et al., Self-replication of DNA by its encoded proteins in liposome-based synthetic cells. *Nat. Commu.* **9**, 1583 (2018)
25. P. Gruner et al., Controlling molecular transport in minimal emulsions. *Nat. Commu.* **7**, 10392 (2016)
26. A. Dupin, F.C. Simmel, Signalling and differentiation in emulsion-based multi-compartmentalized in vitro gene circuits. *Nat. Chem.* **11**, 32–39 (2019)

27. S.-Y. Teh, R. Lin, L.-H. Hung, A.P. Lee, Droplet microfluidics. *Lab Chip* **8**, 198–220 (2008)
28. E. Amstad et al., Robust scalable high throughput production of monodisperse drops. *Lab. Chip* **16**, 4163–4172 (2016)
29. S. Yadavali, H.-H. Jeong, D. Lee, D. Issadore, Silicon and glass very large scale microfluidic droplet integration for terascale generation of polymer microparticles. *Nat. Commun.* **9**, 1222 (2018)
30. D. Pekin et al., Quantitative and sensitive detection of rare mutations using droplet-based microfluidics. *Lab. Chip* **11**, 2156–2166 (2011)
31. M. Morita et al., Droplet-shooting and size-filtration (DSSF) method for synthesis of cell-sized liposomes with controlled lipid compositions. *Chem. Bio. Chem.* **16**, 2029–2035 (2015)
32. K. Kamiya, R. Kawano, T. Osaki, K. Akiyoshi, S. Takeuchi, Cell-sized asymmetric lipid vesicles facilitate the investigation of asymmetric membranes. *Nat. Chem.* **8**, 881–889 (2016)
33. N. Ichihashi et al., Constructing partial models of cells. *CSH Perspect. Biol.* **2**, a004945 (2010)
34. V. Noireaux, Y.T. Maeda, A. Libchaber, Development of an artificial cell, from self-organization to computation and self-reproduction. *PNAS* **108**, 3473–3480 (2011)
35. Y. Rondelez et al., Microfabricated arrays of femtoliter chambers allow single molecule enzymology. *Nat. Biotechnol.* **23**, 361–365 (2005)
36. A.M. Tayar, E. Karzbrun, V. Noireaux, R.H. Bar-Ziv, Propagating gene expression fronts in a one-dimensional coupled system of artificial cells. *Nat. Phys.* **11**, 1037–1041 (2015)
37. F. Naqib, et al., Tunable oscillations and chaotic dynamics in systems with localized synthesis. *Phys. Revi. E, Statis., Nonlinear, and Soft Matter Phys.* **85**, 046210 (2012)
38. S. Sakakihara, S. Araki, R. Iino, H. Noji, A single-molecule enzymatic assay in a directly accessible femtoliter droplet array. *Lab. Chip* **10**, 3355 (2010)
39. J. Horváth, I. Szalai, P.D. Kepper, An experimental design method leading to chemical Turing patterns. *Science* **324**, 772–775 (2009)
40. J. Boissonade, P.D. Kepper, Transitions from bistability to limit cycle oscillations. Theoretical analysis and experimental evidence in an open chemical system. *J. Phys. Chem.*, 1–6 (1980)
41. G. Gines, et al., Microscopic agents programmed by DNA circuits. *Nat. Nanotech.*, 1–11 (2017). <https://doi.org/10.1038/nnano.2016.299>
42. A. Estevez-Torres, Un microlaboratoire électrophorétique pour l'étude du couplage entre transport et cinétique chimique : application à la réaction d'hybridation d'oligonucléotides.
43. M. Loose, E. Fischer-Friedrich, J. Ries, K. Kruse, P. Schwill, Spatial regulators for bacterial cell division self-organize into surface waves in vitro. *Science* **320**, 789–792 (2008)
44. A. Padirac, T. Fujii, A. Estévez-Torres, Y. Rondelez, Spatial waves in synthetic biochemical networks. *J. Am. Chem. Soc.* **135**, 14586–14592 (2013)
45. A.S. Zadorin et al., Synthesis and materialization of a reaction–diffusion French flag pattern. *Nat. Chem.* **9**, 990–996 (2017)
46. S.M. Chirieleison, P.B. Allen, Z.B. Simpson, A.D. Ellington, X. Chen, Pattern transformation with DNA circuits. *Nat. Chem.* **5**, 1000–1005 (2013)
47. A. Zambrano, A.S. Zadorin, Y. Rondelez, A. Estévez-Torres, J.-C. Galas, Pursuit-and-evasion reaction-diffusion waves in microreactors with tailored geometry. *J. Phys. Chem. B* **119**, 5349–5355 (2015)
48. A.S. Zadorin, Y. Rondelez, J.-C. Galas, A. Estévez-Torres, Synthesis of programmable reaction-diffusion fronts using DNA catalyzers. *Phys. Rev. Lett.* **114**, 068301 (2015)
49. G. Urtel, A. Estevez-Torres, J.-C. Galas, DNA-based long-lived reaction–diffusion patterning in a host hydrogel. *Soft Matt.* **15**, 9343–9351 (2019)
50. M.V.D. Ofstadt, G. Gines, J. Galas, A. Estevez-Torres, DNA- and RNA-based computing systems, 185–212 (2021). <https://doi.org/10.1002/9783527825424.ch10>
51. A. Joesaar et al., DNA-based communication in populations of synthetic protocells. *Nat. Nanotechnol.* **14**, 369–378 (2019)
52. T. Hosoya, I. Kawamata, S.M. Nomura, S. Murata, Pattern formation on discrete gel matrix based on DNA computing. *New Generat. Comput.* **37**, 97–111 (2019)
53. D. Tischer, O.D. Weiner, Illuminating cell signalling with optogenetic tools. *Nat. Rev. Mol. Cell. Bio.* **15**, 551–558 (2014)

54. V. Noireaux, A. Libchaber, A vesicle bioreactor as a step toward an artificial cell assembly. *Proc. Natl. Acad. Sci. USA* **101**, 17669–17674 (2004)
55. S. Fujii, T. Matsuura, T. Sunami, Y. Kazuta, T. Yomo, In vitro evolution of α -hemolysin using a liposome display. *Proc. Nat. Acad. Sci.* **110**, 16796–16801 (2013)
56. Z. Ma et al., Isothermal amplification method for next-generation sequencing. *Proc. Nat. Acad. Sci.* **110**, 14320–14323 (2013)
57. M. de Raad, S.A.A. Kooijmans, E.A. Teunissen, E. Mastrobattista, A solid-phase platform for combinatorial and scarless multipart gene assembly. *ACS Synt. Biol.* **2**, 316–326 (2013)
58. A. Dramé-Maigné, R. Espada, G. McCallum, R. Sieskind, Y. Rondelez, Directed evolution of enzymes based on in vitro programmable self-replication. *Biorxiv* (2021). <https://doi.org/10.1101/2021.04.22.440993>
59. Q.H. Dinh, et al., An effective method for evolving reaction networks in synthetic biochemical systems. *IEEE Trans. Evol. Comp.*, 1–13 (2013)
60. Y. Zhang, et al., Accurate high-throughput screening based on digital protein synthesis in a massively parallel femtoliter droplet array. *Sci. Adv.* **5**, eaav8185 (2019)
61. A. Kulesa, J. Kehe, J.E. Hurtado, P. Tawde, P.C. Blainey, Combinatorial drug discovery in nanoliter droplets. *Proc. Nat. Acad. Sci.* **115**, 201802233 (2018)
62. R.F.-X. Tomasi, S. Sart, T. Champetier, C.N. Baroud, Individual control and quantification of 3D spheroids in a high-density microfluidic droplet array. *Cell Rep.* **31**, 107670 (2020)
63. A.B. Rosenberg et al., Single-cell profiling of the developing mouse brain and spinal cord with split-pool barcoding. *Science* **360**, 176–182 (2018)
64. G. Adornetto, A. Porchetta, G. Palleschi, K.W. Plaxco, F. Ricci, A general approach to the design of allosteric, transcription factor-regulated DNAzymes. *Chem. Sci.* **6**, 3692–3696 (2015)
65. W. Engelen, L.H.H. Meijer, B. Somers, T.F.A. de Greef, M. Merckx, Antibody-controlled actuation of DNA-based molecular circuits. *Nat. Commun.* **8**, 14473 (2017)
66. V.A. Shenshin, C. Lescanne, G. Gines, Y. Rondelez, A small-molecule chemical interface for molecular programs. *Nucleic Acids Res.* **49**, gkab470 (2021)
67. D. Deamer, M. Akeson, D. Branton, Three decades of nanopore sequencing. *Nat. Biotechnol.* **34**, 518–524 (2016)
68. J. Maia, S. Caja, M.C.S. Moraes, N. Couto, B. Costa-Silva, Exosome-based cell-cell communication in the tumor microenvironment. *Frontiers Cell. Dev. Biol.* **6**, 18 (2018)
69. L.M. Adleman, Molecular computation of solutions to combinatorial problems. *Science* **266**, 1021–1024 (1994)
70. M. Cui, et al., Winner(s)-take-all: nonlinear amplification of DNA-encoded library. *Biorxiv*, 744680 (2019). <https://doi.org/10.1101/744680>
71. G. Chatterjee, N. Dalchau, R.A. Muscat, A. Phillips, G. Seelig, A spatially localized architecture for fast and modular DNA computing. *Nat. Nanotechnol.* **12**, 920–927 (2017)
72. F.C. Simmel, DNA-based assembly lines and nanofactories. *Curr. Opin. Biotechnol.* **23**, 516–521 (2012)
73. M. Teichmann, E. Kopperger, F.C. Simmel, Robustness of localized DNA strand displacement cascades. *ACS Nano*. **8**, 8487–8496 (2014)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Social DNA Nanorobots



Ming Yang and John Reif

Abstract We describe *social DNA nanorobots*, which are autonomous mobile DNA devices that execute a series of pair-wise interactions between simple individual DNA nanorobots, causing a desired overall outcome behavior for the group of nanorobots which can be relatively complex. We present various designs for social DNA nanorobots that walk over a 2D nanotrack and collectively exhibit various programmed behaviors. These employ only hybridization and strand-displacement reactions, without use of enzymes. The novel behaviors of *social DNA nanorobots* designed here include: (i) *Self-avoiding random walking*, where a group of DNA nanorobots randomly walk on a 2D nanotrack and avoid the locations visited by themselves or any other DNA nanorobots. (ii) *Flocking*, where a group of DNA nanorobots follow the movements of a designated leader DNA nanorobot, and (iii) *Voting by assassination*, a process where there are originally two unequal size groups of DNA nanorobots; when pairs of DNA nanorobots from distinct groups collide, one or the other will be assassinated (by getting detached from the 2D nanotrack and diffusing into the solution away from the 2D nanotrack); eventually all members of the smaller groups of DNA nanorobots are assassinated with high likelihood. To simulate our social DNA nanorobots, we used a surface-based CRN simulator.

1 Introduction

1.1 Motivation

Due to its simple base-pairing and predictable secondary structure, DNA is an ideal material to construct useful molecular-scale objects and devices. *DNA nanorobots* are molecular-scale synthetic devices composed primarily of DNA that can execute a variety of operations. *Autonomous DNA nanorobots* operate without outside medi-

M. Yang (✉) · J. Reif (✉)
Department of Computer Science, Duke University, Durham, NC 27708, USA
e-mail: myang@cs.duke.edu

J. Reif
e-mail: reif@cs.duke.edu

© The Author(s) 2023

N. Jonoska and E. Winfree (eds.), *Visions of DNA Nanotechnology at 40 for the Next 40*,
Natural Computing Series, https://doi.org/10.1007/978-981-19-9891-1_20

ation. *DNA walkers* are a class of mobile DNA nanorobots which can move over a *nanotrack* composed of DNA stepping stones. The nanotrack may be 1D or 2D and may be either a self-assembled DNA nanostructure or a set of DNA strands affixed to a surface. Autonomous DNA nanorobots have been demonstrated to perform a small number of moderately complex tasks including walking over nanostructures, maze traversal, and cargo delivery activities.

A major remaining challenge in the field of DNA nanoscience is to increase the complexity and diversity of the activities DNA nanorobots can perform in spite of practical limitations on the complexity of each individual DNA nanorobot. How can one design molecular-scale systems with multiple mobile autonomous nanorobots which exhibit a group behavior that is significantly more complex than the behavior of individual nanorobots? We take inspiration from the field of Sociobiology pioneered by Wilson, which demonstrated how social insects such as ants and honeybees perform a wide variety of relatively complex organized group behaviors, even though the individual insects have quite limited brains.

1.2 Summary of Our Results

We describe *social DNA nanorobots*, which are autonomous mobile DNA devices that execute a series of pair-wise interactions between simple individual DNA nanorobots, causing a desired overall outcome behavior for the group of nanorobots which can be relatively complex. We present various designs for social DNA nanorobots that walk over a 2D nanotrack and collectively exhibit various programmed behaviors. In our designs, we employed only hybridization and strand-displacement reactions, without use of enzymes; strand-displacement reactions are used for communication between pairs of nearby DNA nanorobots (where a finite amount of information is transferred between a pair of nearby DNA nanorobots). The novel behaviors of *social DNA nanorobots* designed here include:

- (i) *Self-avoiding random walking*, where a group of DNA nanorobots randomly walk on a 2D nanotrack and avoid the locations visited by themselves or any other DNA nanorobots.
- (ii) *Flocking*, where a group of DNA nanorobots follow the movements of a designated leader DNA nanorobot, and
- (iii) *Voting by assassination*, a process where there are originally two unequal size groups of DNA nanorobots; when pairs of DNA nanorobots from distinct groups collide, one or the other will be assassinated (by getting detached from the 2D nanotrack and diffusing into the solution away from the 2D nanotrack); eventually all members of the smaller groups of DNA nanorobots are assassinated with high likelihood.

We simulated our social DNA nanorobots using the Surface CRN Simulator of Clamons[1, 2].

1.3 Organization

- In Sect. 2, we overview the field of Sociobiology and in particular the behavior of social insects.
- In Sect. 3, we review prior DNA nanorobots, including prior DNA walkers, and programmable DNA nanomachines.
- In Sect. 4, we present our designs and simulations of various social DNA nanorobots.
- In Sect. 5, we conclude with a discussion of future work and challenges, including: further development of simulation software for social nanorobots, experimental demonstrations of social DNA nanorobots, design of further social DNA nanorobot behaviors, and possible use of the diffusion of pheromone-like DNA molecules for communication between social DNA nanorobots.

2 Sociobiology

The concept of Sociobiology was defined by Wilson [3] in 1975. The field of Sociobiology aims to investigate and explain the evolved social behaviors of social animals. Sociobiology studies have been made for example of mating patterns, aggression, nurturance, pack hunting, and the hive society of social insects. High-level social organizations can be found in social insects that have the following three characteristics: cooperative brood care, overlapping generations, and a division of labor into reproductive and non-reproductive groups[4–8]. Social insects include ants and termites, and some social bees and wasps. Social insects gain several advantages by living together. They work together to gain resources, share their findings with each other, defend their home when under attack, and attack other insects for territory and food. Social insect communities are divided into castes by their function and behavior; these include a reproductive caste (e.g., the queen) and the sterile caste (soldiers and various types of workers) [9]. The reproductive caste carries out the basic function of reproduction, and the sterile castes do various types of tasks including taking care of the reproductive members. There are generally multiple subcastes of insects within the sterile castes, which do specialized tasks. For example, the soldiers defend the colony against predators and the workers are responsible for foraging, construction and repair of the nest and feeding the larvae and brood care, all tasks that are typically done by specialized sterile subcastes (which may depend on the insects age or development).

The communication signals between the social insects required for this control can be mechanical, optical, or chemical. Mechanical communication between social insects includes direct physical contact between members of the insect colony. Optical communication between social insects includes visual displays and stylized movements, sometimes termed dances, that communicate discoveries of food and their locations. Olfactory communication between social insects includes chemical tracks

that specify a path to important locations, as well as diffused chemical factors known as pheromones. Pheromones can trigger a social response in members of the same species, also often playing an important role in the development and maintenance of insect society [10]. Also, ants and termites are specialized to perform specific functions (e.g., attacking & foraging) and can communicate and lay down chemical tracks to specify a path to important locations such as food sources. (See Sect. 5 for a discussion of possible use of diffusion of pheromone-like molecules for communication between nanorobots.)

Quite surprisingly, the overall control of these insect communities is not done by the reproductive caste, but instead the control is done distributively within the sterile subcastes via group social interactions. Particularly, complex collective behavior is found in honeybee colonies, where individual honeybees can be specialized for foraging and harvesting. For example, after traveling outside the colony's nest to find possible locations of flowers with pollen, successful honeybee foragers can share information about the direction and distance to a food source [11] by use of a flying dance known as a waggle dance (a particular figure-eight dance) [11–13]. These dances communicate (i) the existence of pollen, (ii) its quantity, and also (iii) its general direction. Also, the honeybees of the species *Apis mellifera* perform tremble dances, which recruit receiver bees to collect nectar from returning foragers [14]. Seeley [15] demonstrated that honeybees also use a form of democratic voting (executed without input from the queen bee) to make important decisions, such as the best source of flowering plants providing pollen or the best location for a new hive home for the honeybee colony.

Our paper is not concerned with evolution per se, but we propose designs (rather than evolution) of nanorobots which are derived from prior Sociobiology studies of these behaviors of social insects. We are particularly inspired by the social behaviors of social insects such as ants and bees, which exhibit complex collective behavior, even though the individual insects have quite limited brains, a property shared also by DNA nanobots. Notable diverse activities of social insects that we aim to mimic using DNA nanobots include:

- *Random walking*: where insects of the colony make random walks.
- *Flocking*: a group of insects of the colony follow a selected leader individual insect.
- *Guarding*: a group of insects of the colony follow, and guard from attack by another group, a particular individual insect of the colony.
- *Attacking*: a group of insects of the colony attack another group.
- *Communication*: between insects of the colony.
- *Democratic group decision making (voting)*: among groups of insects of the colony.
- *Foraging*: a select *foraging group* of insects of the colony leave the colony and attempt to discover new sources of food, and then report back to the colony their discoveries.
- *Harvesting*: a harvesting group of insects of the colony travel (navigating by either (a) following successful foragers or (b) following their chemical trail, or (c) via instruction from successful foragers) to the new sources of food and harvest it.

3 Prior DNA Nanorobots

We use the term *nanorobot* for a molecular-scale device that can execute a variety of operations. A *DNA (or RNA) nanorobot* is a nanorobot composed of nucleic acids. Our overall aim is to take inspiration from the field of Sociobiology to design novel nanorobotic systems, but our designs will employ in part various known nanorobotic design techniques, which we briefly overview in this section.

In 2000, Yurke and Turberfield [16] demonstrated the first DNA device, a DNA tweezer, that used DNA hybridization to power its movements. Their DNA tweezer was nonautonomous; its movements were controlled by adding ssDNA strands. The field of DNA nanorobotics has rapidly evolved from *nonautonomous* molecular devices that each successive movement needed to be controlled externally, to subsequent *autonomous* molecular devices that operate without control from external environment. Autonomous DNA nanorobot devices that executed in-place motions were demonstrated by [17–20].

3.1 Prior DNA Walkers

A DNA nanorobot is *autonomous* if it operates without outside intermediate control, and otherwise is *non-autonomous*. In the past decades, researchers designed and experimentally realized DNA devices that can autonomously conduct complex tasks such as autonomous walking, maze traversal, and cargo delivery activities.

A *mobile DNA nanorobot* is a DNA nanorobot that locomotes in some way. A *DNA walker* (also termed a *Mobile DNA nanorobot*) is a mobile DNA nanorobot which moves over a *nanotrack* composed of ssDNA pads. The nanotrack may be 1D or 2D and may be either self-assembled DNA nanostructure or a set of DNA strands affixed to a surface. An *autonomous DNA walker* is a mobile DNA nanorobot that locomotes autonomously.

The concept of the DNA walker was first defined and named by Reif [21] in 2002, who gave two autonomous designs for bidirectional movement. Sherman and Seeman [22] and Shin and Pierce [23] then experimentally realized the first DNA walkers, which were bipedal walkers that moved along a linear track. But they were nonautonomous walkers that required external control for each step. Yin et al. [24] and Tian and Mao [19] also demonstrated biped non-autonomous walkers that walked foot-over-foot along a linear track.

In 2004, the first autonomous DNA walker was experimentally demonstrated [25, 26] by Reif's group (in collaboration with Turberfield). This autonomous DNA walker and many subsequent DNA walkers [20, 27–31] made use of a series of enzymic reactions to power the locomotion; for example, Sahu [29] demonstrated a DNA nanotransport device which is powered by strand-displacing polymerase $\phi 29$. Other autonomous DNA walkers employed DNAszymes, for example, Pei et al. [32] demonstrated a multipedal DNA walker that moves on a 2D substrate in a biased random walk and Lund [33] demonstrated DNA walkers that traversed paths on a

2D nanostructure guided by landmark molecules affixed to the 2D nanostructure. There are also many known DNA walkers powered by hybridization reactions. Also, autonomous DNA walkers have been demonstrated that navigate networks: Wickham et al. demonstrated a DNA-based molecular motor that can be routed through a network of tracks [34]. Chao et al. [35] demonstrated a DNA walker that conducted single-molecule parallel depth-first search on a 2D DNA origami surface. Reviews of DNA-based walkers are given in [36–38].

3.2 Prior Programmable DNA Nanorobots

Various schemes for programmable autonomous DNA nanorobots that do computations as they walk have been described; those of Yin and Reif et al. [39] use enzymes, and those of Reif and Sahu [40] use DNAzymes. One application of the DNAzyme programmable autonomous DNA nanorobot of [40] was a DNAzyme router for programmable routing of nanostructures on a 2D DNA addressable lattice, where the 2D DNA addressable lattice is embedded with a network of DNAzymes and where the routed path for the input nanostructure could be programmed by modifying its sequence. The input was encoded as a set of hairpins on the walker. The transport of the walker across the surface simulated a finite state machine that switched states based on input, where the state of the automaton was indicated by the DNAzyme that currently binds the walker. The various DNAzymes embedded on the 2D surface consumed the input as the walker moved.

The Seeman group [41] demonstrated a non-autonomous programmable DNA nanorobot that transported a series of molecules to form a molecular-scale assembly line. Their nanorobot picked up cargo in a programmable manner when it walked on a DNA track. This process was non-autonomous since it required addition of appropriate fuel strands at specific time instants.

3.3 Prior Autonomous DNA Walkers that Do Molecular Cargo-Sorting on a 2D Nanostructure

The prior work most similar to that reported here is the work of Thubagere [42], which demonstrated an ingenious molecular-scale system with a group of autonomous DNA nanorobots executing a molecular cargo-sorting task on a 2D nanostructure. The 2D nanostructure initially had various kinds of molecular cargo that needed to be transported to different targeted locations on a DNA nanostructure (a DNA origami surface). Each DNA nanorobot traversed a random walk over the 2D nanostructures and when encountering a molecular cargo, they loaded and transported the cargo to the targeted location (the goal). They used a simple protocol to perform a complex cargo-sorting task. When the robot randomly walked on the surface, if it moved local to a cargo molecule, the nanorobot picked the cargo up and continued walking

randomly. If it moved local to a goal molecule at the targeted location of the cargo molecule, the robot dropped the cargo off. The nanorobot repeated the above process until all cargo molecules were sorted. The picking up and dropping off process used strand displacement reactions.

4 Design and Simulation of Social DNA Nanorobots

Our approach is to adapt collective behavior strategies of social insects into molecular-scale nanorobots, which will be specialized to perform specific functions. We describe *social DNA nanorobots*, which are autonomous mobile DNA devices that execute a series of pair-wise interactions (only between pairs of nearby nanorobots) on a 2D DNA nanotrack that determine an overall desired outcome behavior as a group. We give detailed designs that program social behaviors via interactions between individual DNA molecules.

4.1 Social DNA Nanorobot Behaviors Designed and Simulated

A basic behavior of social DNA nanorobots is *Random Walking*, where a group of DNA nanorobots make random traversals of a 2D nanotrack. For random walking, we will make use of a known design of [42].

The novel behaviors of *social DNA nanorobots* described here include:

- *Self-avoiding random walking*, where a group of DNA nanorobots walk on a 2D nanotrack and avoid the locations visited by themselves or any other DNA nanorobots.
- *Flocking*, where a group of DNA nanorobots follow the movements of a designated leader DNA nanorobot.
- *Voting by assassination*, a process where there are originally two unequal size groups of DNA nanorobots; when pairs of DNA nanorobots from distinct groups collide, one or the other can be assassinated (by getting detached from the 2D nanotrack and diffusing into the solution away from the 2D nanotrack); eventually all members of the smaller groups of DNA nanorobots are assassinated with high likelihood.

4.2 Software for Stochastic Simulations of the Social DNA Nanorobots Behaviors

We made stochastic simulations of the social DNA nanorobots behaviors listed above. Our simulation model is adapted from the Surface CRN Simulator of Clamons [1, 2]. A *chemical reaction network (CRN)* contains chemical reactions and their rates.

In a *2D surface CRN model*, each individual nanorobot to be simulated is a molecule attached at a specific position on a 2D surface, so that the nanorobot can only interact with neighbors and their attachment strands. The behaviors of DNA nanorobots moving on a 2D nanotrack can be modeled in this 2D surface CRN model as a set of chemical reactions between DNA walkers and DNA strands affixed to a surface. State transitions modeled chemical reactions (e.g., toehold-mediated strand displacements and dehybridizations) between DNA walkers and DNA strands affixed to a surface. We applied the Surface CRN Simulator specifically for optimized performance assessment of our social DNA nanorobot designs. Assessment criteria are formulated to specify the performance of the designs of the social DNA nanorobots, and redesigns were made to improve performance.

In the next subsections, we give detailed domain level designs of social DNA nanorobots that conduct random walking, self-avoiding random walking, flocking and voting by assassination.

4.3 A Prior DNA Nanorobot that Autonomously Walks

There are many prior known designs for DNA nanorobots that make random traversals on a 2D DNA nanotrack. Our design for a DNA nanorobot that autonomously walks uses the design for a random DNA walker of Thubagere [42]. A nanotrack with an array of attached ssDNA pads that are self-assembled on a surface is illustrated in Figure 1 (for simplicity it is only illustrated in 1D). The 2D arrangement of pads on a 2D nanotrack is illustrated in Fig. 2. There are two types of pads: (i) ssDNA $p_0 = B^*A^*$ attached at its 3' end and (ii) ssDNA $p_1 = C^*B^*$ attached at its 5' end.

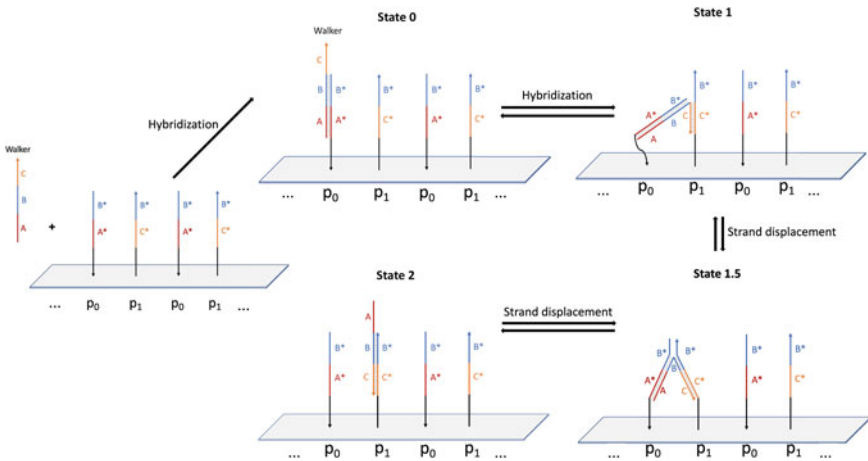
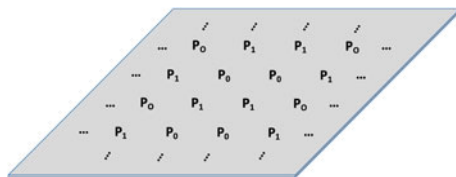


Fig. 1 Known design of a DNA nanorobot that executes a random walk (for simplicity only illustrated in 1D)

Fig. 2 2D locations of the random walker's pads on a 2D Nanotrack



(These ssDNA sequences, and all sequences given subsequently, are written in 5' to 3' order.) There is a single type of ssDNA nanorobot, the *DNA Walker* $W = ABC$ (see Fig. 1), which operates as follows:

- (a) A low concentration of W walker strands are added to the buffer solution containing the 2D nanotrack, and a few of these W strands hybridize to random pads of the nanotrack.
- (b) The buffer solution is replaced, so as to remove the remaining non-hybridized W walker strands from the solution surrounding the nanotrack.
- (c) As illustrated in Fig. 1, at first, the W strand hybridizes with a $p_0 = B^*A^*$ pad, which is the *State 0*. In *State 1*, the unpaired domain C of W hybridizes with domain C^* of p_1 . Then domain B^* of pad p_1 can displace the domain B^* of pad p_0 , so domain B of W hybridizes with both p_0 and p_1 , which is the intermediate *State 1.5*, and this process is reversible. If the strand displacement completes, it enters *State 2*, in which W detaches fully from the p_0 pad and hybridizes with a $p_1 = C^*B^*$ pad. From *State 0* to *State 2*, W walks from p_0 pad to p_1 pad due to hybridization and strand displacement. Similarly, W walks from p_1 pad to p_0 pad when it proceeds from *State 2* to *State 1.5* to *State 1* to *State 0*.
- (d) W walks successively from either (a) the p_0 to the p_1 pad of the nanotrack, or (b) the p_1 to the p_0 pad of the nanotrack, as described above. As result, W walks randomly over the nanotrack.

Figure 3 gives an example simulation (also see mp4 at [43]) run for the W nanorobot (in green) randomly walking on a 2D nanotrack. Where it randomly walks on the nanotrack, the grid turns to orange to show the trace of the W nanorobot. The W Walker traverses uniformly randomly over the nanotrack.

4.4 Prior Demonstrated Technique for Hybridization Inhibition of Short Sequences Within the Hairpin Loops

Our DNA nanorobots will make use of hybridization inhibition of short ssDNA sequences within the hairpin loops; this seems well established. Extensive prior published works (see survey [44]) have demonstrated (in simulation as well experimentally) localized reactions that make use of DNA hairpins for inhibition of hybridization on short ssDNA sequences within the hairpin loops. For example, the hybridization chain reaction (HCR) [45] makes use of a series of strand-displacements to open

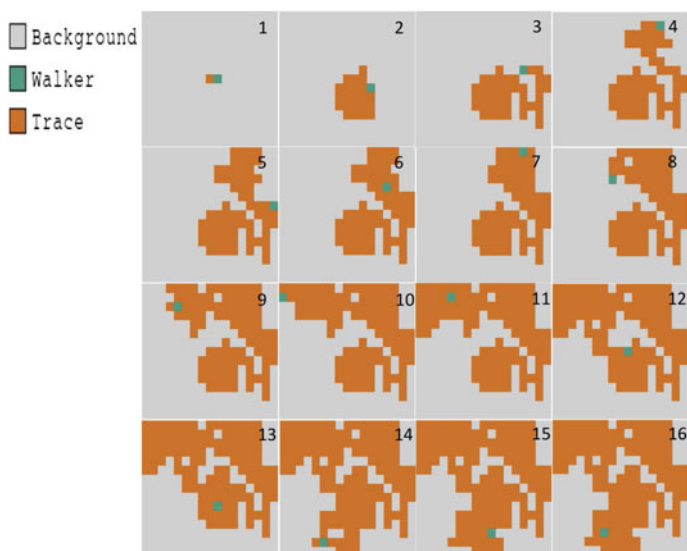


Fig. 3 Simulation of a DNA nanorobot that executes a random walk (with orthogonal steps) on a 2D nanotrack

a series of DNA hairpins; while HCR was initially demonstrated [45] in well-mixed solutions, Reif's group has shown that it also can be made to operate reliably in localized reactions. Reif's group has made extensive probabilistic analysis and simulations (in the studies [46–48]) and has also experimentally implemented (in the paper [49]) a localized version of HCR where the hairpins are localized on DNA Origami.

4.5 A Novel DNA Nanorobot that Executes a Self-Avoiding Walk

A *self-avoiding walk (SAW)* is a sequence of moves on a lattice (a lattice path) that does not visit the same point more than once. SAWs have a number of important applications, e.g., in the modeling of nucleic acids, peptides, and proteins. It is known [50] that a self-avoiding random walk on the 2D square lattice lasts an average of approximately 71 steps before the walker is trapped. (**Note:** While we could modify our design given the below to decrease the likelihood the walker gets trapped at a p_0 pad position, and so increase the average number of steps before the walker is trapped, but then the resulting system would not correspond to the classical self-avoiding random walk on the 2D square lattice.)

Here, we described the design of a DNA nanorobot that makes a random self-avoiding traversal of a 2D DNA nanotrack (which is a 2D square lattice). It is the first

to execute a self-avoiding walk without use of enzymes. There are two types of pads: (i) ssDNA $p_0 = B^*A^*$ attached at its 3' end and (ii) DNA hairpin $p_1 = C^*B^*E_1^*E_2^*B$ attached at its 5' end. Figure 4 illustrates (for simplicity on a 1D nanotrack) the self-assembly of the pads into DNA hairpins. Figure 5 gives the 2D arrangement of these pads on a 2D nanotrack. **Note:** To improve stability of the hairpin p_1 , its loop portion $E_1^*E_2^*$ needs to be relatively short compared to its stem sequences B and B^* . The initial buffer solution S_0 is a conventional buffer solution with the component strands of a DNA nanotrack, and the buffer solution S_1 is S_0 plus a high concentration of another DNA hairpin type $h_1 = E_2E_1BE_1^*$.

The operation of a self-avoiding walker $W = ABC$ is as follows:

- (a) A low concentration of W walker strands are added to the initial buffer solution S_0 containing the nanotrack, and a few of these W strands hybridize to random pads of the nanotrack.
- (b) The buffer solution S_0 is replaced with buffer solution S_1 , so as to remove the remaining non-hybridized W strands from the solution surrounding the nanotrack, and to add the DNA hairpin h_1 .
- (c) As in Fig. 4, the Walker W first hybridizes with a $p_0 = B^*A^*$ pad, which is *State 0*. In *State 1*, the unpaired domain C of W hybridizes with the domain C^* of the p_1 pad. Then the domain B^* of p_1 can displace the domain B^* of p_0 , the domain B of W hybridizes with both p_0 and p_1 and also hairpin p_1 is opened, giving intermediate *State 1.5*. (Note this transition process is reversible.) If and when the strand displacement finishes, it enters *State 2*, in which W hybridizes with a p_1 pad. In summary, transitioning from *State 0* to *State 2*, the W nanorobot walks from p_0 pad to p_1 pad due to the hybridization and strand displacement.
- (d) (Note our design includes a way to keep W from stepping to the next pad, therefore letting the track hairpin reform.) The ssDNA strand h_1 is in high concentration in buffer solution S_1 , with domain E_2 that can hybridize with the newly released E_2^* domain of p_1 , so h_1 is then opened and displaces the domain B of W . The unpaired domain AB of W can hybridize with the next p_0 and moves from p_1 to p_0 , entering *State 1*. After some time, W detaches from p_1 and hybridizes with a p_0 pad, taking it back to *State 0*. Then the previously visited p_1 will hybridize with a hairpin h_1 from the buffer solution, which hinders the p_1 's reformation of the hairpin, so W cannot move back to the visited p_1 . (Also, note that the small length of the loop of W deters h_1 from binding to the loop of W in *State 0*.)
- (e) The W nanorobot walks from either (a) p_0 to a non-visited p_1 pad of the nanotrack, or (b) p_1 to a p_0 pad of the nanotrack, as described in part c, d. As a result, the W nanorobot avoids the locations visited by itself or any other DNA nanorobot when it moves on the nanotrack. (A similar design can also be used for foraging nanorobots.)

Figure 6 (also see mp4 at [43]) gives an example simulation run for a W nanorobot (in green) that executes a self-avoiding random walk on a 2D nanotrack; it randomly walks on the nanotrack, and the grid turns to orange to show the trace of the W

nanorobot. The W walker randomly moves over the 2D nanotrack and then eventually stops when all the pads around it are visited.

4.6 Flocking: Novel DNA Nanorobots that Follow a Leader

This is a DNA nanorobot system with two types of DNA nanorobots (the designated leader and the followers): the leader makes a random traversal of a 2D nanotrack and the other DNA nanorobots follow the movements of the leader. Let $B_2 = B_{2a}B_{2b}$. There are two types of pads: (i) $h_1 = B_1B_2^*A_2^*B_{2b}B_1^*A_1^*$ attached at its 3' end and (ii) $h_2 = C_1^*B_1^*B_{2a}C_2^*B_2^*B_1$ attached at its 5' end. Figure 7 illustrates (for simplicity on a 1D nanotrack) the self-assembly of the pads into DNA hairpins. The 2D arrangement of these pads on a 2D nanotrack is illustrated in Fig. 8.

Note: Observe that each of h_1 and h_2 each self-assemble into a type of hairpin with two separate short loops: h_1 has two separate short loops, one with A_2^* and another with B_{2a}^* , whereas h_2 has one short loop with $B_{2b}^*C_2^*$. This use of small hairpin loops is a deliberate design with the goal of inhibiting the hybridization follower W_2 with A_2^* , B_{2a}^* , C_2^* , B_{2b}^* . This will make it more difficult for a follower W_2 to avoid following a leader W_1 . Note that to ensure stability of each of the hairpins, their loop portions need to be relatively short compared to the B_1 and B_1^* portions of their stems.

There are two types of ssDNA nanorobots, the *leader* $W_1 = A_1B_1C_1$ and the *follower* $W_2 = A_2B_2C_2$, which operate as follows:

- (a) W_1 (leader) and W_2 (follower) strands are added to buffer solution containing the nanotrack.
- (b) The leader W_1 hybridizes with h_1 on the domain A_1 and opens the hairpin of h_1 by strand displacement; then a follower W_2 can hybridize with the newly opened h_1 .
- (c) The leader W_1 moves from pad h_1 to pad h_2 by hybridizing with C_1^* of h_2 and opening h_2 by strand displacement; then follower W_2 can hybridize with the newly opened h_2 .
- (d) After W_1 and W_2 Walkers leave these pads, a limited number of further W_2 strands can walk nearby, and similarly follow leader W_1 .
- (e) As in Fig. 7 (for simplicity it is only illustrated in 1D), the leader W_1 walks successively from either (a) a h_0 pad to a h_1 pad of the nanotrack, or (b) a h_1 pad to a h_0 pad of the nanotrack. Whenever a follower W_2 stand hybridizes with a h_1 pad, the hairpin h_1 needs to be open (where the loop of the hairpin was opened by a leader W_1 and the loop closes after some time), so W_2 is forced to follow the leader W_1 . Hence, leader W_1 walks randomly over the nanotrack and is followed by a group of W_2 followers. (The maximum size of group of W_2 followers is limited by the duration that h_1 , h_2 remain open, and this parameter can be set by appropriate DNA strand design.)

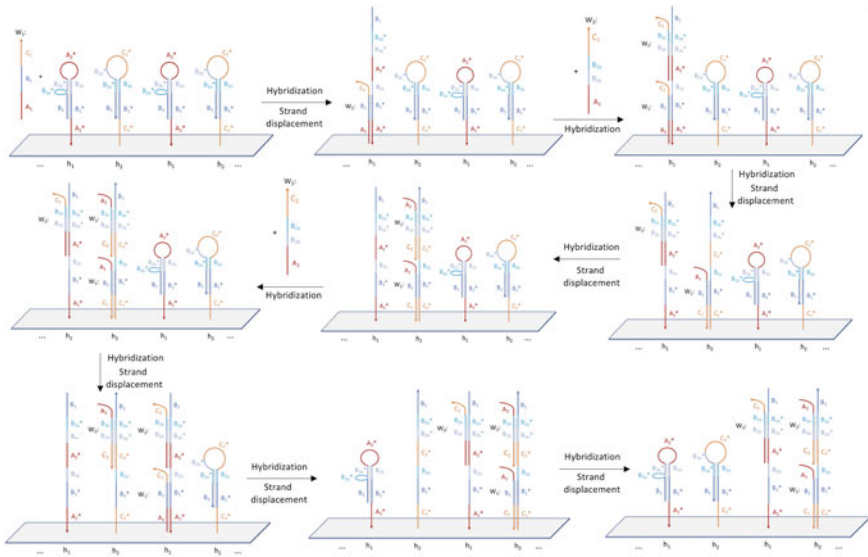


Fig. 7 Design of DNA nanorobots that follow a leader DNA nanorobot

Fig. 8 2D locations of the pads on the 2D nanotrack used for DNA nanorobots that follow a leader DNA nanorobot

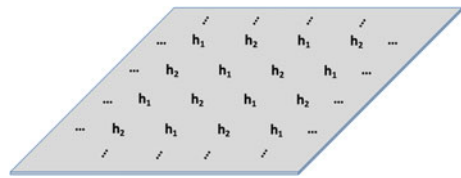


Figure 9 (also see mp4 at [43]) gives an example simulation run for a group of DNA nanorobots that follow a leader DNA nanorobot randomly walk on a 2D nanotrack. The yellow and blue cells are the W_1 (leader) and W_2 (follower) nanorobots, respectively. The W_1 walker can randomly walk on the nanotrack, while the W_2 walker can only follow the W_1 walker or stay in place. If the W_2 walker follows the W_1 walker, the grid it visited turns to green to show the trace of the W_2 Nanorobot; if the W_2 walker does not follow the W_1 walker, it will stay in place and keep blue until the W_1 walker visited W_2 's neighbor again.

4.7 Novel DNA Nanorobots that Vote by Assassination

Distributed voting is essential to many distributed computing and population protocols [51–53], where processors are restricted to pair-wise interactions. For example, distributed voting can be used for leader election which allows a process in a distributed system some special powers in the distributed system, often allowing for

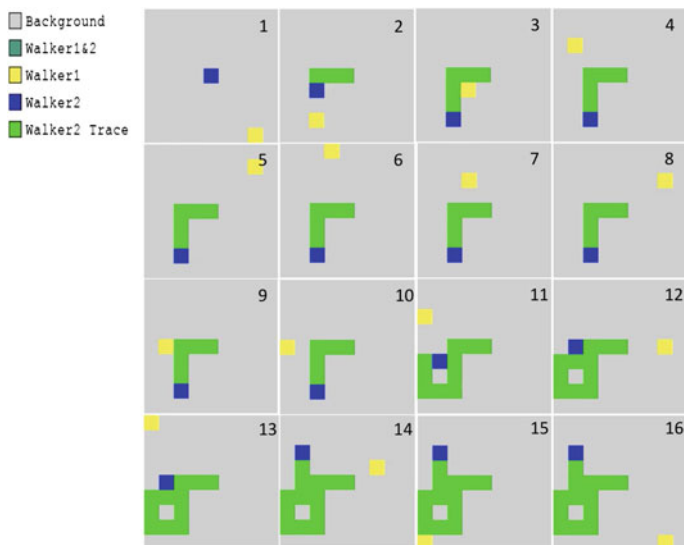


Fig. 9 Simulation of DNA nanorobots that follow a leader DNA nanorobot

simplified protocols, reduced coordination, and improved efficiency. The task of determining *approximate majority* in distributed computing can be reduced to the case where each processor has a binary value in $\{0, 1\}$; assuming an initial margin of disparity $\delta > 1$ between those processors with value 0 and value 1, then the problem is for the set of processors to settle on a majority value in $\{0, 1\}$. A fast randomized distributed protocol for approximate majority was given by Angluin et al. [54, 55]. (Interestingly, Cardelli [56] observed that this approximate majority protocol was used in certain cell cycle switches.) Let an event with size parameter n be *high probability* if it has likelihood $\geq 1 - \frac{1}{n^\alpha}$ for some constant $\alpha \geq 1$. Angluin et al. [54, 55] proved that with high probability, their randomized distributed protocol n processors reached consensus on a majority value after $O(n \log n)$ pair-wise interactions, assuming that the initial margin of disparity is $\geq \delta = c\sqrt{n} \log n$ for a constant $c \geq 1$. A slightly modified version of their protocol proceeds in $O(n \log n)$ stages, where in each stage a random pair of processors compare their values; if their values are the same, they do nothing, and otherwise, a random processor of the pair drops out from subsequent stages of the protocol. Afterward, with high probability, only processors with the same majority value remain. Then the other processors that previously dropped out are informed of that majority value.

Condon et al. [57, 58] have given chemical reaction systems (CRN) for approximate majority, which can in principle be implemented by strand-displacement reactions operating in solution. There has been no previous localized reaction protocol for approximate majority; a localized reaction protocol for approximate majority would likely operate far faster than a solution-based protocol which is delayed by diffusion.

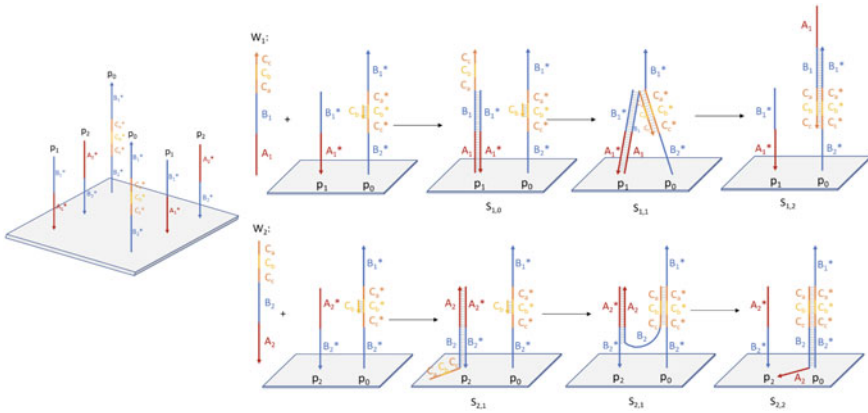
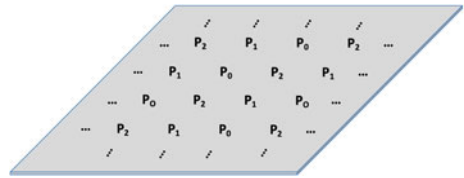


Fig. 10 Separate walks of assassinator nanorobots W_1 and W_2

Fig. 11 2D locations of the assassination protocol's pads on a 2D Nanotrack



We expect distributed voting to be also of central importance to programming complex behavior in social DNA nanorobots. The fast randomized distributed protocol for approximate majority given by Angluin et al. [54, 55] inspired our DNA nanorobots assassination protocol design described here. Our design for distributed voting of DNA nanorobots has the nanorobots exhibit an anti-social behavior to achieve group decision making. The idea is the DNA nanorobots vote by assassination. There are originally two unequal size (with sufficiently large size difference) groups of DNA nanorobots; when pairs of DNA nanorobots from distinct groups collide, one or the other is *assassinated* (it is detached from the 2D nanotrack and then diffused into the solution away from the 2D nanotrack). Eventually, all members of the smaller of the two groups of DNA nanorobots are assassinated with high likelihood.

Let $C = C_a C_b C_c$ where $|C_a| = |C_c|$ are between 3 to 5 bases pairs (sufficient to act as toeholds), and $|C_b| \geq 10$. There are three types of pads: (i) $p_0 = (B_2)^* C^* (B_1)^*$ attached at its 5' end and with the ssDNA sequence C_b hybridized to the complementary subsequence C_b^* of p_0 , (ii) $p_1 = (B_1)^* (A_1)^*$ attached at its 3' end, and (iii) $p_2 = (A_2)^* (B_2)^*$ attached at its 3' end. The 2D arrangement of pads on a 2D nanotrack is illustrated in Fig. 11. There are two types of ssDNA nanorobots: $W_1 = A_1 B_1 C$ and $W_2 = C B_2 A_2$. Figure 10 illustrates the separate walks of assassinator nanorobots W_1 and W_2 (for simplicity it is only illustrated in 1D).

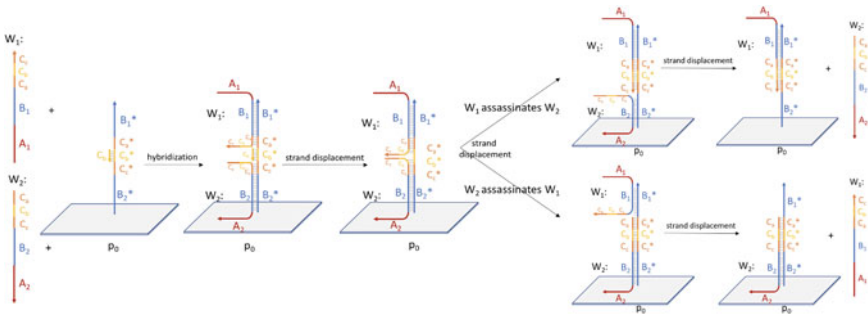


Fig. 12 Design of DNA nanorobots that vote by assassination where $n_1 = 10$ and $n_2 = 6$

There is also a short ssDNA sequence C_b initially hybridized to the complementary subsequence C_b^* of p_0 , which has purpose of substantially increasing the likelihood that W_1 and W_2 will simultaneously bind to a pad p_0 . Once W_1 or W_2 is partly bound to p_0 , it still has to engage in a relatively slow strand-displacement reaction to dislodge the C_b , which was already hybridized to the complementary subsequence C_b^* of p_0 , increasing the likelihood that a second nanorobot also attaches to p_0 . (C_b needs to be in sufficient concentration in the solution, allow it to re-hybridize to C_b^* of p_0 if strand-displaced.)

Initially, a combination of an unequal concentration of W_1 and W_2 strands is added to the buffer solution containing the nanotrack; some W_1 and W_2 strands hybridize to random pads of nanotrack. The buffer solution is replaced, so as to remove the remaining non-hybridized W_1 and W_2 strands from the solution surrounding the nanotrack. Let n_1 and n_2 be the (unknown) numbers of W_1 and W_2 strands initially attached to the nanotrack and let $n = n_1 + n_2$. We assume $n > 0$ and the initial margin of disparity $|n_1 - n_2| \geq \delta$ with $\delta = c\sqrt{n} \log n$ and constant $c \geq 1$. Our goal is: to test if $n_1 > n_2$ or $n_1 < n_2$.

Randomized Assassination Protocol: The nanorobots W_1 and W_2 operate as follows:

- (a) As in Fig. 10, W_1 and W_2 , when separate, walk randomly over the nanotrack, The W_1 nanorobot walks only on the p_0 and p_1 pads of the nanotrack, whereas the W_2 Nanorobot walks only on the p_0 and p_2 pads of the nanotrack.
- (b) As in Fig. 12, whenever both a W_1 and a W_2 nanorobot collide at a common p_0 pad of the nanotrack, a random one of either W_1 or W_2 nanorobot is partially detached via strand-displacement at domain C , and then is melted off to enter the buffer solution. By this process, pairs of W_1 , W_2 duel and randomly one of the nanorobots assassinate the other (which disassociates from the nanotrack).

Note that often only one of W_1 or W_2 will arrive at the p_0 pad of the nanotrack, in which case there will be no competition for hybridization at domain C , and no assassination. However, for the protocol to operate correctly, there just needs to be a constant finite probability that both a W_1 and a W_2 nanorobot collide at a common

p_0 pad of the nanotrack. It is shown below that there is only constant probability of the detached nanorobot W_i reattaching to the 2D nanotrack.

Probabilistic Analysis of Re-attachment Likelihood of a Detached (assassinated) Nanorobot: We assume:

- The protocol is executed over some finite time duration.
- The size of each 2D nanotrack is very small compared to the width of the test tube containing the buffer solution.
- The detached (assassinated) nanorobot W_i takes a random 3D walk within the volume of the test tube after detaching from its 2D nanotrack.
- There is a very small concentration of 2D nanotracks in the buffer solution, so it is very low likelihood $\rho_0 > 0$ that a detached nanorobot W_i later attaches to another 2D nanotrack also in the buffer solution of the test tube during the duration of the protocol.

Consider a sphere S fully containing the 2D nanotrack (and 3 times its diameter) from which a nanorobot W_i is detached from during the assassination protocol. It is easy to see that there is a constant probability $\rho_1 > 0$ that a random 3D walk of W_i takes it outside S . The further random movement of nanorobot W_i can be modeled by a random walk on a 3D grid whose nodes correspond to spheres of the same diameter as S . By Pólya's recurrence theorem [59, 60], in a random walk on a 3D grid starting at a given start node, the likely of never re-visiting that start node is a constant $\rho_2 > 0$. Hence during the duration of the protocol:

- There is at most a constant total likelihood $\leq (1 - \rho_0)\rho_1\rho_2$ that a nanorobot W_i that is detached from a nanotrack during the assassination protocol then is reattached to the same nanotrack or any other nanotrack in the test tube, and so
- There is at least a constant likelihood $\geq 1 - (1 - \rho_0)\rho_1\rho_2$ that the detached nanorobot W_i never re-attaches to any nanotrack.

Probabilistic Analysis of Outcome of Assassinating Nanorobot Protocol: We further assume: whenever both a W_1 and a W_2 nanorobot collide at a common p_0 pad of the nanotrack, it is equally likely that the W_1 or W_2 nanorobot is detached from the nanotrack. From the above, there is at least a constant likelihood that the detached nanorobot never re-attaches to any nanotrack during the duration of the protocol. Observe that as a consequence, ultimately either:

- one or more W_1 remains attached to the nanotrack and all the W_2 are detached, or
- one or more W_2 remains attached to the nanotrack and all the W_1 are detached.

Angluin et al. [54, 55] gave a similar randomized protocol for approximate majority (for distributed computing applications), and their probabilistic analysis implies that if the initial margin of disparity is $|n_1 - n_2| > c\sqrt{n_1 + n_2}$ (for some constant $c > 0$), then ultimately, with high probability:

- if at least one W_1 remains attached to the nanotrack, then $n_1 > n_2$, and
- if at least one W_2 remains attached to the nanotrack, then $n_2 > n_1$.

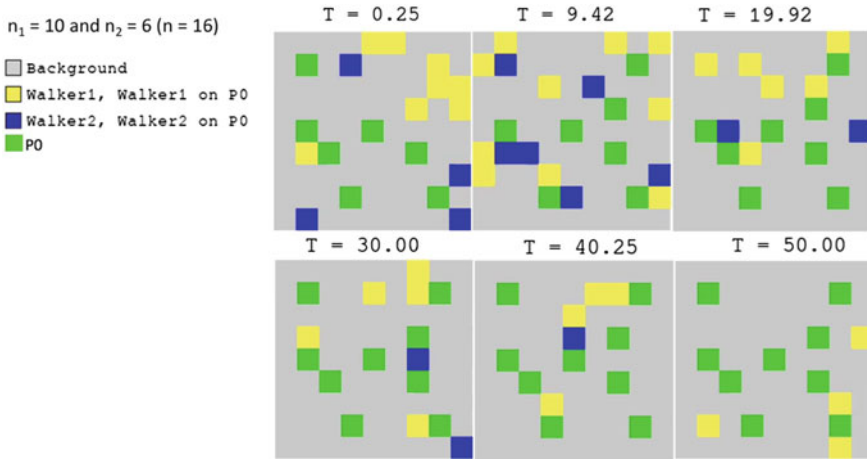


Fig. 13 Simulation of DNA nanorobots that vote by assassination where $n_1 = 10$ and $n_2 = 6$

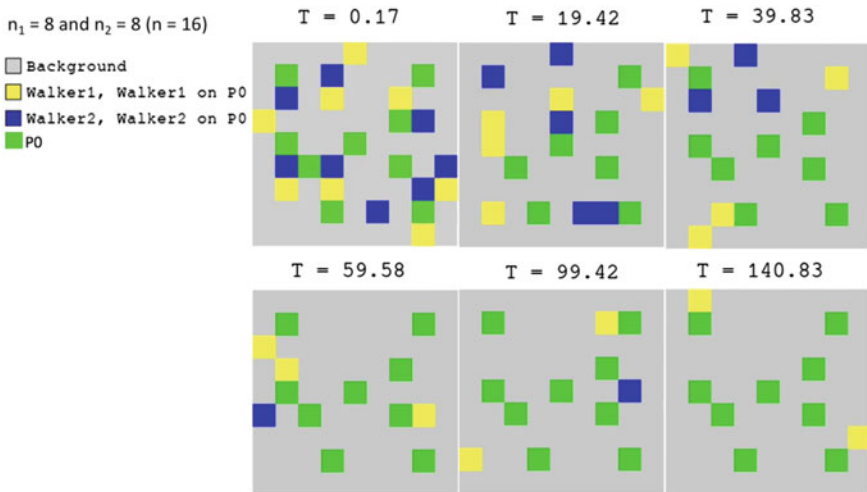


Fig. 14 Simulation of DNA nanorobots that vote by assassination where $n_1 = 8$ and $n_2 = 8$

Simulation of the Assassination Protocol:

- Figure 13 gives an example simulation run for two groups of DNA nanorobots with unequal sizes that vote by assassination on a 2D nanotrack; the number of W_1 is larger than the number of W_2 initially attached to the nanotrack ($n_1 = 10, n_2 = 6$), and eventually all members of W_2 DNA nanorobots are assassinated.
- Figure 14 gives an example simulation run for two groups of DNA nanorobots with equal sizes that vote by assassination on a 2D nanotrack; the numbers of W_1 and W_2 strands initially attached to the nanotrack are equal ($n_1 = n_2 = 8$), so

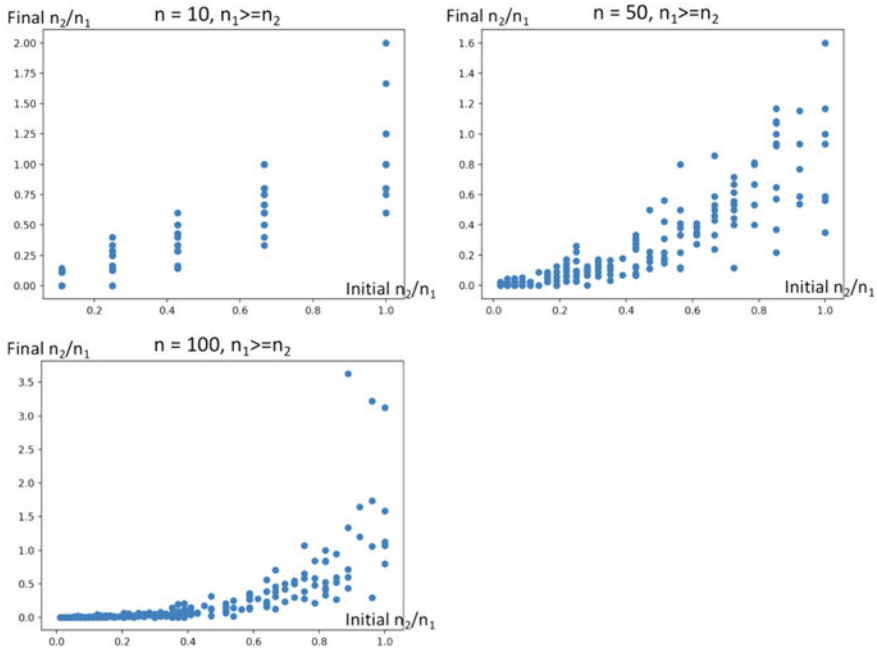


Fig. 15 Results of a collection of simulation runs for DNA Nanorobots voting by assassination on a 2D nanotrack with different initial n , where $n_1 \geq n_2$

they have the same chance to win the game, and in this simulation, eventually all members of W_2 DNA nanorobots are assassinated.

- Figure 15 (also see mp4 at [43]) gives the results of a collection of simulation runs for DNA nanorobots voting by assassination on a 2D nanotrack with different initial n , where the number of W_1 is larger or equal to the number of W_2 ($n_1 \geq n_2$) that are initially attached to the nanotrack. The X-axis and Y-axis represent the initial n_2/n_1 and final n_2/n_1 respectively. Due to $n_1 \geq n_2$ initially, it is more likely that eventually W_1 remains attached to the nanotrack and all the W_2 are detached, and the final n_2/n_1 will convert to 0. With same n , when initial n_2/n_1 goes smaller, the final n_2/n_1 will approach 0 with higher likelihood. When n goes larger, the final n_2/n_1 will approach 0 with higher likelihood.

5 Discussion

We described social DNA nanorobots: these are autonomous mobile DNA nanorobots that execute a series of pair-wise interactions that determine an overall desired outcome behavior for the group of nanorobots. Our goal was to increase the complexity

of the various tasks the nanorobots can execute and at the same time preserve a low design complexity for individual nanorobots. We presented detailed designs for social DNA nanorobots that perform novel behaviors of *self-avoiding walking*, *flocking*, and *voting by assassination*, and their behaviors were simulated in the 2D surface CRN model.

5.1 Further Development of Simulation Software for Social Nanorobots

We are developing software extending the Surface CRN Simulator of Clamons [1, 2] specifically for use with social DNA nanorobots. The software will allow high-level specification and visualizations of state transitions (modeled by chemical reactions such as toehold-mediated strand displacements and dehybridizations) between DNA walkers and DNA strands affixed to a 2D surface. The software could provide an editable catalog of DNA nanorobot devices, improved visualization, and allow automatic incrementally optimized performance assessment. This software should significantly improve performance assessments & design optimizations. We are also using the Visual DSD [61] to simulate the DNA hybridization and strand-displacement reactions of the individual DNA nanorobots and between pairs of DNA nanorobots.

5.2 Experimental Demonstrations of Social DNA Nanorobots

We are planning to experimentally demonstrate the social DNA nanorobots behaviors on 1-dimensional DNA nanotracks and 2-dimensional DNA origami. After experimental demonstrations are made of each design and assessed, they may also be redesigned for further optimization.

5.3 Further Social DNA Nanorobot Behaviors

Our novel designs presented here for DNA nanorobots (self-avoiding walking, flocking, and voting by assassinations) can be employed in designs for even more complex behavior. For example, other behaviors of interest for DNA nanorobots include:

- *Guarding*, where a group of DNA nanorobots follow and guard a particular DNA nanorobot from attack by another group of DNA nanorobots. Here we expect we can employ parts of our flocking design.
- *Attacking*, where a group of DNA nanorobots attack another group of DNA nanorobots. Here we expect we can employ a simplification of the assassination design.

- *Foraging and Harvesting.* In *foraging*, a group of designated *foraging* DNA nanorobots randomly walk on the 2D nanotrack, and can transform to a “discovery state” when they discover a target molecule (e.g., a group of gold nanoparticles attached to 2D surface) (this makes use of our designs for self-avoiding random walking). In *harvesting*, a group of *harvesting* DNA nanorobots which follow the trail of foraging DNA nanorobots in discovery state and pick up the detected target molecules and deliver the target molecules to a designated region of the 2D nanotrack. Designs for *foraging* nanorobots may employ our designs for self-avoiding random walking, and designs for *harvesting* nanorobots may employ our design for flocking.

5.4 Communication Between Distant Social Nanorobots

Prior Use of Potential Fields for Generating Autonomous Group Social Activities: Another source of inspiration for collective behavior strategies by groups can be found in biology: for example, flocking of animals such as birds and schooling of amphibious animals. The behavior of these animals has been modeled by mathematical models and computer programs. In 1989, Beni [62, 63] developed one of the first such flocking model, which he called *swarm intelligence* and made applications to multi-robot motion planning systems. Subsequently the field of swarm intelligence [64, 65] and artificial flocking grew rapidly and found applications to many applied areas in addition to robotics, such as for computer graphics. In 1994, Reif [66, 67] developed a general programmable scheme for multi-robot motion planning, termed *Social Potential Fields*, which made use of artificially defined potential fields that controlled the individual robots by weighted sum of decreasing functions of the distance and direction of other local robots; he demonstrated various autonomous group social activities, including flocking, attacking, and guarding, using the Social Potential Fields technique. Unfortunately, the potential field models assume far-distance field effects that are not easy to implement using local interactions between co-located DNA nanorobots.

Using Instead Diffusion of Pheromone-like DNA Molecules for Communication Between Social Nanorobots: Recall that the communication signals between social insects include pheromones; these are chemical factors that can trigger a social response in members of the same species [10]. We are exploring the use of diffusion of DNA molecules for communication between social nanorobots in a manner similar to the use of pheromones in social insects. For example, this technique may be employed by foraging and harvesting nanorobots. Suppose the goal is to detect and harvest a particular target molecule on the 2D surface on which the foraging and harvesting nanorobots walk on. The 2D surface would be decorated with additional hairpins that when opened by a foraging nanorobot (to announce the discovery of a nearby target molecule), would act as pheromones for the foraging nanorobots (e.g., the speed of the motion of nearby harvesting nanorobots could temporarily increased when encountering such an opened hairpin).

Acknowledgements This work was supported by NSF Grants CCF-2113941 and CCF-1909848. Many thanks for editorial suggestions by the referees, as well as from Dan Fu and Xin Song.

References

1. L. Qian and E. Winfree, Parallel and scalable computation and spatial dynamics with dna-based chemical reaction networks on a surface, in *International Workshop on DNA-Based Computers*, pp. 114–131 (Springer, 2014)
2. S. Clamons, L. Qian, E. Winfree, Programming and simulating chemical reaction networks on a surface. *J. R. Soc. Interface* **17**(166), 20190790 (2020)
3. E. O. Wilson, in *Sociobiology: The New Synthesis*, 1st edn (Harvard University Press, Cambridge, MA, 1975)
4. S. Batra, Behavior of some social and solitary halictine bees within their nests: a comparative study (hymenoptera: Halictidae). *J. Kansas Entomol. Soc.*, pp. 120–133 (1968)
5. C.D. Michener, Comparative social behavior of bees. *Annu. Rev. Entomol.* **14**(1), 299–342 (1969)
6. E.O. Wilson, *The Insect Societies* (Harvard University Press, Cambridge, Massachusetts, USA, 1971)
7. B. Crespi, D. Yanega, The definition of eusociality. *Behav. Ecol.* **6**(1), 109–115 (1995)
8. E. Wilson, B. Hölldobler, Eusociality: origin and consequences. *Proc. Natl. Acad. Sci.* **102**(38), 13367–13371 (2005)
9. B. Hölldobler, E. Wilson, *The Ants* (Harvard University Press, 1990)
10. P. Karlson, M. Lüscher, Pheromones: a new term for a class of biologically active substances. *Nature* **183**(4653), 55–56 (1959)
11. K. von Frisch, *The Dance Language and Orientation of Bees*, 1st edn. (The Belknap Press of Harvard University Press, 1967)
12. T. Seeley, P. Visscher, K. Passino, Group decision making in honey bee swarms: when 10,000 bees go house hunting, how do they cooperatively choose their new nesting site? *Am. Sci.* **94**(3), 220–229 (2006)
13. M. Girard, H. Mattila, T. Seeley, Recruitment-dance signals draw larger audiences when honey bee colonies have multiple patrines. *Insectes Sociaux* **58**(1), 77–86 (2011)
14. T. Seeley, S. Kühnholz, A. Weidenmüller, The honey bee's tremble dance stimulates additional bees to function as nectar receivers. *Behavioral Ecol. Sociobiol.* **39**(6), 419–427 (1996)
15. T.D. Seeley, *Honeybee Democracy*, 1st edn. (Princeton University Press, Princeton, NJ, 2010)
16. B. Yurke, A. Turberfield, A. Mills, F. Simme, J. Neumann, A DNA-fuelled molecular machine made of DNA. *Nature* **406**, 605–608 (2000)
17. Y. Chen, M. Wang, C. Mao, An autonomous DNA nanomotor powered by a DNA enzyme. *Angew. Chem. Int. Ed.* **43**(27), 3554–3557 (2004)
18. J. Bath, S. Green, A. Turberfield, A Free-Running Motor DNA Powered by a nicking enzyme. *Angew. Chem. Int. Ed.* **44**(28), 4358–4361 (2005)
19. Y. Tian, C. Mao, Molecular gears: a pair of DNA circles continuously rolls against each other. *J. Am. Chem. Soc.* **126**(37), 11410–11411 (2004)
20. J. Bishop, E. Klavins, An improved autonomous DNA nanomotor. *Nano Lett.* **7**(9), 2574–2577 (2007)
21. J. Reif, The design of autonomous DNA nanomechanical devices: walking and rolling DNA, in *Lecture Notes in Computer Science book series (LNCS): DNA Based Computers (DNA8)*, Sapporo, Japan, ed. by M. Hagiya, A. Ohuchi, no. 3384, pp. 22–37 (Springer, New York, 2002)
22. W. Sherman, N. Seeman, A precisely controlled DNA biped walking device. *Nano Lett.* **4**, 1203–1207 (2004)

23. J.S. Shin, N.A. Pierce, A synthetic DNA walker for molecular transport. *J. Am. Chem. Soc.* **126**(35), 10834–10835 (2004)
24. P. Yin, H. Choi, C. Calvert, N. Pierce, Programming biomolecular self-assembly pathways. *Nature* **451**(7176), 318–322 (2008)
25. P. Yin, H. Yan, X. Daniel, A. Turberfield, J. Reif, A unidirectional dna walker moving autonomously along a linear track. *Angew. Chem. Int. Ed.* **43**(37), 4906–4911 (2004)
26. P. Yin, A. Turberfield, S. Sahu, J. Reif, Designs for autonomous unidirectional walking DNA devices, in *Lecture Notes in Computer Science book series (LNCS): Tenth International Meeting on DNA Based Computers (DNA10), Milano, Italy*, ed. by C. Ferretti, G. Mauri, C. Zandron, no. 3384, pp. 410–425 (Springer, New York, 2004)
27. Y. Tian, Y. He, Y. Chen, P. Yin, C. Mao, A DNAzyme that walks processively and autonomously along a one-dimensional track. *Angew. Chem. Int. Ed.* **44**(28), 4355–4358 (2005)
28. S. Venkataraman, R. Dirks, P.W. Rothmund, E. Winfree, N. Pierce, An autonomous polymerization motor powered by dna hybridization. *Nat. Nanotechnol* **2**(8), 490–494 (2007)
29. S. Sahu, T. LaBean, J. Reif, A DNA nanotransport device powered by polymerase phi29. *Nano Lett.* **8**(11), 3870–3878 (2008)
30. S. Green, J. Bath, A. Turberfield, Coordinated chemomechanical cycles: a mechanism for autonomous molecular motion. *Phys. Rev. Lett.* **101**(23), 238101 (2008)
31. F. Simmel, Processive motion of bipedal DNA walkers. *Chem. Phys. Chem.* **10**(15), 2593–2597 (2009)
32. R. Pei, S. Taylor, D. Stefanovic, S. Rudchenko, T. Mitchell, M.N. Stojanovic, Behavior of polycatalytic assemblies in a substrate-displaying matrix. *J. Am. Chem. Soc.* **128**(39), 12693–12699 (2006)
33. K. Lund, A. Manzo, N. Dabby, N. Michelotti, A. Johnson-Buck, J. Nangreave, S. Taylor, R. Pei, M. Stojanovic, N. Walter, E. Winfree, H. Yan, Molecular robots guided by prescriptive landscapes. *Nature* **465**(7295), 206–210 (2010)
34. S.F. Wickham, J. Bath, Y. Katsuda, M. Endo, K. Hidaka, H. Sugiyama, A.J. Turberfield, A dna-based molecular motor that can navigate a network of tracks. *Nat. Nanotechnol.* **7**(3), 169–173 (2012)
35. J. Chao, J. Wang, F. Wang, X. Ouyang, E. Kopperger, H. Liu, Q. Li, J. Shi, L. Wang, J. Hu, L. Wang, W. Huang, F.C. Simmel, C. Fa, Solving mazes with single-molecule dna navigators. *Nat. Mater.* **18**(3), 273–279 (2019)
36. H. Chandran, N. Gopalkrishnan, J. Reif, DNA nanoRobotics, in *Nanorobotics: Current Approaches and Techniques*, ed. by C. Mavroidis, A. Ferreira, pp. 355–382 (Springer, New York, NY, 2013)
37. D. Leigh, Synthetic DNA walkers. *Top. Curr. Chem.* **354**, 111–138 (2014)
38. J. Pan, Recent progress on DNA based walkers. *Curr. Opin. Biotechnol.* **34**, 56–64 (2015)
39. P. Yin, A. Turberfield, S. Sahu, J. Reif, Design of an autonomous DNA nanomechanical device capable of universal computation and universal translational motion, in *Lecture Notes in Computer Science book series (LNCS): Tenth International Meeting on DNA Based Computers (DNA10), Milano, Italy*, ed. by C. Ferretti, G. Mauri, C. Zandron, no. 3384, pp. 426–444 (Springer, New York, 2004)
40. J. Reif, S. Sahu, Autonomous programmable DNA nanorobotic devices using dnazymes. *Theoret. Comput. Sci.* **410**, 1428–1439 (2009)
41. H. Gu, J. Chao, S. Xiao, N.C. Seeman, A proximity-based programmable DNA nanoscale assembly line. *Nature* **465**, 202–205 (2010)
42. A. Thubagere, A. Thubagere, W. Li, R. Johnson, Z. Chen, S. Doroudi, Y. Lee, G. Izatt, S. Wittman, N. Srinivas, D. Woods, E. Winfree, Q. Qian, A cargo-sorting DNA robot. *Science* **357**(6356) (2017)
43. M. Yang, J.H. Reif, *Simulations of Social DNA Nanorobots*, in mp4 format. <https://users.cs.duke.edu/~reif/paper/SocialDNAAnanorobots/Simulations>
44. H. Bui, T. Song, J.H. Reif, Localized dna computation, in *Invited Chapter, From Parallel to Emergent Computing*, ed. by A. Adamatzky (CRC Press, 2019)

45. R. Dirks, N. Pierce, Triggered amplification by hybridization chain reaction. *Proc. Natl. Acad. Sci.* **101**, 1527–15278 (2004)
46. H. Chandran, N. Gopalkrishnan, A. Phillips, J.H. Reif, Localized hybridization circuits, in *DNA Computing and Molecular Programming*, pp. 64–83 (2011)
47. N. Dalchau, H. Chandran, N. Gopalkrishnan, A. Phillips, J.H. Reif, Probabilistic analysis of localized DNA hybridization circuits. *ACS Synth. Biol.* **4**(8), 898–913 (2015)
48. H. Bui, V. Miao, S. Garg, R. Mokhtar, T. Song, J.H. Reif, Design and analysis of localized DNA hybridization chain reactions. *Small* **13**(12) (2017)
49. H. Bui, S. Shah, R. Mokhtar, T. Song, S. Garg, J.H. Reif, Localized dna hybridization chain reactions on dna origami. *ACS Nano* (2018)
50. S. Hemmer, P. Hemmer, An average self-avoiding random walk on the square lattice lasts 71 steps. *J. Chem. Phys.* **81**(1), 584–585 (1984)
51. N. Lynch, *Distributed algorithms*, 1st edn. (Morgan Kaufmann, San Francisco, CA, 1996)
52. A. Kshemkalyani, M. Singhal, *Distributed Computing: Principles, Algorithms, and Systems*, 1st edn. (Cambridge University Press, Cambridge, UK, 2011)
53. J. Aspnes, E. Ruppert, An introduction to population protocols, in *Middleware for Network Eccentric and Mobile Applications*, ed. by B. Garbinato, H. Miranda, L. Rodrigues, pp. 97–120 (Springer, 2009)
54. D. Angluin, J. Aspnes, D. Eisenstat, A simple population protocol for fast robust approximate majority, in *Proceedings of the 21st International Conference on Distributed Computing (DISC07)*, pp. 20–32 (ACM, 2007)
55. D. Angluin, J. Aspnes, D. Eisenstat, A simple population protocol for fast robust approximate majority. *Distrib. Comput.* **21**, 87–102 (2008). (July)
56. L. Cardelli, A. Csikász-Nagy, The cell cycle switch computes approximate majority. *Sci. Rep.* **2**(1), 1–9 (2012)
57. A. Condon, M. Hajiaghayi, D. Kirkpatrick, J. Mañuch, Simplifying analyses of chemical reaction networks for approximate majority, in *23rd International Conference on DNA Computing and Molecular Programming (DNA23), Lecture Notes in Computer Science*, ed. by R. Brijder, L. Qia, vol. 10467, pp. 189–209 (Springer, 2017)
58. A. Condon, M. Hajiaghayi, D.K. Kirkpatrick, J. Mañuch, Approximate majority analyses using tri-molecular chemical reaction networks. *Nat. Comput.* **19**, 249–270 (2019)
59. G. Pólya, Über eine aufgabe betreffend die irrfahrt im strassennetz. *Math. Ann.* **84**, 149–160 (1921)
60. M. Barber, B. Ninham, *Random and Restricted Walks: Theory and Applications* (Gordon and Breach Science Publishers, 1970)
61. M. Lakin, S. Youssef, F. Polo, S. Emmott, A. Phillips, Visual DSD: a design and analysis tool for DNA strand displacement systems. *Bioinformatics* **27**(22), 3211–3213 (2011)
62. G. Beni, J. Wan, Swarm intelligence in cellular robotic systems, in *Proceedings of NATO Advanced Workshop on Robots and Biological Systems*, pp. 26–30 (NATO, Tuscany, 1989)
63. G. Beni, Distributed robotic system and swarm intelligence. *J. Robot Soc. Japan.* **10**(4), 457–463 (1992)
64. E. Bonabeau, M. Dorigo, G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, 1st edn. (Oxford University Press, New York, 1999)
65. J. Kennedy, R. Eberhart, Y. Shi, *Swarm Intelligence*, 1st edn. (Morgan Kaufman, San Mateo, CA, 2001)
66. J. Reif, H. Wang, Social potential fields: a distributed behavioral control for autonomous robots, in *Workshop on Algorithmic Foundations of Robotics (WAFR'94), San Francisco* (ACM, California, 1994)
67. J. Reif, H. Wang, Social potential fields: a distributed behavioral control for autonomous robots. *Robot. Auton. Syst.* **27**(3), 171–194 (1999)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Models of Gellular Automata



Masami Hagiya and Taiga Hongu

Abstract We summarize our work on gellular automata, which are cellular automata we intend to implement with gel materials. If cellular automata are implemented as materials, it will become possible to realize smart materials with abilities such as self-organization, pattern formation, and self-repair. Furthermore, it may be possible to make a material that can detect the environment and adapt to it. In this article, we present three models of gellular automata, among which the first two have been proposed previously and the third one is proposed here for the first time. Before presenting the models, we briefly discuss why cellular automata are a research target in DNA computing, a field which aims to extract computational power from DNA molecules. Then, we briefly describe the first model. It is based on gel walls with holes that can open and exchange the solutions that surround them. The second model is also based on gel walls but differs in that the walls allow small molecules to diffuse. In presenting the second model, we focus on self-stability, which is an important property of distributed systems, related to the ability to self-repair. Finally, we report our recent attempt, in the third model, to design gellular automata that learn Boolean circuits from input–output sets, i.e., examples of input signals and their expected output signals.

1 Introduction: Why Cellular Automata?

DNA computing is a research field that focuses on extracting computational power from DNA molecules and their reactions. Implementing a Turing machine using DNA is a typical example of research in this field. Research to realize cellular automata using molecules such as DNA has also been actively conducted. Why are cellular automata a target of research in DNA computing?

M. Hagiya (✉) · T. Hongu
Institute for AI and Beyond, University of Tokyo, Tokyo, Japan
e-mail: hagiya@g.ecc.u-tokyo.ac.jp

1.1 *Computation by Molecules*

The first attempt to perform computations using molecules was a thought experiment in which molecules were used to make Turing machines. Research on molecular Turing machines began with Bennett's concept of a biological computer [1]. Since then, proposals to realize Turing machines and various kinds of automata by molecules have been repeated, including the recent work on Turing machines by King et al. [2]. The work by Benenson et al. is a typical example of implementing finite automata [3].

However, the computational power of a single molecule is limited because it is not easy for different parts of a molecule to interact and exchange information. Therefore, the possibility of computing with a chemical solution containing a huge number of molecules has been explored. Adleman's research on data parallel computation by molecules sits in this context [4].

Since then, in DNA computing, research on computation by a chemical solution has progressed from logic circuits to neural networks. For example, some studies implemented logic circuits and neural networks using seesaw gates [5, 6].

In parallel with such efforts, the concept of a chemical reaction network (CRN) has been widely adopted as a unified standard model for computation using a chemical solution, and theoretical research on computation by CRN is being conducted [7].

However, the computational power of a chemical solution that is uniform in space is also limited. Information should be encoded as concentrations of molecular species. Therefore, many researchers in the field became interested in cellular automata, a computational model that divides space into cells. Researchers began implementing cellular automata using molecules. Historically, the self-assembly of DNA tiles, pioneered by Seeman and Winfree, was modeled using cellular automata [8]. In their model, the position where each tile is placed is considered a cell. In the process of self-assembly, each cell transitions from an empty state to a state in which it is filled with a tile.

1.2 *Smart Materials*

Researchers in DNA computing have continually asked this question: What are the possible applications of the computational power extracted from molecules? Among the various proposals, smart materials, i.e., those that exhibit intelligent behaviors, are considered promising [9].

If cellular automata are implemented as materials, it will become possible to realize materials with abilities such as self-organization, pattern formation, and self-repair. For example, suppose such a smart material is used to implement an artificial blood vessel. The vessel would form autonomously at an appropriate place in the

living body, and it would recognize any deformation or occlusion of itself and self-repair. Note that cellular automata capable of simple self-assembly, such as those consisting of DNA tiles, are not sufficiently complex to make such smart materials.

With more sophisticated cellular automata, it is possible to envision smart materials that learn appropriate functions from external stimuli. For example, as we report later in this article, we have designed cellular automata that form a logic circuit from input–output sets in a three-dimensional (3D) cellular space.

1.3 Why Discrete?

When envisioning smart materials, such as the ones above, one may wonder why these should be discrete. It might be possible to realize the same functions in a continuous medium, for example, by taking advantage of Turing patterns and developing design principles for continuous media [10–12]. However, using current methods, the rational design of continuous media is challenging. Tuning a huge number of continuous parameters to produce target shapes or behaviors is a nontrivial task in general because these parameters are intricately intertwined.

By contrast, we can take advantage of a sizeable accumulation of research on cellular automata. Therefore, we consider it an appropriate approach to design a discrete model while implementing the model in a continuous medium, such as a reaction–diffusion system. In addition, as we shall see in the next section, implementation at the molecular level is naturally discrete, so this has a high affinity with cellular automata.

2 Implementation of Cellular Automata

Methods to implement cellular automata have been actively investigated in DNA computing, as we shall see below, but no optimal method has been established. Proposed methods can be classified into two groups: those at the molecular level and those using reaction–diffusion systems.

2.1 Molecular Level

Yin, Sahu, Turberfield, and Reif proposed a molecular-level implementation [13]. Each cell is represented by a strand of DNA, doubled in part, which is attached to a 1D track. Each partially double strand has a single-stranded portion that represents its state and is changed by a restriction enzyme. The entire reaction system is complex, and therefore, its implementation is very challenging.

Qian and Winfree proposed CRN on a surface [14]. Again, each cell is represented by a partially double strand of DNA with a single-stranded portion. It makes a state transition by replacing its single strand with a new one. Two neighboring cells make state transitions at the same time. That is, by the reaction $A + B \& \rightarrow C + D$, a cell in state A transitions to state C, and a neighboring cell in state B transitions to state D. The reaction $A + B \& \rightarrow C + D$ is a basic form of CRN called a population protocol. In other words, Qian and Winfree proposed realizing population protocols in a 2D cellular space, i.e., on a surface. Note that population protocols have been investigated as a standard model of distributed computing, and various distributed algorithms can be represented in this form.

For implementing population protocols on a surface, Qian and Winfree proposed a very sophisticated system consisting of many reactions, called a strand displacement system, but it is not known whether its implementation by DNA is feasible. Despite the apparent difficulty of its implementation, theoretical research on CRNs on a surface is progressing [15].

A new design for molecular-level implementation by data parallel computation based on strand displacement was recently proposed with a simulation of Rule 110 elementary cellular automata [16].

The self-assembly of DNA tiles, mentioned above, realizes a restricted class of cellular automata in which each cell can make a transition only once. This restriction is not appropriate for smart materials that are expected to repair themselves after external disturbances, although 2D self-assembly can simulate the spatiotemporal evolution of 1D cellular automata [17–19].

Compared to reaction–diffusion systems, molecular-level proposals of cellular automata have advantages including their size and the speed of state transitions. However, all proposals other than those for self-assembly are based on complex reactions of DNA and are inherently difficult to implement. In addition, the size of molecular-level implementations can be a disadvantage for expressing the macroscopic behaviors needed in smart materials.

2.2 *Reaction–Diffusion Systems*

Proposals have also been made to implement cellular automata using reaction–diffusion systems, as discussed above. For example, in the proposal by Scalise and Schulman, a cell in cellular automata is implemented by a region with a high concentration of a molecule that is attached to the region and cannot diffuse [20]. Each cell produces a molecule corresponding to its current state, and this molecule diffuses to the neighboring cells; in this paper, we call this a signal molecule.

In the molecular robotics project led by Hagiya, Murata et al. actively attempted to implement cellular automata by a reaction–diffusion system in a gel [21, 22]. Cells were implemented with solutions separated by gel walls. Signal molecules can diffuse through gel walls while other molecules stay within each cell’s solution [23].

In this project, Hagiya started to use the phrase “gellular automata” for cellular automata that are suitable for implementation with gels. Based on the implementation mentioned above, the second model of gellular automata was proposed, as explained in the next section.

However, it is very challenging to implement reaction–diffusion systems in gel materials so that cellular automata work as expected. One serious problem is that the diffusion of signal molecules is not limited to the immediate neighbors of the cell that generated the signal but instead may diffuse to a neighbor of a neighbor. To overcome this, we should decompose signals at an appropriate rate to prevent them from diffusing beyond the immediate neighbors. Consequently, we should constantly supply energy and ingredients from an external environment.

Further, if we allow an external environment to actively interact with a reaction–diffusion system, we can control diffusion via a global effect such as an electric or magnetic field or chemical gradient. Then, we can enhance models of cellular automata using such effects, as in our third model of gellular automata, proposed below, where neighboring cells can be distinguished by their directions.

3 Gellular Automata

In this article, the phrase “gellular automata” describes several models of cellular automata that we have proposed and investigated and which we intend to implement with gel materials. In this section, we briefly introduce three such models. The first is based on gel walls with holes that can open and exchange neighboring solutions. It is somewhat rudimentary in that precise control of the interactions between cells is difficult. The second model is also based on gel walls but differs in that the walls allow small molecules to diffuse. In describing the second model, we focus on the self-stability of the model. Note that this is an important property of distributed systems that is crucial for realizing smart materials. In the subsequent section, we report our ongoing work on the third model, which learns Boolean circuits from input–output sets, i.e., examples of input signals and their expected output signals.

3.1 *Gellular Automata with Holes*

The first model we investigated is based on gel walls separating cells of solutions. The walls are assumed to have holes that are opened and closed by the solutions that surround them [24–27].

This model is both continuous and discrete in the sense that, while concentrations of molecules are continuous with respect to time, a hole is either open or closed depending on a continuous parameter of the hole. The parameter is controlled by molecules called composers and decomposers (Fig. 1).

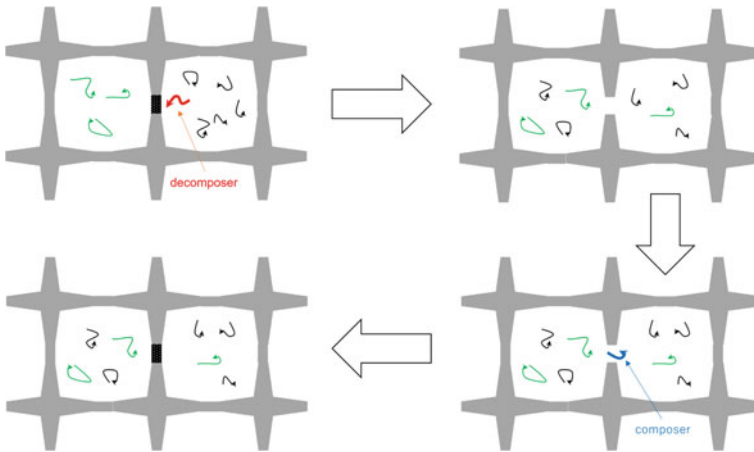


Fig. 1 Gel walls separate solutions. One wall has a hole. A molecule called a decomposer changes a parameter of the hole, and the hole eventually opens (top left to top right). When a hole is open, adjacent cells are connected, and their solutions mix. A molecule called a composer then changes the parameter in the opposite direction and the hole eventually closes (bottom right to bottom left). It is assumed that the composer is generated in the mixed solution (top right)

We demonstrated the Turing computability of the model by encoding rotary elements in it [24]. Note that it is possible to implement any reversible Turing machine using only rotary elements [28].

We conducted some preliminary experiments using polyacrylamide gels with DNA bridges. We implemented holes that could open once or close once, but we could not implement a hole that could open and close repeatedly. Since then, various kinds of DNA gels have been developed. It will be a challenge to develop DNA gels that can shrink and swell repeatedly and thus be used as valves for opening and closing holes.

3.2 Boolean Total and Non-Camouflage Gellular Automata

The second model is much simpler than the first and more suited to implementation with gels. In this model, gel walls also separate cells of solutions, but communication between them is realized by the diffusion of signal molecules, as implemented by Abe et al. [23].

With such an implementation in mind, we have defined this model as a limited class of asynchronous cellular automata [29, 30]. Each cell undergoes a state transition only if there are neighboring cells in specified states (Boolean totality). However, each cell cannot recognize a cell in the same state as itself (non-camouflage). A transition rule can be represented in the following form:

$$S \rightarrow T (P \text{ and } Q \text{ and not } R \text{ and } \dots).$$

which means that a cell in state S will transition to state T if there is a neighboring cell in state P and a neighbor in state Q but no neighbor in state R ; due to being non-camouflaged, S must be different from P , Q , and R .

In addition, state transitions are asynchronous in the sense that each cell may or may not make a transition at each instance of discrete time.

Note that the above restrictions occur naturally in implementations of cells by solutions separated by gel walls. Each cell is assumed to transmit a signal corresponding to its current state, and therefore, a cell cannot see a neighbor in the same state. Due to Boolean totality, it is not necessary to control the direction of diffusion or consider the number of neighboring cells transmitting the same signal.

This second model of gellular automata also has sufficient computational powers. Turing computability was demonstrated [29, 30] as was the fact that the model can simulate population protocols [31]. This means that the distributed algorithms that population protocols can express can also be realized in the model.

Self-stability is an important property of distributed algorithms; it is essential for smart materials. If a material is implemented by gellular automata that realize a self-stable distributed algorithm, it will eventually reach a desired configuration from any initial configuration. Here, a configuration means a global state of the gellular automata, i.e., a mapping of cells to states. Even if an external disturbance damages the material, i.e., changes the states of some of its cells, it can self-repair from the damaged configuration. Therefore, we tried to show the self-stability of various algorithms realized in the model.

Self-stability consists of safety and reachability. Reachability means that it is possible to reach a desired configuration from any initial configuration; this is usually demonstrated under the assumption of fair scheduling. Safety means staying in a desired configuration forever once it has been reached.

We took the following approach to demonstrating self-stability. First, the conditions for desired configurations are expressed by the local conditions of each cell. Next, we show that state transitions of cells in a desired configuration preserve the local conditions; this derives safety. Furthermore, we showed that, with fairness, a desired configuration that satisfies the local conditions is always achieved; this derives reachability. Taking the above approach, we have proved the self-stability of the gellular automata for solving mazes (Fig. 2), two-distance coloring, and the formation of spanning trees [32–34].

We are currently working on 3D gellular automata. In our first attempt, we formulated an algorithm for solving maze problems in 3D. Then, we extended it to solve matching problems in which paths are formed between pairs of cells in 3D (Fig. 3). It is hoped that such paths will be able to transport substances or transmit information in smart materials. Our next target is to realize more intelligent behaviors by extending paths to circuits. This target led us to introduce our third model of gellular automata.

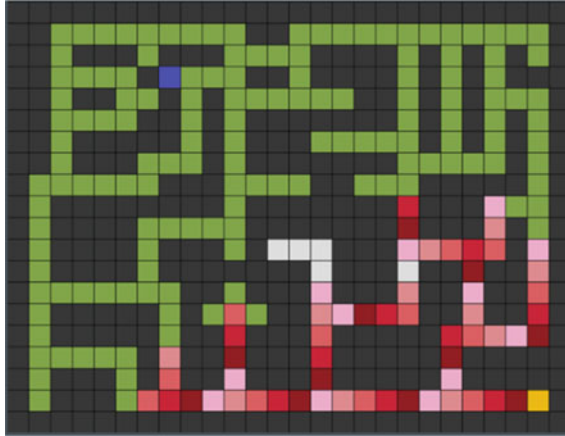


Fig. 2 Self-stable gellular automata solve a maze problem. The orange cell (bottom right) is the start, and the blue cell is the goal. Black cells are walls. First, green cells spread from the goal. When they reach the start, red cells with different levels spread from it. Eventually, a single path consisting of red cells forms from the start to the goal. External disturbances can change the states of some cells. For example, some non-black cells may be changed to black ones. Provided that there is one start and one goal, a single path from the start to the goal forms again

3.3 *Three-Dimensional Gellular Automata That Learn Boolean Circuits*

Recently, we formulated a 3D model of cellular automata that forms Boolean circuits from input–output sets. In this problem, Boolean input signals (0 or 1) are given to some cells (input nodes) and their states are changed. Expected Boolean output signals (0 or 1) are also given to some cells (output nodes). This forms one input–output set. More input–output sets are given to the model successively and repeatedly. For each set, a circuit gradually forms that produces the expected output signals from the input signals. In this way, supervised learning is realized as a kind of pattern formation. If smart materials gain such an ability, they can learn by signals from their external environment.

A cell is placed at each lattice point in a 3D space. Each cell therefore has six neighbors in its von Neumann neighborhood. Some cells are specified as input or output nodes. Other cells are either active or inactive; if a cell is active, it works as an OR gate. Signals are fed to input nodes and transmitted from these to output nodes along the OR gates in the direction of increasing coordinates. This means that from a cell at (x, y, z) , a signal can transmit to the cells at $(x + 1, y, z)$, $(x, y + 1, z)$, and $(x, y, z + 1)$. For example, if OR gates are placed at (x, y, z) and $(x + 1, y, z)$, then a signal transmits from the former to the latter. Note that this directionality prohibits loops of signals and makes the construction of circuits tractable.

However, in this way, we have abandoned Boolean totality in this model: The signals are transmitted from input nodes only in certain directions. Thus, each cell

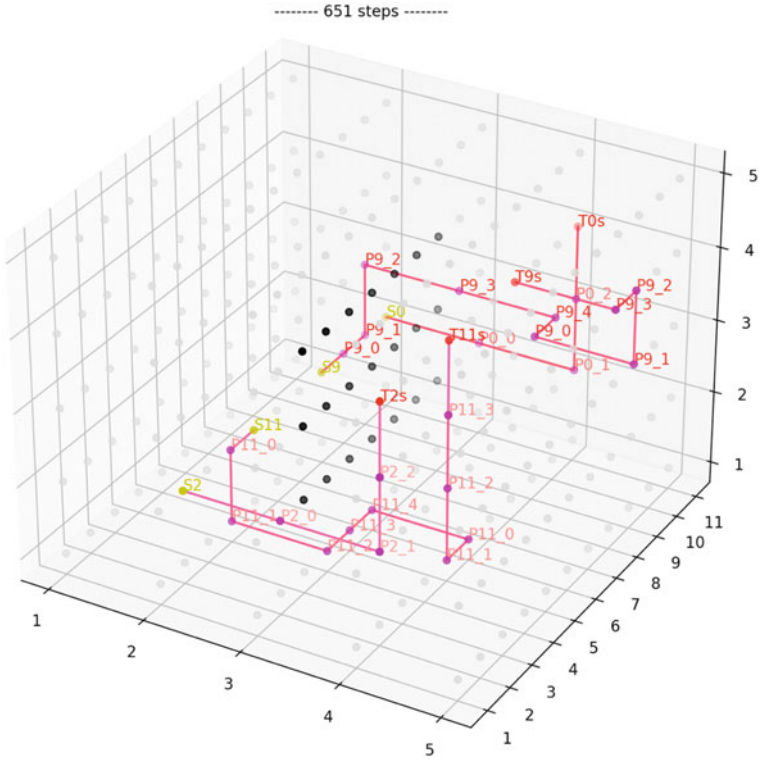


Fig. 3 Gellular automata for solving maze problems have been extended to solve matching problems in three dimensions. Paths are formed between S_i and T_i

is assumed to recognize each neighbor. To implement this model with gel materials, we must provide a global effect, such as that from an electric field, to provide directionality, as mentioned above.

4 Supervised Learning of Boolean Circuits

Under our third model of gellular automata, we are currently developing an algorithm for constructing Boolean circuits consisting of OR gates.

4.1 Assumption

With given Boolean signals at the input nodes, the expected Boolean signals at the output nodes are specified as teacher signals. In other words, an example of supervised

learning consists of given signals at the input nodes and expected signals at the output nodes. With each set of these, cells make state transitions and form a Boolean circuit consisting of OR gates between the input and output nodes.

For a cell at (x, y, z) , the cells at $(x-1, y, z)$, $(x, y-1, z)$, and $(x, y, z-1)$ are called its “fan-in” cells, and the cells at $(x+1, y, z)$, $(x, y+1, z)$, and $(x, y, z+1)$ are called its “fan-out” cells.

4.2 States

The state of each cell consists of the following components. The type can be *input*, *output*, *OR*, or *none*. Input and output nodes are cells of those respective types. Cells have type *OR* if they are active, working as OR gates in a circuit. They have type *none* if they are inactive and do not belong to a circuit.

Each cell has a Boolean value (0 or 1). This component is called the value component or value and is determined by signals transmitted from input nodes via the OR nodes. The value component of an input node is set by the Boolean signal given to it. If the type of a cell is *OR* or *output* and it has a fan-in cell whose value is 1, then its value component is set to 1. If there is no such fan-in cell, it is set to 0.

In addition, the state of each cell has a component containing information for fixing the type of the cell. This component is called the fix component and can be one of *FIX0*, *FIX1*, and *none*. *FIX0* means that the cell has the value component 1, but it should be 0. *FIX1* means the other way around. In the case of *FIX1*, the component further specifies one of the fan-in cells of the cell as described in the algorithm below.

In summary, a state of a cell is a tuple in one of the forms $(input, v)$, (t, v, f) , and $(t, v, FIX1, n)$, where v is 0 or 1; t is *none*, *OR*, or *output*; f is *none* or *FIX0*; and n is a number that specifies a fan-in cell.

4.3 Algorithm

The algorithm for constructing Boolean circuits is formulated as a set of state transition rules for changing the fix component and type of each cell. These rules are executed for one set of input signals and expected output signals. Before fix components and types are changed by these rules, the value components of the input nodes are set according to the input signals in the example set. The values of other cells are set according to their current types and the values of their fan-in cells. The fix components of the output nodes are set according to their values and expected output signals.

The transition rules are described in pseudo-code as follows.

```

if the type is output then
    if the value component is 1 and the expected signal is 0 then
        the fix component is set to FIX0
  
```

if the value component is 0 **and** the expected signal is 1 **then**
 the fix component is set to *FIX1*
 a fan-in cell is randomly chosen and specified (*)

if the type is *OR* **and** the fix component is *none* **and**
 the value component is 1 **and** one of its fan-out cells has *FIX0* **then**
 if there is a fan-out cell whose type is *OR* or *output* with *FIX0* **or**
 there is no fan-out cell whose type is *OR* or *output* **then**
 the fix component is changed to *FIX0*
 the type is changed to *none*

if the type is *none* or *OR* **and** the fix component is *none* **and**
 the value component is 0 **and**
 specified by a fan-out cell with *FIX1* **then**
 the fix component is changed to *FIX1*
 the type is changed to *OR* (it may already be *OR*)
 a fan-in cell is randomly chosen and specified (*)

At (*), when the fix component of a cell is changed to *FIX1*, one of its fan-in cells is randomly chosen and specified. This choice greatly affects the number of steps taken for the algorithm to construct the expected circuit. We currently impose the following constraints on the chosen cell as heuristics:

- It is possible to reach the chosen cell from an input node with the value 1 in the direction with increasing coordinates.
- All the fan-in cells of the chosen cell that are OR gates or inputs nodes have the value 1.
- All the fan-out cells of the chosen cell that are OR gates or output nodes have the value 1.

Note that checking these constraints requires more components in the state of each cell. We omit description of those components and the rules for changing them in this article.

The state transitions defined above are repeated until no cell can make a transition. Then, we eliminate dangling branches of the formed circuit, i.e., those branches that do not lie between an input node and an output node. This step can also be realized by state transitions, but its description is omitted here.

The above algorithm is executed for each set of input and expected output signals, and a Boolean circuit of OR gates is formed or modified. After execution for each set, the value and fix components of all cells are reset, and the next set is prepared. After all of the sets have been processed, the first set is processed again. This process is repeated until no change is made to the Boolean circuit for any of the sets.

We had previously tried several algorithms, from which we formulated the one above. The proposed algorithm finds a Boolean circuit that satisfies the input–output sets, as in Fig. 4.

It is possible to define self-stability over multiple sets; however, we have yet to prove the self-stability of the algorithm.

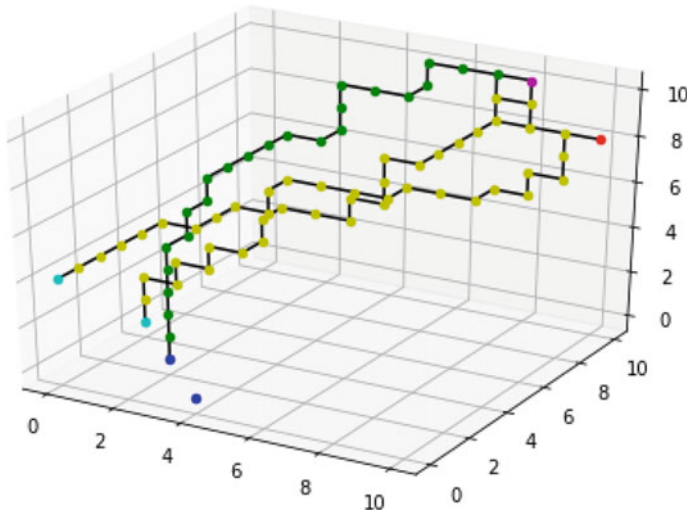


Fig. 4 Algorithm constructs a Boolean circuit composed of OR gates, four input nodes, and two output nodes. The input nodes are shown in blue (value 1) and cyan (value 0), and the output nodes are shown in magenta (value 1) and red (value 0). The OR gates are shown in green (value 1) and yellow (value 0). This circuit was constructed from four input–output sets, of which one is shown

5 Concluding Remark

In this article, we first reviewed the research efforts in DNA computing that have led to the implementation of cellular automata. Then, we summarized our work on gellular automata, focusing on self-stability. Finally, we proposed our third model of gellular automata that can learn Boolean circuits in 3D from sets of input–output signals. More work remains, including proving the self-stability of the algorithm.

Although gellular automata are based on implementing cellular automata by reaction–diffusion systems, the approach at the molecular level is also attractive and has various applications. It may be possible to combine the two approaches.

Regardless of which approach is taken, when a new concrete implementation method is developed, it may be necessary to update the existing models. It is also interesting to extend existing models assuming possible future implementation methods. One possibility is the introduction of directionality into a cellular space, as we assumed in our third model. Directionality can be introduced by electric or magnetic fields or chemical gradients. With such novel factors, communication between cells will be enhanced, with corresponding enrichment of cellular automata models.

Acknowledgements We would like to thank Ibuki Kawamata, Ferdinand Peper, and Damien Woods for their valuable comments for improving the article. The research reported in this article was partly supported by Kakenhi 17K19961 from the Ministry of Education, Culture, Sports, Science and Technology, Japan.

References

1. C.H. Bennett, Logical reversibility of computation. *IBM J. Res. Dev.* **17**, 525–532 (1973)
2. A. Currin, K. Korovin, M. Ababi, K. Roper, D.B. Kell, P.J. Day, R.D. King, Computing exponentially faster: implementing a non-deterministic universal Turing machine using DNA. *J. R. Soc. Interface*, 1420160990 (2017)
3. Y. Benenson, T. Paz-Elizur, R. Adar, E. Keinan, Z. Livneh, E. Shapiro, Programmable and autonomous computing machine made of biomolecules. *Nature* **414**, 430–434 (2001)
4. L.M. Adleman, Molecular computation of solutions to combinatorial problems. *Science* **266**, 1021–1024 (1994)
5. L. Qian, E. Winfree, Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**(6034), 1196–1201 (2011)
6. L. Qian, E. Winfree, J. Bruck, Neural network computation with DNA strand displacement cascades. *Nature* **475**, 368–372 (2011)
7. H.-L. Chen, D. Doty, D. Soloveichik, Deterministic function computation with chemical reaction networks. *Nat. Comput.* **13**, 517–534 (2014)
8. E. Winfree, F. Liu, L.A. Wenzler, N.C. Seeman, Design and self-assembly of two-dimensional DNA crystals. *Nature* **394**, 539–544 (1998)
9. M. McEvoy, N. Correll, Materials that couple sensing, actuation, computation, and communication. *Science* **347**(6228), 1261689 (2015)
10. A. Padirac, T. Fujii, A. Estévez-Torres, Y. Rondelez, Spatial waves in synthetic biochemical networks. *J. Am. Chem. Soc.* **135**(39), 14586–14592 (2013)
11. A.S. Zadorin, Y. Rondelez, G. Gines, V. Dilhas, G. Urtel, A. Zambrano, J.-C. Galas, A. Estevez-Torres, Synthesis and materialization of a reaction–diffusion French flag pattern. *Nat. Chem.* **9**, 990–996 (2017)
12. A. Cangialosi, C.K. Yoon, J. Liu, Q. Huang, J. Guo, T.D. Nguyen, D.H. Gracias, R. Schulman, DNA sequence–directed shape change of photopatterned hydrogels via high-degree swelling. *Science* **357**(6356), 1126–1130 (2017)
13. P. Yin, S. Sahu, A.J. Turberfield, J.H. Reif, Design of autonomous DNA cellular automata. *Lect. Notes Comput. Sci.* **3892**, 399–416 (2005)
14. L. Qian, E. Winfree, Parallel and scalable computation and spatial dynamics with DNA-based chemical reaction networks on a surface. *Lect. Notes Comput. Sci.* **8727**, 114–131 (2014)
15. S. Clamons, L. Qian, E. Winfree, Programming and simulating chemical reaction networks on a Surface. *J. R. Soc. Interface* **17**, 20190790 (2020)
16. B. Wang, C. Chalk, D. Soloveichik, “SIMD, DNA: Single instruction, multiple data computation with DNA strand displacement cascades”. *Lect. Notes Comput. Sci.* **11648**, 219–235 (2019)
17. P.W.K. Rothmund, N. Papadakis, E. Winfree, Algorithmic self-assembly of DNA Sierpinski triangles. *PLoS Biol.* **2**(12), e424 (2004)
18. K. Fujibayashi, R. Hariadi, S.H. Park, E. Winfree, S. Murata, Toward reliable algorithmic self-assembly of DNA tiles: a fixed-width cellular automaton pattern. *Nano Lett.* **8**(7), 1791–1797 (2008)
19. D. Woods, D. Doty, C. Myhrvold, J. Hui, F. Zhou, P. Yin, E. Winfree, Diverse and robust molecular algorithms using reprogrammable DNA self-assembly. *Nature* **567**, 366–372 (2019)
20. D. Scalise, R. Schulman, Emulating cellular automata in chemical reaction-diffusion networks. *Lect. Notes Comput. Sci.* **8727**, 67–83 (2014)
21. S. Murata, A. Konagaya, S. Kobayashi, H. Saito, M. Hagiya, Molecular robotics: a new paradigm for artifacts. *New Gener. Comput.* **31**(1), 27–45 (2013)
22. M. Hagiya, A. Konagaya, S. Kobayashi, H. Saito, S. Murata, Molecular robots with sensors and intelligence. *Acc. Chem. Res.* **47**(6), 1681–1690 (2014)
23. K. Abe, S. Murata, Programming methods for DNA-based reaction-diffusion systems. *New Gener. Comput.* **38**(2), 379–393 (2020)
24. M. Hagiya, S. Wang, I. Kawamata, S. Murata, T. Isokawa, F. Peper, K. Imai, On DNA-based gellular automata. *Lect. Notes Comput. Sci.* **8553**, 177–189 (2014)

25. S. Wang, K. Imai, M. Hagiya, On the composition of signals in gellular automata. *CANDAR*, 499–502 (2014)
26. S. Wang, K. Imai, M. Hagiya, An approach to constructing and simulating block cellular automata by gellular automata. *CANDAR*, 442–448 (2015)
27. M. Hagiya, K. Imai, On the persistency of gellular automata. *Revers. Univers.* 373–383 (2018)
28. K. Morita, A simple universal logic element and cellular automata for reversible computing. *Lect. Notes Comput. Sci.* **2055**, 102–113 (2001)
29. T. Yamashita, T. Isokawa, F. Peper, I. Kawamata, M. Hagiya, Turing-completeness of asynchronous non-camouflage cellular automata. *Lect. Notes Comput. Sci.* **10248**, 187–199 (2017)
30. T. Yamashita, T. Isokawa, F. Peper, I. Kawamata, M. Hagiya, Turing-completeness of asynchronous non-camouflage cellular automata. *Inf. Comput.* **274**, 104539 (2020)
31. T. Yamashita, M. Hagiya, in *Simulating population protocols by gellular automata*. 57th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), pp. 1579–1585 (2018)
32. T. Yamashita, A. Yagawa, M. Hagiya, Self-stabilizing gellular automata. *Lect. Notes Comput. Sci.* **11493**, 272–285 (2019)
33. T. Hongu, M. Hagiya, Self-stabilizing distributed algorithms by gellular automata. *Lect. Notes Comput. Sci.* **12286**, 86–98 (2021)
34. T. Hongu, M. Hagiya, Self-stabilizing distributed algorithms by gellular automata. *Complex Syst.* **30**(2), 159–185 (2021)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.



Patterning DNA Origami on Membranes Through Protein Self-Organization



Beatrice Ramm, Alena Khmelinskaia, Henri G. Franquelim,
and Petra Schwille

Abstract Spatial organization on the atomic scale is one of the key objectives of nanotechnology. The development of DNA nanotechnology is a hallmark of material programmability in 2D and 3D, in which the large variety of available DNA modifications allows it to be interfaced with a number of inorganic and organic materials. Nature's solution to spatiotemporal control has been the evolution of self-organizing protein systems capable of pattern formation through energy dissipation. Here, we show that combining DNA origami with a minimal micron-scale pattern-forming system vastly expands the applicability of DNA nanotechnology, whether for the development of biocompatible materials or as an essential step toward building synthetic cells from the bottom up. We first describe the interaction of DNA origami nanostructures with model lipid membranes and introduce the self-organizing MinDE protein system from *Escherichia coli*. We then outline how we used DNA origami to elucidate diffusiophoresis on membranes through MinDE protein pattern formation. We describe how this novel biological transport mechanism can, in turn, be harnessed to pattern DNA origami nanostructures on the micron scale on lipid membranes. Finally, we discuss how our approach could be used to create the next generation of hybrid materials, through cargo delivery and multiscale molecular patterning capabilities.

B. Ramm · A. Khmelinskaia · H. G. Franquelim · P. Schwille (✉)
Max Planck Institute of Biochemistry, Martinsried, Germany
e-mail: schwille@biochem.mpg.de

B. Ramm
Department of Physics, Princeton University, Princeton, NJ, USA

A. Khmelinskaia
Department of Chemistry, Ludwig Maximilian University of Munich, Munich, Germany

H. G. Franquelim
Interfaculty Centre for Bioactive Matter, Leipzig University, Leipzig, Germany

1 Introduction

The plasma membrane is a rather complex organelle, formed from a bilayer composed mostly of a mixture of phospholipids, sphingolipids, cholesterol, and a variety of peripheral and integral proteins [1, 2]. Long believed to be merely a passive barrier between the cell and its surrounding environment, the plasma membrane plays a fundamental role in a number of biological processes, such as cell signaling and signal transduction, motility, and cellular division. While research of such phenomena is of high interest both on a fundamental and technological level, their complexity often enough requires the use of model systems that allow their interrogation in a controlled environment [3].

Supported lipid bilayers (SLBs) are one particular example of model lipid membranes that are of great interest, as they can not only recapitulate the heterogeneous composition and lateral dynamics of plasma membranes, but can also be patterned in 2D and 3D. While they are traditionally formed on continuous planar surfaces, over the last two decades SLBs have been combined with nanostructured surfaces (Fig. 1a) [4–10], patterned substrates [11–13], microfabricated compartments [14, 15], and 3D printed complex architectures [16].

Beyond controlling membrane topology and topography, several methods can be applied to control molecular distribution and localization in and on SLBs. Lipid phase separation [17, 18] in membranes composed of lipids of different melting temperatures is itself an example of molecular demixing [19–21] (Fig. 1b) and can be harnessed to change the distribution of membrane-coupled proteins [22–24]. On the other hand, external forces can be applied to control molecular distribution: Electric fields can be used to generate gradients of charged lipids [25, 26], while surface acoustic waves have been shown to result in lipid demixing and even protein transport and accumulation [27–29]. Light responsive molecules can also be exploited to control membrane properties—the incorporation of azobenzene groups in a lipidic hydrophobic moiety results in control over lipid phase separation [30–33], while control over protein binding and localization can be achieved by the fusion of light responsive membrane-binding domains to proteins of interest [33–35]. While most of these methods offer reversible control over membrane properties and molecular localization, their applicability can be restricted due to membrane diffusion, introduced perturbations, spatio- and/or temporal resolution or scalability.

DNA-based nanostructures take advantage of the unique and inherent properties of the DNA double helix and are the paragon of a molecular breadboard. With its nanoscale addressability and diversity of possible modifications, complex molecular patterns can be built on the surface of DNA assemblies in 2D and 3D (Fig. 1d) [36–39]. It is thus tempting to employ DNA origami as a tool for patterning lipid membranes. Indeed, a lot of attention has been given in recent years to the development and investigation of membrane-active DNA nanostructures [40]. From the exploration of a variety of membrane-binding strategies, including different hydrophobic moieties [41–44], ligand-receptor type binding [43], covalent conjugation [45], and ionic strength [46], to the development of triggerable membrane-binding nanostructures

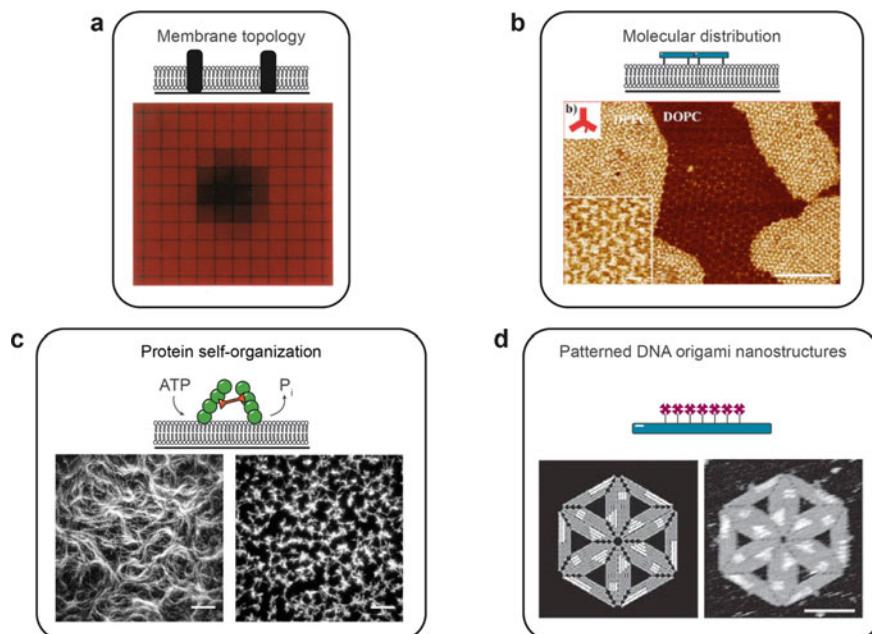
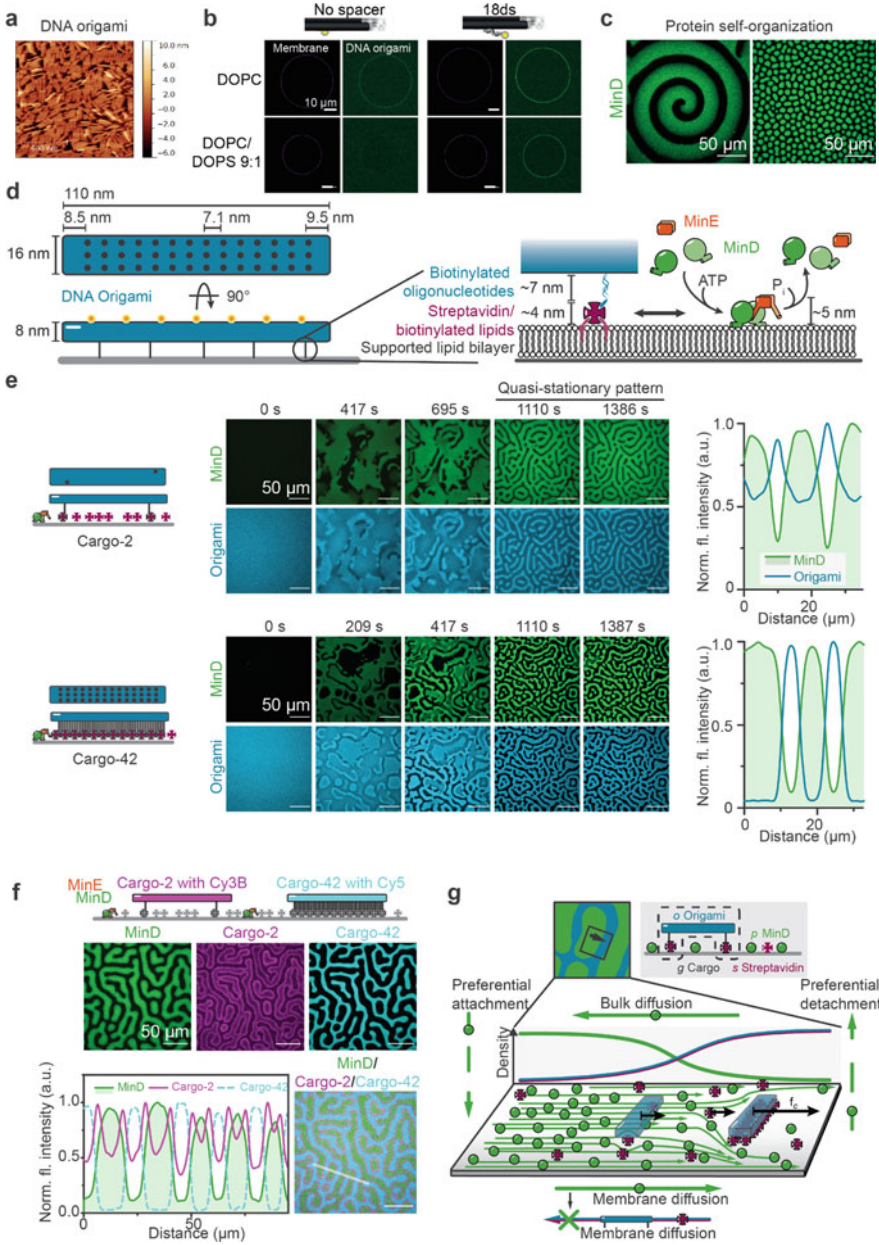


Fig. 1 Molecular patterning strategies, with emphasis on model lipid membranes as mimics of natural membranes. **a** Epifluorescence photographs of a corral array ($20 \times 20 \mu\text{m}$) of a supported lipid bilayer partitioned by a microfabricated grid of chrome lines. Corrals in the center were photo-bleached (dark) with a circular spot highlighting how the barriers (dark lines) prevent diffusive mixing of lipids between corrals. **b** Cholesteryl-TEG modified three-point star DNA tiles preferentially bind and self-assemble on the liquid-ordered phase of phase-separated DOPC/PPC supported lipid bilayers. **c** Myosin II action on F-actin bound to supported lipid bilayers induces distinct patterns on the membrane such as filament bundles (left) and linked, polar asters (right). **d** Model and AFM image of patterned DNA origami triangles combined into a hexagon shape. Scale bars, **b**, 200 nm and inset 25 nm, **c**, 10 μm , **d**, 100 nm. **a**, adapted with permission from [5]. Copyright 1998 American Chemical Society, **b**, adapted with permission from [62]. Copyright 2017 American Chemical Society, **c**, adapted with permission from [66], **d**, reprinted by permission from Springer Nature: Nature, [36], Copyright 2006

[47, 48], channels [49–51] and membrane shaping coats [52–56], we now have a good understanding of how to control membrane binding and dynamics of DNA origami nanostructures [57–59]. For example, we know that not only the number, but also the position and accessibility of membrane anchors, governs the efficiency of membrane binding. Different strategies to attach the membrane anchor to the DNA origami will also result in different levels of sensitivity to variations in local electrostatics (Fig. 2a, b). However, patterning of DNA origami on lipid membranes has so far been limited to the assembly of large-scale arrays or the preferential binding of DNA nanostructures to different lipid phases (Fig. 1b) [42, 60–63].

Biological spatiotemporal organization is achieved by self-organizing protein systems that are capable of pattern formation and large-scale molecular transport



◀**Fig. 2 DNA origami as a tool to elucidate the physical mechanism underlying cargo transport by MinDE self-organization.** **a** AFM image of the bare DNA origami nanostructure (20-helix bundle; $110 \times 16 \times 8$ nm) deposited on mica. **b** Charge sensitivity of membrane-binding DNA origami structures depends on the used attachment strategy of membrane-anchoring moiety. While TEG-chol moieties inserted directly into the structure result in poor DNA origami binding to membranes containing negatively charged lipids, when compared to bare DOPC, the use of an 18 nucleotide long double stranded DNA linker contributes to an efficient membrane binding in both conditions. Representative confocal microscopy images of the equatorial plane of giant unilamellar vesicles (GUVs) incubated with a 3 nM solution of DNA nanostructures modified with two TEG-chol anchors. GUVs contained 0.005 mol% Atto655-DOPE for fluorescence imaging, while each origami structure carried three Atto488 dyes. **c** Examples of patterns formed by MinDE self-organization on planar supported lipid bilayers in vitro (left: $0.75 \mu\text{M}$ MinD, $2 \mu\text{M}$ His-MinE; right: $1 \mu\text{M}$ MinD, $1.5 \mu\text{M}$ MinE-His). **d** Schematic of the MinDE self-organization mechanism and the synthetic membrane-anchored cargo consisting of a DNA origami nanostructure and streptavidin building blocks. The DNA origami nanostructure has 7 dyes at the upper facet and 42 addressable sites for incorporation of biotinylated oligonucleotides at the lower facet which in turn bind to lipid-anchored streptavidin on the SLB. Fueled by ATP hydrolysis, MinDE attach and detach to and from the membrane in a concerted manner. **e** The contrast of the patterns resulting from DNA origami transport by MinDE increases with increasing number of incorporated streptavidin per cargo, as does the size of the MinD minima. Representative time series and line plots for origami nanostructures equipped with 2 or 42 streptavidin building blocks ($1 \mu\text{M}$ MinD (30% EGFP-MinD), $1.5 \mu\text{M}$ MinE-His in presence of 0.1 nM origami-Cy5 with 2 or 42 biotinylated oligonucleotides, streptavidin). **f** MinDE self-organization induces sorting of two cargo species with distinct membrane footprint. Representative images and line plots for simultaneous transport of cargo-2 and cargo-42 by MinDE ($1 \mu\text{M}$ MinD (30% EGFP-MinD), $1.5 \mu\text{M}$ MinE-His, 50 pM origami-Cy5b with two biotinylated oligonucleotides, and 50 pM origami-Cy5 with 42 biotinylated oligonucleotides, non-labeled streptavidin). **g** Schematic of the diffusiophoretic mechanism underlying the molecular transport by MinDE. MinDE reactions and diffusion generate MinDE patterns and density gradients. The diffusive fluxes of MinD exert a frictional force, f_c , on the cargo molecules that depends on the effective size of the cargo molecules. Scale bars **a**, 400 nm , **b**, $10 \mu\text{m}$, **c-f**, $50 \mu\text{m}$; **a**, adapted from [58], **b**, adapted with permission from [59]. Copyright 2018 American Chemical Society. **d-g**, adapted from [89] under a CC BY 4.0 license

through energy dissipation. Much emphasis has been laid on understanding and harnessing complex eukaryotic protein systems based on cytoskeletal and translational motor proteins (Fig. 1c) [64–69]. In contrast, bacterial systems have largely been neglected even though they are often simpler both, on a mechanistic and compositional level [70], making them ideal targets for nanotechnological applications. One such example comes from the bacterium *Escherichia coli*—the Min system. At its core, it consists of two proteins only, the ATPase MinD and the ATPase activating protein MinE, ATP as an energy source, and a lipid membrane as a reaction platform. The two proteins interact with each other and the lipid membrane to form patterns via a reaction-diffusion mechanism. This compositional simplicity paired with its rich dynamics made the system a paradigm for the study of biological pattern formation [71]. Over 30 years of research in vivo, in vitro, and in silico allowed us to obtain a rather detailed understanding of the underlying molecular mechanism [72–79]. In short, MinD binds to the membrane cooperatively upon ATP-induced dimerization, presumably involving the formation of higher order structures. MinE binds

to membrane-bound MinD stimulating its ATPase activity, which leads to protein detachment from the membrane (Fig. 2d). In *E. coli*, MinDE performs pole-to-pole oscillations which generate a time-averaged gradient of the passenger protein MinC with a minimum at midcell. As MinC is an inhibitor of the main divisome protein FtsZ, this time-averaged protein gradient restricts the assembly of the cell division machinery to the middle of the cell. In 2008, MinDE dynamics were reconstituted *in vitro*: The purified proteins MinD and MinE supplied with ATP as an energy source self-organized on supported lipid bilayers in aqueous buffer forming traveling surface waves [77].

Depending on the specific reaction conditions, the system has been shown to exhibit a variety of patterns *in vitro*, such as traveling surface waves, dot and labyrinthine patterns, and also oscillatory behavior (Fig. 2c) [12, 77, 80, 81]. Indeed, while the reaction system is truly nonlinear, *i.e.*, small changes in the reaction conditions can lead to drastic changes in the reaction outcome; considerable effort has been invested in determining the parameters influencing pattern formation in order to achieve control over it. Protein concentration and ratios [77, 81], ionic strength of the buffer [82], molecular crowding [7, 14, 83], membrane charge and fluidity [12, 82, 84], and in particular the reaction space geometry, including the form and size of the membrane as well as the surface to volume ratio, heavily influence the type of obtained patterns. For example, when MinDE is confined in membrane-coated PDMS microcompartments with an elongated, cell-like shape, they recapitulate the oscillatory behavior that occurs in the bacterial cells *in vivo* [12, 14, 78]. On planar rectangular membrane patches, in turn, MinDE traveling surface waves align along the longest axis [7]. Beyond modifications to the reaction surface, engineering of both, MinD and MinE, has revealed that modifications of key molecular motifs, such as the membrane binding and dimerization helices, can change the type of MinDE patterns, such that robust standing waves or MinDE traveling waves with wavelength on a millimeter-scale can be generated [85, 86]. These examples and many others found in the literature [71] clearly demonstrate that the Min system, although compositionally simple, is a striking pattern-forming system whose behavior can be tightly controlled using a varied set of parameters.

Here, we first review how DNA nanotechnology has been previously harnessed to elucidate physical mechanisms governing complex biological phenomena, *i.e.*, the transport of molecular cargo by MinDE dynamics. Building on this work, we then investigate how MinDE-dependent transport of DNA origami nanostructures can be used to generate stable, biologically compatible patterns at the micron to millimeter-scale. We further speculate how combining the rich dynamics of the self-organizing MinDE system with the nanometer-precision addressability of DNA origami nanostructures opens up new possibilities in creating the next generation of hybrid materials with multiscale molecular patterning capabilities.

2 DNA Origami as a Tool to Elucidate Molecular Mechanisms

Recently, we and others have shown that the Min system can not only form mesmerizing patterns on membranes itself, but also induce patterns of other functionally unrelated proteins [87, 88]. These studies showed that MinDE-induced regulation of lipid-bound proteins that have a long membrane dwell-time results in their net transport. This suggested that the non-specific interaction of MinDE with these proteins modulates their diffusion on the membrane, an effect that should depend on the properties of the transported, membrane-bound molecules, such as the area they occupy on the membrane, *i.e.*, the membrane footprint, or their diffusion on the membrane. In order to specifically probe distinct hypotheses regarding the underlying mechanism, it was desirable to modify specific cargo properties over a wide range, while keeping all other parameters comparable. While the properties of membrane-attached proteins cannot easily be tuned in a defined fashion, DNA origami nanostructures are fully programmable, as illustrated across this book. Importantly, binding and diffusion of DNA origami nanostructures on model membranes have been extensively characterized in detail, as described above, rendering them an ideal cargo for the detailed study of MinDE-dependent cargo transport [89].

The DNA origami nanostructure employed was a 20-helix bundle with 42 addressable positions at the bottom facet that can be modified with cholesteryl or biotinyl moieties for membrane anchoring ($110 \times 16 \times 8$ nm) [58] (Fig. 2a, d). While the origami structures with cholesteryl oligonucleotides directly bound to the membrane, the ones with biotinylated oligonucleotides could be decorated with streptavidin which in turn bound to biotinylated lipids on the SLB. The former design allowed us to vary the diffusion of the structures while maintaining similar membrane footprints; the latter was used to vary the membrane footprint and thus, the effective size of the molecule via the amount of incorporated streptavidin molecules. For visualization by fluorescence microscopy, the DNA origami nanostructures were functionalized with seven dyes on the upper facet. In order to test the redistribution of these structures in a controlled manner, we used conditions under which MinDE formed quasi-stationary patterns, *i.e.*, after an initial chaotic self-organization phase, labyrinthine patterns emerge, whose macroscopic appearance is stable over long periods of time, but which are nevertheless maintained by continuous binding and unbinding of the individual proteins. We found that MinDE was also capable of redistributing these rather large structures (compared to the 5 nm-sized MinDE proteins [90] and the previously investigated membrane-bound proteins [87, 88]) leading to an anti-correlated pattern of the DNA origami on the membrane. Comparing the final quasi-stationary patterns for structures bearing different kinds and numbers of membrane anchors, we were able to show that the extent of the molecular transport by MinDE increased with the membrane footprint of the cargo molecule, *i.e.*, the effective size of the cargo molecule on the membrane surface (Fig. 2e), but did not directly correlate with cargo diffusion. In contrast, in experiments with conditions under which MinDE formed

traveling surface waves, we found that both, the cargo's effective size and its diffusion coefficient, play a role in its transport efficiency. Cargo molecules that are too slow in comparison to the wave propagation will not be transported effectively. Even more intriguingly, we showed that MinDE self-organization was capable of spatially sorting DNA origami nanostructures with distinct effective sizes, *i.e.*, one with 2 and one with 42 streptavidin molecules bound to its bottom surface (Fig. 2f).

Our experiments allowed us to rule out several possible physical mechanisms arriving at diffusiophoresis as the simplest one that could fully explain our observations (Fig. 2g). Diffusiophoresis generally describes the transport of particles in fluids along concentration gradients of small solutes and has mostly been reported in a non-biological context such as colloid transport [91, 92]. During MinDE self-organization, density gradients of the proteins on the membrane are established which lead to diffusive protein fluxes on the membrane toward low protein densities. Due to the high density of proteins on the membrane, MinDE proteins directly interact with other molecules on the membrane, generating a frictional force on these "cargo" molecules. As a result, the diffusive fluxes of MinDE and cargo couple on the membrane, leading to their transport toward, and subsequent accumulation in, areas of low MinDE density. As the frictional force increases with the effective size of the molecule, molecules with a larger membrane footprint experience a stronger redistribution than smaller ones. Although this first description of diffusiophoresis by a biological self-organizing system was achieved *in vitro*, diffusiophoresis could be more widespread in cellular systems, but might be masked by the stronger, specific molecular interactions within the cell.

3 Stable DNA Origami Patterns on Lipid Membranes

Harnessing the established experimental framework, we set out to explore the possibilities arising from the ability to pattern DNA origami nanostructures by MinDE self-organization. In this manuscript, we started by testing whether the resulting patterns could be stabilized after they had been established. DNA origami distribution out of equilibrium is maintained by the energy dissipation upon MinDE self-organization and is as such reversible: DNA origami patterns respond to changes in MinDE patterns induced, for example, by the addition of more MinE [89] and DNA origami patterns disappear altogether by thermal mixing when MinDE activity subsides [87, 88]. This reversibility may become inconvenient for any downstream applications, considering the dynamic nature of lipid membranes. Thus, it would be desirable to "freeze" MinDE-induced cargo patterns, *e.g.*, through crosslinking. Previous work has described two main strategies for hierarchical self-assembly of large DNA-based structures (for a more detailed review, refer to [93]): "sticky" interactions [94–96], based on DNA base-pairing and "stacking" interactions [97–99], based on blunt-end interactions and shape complementarity of building blocks. While the first one can be accomplished by programming complementary single-stranded DNA extensions between building blocks or by addition of a single-stranded DNA

oligonucleotide complementary to single-stranded portions on the DNA structures, the second one can be controlled by the concentration of Mg^{2+} in solution. As MinDE self-organization is sensitive to the ionic strength of the buffer [82], we here opted to modify our DNA origami nanostructures to contain 8 nucleotide long poly-A ssDNA extensions on each end. The addition of a complementary 14 nucleotide long poly-T polymerization staple would result in the crosslinking of DNA nanostructures and their polymerization in an end-to-end fashion.

First, we verified the behavior of a priori polymerized DNA origami nanostructures upon MinDE self-organization (Fig. 3a). A homogeneous, membrane-bound layer of DNA origami was crosslinked before MinDE self-organization was started with ATP. Upon ATP addition, MinDE self-organized into labyrinthian patterns, as did the excess membrane-bound streptavidin. However, polymerized DNA origami did not get patterned, remaining homogeneously distributed. This shows that the DNA origami nanostructures indeed formed large stable polymers on the membrane whose diffusional behavior was virtually frozen and as such could not be transported by MinDE, as the diffusiophoretic transport requires diffusive mobility of both the cargo and MinDE proteins.

In contrast, when we started MinDE self-organization in presence of these DNA origami structures prior to the addition of polymerization staples, MinDE dynamics induced anti-correlated DNA origami patterns similar to those observed for non-modified DNA origami nanostructures (Fig. 3b). After MinDE patterns entered the quasi-stationary phase, we added the polymerization staple to “freeze” the patterns in place. Indeed, when we added more MinE, only the MinDE patterns changed, whereas the DNA origami retained their original pattern. Intriguingly, we could observe the uncorrelated patterning of two cargo molecules: While the crosslinked DNA origami retained the original pattern, the excess membrane-anchored protein streptavidin, here used to bind DNA origami to the membrane, was transported by the reorganized MinDE patterns. In the control experiment, in which the polymerization staple was added to a DNA origami structure that cannot be crosslinked (*i.e.*, without single-stranded overhangs), the addition of MinE resulted in concomitant and anti-correlated pattern reorganization of MinDE and DNA origami (Fig. 3c).

Having shown that MinDE-induced DNA origami patterns could be spatially stabilized, we next asked whether other molecules could be transported or targeted using DNA origami as a shuttle or scaffold. Over the years, many studies have shown that molecules can be targeted to DNA origami nanostructures with nanometer precision via a wide variety of different chemistries [100]. As a proof of concept, we used a DNA origami structure that contains 15 cholesteryl moieties for membrane binding and which is strongly redistributed by MinDE self-organization [89]. We replaced the fluorescent dye staples on the upper facet of these structures with biotinyl moieties which allowed the coupling of fluorescently labeled streptavidin to the upper facet of the DNA origami (Fig. 3d). Indeed, MinDE was also capable of transporting this streptavidin-loaded DNA origami, resulting in high-contrast, anti-correlated patterns (Fig. 3e). When adding biotinylated stabilized actin filaments to such pre-patterned streptavidin-bearing DNA origami structures, actin preferentially

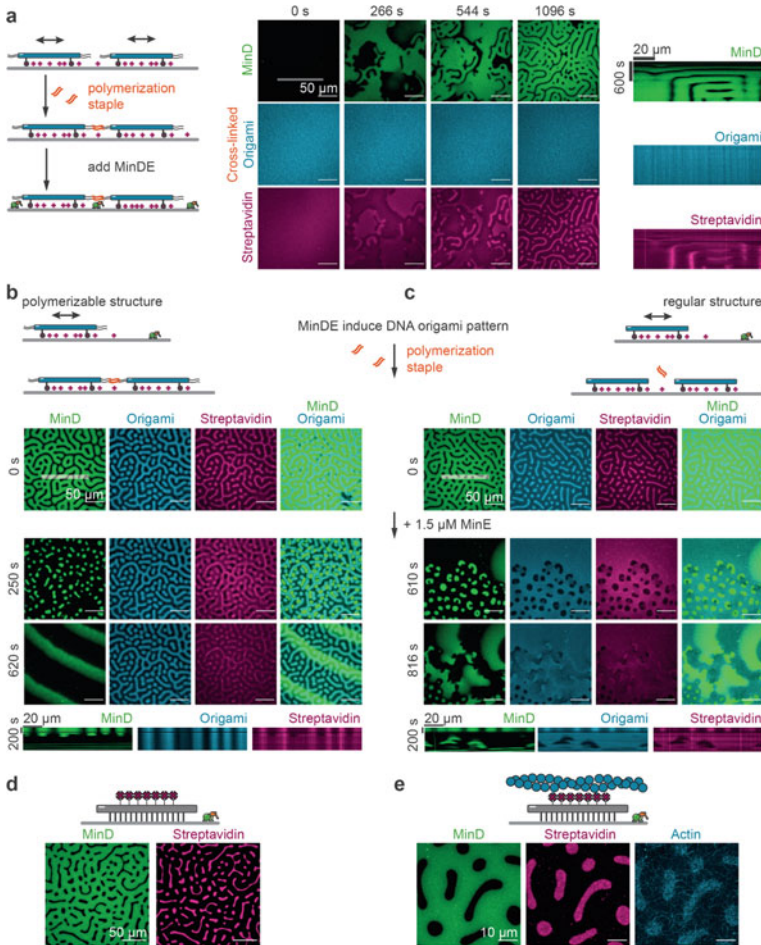


Fig. 3 DNA origami patterns induced by MinDE self-organization can be “frozen” on lipid membranes. **a** Pre-polymerized DNA origami cannot be reorganized by the MinDE system. Representative time series and kymographs of pre-polymerized DNA origami and MinD channels after initiating the start of MinDE self-organization by addition of ATP ($1 \mu\text{M}$ MinD, $1.5 \mu\text{M}$ MinE-His, 0.1 nM cargo-2 pre-polymerized by addition of $17 \mu\text{M}$ polymerization staple, Alexa568-streptavidin). **b** MinDE-induced DNA origami patterns are stabilized by DNA origami crosslinking upon addition of polymerization staple, **c** but not if complementary overhangs are absent in the DNA nanostructure. Representative images and kymographs of DNA origami, MinD and streptavidin patterns after addition of polymerization staple and subsequent addition of MinE for a structure with extensions that can be polymerized (**b**, left) and a regular structure (**c**, right) (initial conditions: $1 \mu\text{M}$ MinD, $1.5 \mu\text{M}$ MinE-His, 0.1 nM (polymerizable) cargo-2, Alexa568-streptavidin; addition of $17 \mu\text{M}$ polymerization staple and subsequently $1.5 \mu\text{M}$ MinE-His). **d** MinDE induces patterns of streptavidin-decorated origami ($1 \mu\text{M}$ MinD (30% EGFP-MinD), $1.5 \mu\text{M}$ MinE-His, 0.1 nM cargo-cho15 with biotin extension on upper facet, Alexa568-streptavidin), **e** which can be crosslinked upon addition of biotinylated stabilized actin filaments, which preferentially localize to the regions enriched in streptavidin-decorated origami ($1 \mu\text{M}$ MinD, $1.5 \mu\text{M}$ MinE-His, 0.1 nM cargo-cho15 with biotin extension, Alexa568-streptavidin, Alexa-647-Phalloidin-labeled, biotinylated actin). Scale bars **a–d**, $50 \mu\text{m}$; **e**, $10 \mu\text{m}$

bound to the regions enriched in DNA origami, *i.e.*, the MinDE minima. Moreover, the presence of multiple binding sites per filament resulted in pattern crosslinking.

Taken together, the new results presented herein are a proof of concept of how DNA origami, when combined with the MinDE system, can be used to create stable and spatially uncorrelated patterns of distinct molecules on a dynamic support, such as a lipid membrane.

4 Challenges and Opportunities

Since the birth of DNA nanotechnology, the biophysical community has been exploiting the well understood behavior of DNA and its versatility regarding conjugation/modification [100] to interrogate a number of fundamental biological processes [101]. From single molecule methods to cellular studies, researchers used DNA origami to investigate enzymatic cascades [102–104], motor proteins [105, 106], DNA binding proteins [107–109], and even receptor-mediated cellular responses, *e.g.*, apoptosis, phagocytosis, B cell activation, T cell stimulation [110–114], to name only a few. Here, we reviewed how our detailed knowledge and fine control over membrane binding and diffusion of DNA origami nanostructures allowed us to unravel a new unspecific transport mechanism by the self-organizing MinDE system of *E. coli* [89]. Our recent results reiterate how the power of DNA origami can be used to interrogate complex dynamic phenomena in a controlled fashion and further expand the available *in vitro* toolkit.

Non-specific transport based on diffusiophoresis should in principle occur in any system that generates and/or maintains concentration gradients. We have shown that (membrane-bound) DNA origami is the ideal tool to characterize such phenomena in detail. For example, similar approaches could be used to explore diffusiophoretic transport or related effects in other self-organizing membrane systems such as small GTPases [115], membrane-bound kinase/phosphatase networks [10], or minimal actin cortices [64–66]. Soluble DNA origami structures might even be harnessed to study incorporation and molecular transport that occur during liquid–liquid phase separation of biomolecules, a phenomena that recently gained considerable attention in cell biology [116]. With the routine establishment of uncorrelated patterns, the complexity of such assays can be increased, allowing to explore the interplay between simultaneously occurring dynamics of distinct systems, as observed in cells.

To date, in the quest for biomolecular spatiotemporal patterning and transport, the focus has mostly been laid on using eukaryotic active matter elements on solid supports, which mediate cargo transport via specific interactions, such as the ParMRC system, actin, microtubules, and motor proteins [67–69]. Despite not being a “conventional” biological motor system, diffusiophoretic cargo transport by MinDE self-organization has the potential to enable complex biomolecular patterning on membranes with particularly high modularity when coupled to DNA origami, the model tool of molecular nanopatterning [117]. The variety of modification chemistries for membrane binding or molecular targeting, crosslinking and

actuation strategies of DNA origami, paired with the diversity of possible patterns and various control elements for the Min system, as well as the ease of purification of the MinDE proteins (small proteins as compared to eukaryotic molecular motors) and the non-specific nature of the MinDE-mediated transport that overcomes the need for additional adapter proteins, are all factors that contribute to making this an attractive combination for a variety of applications (Fig. 4a).

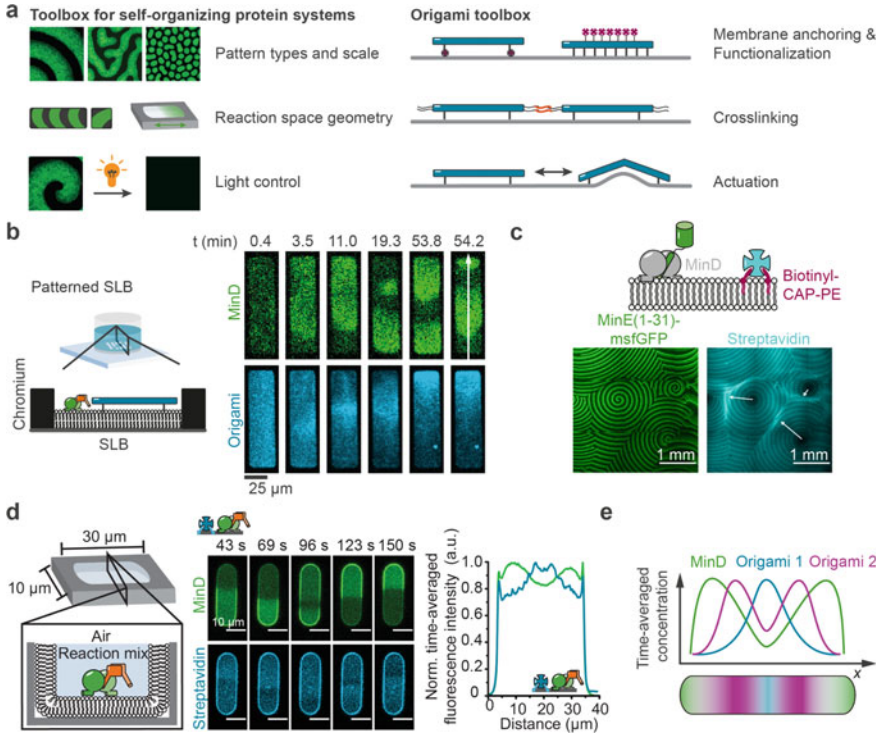


Fig. 4 Combination of DNA origami nanotechnology with self-organizing protein systems opens up new possibilities for molecular transport and patterning. **a** Schematic highlighting the variety of control elements available for self-organizing protein systems such as MinDE and DNA origami nanostructure toolboxes. **b** MinDE accumulates DNA origami structures along the long axis of patterned lipid membranes. Representative time series of MinDE traveling surface waves transporting DNA origami along the wave vector on chromium-patterned SLBs. **c** Large-scale gradients of cargo molecules established by a minimal MinDE system (indicated by white arrows). Representative images of MinE(1-31)-msfGFP and MinD self-organization in the presence of membrane-bound streptavidin (2.5 μ M MinD (20% Alexa647-MinD), 200 nM MinE(1-31)-msfGFP, Alexa568-streptavidin). **d** MinDE oscillations in cell-shaped compartments result in preferential localization of the cargo molecules toward the center of the compartment. Representative time-lapse images and time-averaged fluorescence intensity profile of MinDE oscillations and streptavidin counter-oscillations in PDMS microcompartments (1 μ M MinD, 2 μ M MinE, streptavidin-Alexa647). **e** Schematic highlighting the potential of generating two distinct time-averaged gradients of DNA origami in cell-shaped compartments simultaneously. **b** was adapted from [89] and **d** from [87] under a CC BY 4.0 license. Scale bars **b**, 25 μ m, **c**, 1 mm, **d**, 10 μ m

For example, by combining the ability to direct MinDE traveling waves on planar patterned bilayers [7] with DNA origami, reproducible directed molecular transport can be achieved, resulting in a gradient of the cargo of interest along the longest axis of a membrane patch [89] (Fig. 4b). On the other hand, we show here that MinDE-dependent net transport of molecules can occur over several millimeters and establish millimeter-sized gradients on the membrane (Fig. 4c), when MinE mutant peptides are employed [85]. As such, arbitrary cargo tethered to a DNA origami shuttle could be transported directionally over several millimeters and could even be controlled by light [33]. By harnessing the current technologies of microfabrication and membrane-coating of surfaces (see Introduction for more detail), exciting possibilities arise for macromolecular “writing” at a scale visible to the naked eye, resulting in a new generation of biohybrid devices. Hence, directed and controlled diffusiophoretic transport of DNA origami structures by MinDE could potentially be explored for molecular patterning, molecular delivery, biocomputation, or molecular separation based on membrane footprint.

Besides its potential application in nanotechnology, DNA origami transport by MinDE also offers new possibilities to achieve spatiotemporal organization in synthetic cells. We have shown in the past that MinDE self-organization can spatiotemporally position molecules via diffusiophoresis to the center of cell-shaped compartments [87, 88] (Fig. 4d). Using DNA origami structures with distinct membrane footprints, it should now be possible to target distinct molecules to at least three different regions of such a cell-like compartment (Fig. 4e). Furthermore, the concentration contrast between these regions can be controlled by adjusting the differences between DNA origami footprints. Combined with the recent strides that have been made to encapsulate a functional Min system into 3D compartments, such as monolayer sealed microfabricated compartments [15], water-in-oil droplets [118] or deformable giant unilamellar vesicles (GUV) [119, 120], DNA origami transport by MinDE could then be exploited for basic spatial compartmentalization of synthetic cells, *e.g.*, to tether and segregate genetic material or compartmentalize the activity of molecular machines. Using functional DNA origami nanostructures that can be actuated by a number of mechanisms, *e.g.*, strand displacement or insertion, pH, temperature, electric field, or light [121–127], even a functional divisome could potentially be generated, a long-standing goal of the field.

In conclusion, the versatility of DNA nanotechnology and the rich dynamics of self-organizing proteins such as the MinDE system can be harnessed together to transport, deliver, pattern, and sort functionalized cargo on membranes, thereby creating a new class of hybrid biomaterials for applications in nanotechnology and the bottom-up construction of synthetic cells.

5 Materials and Methods

Most of the experimental methods and materials for this chapter have been described in detail before [58, 89, 128], but are described in brief below, highlighting modifications and new methods.

MinDE plasmids and proteins—the plasmids pET28a-His-MinD_MinE and pET28a-His-MinE [77], pET28a-His-EGFP-MinD [11], pET28a-MinE-His [11, 77, 80, 81] and pET28a-MinE(1-31)-msfGFP [85] were used for purification of His-MinD, His-EGFP-MinD, His-MinE, MinE-His, and MinE(1-31)-msfGFP, respectively, as described in detail before [128].

DNA origami nanostructures—the modifications and the functionalization of the previously designed elongated DNA origami nanostructure [58] have been described in detail in [89]. For polymerizable DNA origami structures, end staples with 8A extensions are added upon DNA origami folding. Biotin-top-labeled DNA origami structures contained 7 modified oligonucleotides attached to extended staples on the upper facet. The assembly of the origami structure was performed in a one-pot reaction mix as described previously [58].

Self-organization assay on SLBs—SLBs were prepared as described before with a lipid composition of 30 mol% DOPG, 69 or 70% DOPC, and 1% CAP-Biotinyl-PEG in case of DNA origami with biotinyl moieties [128]. Binding of DNA origami to lipid membranes via streptavidin interactions or cholesteryl anchors has been previously described [89]. Self-organization assays were performed essentially as detailed in [128]. For large-scale MinDE traveling waves a MinE peptide [85], 200 nM MinE(1-31)-msfGFP, and 2.5 μ M MinD (20% Alexa647-MinD) were used in the self-organization assay that was performed in sticky-Slide VI 0.4 chambers (ibidi GmbH, Gräfelfing, Germany).

Crosslinking of DNA origami by strand hybridization—to trigger origami crosslinking, polymerization staple (14 T) was added to the chamber at a final concentration of 17 μ M, and the reaction mixture was mixed by pipetting. The large volume addition (40 μ l) induced changes to MinDE patterns in some experiments. To further change the MinDE pattern, 1.5 μ M MinE-His was added to the reaction mixture.

Attachment of actin to DNA origami—for attachment of labeled streptavidin to DNA origami, cholesteryl-modified DNA nanostructures were first bound to the membrane. Subsequently, the chamber was incubated with Alexa568-labeled streptavidin (ThermoFisher Scientific) at a final concentration of 1 μ g/ml. After 5–10 min incubation, unbound streptavidin was removed by gently washing 3 times with a total volume of 600 μ l reaction buffer. MinDE proteins were added to the reaction mixture, and the self-organization assay was started by addition of ATP. After pattern establishment, Alexa-647-Phalloidin-labeled biotinylated actin (produced as described in [65]) was added at a final concentration of 0.1 μ M and allowed to bind to the DNA origami before image acquisition.

Acknowledgements The authors thank Sven Vogel for providing stabilized actin filaments and the MPIB Core Facility for assistance in protein purification. The authors acknowledge financial support by the German Research Foundation (DFG, German Research Foundation) through

SFB-863-Project ID 111166240 and TRR 174-Project ID 269423233. Additional support from the Graduate School of Quantitative Biosciences Munich to A.K. and B.R., Max Planck Society to P.S., and Alexander von Humboldt Foundation to H.G.F (Humboldt Research Fellowship PTG/1152511/STP) is also acknowledged, as well as the support by the Center for NanoScience Munich. The authors further thank the Munich DNA Node for fruitful discussions.

References

1. T. Harayama, H. Riezman, Understanding the diversity of membrane lipid composition. *Nat. Rev. Mol. Cell Biol.* **19**, 281–296 (2018)
2. T.R. Shaw, S. Ghosh, S.L. Veatch, Critical phenomena in plasma membrane organization and function. *Ann. Rev. Phys. Chem.* **72**, 51–72 (2021)
3. E. Sezgin, P. Schwille, Model membrane platforms to study protein-membrane interactions. *Mol. Membr. Biol.* **29**, 144–154 (2012)
4. J.T. Groves, N. Ulman, S.G. Boxer, Micropatterning fluid lipid bilayers on solid supports. *Science* **275**, 651–653 (1997)
5. J.T. Groves, N. Ulman, P.S. Cremer, S.G. Boxer, Substrate–membrane interactions: mechanisms for imposing patterns on a fluid bilayer membrane. *Langmuir* **14**, 3347–3350 (1998)
6. H. Nakashima, K. Furukawa, Y. Kashimura, K. Sumitomo, Y. Shinozaki, K. Torimitsu, Pattern formation and molecular transport of histidine-tagged GFPs using supported lipid bilayers. *Langmuir* **26**, 12716–12721 (2010)
7. J. Schweizer, M. Loose, M. Bonny, K. Kruse, I. Mönch, P. Schwille, Geometry sensing by self-organized protein patterns. *Proc. Natl. Acad. Sci. USA* **109**, 15283–15288 (2012)
8. P.R. Spycher, H. Hall, V. Vogel, E. Reimhult, Patterning of supported lipid bilayers and proteins using material selective nitrodopamine-mPEG. *Biomater. Sci.* **3**, 94–102 (2015)
9. R.G. Moullick, G. Panaitov, S.-E. Choi, D. Mayer, A. Offenhäuser, Patterning artificial lipid bilayer on nanostructured surfaces. *Int. J. Nanomedicine* **13**, 55–58 (2018)
10. S.D. Hansen, W.Y.C. Huang, Y.K. Lee, P. Bieling, S.M. Christensen, J.T. Groves, Stochastic geometry sensing and polarization in a lipid kinase-phosphatase competitive reaction. *Proc. Natl. Acad. Sci. USA* **116**, 15013–15022 (2019)
11. K. Zieske, J. Schweizer, P. Schwille, Surface topology assisted alignment of Min protein waves. *FEBS Lett.* **588**, 2545–2549 (2014)
12. K. Zieske, P. Schwille, Reconstitution of self-organizing protein gradients as spatial cues in cell-free systems. *Elife* **3** (2014). <https://doi.org/10.7554/eLife.03949>
13. D.A. Ramirez-Diaz, A. Merino-Salomón, F. Meyer, M. Heymann, G. Rivas, M. Bramkamp, P. Schwille, FtsZ induces membrane deformations via torsional stress upon GTP hydrolysis. *Nat. Commun.* **12**, 3310 (2021)
14. Y. Caspi, C. Dekker, Mapping out Min protein patterns in fully confined fluidic chambers. *Elife* **5** (2016). <https://doi.org/10.7554/eLife.19271>
15. H. Eto, N. Soga, H.G. Franquelim, P. Glock, A. Khmelinskaia, L. Kai, M. Heymann, H. Noji, P. Schwille, Design of sealable custom-shaped cell mimics based on self-assembled monolayers on CYTOP polymer. *ACS Appl. Mater. Inter.* **11**, 21372–21380 (2019)
16. H. Eto, H.G. Franquelim, M. Heymann, P. Schwille, Membrane-coated 3D architectures for bottom-up synthetic biology. *Soft Matt.* **17**, 5456–5466 (2021)
17. F.A. Heberle, G.W. Feigenson, Phase separation in lipid membranes. *Cold Spring Harb. Perspect. Biol.* **3** (2011). <https://doi.org/10.1101/cshperspect.a004630>
18. E. Sezgin, I. Levental, S. Mayor, C. Eggeling, The mystery of membrane organization: composition, regulation and roles of lipid rafts. *Nat. Rev. Mol. Cell. Biol.* **18**, 361–374 (2017)

19. R.F.M. de Almeida, A. Fedorov, M. Prieto, Sphingomyelin/phosphatidylcholine/cholesterol phase diagram: boundaries and composition of lipid rafts. *Biophys J.* **85**, 2406–2416 (2003)
20. K. Bacia, P. Schwille, T. Kurzchalia, Sterol structure determines the separation of phases and the curvature of the liquid-ordered phase in model membranes. *Proc. Natl. Acad. Sci. USA* **102**, 3272–3277 (2005)
21. A.J. García-Sáez, S. Chiantia, P. Schwille, Effect of line tension on the lateral organization of lipid membranes*. *J. Biol. Chem.* **282**, 33537–33544 (2007)
22. A. Choucair, M. Chakrapani, B. Chakravarthy, J. Katsaras, L.J. Johnston, Preferential accumulation of A β (1–42) on gel phase domains of lipid bilayers: an AFM and fluorescence study. *Biochimica et Biophysica Acta (BBA). Biomemb.* **1768**, 146–154 (2007)
23. K. Weise, G. Triola, L. Brunsveld, H. Waldmann, R. Winter, Influence of the lipidation motif on the partitioning and association of N-Ras in model membrane subdomains. *J. Am. Chem. Soc.* **131**, 1557–1564 (2009)
24. S.K. Vogel, F. Greiss, A. Khmelinskaia, P. Schwille, Control of lipid domain organization by a biomimetic contractile actomyosin cortex. *Elife* **6** (2017). <https://doi.org/10.7554/eLife.24350>
25. A. van Oudenaarden, S.G. Boxer, Brownian ratchets: molecular separations in lipid bilayers supported on patterned arrays. *Science* **285**, 1046–1048 (1999)
26. S. Daniel, A.J. Diaz, K.M. Martinez, B.J. Bench, F. Albertorio, P.S. Cremer, Separation of membrane-bound compounds by solid-supported bilayer electrophoresis. *J. Am. Chem. Soc.* **129**, 8072–8073 (2007)
27. A. Wixforth, C. Strobl, C. Gauer, A. Toegl, J. Scriba, Z. Guttenberg, Acoustic manipulation of small droplets. *Anal. Bioanal. Chem.* **379**, 982–991 (2004)
28. M. Hennig, J. Neumann, A. Wixforth, J.O. Rädler, M.F. Schneider, Dynamic patterns in a supported lipid bilayer driven by standing surface acoustic waves. *Lab. Chip.* **9**, 3050–3053 (2009)
29. J. Neumann, M. Hennig, A. Wixforth, S. Manus, J.O. Rädler, M.F. Schneider, Transport, separation, and accumulation of proteins on supported lipid bilayers. *Nano. Lett.* **10**, 2903–2908 (2010)
30. K. Yasuhara, Y. Sasaki, J.-I. Kikuchi, A photo-responsive cholesterol capable of inducing a morphological transformation of the liquid-ordered microdomain in lipid bilayers. *Colloid Polym. Sci.* **286**, 1675 (2008)
31. T. Hamada, R. Sugimoto, T. Nagasaki, M. Takagi, Photochemical control of membrane raft organization. *Soft Matt.* **7**, 220–224 (2011)
32. J.A. Frank, H.G. Franquelim, P. Schwille, D. Trauner, Optical control of lipid rafts with photoswitchable ceramides. *J. Am. Chem. Soc.* **138**, 12981–12986 (2016)
33. P. Glock, J. Broichhagen, S. Kretschmer, P. Blumhardt, J. Mücksch, D. Trauner, P. Schwille, Optical control of a biological reaction-diffusion system. *Angew Chem. Int. Ed. Engl.* **57**, 2362–2366 (2018)
34. S.T. Glantz, E.E. Berlew, Z. Jaber, B.S. Schuster, K.H. Gardner, B.Y. Chow, Directly light-regulated binding of RGS-LOV photoreceptors to anionic membrane phospholipids. *Proc. Natl. Acad. Sci. USA* **115**, E7720–E7727 (2018)
35. H. Jia, L. Kai, M. Heymann, D.A. García-Soriano, T. Härtel, P. Schwille, Light-induced printing of protein structures on membranes in vitro. *Nano. Lett.* **18**, 7133–7140 (2018)
36. P.W.K. Rothmund, Folding DNA to create nanoscale shapes and patterns. *Nature* **440**, 297–302 (2006)
37. J. Schnitzbauer, M.T. Strauss, T. Schlichthaerle, F. Schueder, R. Jungmann, Super-resolution microscopy with DNA-PAINT. *Nat. Protoc.* **12**, 1198–1228 (2017)
38. G. Tikhomirov, P. Petersen, L. Qian, Fractal assembly of micrometre-scale DNA origami arrays with arbitrary patterns. *Nature* **552**, 67–71 (2017)
39. S.F.J. Wickham, A. Auer, J. Min, N. Ponnuswamy, J.B. Woehrstein, F. Schueder, M.T. Strauss, J. Schnitzbauer, B. Nathwani, Z. Zhao, S.D. Perrault, J. Hahn, S. Lee, M.M. Bastings, S.W. Helmig, A.L. Kodal, P. Yin, R. Jungmann, W.M. Shih, Complex multicomponent patterns rendered on a 3D DNA-barrel pegboard. *Nat. Commun.* **11**, 5768 (2020)

40. W. Wang, D.S. Arias, M. Deserno, X. Ren, R.E. Taylor, Emerging applications at the interface of DNA nanotechnology and cellular membranes: Perspectives from biology, engineering, and physics. *APL Bioeng.* **4**, 041507 (2020)
41. J.R. Burns, K. Göpfrich, J.W. Wood, V.V. Thacker, E. Stulz, U.F. Keyser, S. Howorka, Lipid-bilayer-spanning DNA nanopores with a bifunctional porphyrin anchor. *Angew Chem. Int. Ed. Engl.* **52**, 12069–12072 (2013)
42. A. Czogalla, E.P. Petrov, D.J. Kauert, V. Uzunova, Y. Zhang, R. Seidel, P. Schwille, Switchable domain partitioning and diffusion of DNA origami rods on membranes. *Farad. Discuss.* **161**, 31–43; discussion 113–150 (2013)
43. S. Krishnan, D. Ziegler, V. Arnaut, T.G. Martin, K. Kapsner, K. Henneberg, A.R. Bausch, H. Dietz, F.C. Simmel, Molecular transport through large-diameter DNA nanopores. *Nat. Commun.* **7**, 12787 (2016)
44. M.W. Grome, Z. Zhang, F. Pincet, C. Lin, Vesicle tubulation with self-assembling DNA nanosprings. *Angew. Chem. Int. Ed. Engl.* **57**, 5330–5334 (2018)
45. S.D. Perrault, W.M. Shih, Virus-inspired membrane encapsulation of DNA nanostructures to achieve in vivo stability. *ACS Nano.* **8**, 5132–5140 (2014)
46. Y. Suzuki, M. Endo, H. Sugiyama, Lipid-bilayer-assisted two-dimensional self-assembly of DNA origami nanostructures. *Nat. Commun.* **6**, 8052 (2015)
47. A. Johnson-Buck, S. Jiang, H. Yan, N.G. Walter, DNA–cholesterol barges as programmable membrane-exploring agents. *ACS Nano.* **8**, 5641–5649 (2014)
48. J. List, M. Weber, F.C. Simmel, Hydrophobic actuation of a DNA origami bilayer structure. *Angew Chem. Int. Ed. Engl.* **53**, 4236–4239 (2014)
49. M. Langecker, V. Arnaut, T.G. Martin, J. List, S. Renner, M. Mayer, H. Dietz, F.C. Simmel, Synthetic lipid membrane channels formed by designed DNA nanostructures. *Science* **338**, 932–936 (2012)
50. K. Göpfrich, C.-Y. Li, M. Ricci, S.P. Bhamidimarri, J. Yoo, B. Gyenes, A. Ohmann, M. Winterhalter, A. Aksimentiev, U.F. Keyser, Large-conductance transmembrane Porin made from DNA origami. *ACS Nano* **10**, 8207–8214 (2016)
51. A. Fragasso, N. De Franceschi, P. Stömmmer, E.O. van der Sluis, H. Dietz, C. Dekker, Reconstitution of ultrawide DNA origami pores in liposomes for transmembrane transport of macromolecules. *ACS Nano* (2021). <https://doi.org/10.1021/acsnano.1c01669>
52. A. Czogalla, D.J. Kauert, H.G. Franquelim, V. Uzunova, Y. Zhang, R. Seidel, P. Schwille, Amphipathic DNA origami nanoparticles to scaffold and deform lipid membrane vesicles. *Angew Chem. Int. Ed. Engl.* **54**, 6501–6505 (2015)
53. Z. Zhang, Y. Yang, F. Pincet, M.C. Llaguno, C. Lin, Placing and shaping liposomes with reconfigurable DNA nanocages. *Nat. Chem.* **9**, 653–659 (2017)
54. H.G. Franquelim, A. Khmelinskaia, J.-P. Sobczak, H. Dietz, P. Schwille, Membrane sculpting by curved DNA origami scaffolds. *Nat. Commun.* **9**, 811 (2018)
55. M.W. Grome, Z. Zhang, C. Lin, Stiffness and membrane anchor density modulate DNA-nanospring-induced vesicle tubulation. *ACS Appl. Mater. Inter.* **11**, 22987–22992 (2019)
56. H.G. Franquelim, H. Dietz, P. Schwille, Reversible membrane deformations by straight DNA origami filaments. *Soft Matt.* **17**, 276–287 (2021)
57. A. Czogalla, D.J. Kauert, R. Seidel, P. Schwille, E.P. Petrov, DNA origami nanoneedles on freestanding lipid membranes as a tool to observe isotropic-nematic transition in two dimensions. *Nano Lett.* **15**, 649–655 (2015)
58. A. Khmelinskaia, H.G. Franquelim, E.P. Petrov, P. Schwille, Effect of anchor positioning on binding and diffusion of elongated 3D DNA nanostructures on lipid membranes. *J. Phys. D Appl. Phys.* **49**, 194001 (2016)
59. A. Khmelinskaia, J. Mücksch, E.P. Petrov, H.G. Franquelim, P. Schwille, Control of Membrane Binding and Diffusion of Cholesteryl-Modified DNA Origami Nanostructures by DNA Spacers. *Langmuir* **34**, 14921–14931 (2018)
60. M. Endo, T. Sugita, A. Rajendran, Y. Katsuda, T. Emura, K. Hidaka, H. Sugiyama, Two-dimensional DNA origami assemblies using a four-way connector. *Chem. Comm.* **47**, 3213–3215 (2011)

61. S. Kocabey, S. Kempter, J. List, Y. Xing, W. Bae, D. Schiffels, W.M. Shih, F.C. Simmel, T. Liedl, Membrane-assisted growth of DNA origami nanostructure arrays. *ACS Nano*. **9**, 3530–3539 (2015)
62. N. Avakyan, J.W. Conway, H.F. Sleiman, Long-range ordering of blunt-ended DNA tiles on supported lipid bilayers. *J. Am. Chem. Soc.* **139**, 12027–12034 (2017)
63. Y. Sato, M. Endo, M. Morita, M. Takinoue, H. Sugiyama, S. Murata, S.-I.M. Nomura, Y. Suzuki, Environment-dependent self-assembly of DNA origami lattices on phase-separated lipid membranes. *Adv. Mater. Interfaces* **5**, 1800437 (2018)
64. M.P. Murrell, M.L. Gardel, F-actin buckling coordinates contractility and severing in a biomimetic actomyosin cortex. *Proc. Natl. Acad. Sci. USA* **109**, 20820–20825 (2012)
65. S.K. Vogel, Z. Petrasek, F. Heinemann, P. Schwille, Myosin motors fragment and compact membrane-bound actin filaments. *Elife* **2**, e00116 (2013)
66. D.V. Köster, K. Husain, E. Iljazi, A. Bhat, P. Bieling, R.D. Mullins, M. Rao, S. Mayor, Actomyosin dynamics drive local membrane component organization in an in vitro active composite layer. *Proc. Natl. Acad. Sci. USA* **113**, E1645–E1654 (2016)
67. K. Göpfrich, I. Platzman, J.P. Spatz, Mastering complexity: towards bottom-up construction of multifunctional eukaryotic synthetic cells. *Trends Biotechnol.* **36**, 938–951 (2018)
68. G. Saper, H. Hess, Synthetic systems powered by biological molecular motors. *Chem. Rev.* **120**, 288–309 (2020)
69. C. Reuther, R. Catalano, A. Salhotra, V. Vemula, T. Korten, S. Diez, A. Månsson, Comparison of actin- and microtubule-based motility systems for application in functional nanodevices. *New. J. Phys.* **23**, 075007 (2021)
70. I.V. Surovtsev, C. Jacobs-Wagner, Subcellular organization: a critical feature of bacterial cell replication. *Cell* **172**, 1271–1293 (2018)
71. B. Ramm, T. Heermann, P. Schwille, The E. coli MinCDE system in the regulation of protein patterns and gradients. *Cell. Mol. Life Sci.* **76**, 4245–4273 (2019)
72. P.A. de Boer, R.E. Crossley, L.I. Rothfield, A division inhibitor and a topological specificity factor coded for by the minicell locus determine proper placement of the division septum in E. coli. *Cell* **56**, 641–649 (1989)
73. Z. Hu, J. Lutkenhaus, Topological regulation of cell division in Escherichia coli involves rapid pole to pole oscillation of the division inhibitor MinC under the control of MinD and MinE. *Mol. Microbiol.* **34**, 82–90 (1999)
74. D.M. Raskin, P.A. de Boer, Rapid pole-to-pole oscillation of a protein required for directing division to the middle of Escherichia coli. *Proc. Natl. Acad. Sci. USA* **96**, 4971–4976 (1999)
75. K. Kruse, A dynamic model for determining the middle of Escherichia coli. *Biophys. J.* **82**, 618–627 (2002)
76. K.C. Huang, Y. Meir, N.S. Wingreen, Dynamic structures in Escherichia coli: spontaneous formation of MinE rings and MinD polar zones. *Proc. Natl. Acad. Sci. USA* **100**, 12724–12728 (2003)
77. M. Loose, E. Fischer-Friedrich, J. Ries, K. Kruse, P. Schwille, Spatial regulators for bacterial cell division self-organize into surface waves in vitro. *Science* **320**, 789–792 (2008)
78. F. Wu, B.G.C. van Schie, J.E. Keymer, C. Dekker, Symmetry and scale orient Min protein patterns in shaped bacterial sculptures. *Nat. Nanotechnol.* **10**, 719–726 (2015)
79. J. Halatek, E. Frey, Rethinking pattern formation in reaction–diffusion systems. *Nat. Phys.* **14**, 507–514 (2018)
80. A.G. Vecchiarelli, M. Li, M. Mizuuchi, L.C. Hwang, Y. Seol, K.C. Neuman, K. Mizuuchi, Membrane-bound MinDE complex acts as a toggle switch that drives Min oscillation coupled to cytoplasmic depletion of MinD. *Proc. Natl. Acad. Sci. USA* **113**, E1479–E1488 (2016)
81. P. Glock, B. Ramm, T. Heermann, S. Kretschmer, J. Schweizer, J. Mücksch, G. Alagöz, P. Schwille, Stationary patterns in a two-protein reaction-diffusion system. *ACS Synth. Biol.* **8**, 148–157 (2019)
82. A.G. Vecchiarelli, M. Li, M. Mizuuchi, K. Mizuuchi, Differential affinities of MinD and MinE to anionic phospholipid influence Min patterning dynamics in vitro. *Mol. Microbiol.* **93**, 453–463 (2014)

83. A. Martos, Z. Petrasek, P. Schwille, Propagation of MinCDE waves on free-standing membranes. *Environ. Microbiol.* **15**, 3319–3326 (2013)
84. L.L. Lackner, D.M. Raskin, P.A.J. de Boer, ATP-dependent interactions between *Escherichia coli* Min proteins and the phospholipid membrane in vitro. *J. Bacteriol.* **185**, 735–749 (2003)
85. P. Glock, F. Brauns, J. Halatek, E. Frey, P. Schwille, Design of biochemical pattern forming systems from minimal motifs. *Elife* **8** (2019). <https://doi.org/10.7554/eLife.48646>
86. S. Kretschmer, T. Heermann, A. Tassinari, P. Glock, P. Schwille, Increasing MinD's membrane affinity yields standing wave oscillations and functional gradients on flat membranes. *ACS Synth. Biol.* **10**, 939–949 (2021)
87. B. Ramm, P. Glock, J. Mücksch, P. Blumhardt, D.A. García-Soriano, M. Heymann, P. Schwille, The MinDE system is a generic spatial cue for membrane protein distribution in vitro. *Nat. Comm.* **9**, 3942 (2018)
88. Y.-L. Shih, L.-T. Huang, Y.-M. Tu, B.-F. Lee, Y.-C. Bau, C.Y. Hong, H.-L. Lee, Y.-P. Shih, M.-F. Hsu, Z.-X. Lu, J.-S. Chen, L. Chao, Active transport of membrane components by self-organization of the min proteins. *Biophys J.* **116**, 1469–1482 (2019)
89. B. Ramm, A. Goychuk, A. Khmelinskaia, P. Blumhardt, H. Eto, K.A. Ganzinger, E. Frey, P. Schwille, A diffusiophoretic mechanism for ATP-driven transport without motor proteins. *Nat. Phys.* **17**, 850–858 (2021)
90. K.-T. Park, W. Wu, K.P. Battaile, S. Lovell, T. Holyoak, J. Lutkenhaus, The Min oscillator uses MinD-dependent conformational changes in MinE to spatially regulate cytokinesis. *Cell* **146**, 396–407 (2011)
91. S. Marbach, L. Bocquet, Osmosis, from molecular insights to large-scale applications. *Chem. Soc. Rev.* **48**, 3102–3144 (2019)
92. R. Golestanian, Phoretic Active Matter. *arXiv [cond-mat.soft]* (2019)
93. F. Hong, F. Zhang, Y. Liu, H. Yan, DNA origami: scaffolds for creating higher order structures. *Chem. Rev.* **117**, 12584–12640 (2017)
94. Y. He, T. Ye, M. Su, C. Zhang, A.E. Ribbe, W. Jiang, C. Mao, Hierarchical self-assembly of DNA into symmetric supramolecular polyhedra. *Nature* **452**, 198–201 (2008)
95. R. Jungmann, M. Scheible, A. Kuzyk, G. Pardatscher, C.E. Castro, F.C. Simmel, DNA origami-based nanoribbons: assembly, length distribution, and twist. *Nanotech.* **22**, 275301 (2011)
96. W. Liu, H. Zhong, R. Wang, N.C. Seeman, Crystalline two-dimensional DNA-origami arrays. *Angew Chem. Int. Ed. Engl.* **50**, 264–267 (2011)
97. H. Yan, T.H. LaBean, L. Feng, J.H. Reif, Directed nucleation assembly of DNA tile complexes for barcode-patterned lattices. *Proc. Natl. Acad. Sci. USA* **100**, 8103–8108 (2003)
98. T. Gerling, K.F. Wagenbauer, A.M. Neuner, H. Dietz, Dynamic DNA devices and assemblies formed by shape-complementary, non-base pairing 3D components. *Science* **347**, 1446–1452 (2015)
99. K.F. Wagenbauer, C. Sigl, H. Dietz, Gigadalton-scale shape-programmable DNA assemblies. *Nature* **552**, 78–83 (2017)
100. K. Tapio, I. Bald, The potential of DNA origami to build multifunctional materials. *Multifunct. Mater.* **3**, 032001 (2020)
101. W. Engelen, H. Dietz, Advancing biophysics using DNA origami. *Ann. Rev. Biophys.* **50**, 469–492 (2021)
102. J. Fu, M. Liu, Y. Liu, N.W. Woodbury, H. Yan, Interenzyme substrate diffusion for an enzyme cascade organized on spatially addressable DNA nanostructures. *J. Am. Chem. Soc.* **134**, 5516–5519 (2012)
103. J. Fu, Y.R. Yang, A. Johnson-Buck, M. Liu, Y. Liu, N.G. Walter, N.W. Woodbury, H. Yan, Multi-enzyme complexes on DNA scaffolds capable of substrate channelling with an artificial swinging arm. *Nat. Nanotechnol.* **9**, 531–536 (2014)
104. Z. Zhao, J. Fu, S. Dhakal, A. Johnson-Buck, M. Liu, T. Zhang, N.W. Woodbury, Y. Liu, N.G. Walter, H. Yan, Nanocaged enzymes with enhanced catalytic activity and increased stability against protease digestion. *Nat. Commun.* **7**, 10619 (2016)

105. N.D. Derr, B.S. Goodman, R. Jungmann, A.E. Leschziner, W.M. Shih, S.L. Reck-Peterson, Tug-of-war in motor protein ensembles revealed with a programmable DNA origami scaffold. *Science* **338**, 662–665 (2012)
106. M. Iwaki, S.F. Wickham, K. Ikezaki, T. Yanagida, W.M. Shih, A programmable DNA origami nanospring that reveals force-induced adjacent binding of myosin VI heads. *Nat. Commun.* **7**, 13715 (2016)
107. J.J. Funke, P. Ketterer, C. Lieleg, S. Schunter, P. Korber, H. Dietz, Uncovering the forces between nucleosomes using DNA origami. *Sci. Adv.* **2**, e1600974 (2016)
108. P.C. Nickels, B. Wünsch, P. Holzmeister, W. Bae, L.M. Kneer, D. Grohmann, P. Tinnefeld, T. Liedl, Molecular force spectroscopy with a DNA origami-based nanoscopic force clamp. *Science* **354**, 305–307 (2016)
109. K. Kramm, T. Schröder, J. Gouge, A.M. Vera, K. Gupta, F.B. Heiss, T. Liedl, C. Engel, I. Berger, A. Vannini, P. Tinnefeld, D. Grohmann, DNA origami-based single-molecule force spectroscopy elucidates RNA Polymerase III pre-initiation complex stability. *Nat. Commun.* **11**, 2828 (2020)
110. A. Shaw, V. Lundin, E. Petrova, F. Fördös, E. Benson, A. Al-Amin, A. Herland, A. Blokzijl, B. Högberg, A.I. Teixeira, Spatial control of membrane receptor function using ligand nanocalipers. *Nat. Methods* **11**, 841–846 (2014)
111. R. Veneziano, T.J. Moyer, M.B. Stone, E.-C. Wamhoff, B.J. Read, S. Mukherjee, T.R. Shepherd, J. Das, W.R. Schief, D.J. Irvine, M. Bathe, Role of nanoscale antigen organization on B-cell activation probed using DNA origami. *Nat. Nanotechnol.* **15**, 716–723 (2020)
112. R.M.L. Berger, J.M. Weck, S.M. Kempe, O. Hill, T. Liedl, J.O. Rädler, C. Monzel, A. Heuer-Jungemann, Nanoscale FasL organization on DNA origami to decipher apoptosis signal activation in cells. *Small* **17**, e2101678 (2021)
113. J. Hellmeier, R. Platzer, A.S. Eklund, T. Schlichthaerle, A. Karner, V. Motsch, M.C. Schneider, E. Kurz, V. Bamieh, M. Brameshuber, J. Preiner, R. Jungmann, H. Stockinger, G.J. Schütz, J.B. Huppa, E. Sevcsik, DNA origami demonstrate the unique stimulatory power of single pMHCs as T cell antigens. *Proc. Natl. Acad. Sci. USA* **118** (2021). <https://doi.org/10.1073/pnas.2016857118>
114. N. Kern, R. Dong, S.M. Douglas, R.D. Vale, M.A. Morrissey, Tight nanoscale clustering of Fcγ receptors using DNA origami promotes phagocytosis. *Elife* **10** (2021). <https://doi.org/10.7554/eLife.68311>
115. U. Bezeljak, H. Loya, B. Kaczmarek, T.E. Saunders, M. Loose, Stochastic activation and bistability in a Rab GTPase regulatory network. *Proc. Natl. Acad. Sci. USA* **117**, 6540–6549 (2020)
116. A.A. Hyman, C.A. Weber, F. Jülicher, Liquid-liquid phase separation in biology. *Ann. Rev. Cell Dev. Biol.* **30**, 39–58 (2014)
117. S. Dey, C. Fan, K.V. Gothelf, J. Li, C. Lin, L. Liu, N. Liu, M.A.D. Nijenhuis, B. Saccà, F.C. Simmel, H. Yan, P. Zhan, DNA origami. *Nat. Rev. Methods Prim.* **1**, 1–24 (2021)
118. K. Zieske, G. Chwastek, P. Schwille, Protein patterns and oscillations on lipid monolayers and in microdroplets. *Angew Chem. Int. Ed. Engl.* **55**, 13455–13459 (2016)
119. T. Litschel, B. Ramm, R. Maas, M. Heymann, P. Schwille, Beating vesicles: encapsulated protein oscillations cause dynamic membrane deformations. *Angew Chem. Int. Ed. Engl.* **57**, 16286–16290 (2018)
120. E. Godino, J.N. López, D. Foschepoth, C. Cleij, A. Doerr, C.F. Castellà, C. Danelon, De novo synthesized Min proteins drive oscillatory liposome deformation and regulate FtsA-FtsZ cytoskeletal patterns. *Nat. Commun.* **10**, 4969 (2019)
121. A.E. Marras, L. Zhou, H.-J. Su, C.E. Castro, Programmable motion of DNA origami mechanisms. *Proc. Natl. Acad. Sci. USA* **112**, 713–718 (2015)
122. A. Kuzyk, Y. Yang, X. Duan, S. Stoll, A.O. Govorov, H. Sugiyama, M. Endo, N. Liu, A light-driven three-dimensional plasmonic nanosystem that translates molecular motion into reversible chiroptical function. *Nat. Commun.* **7**, 10591 (2016)
123. C. Lee, J.Y. Lee, D.-N. Kim, Polymorphic design of DNA origami structures through mechanical control of modular components. *Nat. Commun.* **8**, 2067 (2017)

124. A. Kuzyk, M.J. Urban, A. Idili, F. Ricci, N. Liu, Selective control of reconfigurable chiral plasmonic metamolecules. *Sci. Adv.* **3**, e1602803 (2017)
125. J. Song, Z. Li, P. Wang, T. Meyer, C. Mao, Y. Ke, Reconfiguration of DNA molecular arrays driven by information relay. *Science* **357** (2017). <https://doi.org/10.1126/science.aan3377>
126. E. Kopperger, J. List, S. Madhira, F. Rothfischer, D.C. Lamb, F.C. Simmel, A self-assembled nanoscale robotic arm controlled by electric fields. *Science* **359**, 296–301 (2018)
127. F.N. Gür, S. Kempfer, F. Schueder, C. Sikeler, M.J. Urban, R. Jungmann, P.C. Nickels, T. Liedl, Double- to single-strand transition induces forces and motion in DNA origami nanostructures. *Adv. Mater.*, e2101986 (2021)
128. B. Ramm, P. Glock, P. Schwille, In vitro reconstitution of self-organizing protein patterns on supported lipid bilayers. *J. Vis. Exp.* (2018). <https://doi.org/10.3791/58139>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

