SpringerBriefs in Applied Sciences and Technology
PoliMI SpringerBriefs

Francesco Amigoni *Editor*

Special Topics in Information Technology

POLITECNICO
MILANO 1863

OPEN ACCESS

SpringerBriefs in Applied Sciences and Technology

PoliMI SpringerBriefs

Springer, in cooperation with Politecnico di Milano, publishes the PoliMI SpringerBriefs, concise summaries of cutting-edge research and practical applications across a wide spectrum of fields. Featuring compact volumes of 50 to 125 (150 as a maximum) pages, the series covers a range of contents from professional to academic in the following research areas carried out at Politecnico:

- Aerospace Engineering
- Bioengineering
- Electrical Engineering
- Energy and Nuclear Science and Technology
- Environmental and Infrastructure Engineering
- Industrial Chemistry and Chemical Engineering
- Information Technology
- Management, Economics and Industrial Engineering
- Materials Engineering
- Mathematical Models and Methods in Engineering
- Mechanical Engineering
- Structural Seismic and Geotechnical Engineering
- Built Environment and Construction Engineering
- Physics
- Design and Technologies
- Urban Planning, Design, and Policy

http://www.polimi.it

Francesco Amigoni
Editor

Special Topics in Information Technology

Springer

Editor
Francesco Amigoni
DEIB
Politecnico di Milano
Milan, Italy

ISSN 2191-530X ISSN 2191-5318 (electronic)
SpringerBriefs in Applied Sciences and Technology
ISSN 2282-2577 ISSN 2282-2585 (electronic)
PoliMI SpringerBriefs
ISBN 978-3-031-51499-9 ISBN 978-3-031-51500-2 (eBook)
https://doi.org/10.1007/978-3-031-51500-2

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Paper in this product is recyclable.

Preface

This volume collects the best results from students who obtained a Ph.D. in Information Technology (IT) at the Dipartimento di Elettronica, Informazione e Bioingegneria of the Politecnico di Milano, during the academic year 2022–2023. Among more than 65 students who graduated in 2022–2023, the IT Ph.D. Board selected the authors of the following chapters and awarded them the IT Ph.D. Award.

As the topics covered in this volume highlight, the IT Ph.D. Program covers a wide range of domains, according to the broad articulation of the program in the areas of computer science and engineering, electronics, systems and control, and telecommunications. The theoretical-oriented and application-oriented contributions illustrated in the following emphasize the interdisciplinary nature of IT.

Doctoral studies in the IT Ph.D. Program pursue excellence in research through the development of innovative cutting-edge methodologies, methods, and technologies, and aim at preparing generations of young researchers and professionals that will shape future innovation both in academia and in industry, as the authors of this volume will undoubtedly do. Overall, the volume gives an overview of some of the most exciting research trends in IT followed at the Politecnico di Milano, presenting them in a mostly easy-to-read format that can be enjoyed also by non-specialists.

Milan, Italy
October 2022

Francesco Amigoni

Contents

Computer Science and Engineering

Reducing the Gap Between Theory and Applications in Algorithmic Bayesian Persuasion

Matteo Castiglioni

Abstract This work focuses on the following question: *is it possible to influence the behavior of self-interested agents through the strategic provision of information*? This 'sweet talk' is ubiquitous among all sorts of economics and non-economics activities. In this work, we model these multi-agent systems as games between an informed sender and one or multiple receivers. We study the computational problem faced by an informed sender that wants to use his information advantage to influence rational receivers with the partial disclosure of information. In particular, the sender faces an information structure design problem that amounts to deciding 'who gets to know what'. Bayesian persuasion provides a formal framework to model these settings as asymmetric-information games. In recent years, much attention has been given to Bayesian persuasion in the economics and artificial intelligence communities due also to the applicability of this framework to a large class of scenarios like online advertising, voting, traffic routing, recommendation systems, security, and product marketing. However, there is still a large gap between the theoretical study of information in games and its applications in real-world scenarios. This work contributes to close this gap along two directions. First, we study the persuasion problem in real-world scenarios, focusing on voting, routing, and auctions. While the Bayesian persuasion framework can be applied to all these settings, the algorithmic problem of designing optimal information disclosure polices introduces computational challenges related to the specific problem under study. Our goal is to settle the complexity of computing optimal sender's strategies, showing when an optimal strategy can be implemented efficiently. Then, we relax stringent assumptions that limit the applicability of the Bayesian persuasion framework in practice. In particular, the classical model assumes that the sender has perfect knowledge of the receiver's utility. We remove this assumption initiating the study of an online version of the persuasion problem. This is the first step in designing adaptive information disclosure policies that deal with the uncertainty intrinsic in all real-world applications.

M. Castiglioni (✉)
Politecnico di Milano, Piazza Leonardo da Vinci 32, Milan, Italy
e-mail: matteo.castiglioni@polimi.it

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_1

1 Introduction

This work considers the following question: *is it possible to influence the behavior of self-interested agents through the strategic provision of information*? This 'sweet talk' is ubiquitous among all sorts of economic activities, and it was famously attributed to 30% of the GDP in the United States [3]. Moreover, information is the foundation of any democratic election, as it allows voters for better choices. In many settings, uninformed voters have to rely on inquiries of third party entities to make their decision. With the advent of modern media environments, malicious actors have unprecedented opportunities to garble this information and influence the outcome of the election via misinformation and fake news [1]. Reaching voters with targeted messages has never been easier. As another example, consider a multi-agent routing problem in which agents seek to minimize their own costs selfishly. In real-world problems, the state of the network may be uncertain, and not known to its users (e.g., drivers may not be aware of road works and accidents in a road network). A central authority or a navigation app may mitigate inefficiencies and reduce the social cost providing players with partial information about the state of the network.

Bayesian persuasion [28] provides a framework to model the problem faced by an informed sender trying to influence the behavior of self-interested receivers. In particular, the sender faces an information structure design problem which amounts to deciding 'who gets to know what' about some exogenous parameters collectively termed state of nature. Since the seminal work of [28], a large attention has been given to the Bayesian persuasion framework in the economics and artificial intelligence community due also to the applicability of this framework to a large class of scenarios like online advertising [7, 8, 12, 27], voting [2, 25], traffic routing [11, 33], recommendation systems [29], security [31, 35], and product marketing [5, 13]. However, there is still a large gap between the theoretical study of information in games and its applications in real-world scenarios. This work contributes to close this gap along two directions. First, we study the Bayesian persuasion framework in real-world scenarios, focusing on voting, routing, and auctions. While the Bayesian persuasion framework can be applied to all these settings, the algorithmic problem of designing optimal information disclosure polices introduces computational challenges related to the specific problem under study. Then, we relax stringent assumptions that limit the applicability of the classical bayesian persuasion framework in practice. In particular, one of the most limiting assumption is, arguably, that the sender is required to know the receiver's utility function to compute an optimal signaling scheme. We remove this assumption by studying a repeated Bayesian persuasion problem in an online learning framework where, at each round, the receiver's type is adversarially chosen from a finite set of types. This is the first step in designing adaptive information disclosure policies that deals with the uncertainty intrinsic in all real-world applications.

2 The Bayesian Persuasion Framework

Bayesian persuasion [28] studies the problem faced by an informed agent (the *sender*) trying to influence the behavior of other self-interested agents (the *receivers*) via the partial disclosure of payoff-relevant information. Agents' payoffs are determined by the actions played by the receivers and by an exogenous parameter represented as a *state of nature*, which is drawn by a known prior probability distribution and observed by the sender only. The sender commits to a public randomized information-disclosure policy, which is customarily called *signaling scheme*. In particular, it defines how the sender should send signals to the receivers. Depending on the application various types of signaling schemes have been introduced to represent the possible communication constraints between the sender and the receivers. In a private signaling scheme, the sender can use a private communication channel per receiver, in a public signaling scheme the sender can use a single communication channel for all the receivers, while we introduce semi-public signaling schemes in which the sender can use a single communication channel for a subset of the receivers.

Arguably, one of the most severe obstacle to the application of the classical bayesian persuasion model by [28] to real-world scenarios is that the sender must know exactly the receiver's utility function to compute an optimal signaling scheme. This assumption is unreasonable in practice. However, only recently some works tries to relax this assumption. In particular, [6] study a game with a single receiver and binary-actions in which the sender does not know the receiver utility, focusing on the problem of designing a signaling scheme that perform well for each possible receiver's utility. Zu et al. [37] relax the perfect knowledge assumption assuming that the sender and the receiver do not know the prior distribution over the states of nature. They study the problem of computing a sequence of persuasive signaling schemes that achieve small regret with respect to the optimal signaling scheme with the knowledge of the prior distribution. Bernasconi et al. [10] extends the analysis to sequential settings. In this work, we follow a different approach and we deal with uncertainty about the receiver's utility by framing the Bayesian persuasion problem in an online learning framework [9]. In particular, we advance the state of the art on algorithmic Bayesian persuasion along two directions. First, we study Bayesian persuasion in games with structure, focusing on voting, routing, and auctions. Then, we initiate the study of Bayesian persuasion with payoff uncertainly.

3 Persuading in Election

In this section, we study Bayesian persuasion in voting scenarios. Information is the foundation of any democratic election, as it allows voters for better choices. In many settings, uninformed voters have to rely on inquiries of third-party entities to make their decision. For example, in most trials, jurors are not given the possibility of choosing which tests to perform during the investigation or which questions are

asked to witnesses. They have to rely on the prosecutor's investigation and questions. The same happens in elections, in which voters gather information from third-party sources. Hence, we pose the question: *can a malicious actor influence the outcome of a voting process only by the provision of information to voters who update their beliefs rationally?* We study majority voting, plurality voting and district-based elections, showing a sharp contrast in term of efficiency in manipulating elections and computational tractability between the case in which private signals are allowed and the more restrictive setting in which only public signals are allowed. In particular, we show that it is possible to compute an optimal private signaling scheme in polynomial time in all the elections that we considered, while the problem of approximating the optimal public signaling scheme is NP-hard even for majority voting. Moreover, we show that, assuming the Exponential Time Hypothesis (ETH), the problem of approximating the optimal public signaling scheme in majority voting requires quasi-polynomial time even relaxing persuasiveness. In doing so, we provide some insights on the complexity of general persuasion problems, such as the characterization of bi-criteria approximations in public signaling problems. A complete version of our results appears in [14, 15, 19].

4 Persuading in Routing

The study of how to influence traffic congestion has receive an increasing attention in recent years [22, 30, 33, 34]. *Network congestion games*, where players seek to minimize their own costs selfishly, are a canonical example of a setting where externalities may induce socially inefficient outcomes [32]. In real-world problems, the state of the network may be uncertain, and not known to its users (e.g., drivers may not be aware of road works and accidents in a road network). This setting is modeled via *Bayesian network congestion games* (BNCGs). Here, we explore how information can be used to reduce the social cost in routing games. In particular, we study Bayesian games with atomic players, where network vagaries are modeled via a (random) state of nature which determines the costs incurred by the players. We investigate whether it is possible to efficiently compute optimal, i.e., minimizing the social cost, *ex ante* persuasive signaling schemes in BNCGs, showing that symmetry is a crucial property for its solution. We focus on the notion of *ex ante persuasiveness*, as introduced by [24, 36], where the receivers are incentivized to follow the sender's recommendations having observed only the signaling scheme. We show that an optimal *ex ante* persuasive signaling scheme can be computed in polynomial time in symmetric BNCGs (i.e., where all the players share the same source and destination pair) with edge costs defined as affine functions of the edge congestion. Then, we show that *symmetry* is a crucial property for efficient signaling by proving that it is NP-hard to compute an optimal *ex ante* persuasive signaling scheme in *asymmetric* BNCGs. Our reduction proves an even stronger hardness result, as it works for non-Bayesian singleton congestion games with affine costs, which is arguably the simplest class of asymmetric congestion games. Furthermore, in such

setting, a solution to our problem is an optimal coarse correlated equilibrium and, thus, computing optimal coarse correlated equilibria is NP-hard. A complete version of our results appears in [17].

5 Persuading in Auctions

In this section, we study persuasion in posted price auctions. In these auctions a seller tries to sell an item by proposing take-it-or-leave-it prices to buyers arriving sequentially. Each buyer has to choose between declining the offer—without having the possibility of coming back—or accepting it, thus ending the auction. We study Bayesian posted price auctions, where the buyers valuations for the item depend on a random state of nature, which is known to the seller only. Thus, the seller does not only have to decide price proposals for the buyers, but also how to partially disclose information about the state so as to maximize revenue. Our model finds application in several real-world scenarios. For instance, in an e-commerce platform, the state of nature may reflect the condition (or quality) of the item being sold and/or some of its features. These are known to the seller only since the buyers cannot see the item given that the auction is carried out on the web. We focus on two different settings: *public signaling*, where the signals are publicly visible to all buyers, and *private signaling*, in which the seller can send a different signal to each buyer through private communication channels. As a first negative result, we prove that, in both public and private signaling, the problem of computing an optimal seller's strategy does not admit an FPTAS unless P = NP. Indeed, the result holds for basic instances with a single buyer. Then, we provide tight positive results by designing a PTAS for each setting. To do so, we provide a preliminary result that allows us to assume without loss of generality that the seller commits to price functions with specific structures. Indeed, in a Bayesian posted price auction, the seller may commit to a price function that selects the prices to be proposed to the buyers *stochastically* on the basis of the signals being sent to *all* the buyers. This introduces considerable additional challenges compared to standard posted price auctions. In order to overcome such difficulties, we show that the seller can commit to a price function that *deterministically* proposes a price to each buyer on the basis of the signal being sent to *that* buyer only, without incurring in any revenue loss. This holds in both the public and the private signaling settings. Finally, we conclude the analysis comparing the effectiveness of different classes of signaling schemes. We show that the seller can increase their revenue by revealing information on the state of nature through signaling, with respect to the case in which they do not disclose anything. Moreover, we shows that the seller may get an higher revenue by using private signaling rather than public signaling. A complete version of our results appears in [23].

6 Online Bayesian Persuasion

In this section, we study Bayesian persuasion with payoff uncertainty. First, we consider the setting with a single receiver and we deal with uncertainty about the receiver's type by framing the Bayesian persuasion problem in an online learning framework. In particular, we study a repeated Bayesian persuasion problem where, at each round, the receiver's type is adversarially chosen from a finite set of types. Our goal is the design of an online algorithm that recommends a signaling scheme at each round, guaranteeing an expected utility for the sender close to that of the best-in-hindsight signaling scheme. We study this problem under two models of feedback: in the full information model, the sender selects a signaling scheme and later observes the type of the receiver; in the partial information model, the sender only observes the actions taken by the receiver. First, we study the computational complexity of the online Bayesian persuasion problem. We provide a negative result that rules out, even in the full information setting, the possibility of designing a no-regret algorithm with polynomial per-round running time. The same hardness result holds when employing the notion of no-α-regret (in the additive sense) for any $\alpha < 1$. Formally, we show that for any $\alpha \leq 1$, a no-α-regret algorithm for the online Bayesian persuasion problem requiring a per-round running time polynomial in the size of the instance cannot exist, unless $\mathsf{NP} \subseteq \mathsf{RP}$. In order to prove this negative result we show, as an intermediate step, that the problem of approximating an optimal signaling scheme is NP-Hard even in the offline Bayesian persuasion problem in which the sender knows the probability distribution according to which receiver's types are selected.

Then, we study whether it is possible to devise a no-regret algorithm for the online Bayesian persuasion problem by relaxing the (per-round) running time constraint. This is not a trivial problem even in the full information feedback setting since, at each round, the sender has to choose a signaling scheme among an infinite number of alternatives. Moreover, the sender's utility depends on the receiver's best response, which yields an objective function which is not linear nor convex (or even continuous in the space of the signaling schemes). In the full information feedback setting, we show how to construct an algorithm that guarantees a regret polynomial in the size of the problem instance, and sublinear in the number of rounds T with order $O(T^{1/2})$. In the partial information feedback setting, we develop an algorithm guaranteeing a regret polynomial in the size of the problem instance, and sublinear in T with order $O(T^{4/5})$. In this case, the main idea is to use a full-information no-regret algorithm in combination with a mechanism to estimate the sender's utilities corresponding to signaling schemes different from the one recommended by the algorithm. Finally, we show that, relaxing the persuasiveness constraints, we can design polynomial-time algorithms with small regret.

Finally, we extend the online Bayesian persuasion framework to include multiple receivers. We focus on the case with no-externalities and binary actions. Moreover, to focus only on the receivers' coordination problem, we overcome the intractability of the single-receiver problem assuming that each receiver has a constant number of

types. First, we prove a negative result: for any $0 < \alpha \leq 1$, there is no polynomial-time no-α-regret algorithm when the sender's utility function is supermodular or anonymous. Then, we focus on the case of submodular sender's utility functions and we show that, in this case, it is possible to design a polynomial-time no-$(1 - 1/e)$-regret algorithm, which is tight. A complete version of our results appears in [16, 18, 20].

7 Efficient Online Learning Through Mechanism Design

In the previous section, we show that, both for the setting with a single and multiple receivers, the design of polynomial-time no-regret algorithms is impossible due to the NP-Hardness of the underline offline problems in which the distribution over the types is known. Hence, the design of efficient algorithms for the offline problem is the bottleneck to the design of efficient online learning algorithms. In this section, we show how to circumvent this issue by leveraging ideas from mechanism design. In particular, we introduce a type reporting step in which the receiver is asked to report her type to the sender, after the latter has committed to a menu defining a signaling scheme for each possible receiver's type. Surprisingly, we prove that, with a single receiver, the addition of this type reporting stage makes the sender's computational problem tractable. Our main result is to show the existence of a menu of *direct* and *persuasive* signaling schemes. In the classical model in which the sender perfectly knows the receiver payoff, a signaling scheme is direct if signals represent action recommendations and persuasive if the receiver is incentivized to follow the recommendations. We extend this definition to menus of signaling schemes. In particular, a menu is direct if the signals used by all the signaling schemes are action recommendations, and it is persuasive if a receiver has an incentive to follow the action recommendation if they reported their true type. Using this result, an optimal menu of signaling schemes can be computed efficiently by a linear program of polynomial size.

Then, we extend our Bayesian persuasion framework with type reporting to settings with multiple receivers, focusing on the widely-studied case of no-externalities and binary actions. Moreover, we focus on most common classes of sender's utility functions: supermodular, submodular and anonymous [4, 5, 26, 36]. In such setting, we show that it is possible to find a sender-optimal solution in polynomial-time for supermodular and anonymous sender's utility functions. As for the case of submodular sender's utility functions, we provide a $(1 - 1/e)$-approximation to the problem, which is tight. A complete version of our results appears in [21].

8 Conclusions and Future Research

In this work, we significantly advance the state of the art on algorithmic Bayesian persuasion along two different directions. First, we study the algorithmic problem of designing optimal information disclosure policies in real-world scenarios. In particular, we study several voting problems, including majority voting, plurality voting and district-based elections characterizing the computational complexity of each problem under private and public signaling. In doing so, we provide some insights on the complexity of general persuasion problems, such as the characterization of bi-criteria approximations in public signaling problems. Moreover, we show how the partial disclosure of information can be used to reduce the social cost in routing games and to increase the revenue in posted price auctions. Then, we relax the assumptions that the sender knows the receiver's utility function, initiating the study of online Bayesian persuasion. This is the first step in designing adaptive information disclosure policies that deals with the uncertainty intrinsic in all real-world applications.

We conclude proposing some future research directions. Despite the great attention received by the economics and artificial intelligence communities and the large class of potential real-world applications, the use of Bayesian persuasion in the real world is still limited. We believe that one of the main obstacle to the design of information disclosure policies in practice is the perfect knowledge assumption. An interesting direction is to study how the general online Bayesian persuasion framework introduced in this work can be applied to structured games. This posses various challenges. First, despite the design of no-regret algorithms is computationally intractable in general, it would be interesting to find some structured games for which it is possible to design *efficient* no-regret algorithms. As a second point, while for the single-receiver online Bayesian persuasion problem we provide no-regret algorithms with both full information and partial information feedback, our analysis of settings with multiple-receiver is limited to the case with full feedback and no externalities. While this assumptions are reasonable in some settings, they do not fit with some applications. For instance, routing games requires to take in account externalities among the players. Another interesting direction is to deal with the computational challenges introduced by the online learning framework. In particular, we showed that the computation of no-regret algorithms in the online Bayesian persuasion problem is often computational intractable, making it difficult to apply in practice. We propose a way to solve this problem, showing that the intractability of an offline version of the problem can be circumvented with a type reporting step. It remains an open question if a type reporting step can be used to design *efficient* online learning algorithms. Moreover, in our online learning framework we assume that the receivers have a *finite* number of known possible types. Despite this is a significant improvement over the perfect knowledge of the receivers' utilities, this approach assumes some prior knowledge of the receivers. It would be interesting to extend our results to the case in which the receivers can have arbitrary utilities and hence an *infinite* number of possible types. Finally, we show how to deal with uncertainly over the receivers' utility functions. However, this is not the only unreasonable

assumption of the classical Bayesian persuasion framework. For instance, another important assumption is that the sender and receivers share the same prior belief. In practice, these beliefs come from past observations, and thus are uncertain and approximated. References [10, 37] study a game between a sender and a receiver that do not know the prior distribution. It would be interesting to consider uncertainly on the receiver's payoffs and the prior belief simultaneously.

References

1. Allcott H, Gentzkow M (2017) Social media and fake news in the 2016 election. J Econ Perspect 31(2):211–236
2. Alonso R, Câmara O (2016) Persuading voters. Am Econ Rev 106(11):3590–3605
3. Antioch G, et al (2013) Persuasion is now 30 per cent of US GDP: revisiting McCloskey and Klamer after a quarter of a century. Econ Round-Up 1:1
4. Arieli I, Babichenko Y (2019) Private Bayesian persuasion. J Econ Theory 182:185–217
5. Babichenko Y, Barman S (2017) Algorithmic aspects of private Bayesian persuasion. In: Innovations in theoretical computer science conference
6. Babichenko Y, Talgam-Cohen I, Xu H, Zabarnyi K (2021) Regret-minimizing Bayesian persuasion. In: EC '21: the 22nd ACM conference on economics and computation, Budapest, Hungary, July 18–23, 2021. ACM, p 128. https://doi.org/10.1145/3465456.3467574
7. Bacchiocchi F, Castiglioni M, Marchesi A, Romano G, Gatti N (2022) Public signaling in Bayesian ad auctions. In: Proceedings of the thirty-first international joint conference on artificial intelligence, IJCAI 2022, Vienna, Austria, 23–29 July 2022. pp 39–45. ijcai.org. https://doi.org/10.24963/ijcai.2022/6
8. Badanidiyuru A, Bhawalkar K, Xu H (2018) Targeting and signaling in ad auctions. In: Proceedings of the twenty-ninth annual ACM-SIAM symposium on discrete algorithms, pp 2545–2563
9. Bernasconi M, Castiglioni M, Celli A, Marchesi A, Gatti N, Trovò F (2023) Optimal rates and efficient algorithms for online Bayesian persuasion. In: Proceedings of the 40th international conference on machine learning
10. Bernasconi M, Castiglioni M, Marchesi A, Gatti N, Trovò F (2022) Sequential information design: learning to persuade in the dark. In: NeurIPS
11. Bhaskar U, Cheng Y, Ko YK, Swamy C (2016) Hardness results for signaling in Bayesian zero-sum and network routing games. In: Proceedings of the 2016 ACM conference on economics and computation, pp 479–496
12. Bro Miltersen P, Sheffet O (2012) Send mixed signals: earn more, work less. In: Proceedings of the 13th ACM conference on electronic commerce, pp 234–247
13. Candogan O (2019) Persuasion in networks: public signals and k-cores. In: Proceedings of the 2019 ACM conference on economics and computation, pp 133–134
14. Castiglioni M, Celli A, Gatti N (2020) Persuading voters: it's easy to whisper, it's hard to speak loud. In: The thirty-fourth AAAI conference on artificial intelligence, pp 1870–1877
15. Castiglioni M, Celli A, Gatti N (2023) Public Bayesian persuasion: being almost optimal and almost persuasive. Algorithmica 1–37
16. Castiglioni M, Celli A, Marchesi A, Gatti N (2020) Online Bayesian persuasion. In: Advances in neural information processing systems, vol 33, pp 16188–16198
17. Castiglioni M, Celli A, Marchesi A, Gatti N (2021) Signaling in Bayesian network congestion games: the subtle power of symmetry. In: The thirty-fifth AAAI conference on artificial intelligence
18. Castiglioni M, Celli A, Marchesi A, Gatti N (2023) Regret minimization in online Bayesian persuasion: handling adversarial receiver's types under full and partial feedback models. Artif Intell 314:103821. https://doi.org/10.1016/j.artint.2022.103821

19. Castiglioni M, Gatti N (2021) Persuading voters in district-based elections. In: The AAAI conference on artificial intelligence, AAAI. AAAI Press
20. Castiglioni M, Marchesi A, Celli A, Gatti N (2021) Multi-receiver online Bayesian persuasion. In: Meila M, Zhang T (eds.) Proceedings of the 38th international conference on machine learning. Proceedings of machine learning research, vol 139, pp 1314–1323. PMLR. Accessed 18–24 Jul 2021
21. Castiglioni M, Marchesi A, Gatti N (2022) Bayesian persuasion meets mechanism design: going beyond intractability with type reporting. In: The 21st international conference on autonomous agents and multiagent systems (2022)
22. Castiglioni M, Marchesi A, Gatti N, Coniglio S (2019) Leadership in singleton congestion games: what is hard and what is easy. Artif Intell 277 (2019). https://doi.org/10.1016/j.artint.2019.103177
23. Castiglioni M, Romano G, Marchesi A, Gatti N (2022) Signaling in posted price auctions. In: The thirty-sixth AAAI conference on artificial intelligence
24. Celli A, Coniglio S, Gatti N (2020) Private Bayesian persuasion with sequential games. In: The thirty-fourth AAAI conference on artificial intelligence, pp 1886–1893
25. Cheng Y, Cheung HY, Dughmi S, Emamjomeh-Zadeh E, Han L, Teng SH (2015) Mixture selection, mechanism design, and signaling. In: 56th annual symposium on foundations of computer science, pp 1426–1445
26. Dughmi S, Xu H (2017) Algorithmic persuasion with no externalities. In: ACM EC, pp 351–368
27. Emek Y, Feldman M, Gamzu I, PaesLeme R, Tennenholtz M (2014) Signaling schemes for revenue maximization. ACM Trans Econ Comput 2(2):1–19
28. Kamenica E, Gentzkow M (2011) Bayesian persuasion. Am Econ Rev 101(6):2590–2615
29. Mansour Y, Slivkins A, Syrgkanis V, Wu Z (2016) Bayesian exploration: incentivizing exploration in Bayesian games. In: ACM EC, p 661
30. Massicot O, Langbort C (2019) Public signals and persuasion for road network congestion games under vagaries. IFAC-PapersOnLine 51(34):124–130
31. Rabinovich Z, Jiang AX, Jain M, Xu H (2015) Information disclosure as a means to security. In: Proceedings of the 2015 international conference on autonomous agents and multiagent systems, pp 645–653
32. Roughgarden T (2005) Selfish routing and the price of anarchy, vol 174. MIT Press Cambridge
33. Vasserman S, Feldman M, Hassidim A (2015) Implementing the wisdom of Waze. In: Twenty-fourth international joint conference on artificial intelligence, pp 660–666
34. Wu M, Amin S, Ozdaglar AE (2018) Value of information systems in routing games. arXiv:1808.10590
35. Xu H, Freeman R, Conitzer V, Dughmi S, Tambe M (2016) Signaling in Bayesian stackelberg games. In: AAMAS, pp 150–158
36. Xu H (2020) On the tractability of public persuasion with no externalities. In: Proceedings of the fourteenth annual ACM-SIAM symposium on discrete algorithms. SIAM, pp 2708–2727
37. Zu Y, Iyer K, Xu H (2021) Learning to persuade on the fly: robustness against ignorance. In: EC '21: the 22nd ACM conference on economics and computation, Budapest, Hungary, July 18–23, 2021. ACM, pp 927–928. https://doi.org/10.1145/3465456.3467593

Modern High-Level Synthesis: Improving Productivity with a Multi-level Approach

Serena Curzel

Abstract High-Level Synthesis (HLS) tools simplify the design of hardware accelerators by automatically generating Verilog/VHDL code starting from a general-purpose software programming language. Because of the mismatch between the requirements of hardware descriptions and the characteristics of input languages, HLS tools still require hardware design knowledge and non-trivial design space exploration, which might be an obstacle for domain scientists seeking to accelerate applications written, for example, in Python-based programming frameworks. This research proposes a modern approach based on multi-level compiler technologies to bridge the gap between HLS and high-level frameworks, and to use domain-specific abstractions to solve domain-specific problems. The key enabling technology is the Multi-Level Intermediate Representation (MLIR), a framework that supports building reusable compiler infrastructure. The proposed approach uses MLIR to introduce new optimizations at appropriate levels of abstraction outside the HLS tool while still relying on years of HLS research in the low-level hardware generation steps; users and developers of HLS tools can thus increase their productivity, obtain accelerators with higher performance, and not be limited by the features of a specific (possibly closed-source) backend. The presented tools and techniques were designed, implemented, and tested to synthesize machine learning algorithms, but they are broadly applicable to any input specification written in a language that has a translation to MLIR. Generated accelerators can be deployed on Field Programmable Gate Arrays or Application-Specific Integrated Circuits, and they can reach high energy efficiency without any manual optimization of the code.

This research was partially supported by Pacific Northwest National Laboratory's DMC and ATSCALE Laboratory-Directed R&D Initiatives.

S. Curzel (✉)
Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milano, Italy
e-mail: serena.curzel@polimi.it

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_2

1 Introduction

The exponential growth of data science and machine learning (ML), coupled with the diminishing performance returns of silicon at the end of Moore's law and Dennard scaling, is leading to widespread interest in domain-specific architectures and accelerators [16]. Field Programmable Gate Arrays (FPGAs) and Application-Specific Integrated Circuits (ASICs) can provide the necessary hardware specialization with higher performance and energy efficiency than multi-core processors or Graphic Processing Units (GPUs). ASICs are the best solution in terms of performance, but they incur higher development costs; FPGAs are more accessible and can be quickly reconfigured, allowing to update accelerators according to the requirements of new applications or to try multiple configurations in a prototyping phase before committing to ASIC manufacturing.

ASICs and FPGAs are designed and programmed through hardware description languages (HDLs) such as Verilog or VHDL, which require developers to identify critical kernels, build specialized functional units and memory components, and explicitly manage low-level concerns such as clock and reset signals or wiring delays. The distance between traditional software programming and HDLs creates significant productivity and time-to-market gaps [19, 20] and traditionally required manual coding from expert hardware developers. The introduction of High-Level Synthesis (HLS) simplified this process, as HLS tools allow to automatically translate general-purpose software specifications, primarily written in C/C++, into an HDL description ready for logic synthesis and implementation [7, 8]. Thanks to HLS, developers can describe the kernels they want to accelerate at a high level of abstraction and obtain efficient designs without being experts in low-level circuit design.

Due to the mismatch between the levels of abstraction of hardware descriptions and general-purpose programming languages, HLS tools often require users to augment their input code through *pragma* annotations (i.e., compiler directives) and configuration options that guide the synthesis process, for example, towards a specific performance-area trade-off. Different combinations of pragmas and options result in accelerator designs with different latency, resource utilization, or power consumption. An exhaustive exploration of the design space requires few modifications to the input code, and it does not change the functional correctness of the algorithm, but it is still not a trivial process: the effect of combining multiple optimization directives can be unpredictable, and the HLS user needs a good understanding of their impact on the generated hardware.

Data scientists who develop and test algorithms in high-level, Python-based programming frameworks (e.g., TensorFlow [1] or PyTorch [18]) typically do not have any hardware design expertise: therefore, the abstraction gap that needs to be overcome is not anymore from C/C++ software to HDL (covered by mature commercial and academic HLS tools), but from Python to annotated C/C++ for HLS. The issue is exacerbated by the rapid evolution of data science and ML, as no accelerator can be general enough to support new methods efficiently, and a manual translation of each algorithm into HLS code is highly impractical.

The aim of this research is to bridge the gap between high-level frameworks and HLS through a multi-level, compiler-based approach. The key enabling technology is the Multi-Level Intermediate Representation (MLIR) [17], a reusable and extensible infrastructure in the LLVM project for the development of domain-specific compilers. MLIR allows defining specialized intermediate representations (IRs) called *dialects* to implement analysis and transformation passes at different levels of abstraction, and it can interface with multiple software programming frameworks. An MLIR-based approach is a "modern" solution to automate the design of hardware accelerators for high-level applications through HLS, as opposed to "classic" approaches that rely on hand-written template libraries [4, 11, 14].

A practical realization of the proposed approach is the SOftware Defined Architectures (SODA) Synthesizer [2, 6], an open-source hardware compiler composed of an MLIR frontend [5] and an HLS backend [15]. SODA provides an end-to-end agile development path from high-level software frameworks to FPGA and ASIC accelerators, supports the design of complex systems, and allows to introduce and explore optimizations at many different levels of abstraction, from high-level algorithmic transformations to low-level hardware-oriented ones. Translation across different levels of abstraction is performed through progressive lowering between IRs, allowing each step to leverage information gathered in other phases of the compilation. In the frontend, domain-specific MLIR dialects allow developers to work on specialized abstractions to address system-level concerns and pre-optimize the code. The integration of an open-source tool in the backend allows to exploit years of HLS research and to introduce new features in the low-level hardware generation steps when necessary. The rest of the paper will focus on the main features of SODA (Sect. 2) and describe the results it allowed to obtain (Sect. 3).

2 The SODA Synthesizer

The SODA Synthesizer (Fig. 1) is an open-source, modular, compiler-based toolchain that uses a multi-level approach, able to generate optimized FPGA and ASIC accelerators for ML through MLIR and HLS. It can accept as inputs pre-trained ML models developed in a high-level framework such as TensorFlow or PyTorch and translated into an MLIR representation. The SODA frontend (SODA-OPT) provides a search and outlining methodology to automatically extract accelerator kernels and their data dependencies from the input specification; the kernels are then optimized through a set of compiler passes that can be tuned to explore different design points, while host code containing calls to the kernel functions can be compiled by a standard LLVM compiler. SODA-OPT provides a default optimization pipeline that privileges passes resulting in faster accelerators (e.g., passes that increase instruction- and data-level parallelism or remove unnecessary operations), but many others exist that can be individually enabled or disabled, such as the ones listed in Table 1. Optimized kernels are synthesized by the backend HLS tool to generate FSMD accelerators and later composed in multi-accelerator systems; when using the Bambu HLS backend, the

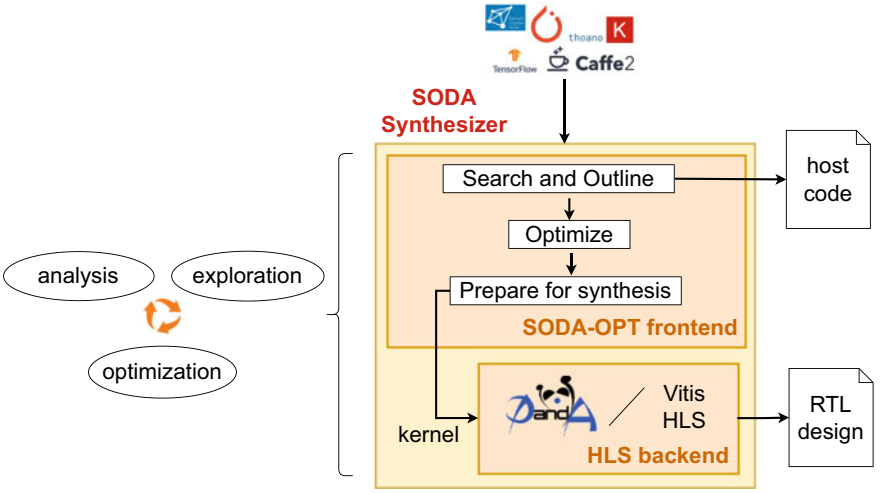


Fig. 1 The SODA Synthesizer: and end-to-end toolchain from ML algorithms to hardware accelerators through MLIR and HLS

Table 1 Partial list of high-level optimizations available in SODA-OPT

Optimization pass	Effect	Default
Loop unrolling	Expose instruction-level parallelism	Yes
Loop tiling	Balance computation and memory transfer	No
Loop pipelining	Parallelize loop iterations	No
If-conversion	Speculative execution of if-else blocks	Yes
Results forwarding	Remove unnecessary memory transfers	Yes
Temporary buffer allocation	Reduce accesses to external memory	Yes
Common sub-expression elimination	Remove unnecessary operations	Yes

SODA Synthesizer is fully open-source from the algorithm to the HDL description. The outputs of SODA-OPT are fully tool-agnostic LLVM IRs that do not contain anything specific to Bambu, so they can also be synthesized through recent versions of Vitis HLS [3].

The multi-level, MLIR-based structure of the SODA Synthesizer provides ample opportunities to explore high-level compiler transformations that can improve the quality of HLS results without needing to modify the HLS tool itself [12]. Such "higher-level" optimizations can improve the performance of the generated accel-

erators, the portability across HLS tools (since they do not introduce tool-specific annotations or code patterns), and the productivity of users and developers: optimizations can be explored more easily and safely through compiler passes than through manual code rewriting, and there is no need to access the backend HLS code nor to be expert in low-level synthesis techniques. Moreover, dedicated MLIR dialects can be built and exploited to solve domain-specific optimization problems: for example, the `soda` dialect has been introduced to support the outlining process for accelerator kernels, and many SODA-OPT passes exploit the `affine` dialect to apply loop optimizations.

Following this approach, a new loop pipelining pass has been introduced in SODA-OPT leveraging the MLIR `affine` dialect, implementing high-level code optimizations that provide a pre-scheduled input description to HLS [13]. The `affine` dialect provides structures and methods to analyze and transform loops (in fact, it was initially introduced to support polyhedral optimizations for ML frameworks), and the higher level of abstraction allows to identify more complex dependencies than what is possible on an LLVM IR or low-level HLS IR. The proposed implementation can analyze dependencies between operations in the loop body of an `affine.for` operation and schedule them to overlap the execution of loop iterations, following standard software pipelining techniques; it can forward results from one iteration to the other, support loops with variable bounds, and speculate execution of if-else blocks.

The SODA Synthesizer also integrates a low-level synthesis methodology for the generation of complex system-on-chip (SoC) architectures composed of multiple kernels, either connected to a central microcontroller, or directly to each other in a custom dataflow architecture [9]. In fact, large and compute-intensive deep neural networks frequently represent a challenge for HLS tools, and they need to be manually broken down into smaller kernels; the issue is especially evident when the model needs to process streaming inputs in a pipelined fashion, as the complexity of the finite state machine (FSM) driving the execution becomes unmanageable. In a SoC with a central general-purpose microcontroller driving multiple accelerators, the data movement between the host microcontroller, the accelerators, and memory quickly becomes a performance bottleneck. For this reason, the SODA Synthesizer has been extended to support the generation of a second type of system: a dynamically scheduled architecture where custom accelerators are composed in a dataflow system and are driven by a distributed controller. In this architecture, multiple accelerators can perform computations in parallel on different portions of streaming input data without requiring orchestration from the host microcontroller, and can communicate with each other without going through external memory. Analysis and transformation passes in the MLIR frontend have access to high-level representations that explicitly describe the flow of data through operators and memory in a computational graph, removing the need for complex alias analysis in the HLS backend and thus simplifying the low-level generation steps.

3 Experimental Results

A multi-level approach to HLS improves productivity, portability, and performance for users that want to accelerate high-level applications and do not have hardware design expertise. While productivity is not a feature that can be precisely measured, there are evident advantages when comparing the SODA Synthesizer with other state-of-the-art design flows based on HLS: unlike hls4ml [14] and FINN [4], SODA does not require to maintain a library of templated operators, so it is more easily adapted to new classes of input applications; SODA also generates backend-agnostic low-level code, while ScaleHLS [21] focuses on extracting performance from one specific HLS tool.

Table 2 presents execution times obtained with SODA and ScaleHLS on Poly-Bench kernels,[1] highlighting for every kernel and every input size which is the frontend/backend combination that resulted in the lowest number of clock cycles (more results are available in [5]). To avoid focusing on performance differences that derive solely from capabilities of different HLS backends, the table also reports separate baselines that are obtained without frontend optimizations. The experiments were run targeting a Xilinx Virtex7 FPGA with 100 MHz frequency; errors sometimes occurred when Verilog code generated by ScaleHLS required more resources than the ones available in the target FPGA.

Looking at absolute numbers of clock cycles, SODA outperforms ScaleHLS in 12 kernels out of 16, through either the Bambu or the Vitis HLS backend. The SODA-OPT optimization pipeline is particularly well suited to kernels with dot product or matrix multiplication structures (providing 66.38× performance increase on *2mm* and 50.43× on *gemm*); its effect is more limited, instead, on kernels that contain irregular loop structures such as *syr2k*. The performance improvement is generally smaller when comparing SODA-OPT for Vitis HLS against the Vitis HLS baseline, because Vitis HLS applies loop optimizations even in absence of user directives, and the optimizations introduced by SODA-OPT can provide only a slight improvement over the default ones. The optimizations introduced by ScaleHLS greatly improve accelerator performance with respect to baseline designs synthesized by Vivado HLS; however, the annotated C++ code it produces is not portable, while the MLIR-based approach of SODA does not rely on pragma annotations and generates designs that can be synthesized with different HLS backends.

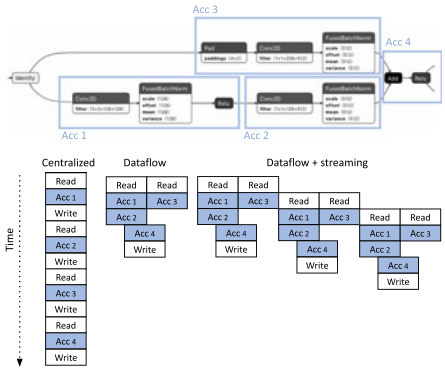
The SODA Synthesizer can generate complex multi-accelerator SoC for neural networks following either a centralized or a dataflow architecture, as presented in [9]. In a centralized architecture individual accelerators are attached to a central bus and a microcontroller drives their execution; all data is stored in and retrieved from external memory. The dataflow architecture, instead, is a system that uses a distributed controller to orchestrate the execution of accelerators accessing shared memory.

Figure 2 shows, on the right, part of the computational graph of a convolutional neural network (CNN) divided into four accelerator kernels. In the centralized archi-

[1] http://web.cse.ohio-state.edu/~pouchet.2/software/polybench/.

Table 2 Execution times of accelerators optimized with different synthesis tools

Kernel	Backend	Frontend	2 × 2	4 × 4	8 × 8	16 × 16	Avg. speedup
2 mm	Bambu	None	176	1375	11218	87842	
		SODA-OPT	25	43	98	784	66.38×
	Vitis HLS	None	43	115	599	4239	
		SODA-OPT	26	48	106	848	3.67×
	Vivado HLS	None	162	1138	9698	75586	
		ScaleHLS	38	63	114	410	72.94×
3 mm	Bambu	None	220	1743	14042	111410	
		SODA-OPT	22	40	320	2560	35.24×
	Vitis HLS	None	37	109	593	4233	
		SODA-OPT	23	45	103	824	3.73×
	Vivado HLS	None	207	1467	12723	99939	
		ScaleHLS	57	97	169	797	54.86×
gemm	Bambu	None	103	794	6538	42514	
		SODA-OPT	16	28	71	568	50.43×
	Vitis HLS	None	24	52	140	5635	
		SODA-OPT	15	29	71	259	6.78×
	Vivado HLS	None	99	669	5593	42801	
		ScaleHLS	19	27	56	Error	43.29×
syr2k	Bambu	None	99	706	4834	35650	
		SODA-OPT	19	270	1417	8835	3.82×
	Vitis HLS	None	97	367	2627	18179	
		SODA-OPT	50	159	509	1785	4.90×
	Vivado HLS	None	73	265	1089	4225	
		ScaleHLS	93	353	1665	Error	0.73×



ResNet50 - 100 inputs stream

Architecture	Computation	Memory
Centralized	114,610,199,200	715,263,486
Dataflow	34,677,385,627	656,320
Speedup	**3.3**	**1089.8**

MobileNetV2 - 100 inputs stream

Architecture	Computation	Memory
Centralized	1,951,706,600	372,630,149
Dataflow	625,392,266	64,345
Speedup	**3.1**	**5,791.2**

Fig. 2 Comparison between the performance of a centralized and a dataflow architecture generated by SODA for convolutional neural network models

tecture, every accelerator communicates with its producers and consumers through external memory, so accelerator execution and memory access are serialized. In the dataflow architecture, instead, only input arguments to the first kernel and output arguments of the last one go through external memory, while intermediate results are kept in a shared on-chip memory with as many ports as there are accelerators in the system, so that the architecture can support conflict-free concurrent accelerator execution, allowing pipelined execution of streaming inputs. The table on the left of Fig. 2 reports the execution time of the two architectures in terms of clock cycles, highlighting the benefits of the dataflow architecture for streaming execution of CNN models. The high cost of communicating between accelerators and external memory is reduced when accelerators can send data to each other through shared memory, and concurrent pipelined execution provides further improvements as the overall latency for streaming inputs is mostly determined by the initiation interval, i.e., the execution of the critical path. Although the accelerators could execute in parallel on different inputs also in the centralized architecture, SODA-OPT currently does not support the generation of host code with non-blocking function calls.

4 Conclusion

In the last few years, High-Level Synthesis has become an invaluable tool to simplify the development of hardware accelerators on FPGA and ASIC, providing higher and higher quality of results to users with little expertise in low-level RTL design. State-of-the-art HLS tools still expect some hardware design knowledge from users, especially when the accelerator needs to be optimized to meet tight application requirements or when different configurations need to be evaluated looking for a specific trade-off between quality metrics.

This requirement prevents widespread adoption of HLS by domain scientists that develop data science and artificial intelligence algorithms in high-level, Python-based programming frameworks. Moreover, research that aims at improving the efficiency of the HLS process itself or the quality of generated accelerators is typically limited by the expressiveness of C/C++ code and by the annotations supported by a specific, closed-source backend tool. This paper proposed to solve the two issues by coupling established HLS tools with the modern compiler infrastructure provided by the MLIR framework, in order to improve the automated synthesis process of accelerators for high-level applications. Such an approach allows seamless integration with high-level ML frameworks, encourages the introduction of innovative optimization techniques at specific levels of abstraction, and can exploit multiple state-of-the-art HLS tools in the backend.

The proposed design flow allows to implement and apply high-level optimizations before HLS, as compiler passes supported by dedicated MLIR abstractions (dialects); such an approach can improve productivity, performance, and portability of optimizations. Loop pipelining has been used as an example of the intrinsic optimization potential in a multi-level design and optimization flow, and it has been seamlessly integrated into the SODA Synthesizer frontend. The availability of multiple levels of abstraction and domain-specific representations opens the door to new possibilities to study and implement innovative design automation methods, ranging from the exploration of techniques that can benefit HLS when applied at a high level of abstraction to the introduction of new synthesis methodologies and architectural models.

The proposed multi-level approach is modular and extensible by design, so different parts can be easily reused and adapted to the needs of different input applications, requirements, and research scenarios. A multi-level compiler-based framework can also adapt more easily to innovative input algorithms and hardware targets. For example, spiking neural networks are built of biologically-inspired integrate-and-fire neurons, and they are usually mapped on analog neuromorphic hardware; a new MLIR dialect has been designed to support the synthesis of SNN models into neuromorphic components [10]. The dialect models concepts from the analog domain of spiking neurons through new types and operations that describe sequences of current spikes as lists of timestamps signaling their arrival.

Experimental results showed strengths and weaknesses of the approach, indicating possible next steps to improve the QoR of generated accelerators and the applicability of the proposed tools and techniques. Code for the tools developed in this research has been released in open-source to foster collaboration[2] parts of them can be easily reused or integrated with future research efforts.

[2] https://github.com/ferrandi/PandA-bambu, https://gitlab.pnnl.gov/sodalite/soda-opt.

References

1. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, et al (2016) TensorFlow: a system for large-scale machine learning. In: 11th USENIX symposium on operating systems design and implementation (OSDI), pp 265–283
2. Agostini NB, Curzel S, Limaye A, Amatya V, Minutoli M, Castellana VG, Manzano J, et al (2022) The SODA approach: leveraging high-level synthesis for hardware/software co-design and hardware specialization. In: Proceedings of the 59th ACM/IEEE design automation conference (DAC), pp 1359–1362
3. AMD-Xilinx: Vitis HLS LLVM 2021.2 (2021). https://github.com/Xilinx/HLS
4. Blott M, Preußer TB, Fraser NJ, Gambardella G, O'brien K, Umuroglu Y, et al (2018) FINN-R: an end-to-end deep-learning framework for fast exploration of quantized neural networks. ACM Trans Reconfigurable Technol Syst 11(3):1–23
5. Bohm Agostini N, Curzel S, Amatya V, Tan C, Minutoli M, Castellana VG, Manzano J, et al (2022) An MLIR-based compiler flow for system-level design and hardware acceleration. In: IEEE/ACM international conference on computer aided design (ICCAD), pp 1–9
6. Bohm Agostini N, Curzel S, Zhang JJ, Limaye A, Tan C, Amatya V, Minutoli M et al (2022) Bridging python to silicon: the SODA toolchain. IEEE Micro 42(5):78–88
7. Cong J, Lau J, Liu G, Neuendorffer S, Pan P, Vissers K, Zhang Z (2022) FPGA HLS today: successes, challenges, and opportunities. ACM Trans Reconfigurable Technol Syst 15(4):1–42
8. Cong J, Liu B, Neuendorffer S, Noguera J, Vissers K, Zhang Z (2011) High-level synthesis for FPGAs: from prototyping to deployment. IEEE Trans Comput-Aided Des Integr Circuits Syst 30(4):473–491. https://doi.org/10.1109/TCAD.2011.2110592
9. Curzel S, Agostini NB, Castellana VG, Minutoli M, Limaye A, Manzano J, Zhang J, Brooks D, Wei GY, Ferrandi F et al (2022) End-to-end synthesis of dynamically controlled machine learning accelerators. IEEE Trans Comput 71(12):3074–3087
10. Curzel S, Agostini NB, Song S, Dagli I, Limaye A, Tan C, Minutoli M, Castellana VG, Amatya V, Manzano J, Das A, Ferrandi F, Tumeo A (2021) Automated generation of integrated digital and spiking neuromorphic machine learning accelerators. In: 2021 IEEE/ACM international conference on computer aided design (ICCAD). pp 1–7. https://doi.org/10.1109/ICCAD51958.2021.9643474
11. Curzel S, Bohm Agostini N, Tumeo A, Ferrandi F (2022) Hardware acceleration of complex machine learning models through modern high-level synthesis. In: Proceedings of the 19th ACM international conference on computing frontiers, pp 209–210
12. Curzel S, Jovic S, Fiorito M, Tumeo A, Ferrandi F (2022) Higher-level synthesis: experimenting with MLIR polyhedral representations for accelerator design. In: 12th international workshop on polyhedral compilation techniques (IMPACT), pp 1–10
13. Curzel S, Jovic S, Fiorito M, Tumeo A, Ferrandi F (2023) Mlir loop optimizations for high-level synthesis: a case study. In: Proceedings of the international conference on parallel architectures and compilation techniques. PACT '22, Association for Computing Machinery, New York, NY, USA, pp 544–545. https://doi.org/10.1145/3559009.3569688
14. Duarte J, Han S, Harris P, Jindariani S, Kreinar E, Kreis B, Ngadiuba J et al (2018) Fast inference of deep neural networks in FPGAs for particle physics. J Instrum 13(07):P07027
15. Ferrandi F, Castellana VG, Curzel S, Fezzardi P, Fiorito M, Lattuada M, Minutoli M, Pilato C, et al (2021) Bambu: an open-source research framework for the high-level synthesis of complex applications. In: Proceedings of the 58th ACM/IEEE design automation conference (DAC), pp 1327–1330
16. Hennessy J, Patterson D (2018) A new golden age for computer architecture: domain-specific hardware/software co-design, enhanced security, open instruction sets, and agile chip development. In: 2018 ACM/IEEE 45th annual international symposium on computer architecture (ISCA), pp 27–29. https://doi.org/10.1109/ISCA.2018.00011
17. Lattner C, Amini M, Bondhugula U, Cohen A, Davis A, Pienaar J, Riddle R, et al (2021) MLIR: scaling compiler infrastructure for domain specific computation. In: IEEE/ACM international symposium on code generation and optimization (CGO), pp 2–14

18. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, et al (2019) PyTorch: an imperative style, high-performance deep learning library. In: Proceedings of the 33rd international conference on neural information processing systems (NeurIPS)
19. Semiconductor Industry Association: International Technology Roadmap for Semiconductors 1999 Edition (1999)
20. Truong L, Hanrahan P (2019) A golden age of hardware description languages: applying programming language techniques to improve design productivity. In: 3rd summit on advances in programming languages (SNAPL), vol 136, pp 7:1–7:21
21. Ye H, Hao C, Cheng J, Jeong H, Huang J, Neuendorffer S, Chen D (2022) ScaleHLS: a new scalable high-level synthesis framework on multi-level intermediate representation. In: 2022 IEEE international symposium on high-performance computer architecture (HPCA), pp 741–755. IEEE

FPGA-Based Design and Implementation of a Code-Based Post-quantum KEM

Andrea Galimberti

Abstract Post-quantum cryptography aims to design cryptosystems that can be deployed on traditional computers and resist attacks from quantum computers, which are widely expected to break the currently deployed public-key cryptography solutions in the upcoming decades. Providing effective hardware support is crucial to ensuring a wide adoption of post-quantum cryptography solutions, and it is one of the requirements set by the USA's National Institute of Standards and Technology within its ongoing standardization process. This research delivers a configurable FPGA-based hardware architecture to support BIKE, a post-quantum QC-MDPC code-based key encapsulation mechanism. The proposed architecture is configurable through a set of architectural and code parameters, which make it efficient, providing good performance while using the resources available on FPGAs effectively, flexible, allowing to support different large QC-MDPC codes defined by the designers of the cryptosystem, and scalable, targeting the whole Xilinx Artix-7 FPGA family. Two separate modules target the cryptographic functionality of the client and server nodes of the quantum-resistant key exchange, respectively, and a complexity-based heuristic that leverages the knowledge of the time and space complexity of the configurable hardware components steers the design space exploration to identify their best parameterization. The proposed architecture outperforms the state-of-the-art reference software that exploits the Intel AVX2 extension and runs on a desktop-class CPU by 1.77 and 1.98 times, respectively, for AES-128- and AES-192-equivalent security instances of BIKE, and it provides a speedup of more than six times compared to the fastest reference state-of-the-art hardware architecture, which targets the same FPGA family.

A. Galimberti (✉)
Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano, Via Ponzio 34/5, 20133 Milano, Italy
e-mail: andrea.galimberti@polimi.it

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_3

1 Introduction

Public-key cryptography (PKC) allows sending encrypted messages over an insecure channel without sharing a secret key, and it has traditionally been a critical component of secure communication protocols such as TLS and SSH. Quantum computing is, however, expected to break the traditional PKC solutions [5, 10, 30] in the upcoming decades, making it mandatory to design new security solutions that can also resist attacks carried out by quantum computers.

Post-quantum cryptography (PQC) aims to design cryptosystems that can be deployed on traditional computers and are based on problems that are computationally hard also for quantum computers, other than traditional ones, thus being able to resist both traditional and quantum attacks.

The USA's National Institute of Standards and Technology (NIST) is currently undertaking a standardization process to define new standards for PQC. Starting from 82 submissions in 2017, it selected as standards four schemes that can be split into key encapsulation mechanisms (KEMs), which are meant to share secret keys confidentially, and digital signatures, which guarantee the authenticity and integrity of a message to the recipient.

All four schemes selected as standards are lattice-based ones [22, 26], i.e., based on the shortest vector problem (SVP), which requires searching for the non-zero vector of a lattice having minimum norm and that is considered NP-hard for both traditional and quantum computers [27].

NIST claimed, therefore, the need to diversify its portfolio of PQC solutions and expects to select one more KEM among the three remaining code-based ones, i.e., BIKE, Classic McEliece, and HQC. Code-based cryptography dates back to the McEliece cryptosystem, introduced in 1978 and based on the difficulty of decoding a generic linear code [21], which is recognized as an NP-hard problem. Code-based cryptosystems in NIST's PQC standardization process are compared in Figs. 1 and 2,

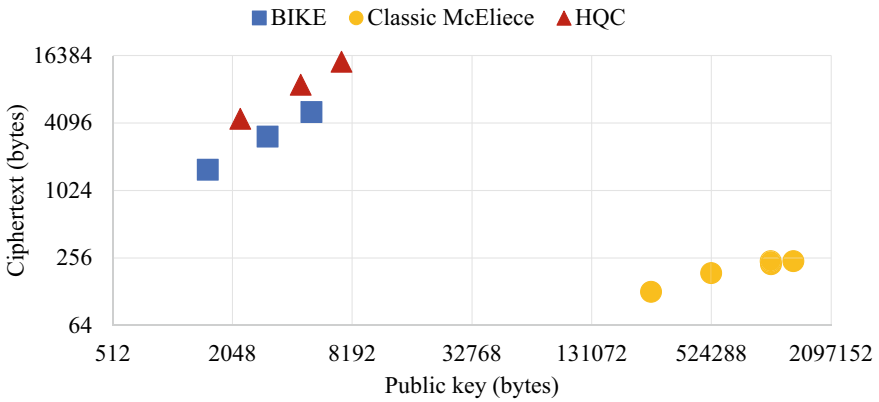


Fig. 1 Size in bytes of the public key and ciphertext of the KEMs advancing to the fourth round of the NIST PQC standardization process [24]

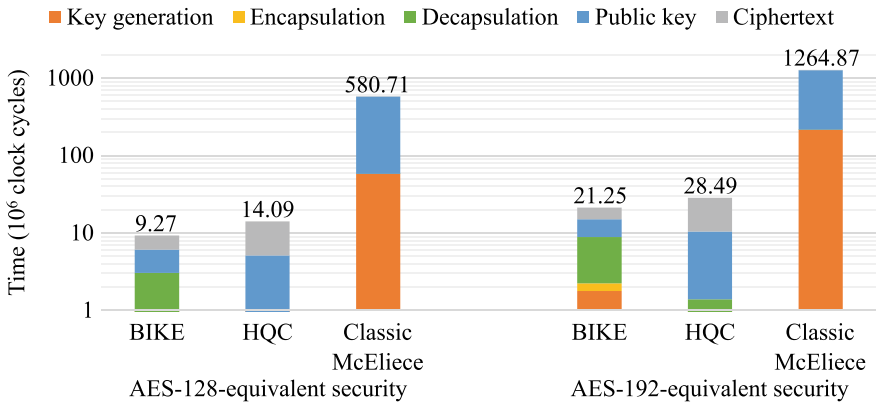


Fig. 2 Performance of NIST Round 4 KEMs on a x86-64 CPU, considering a 2000 cycles/byte transmission cost [25]

respectively, according to their public key and ciphertext sizes, which show how Classic McEliece has a huge public key, in the order of millions of bits, and software performance, which highlights BIKE as the best performing scheme when also considering the cost of transmitting the public keys and ciphertexts between the communicating nodes.

BIKE is a post-quantum code-based KEM using quasi-cyclic moderate-density parity-check (QC-MDPC) codes. These codes are employed in a scheme similar to the well-studied Neiderreiter one, which dates back to the early 1980s. Compared to traditional Niederreiter schemes, whose underlying binary Goppa codes must have sizes in the order of millions of bits to provide quantum resistance, BIKE achieves a significantly smaller public key, in the order of tens of thousands of bits, through its usage of QC-MDPC codes.

Given the complexity of PQC cryptosystems such as BIKE in terms of memory requirements and software performance, providing effective hardware support will be paramount to ensuring a wide adoption and effective deployment of post-quantum security solutions across the computing continuum ranging from embedded devices at the edge to HPC [1]. Indeed, with ever more private, sensitive, and critical data collected and processed in a variety of scenarios, it is mandatory to design computing platforms that not only provide optimal performance for the target applications [13, 33, 34] and the energy and power efficiency required by the specific use case [35] but also guarantee the security of the users' data.

Implementations of BIKE from the literature encompass software, hardware, and hardware-software ones. However, all of them suffer from different drawbacks [16]. Software implementations [3, 7, 8], including those targeting desktop-class Intel CPUs with support for AVX2 instructions and running at more than 4 GHz [2], provide poor performance, whereas hardware ones are custom-tailored to specific target platforms [28, 29].

This research delivers a configurable FPGA-based hardware architecture to support BIKE through two modules dedicated the client- and server-side functionalities of the key exchange. The proposed architecture aims to improve performance over the existing state-of-the-art software and hardware implementations of BIKE, and it is configurable through architectural and code parameters that, through a single parametric design, allow for using the resources available on FPGAs effectively, supporting different large QC-MDPC codes, and targeting the whole Xilinx Artix-7 FPGA family.

2 Components for QC-MDPC Code-Based Cryptography

The hardware components implementing binary polynomial inversion [17], binary polynomial multiplication [4], and Black-Gray-Flip (BGF) decoding [31], i.e., the three most complex operations employed within the BIKE cryptosystem, were specifically designed in a parametric way to exploit parallelism as desired according to the performance requirements and the area constraints given by the target platform. Their designs, meant for FPGA targets, are suitable not only for accelerating the BIKE post-quantum KEM but more in general for other applications making use of large binary polynomials and QC-MDPC codes.

Dense-dense binary polynomial multiplication The dense-dense binary polynomial multiplier [32] performs the multiplication between two large polynomials in $\mathbb{Z}_2[x]/(x^p + 1)$, with degree p in the order of tens of thousands, through a hybrid architecture that mixes the Karatsuba and Comba algorithms [9, 20].

Applying a configurable number of iterations of the Karatsuba algorithm reduces the number of smaller partial products compared to schoolbook multiplication. Each iteration can either compute its three partial products in parallel, on separate internal multipliers, or sequentially, on a shared one. The multipliers employed to compute such partial products either have a Karatsuba architecture themselves or a Comba-based one. At the end of Karatsuba's recursive application, the Comba formula is indeed leveraged to perform the actual computation of the partial products since the size of the operands after the recursive application of the Karatsuba algorithm is still too large to fit into a combinational multiplier. Comba multiplication schedules efficiently the computation of such partial products on a combinational component that performs the carry-less multiplication between two BW-bit digits, where BW corresponds to the datapath bandwidth.

Selecting the number of Karatsuba recursions, whether each computes its partial products sequentially or concurrently, and the datapath bandwidth allows for exploring a variety of performance-area trade-offs.

Binary polynomial exponentiation The exponentiation at the power of k of a polynomial $f(x)$ in $\mathbb{Z}_2[x]/(x^p + 1)$, where k and p are coprime as in QC-MDPC codes

employed by BIKE, corresponds to a permutation in which each i-th bit of the operand $f(x)$ corresponds to the $((i \cdot k) \bmod p)$-th bit of the result $g(x)$.

The exponentiation component [17] implements a two-stage architecture. The first one includes a p-bit memory and outputs E bits per cycle, while the second one contains E p-bit memories, each receiving a bit from the first stage and writing it in the corresponding position. Finally, the contents of the second-stage memories are XORed to produce the actual result of the exponentiation. As an optimization, the usage of lookup tables pre-computed at design time avoids the computation of the bit start addresses and address increments required to obtain the positions of bits in the result polynomial.

The E number of result bits computed per clock cycle, which determines the execution time and area of the exponentiation component, can be selected at design time with any value between 1 and p.

Binary polynomial inversion The binary polynomial inversion component [17] implements a Fermat-based algorithm that computes, by iterating binary polynomial multiplications and exponentiations, the multiplicative inverse of a polynomial in $\mathbb{Z}_2[x]/(x^p + 1)$, which is the most time-consuming operation in BIKE's key generation primitive [19].

The multiplications and exponentiations are carried out on dense-represented operands by two separate parametric components, i.e., the dense-dense binary polynomial multiplication and binary exponentiation components described previously. The two types of operations are computed on their dedicated components by scheduling them in a pipelined fashion, executing independent multiplications and exponentiations concurrently and thus minimizing the execution time of the overall inversion operation.

The dense-dense binary polynomial multiplication and binary polynomial exponentiation components are configurable in their code and architectural parameters, and finding an optimal performance-area trade-off for the inversion one requires balancing their resource utilization and execution time.

Black-Gray-Flip decoding The decoding component implements the BGF decoding algorithm [11], a variant of the baseline QC-MDPC bit-flipping decoding algorithm. The BGF algorithm iterates the computation of two multiplications, performed respectively in the integer and binary domains, between a dense polynomial operand and a sparse one [31]. The two dense-sparse multiplications are performed concurrently in a pipelined fashion, and the number of the bits computed in parallel in both is configurable by the designer [4].

The multiplication between a sparse polynomial $s(x)$ with Hamming weight v, i.e., v coefficients set to 1, and a dense one $d(x)$ corresponds to the addition of v copies of $d(x)$ each shifted by the position of the corresponding 1 in $s(x)$. In the binary domain case, the addition corresponds to XOR, and the result polynomial has binary coefficients, i.e., either 0 or 1. On the contrary, in the integer domain case, it corresponds to integer arithmetic addition, and the result's coefficients are thus integer values comprised between 0 and v. The two integer- and binary-domain

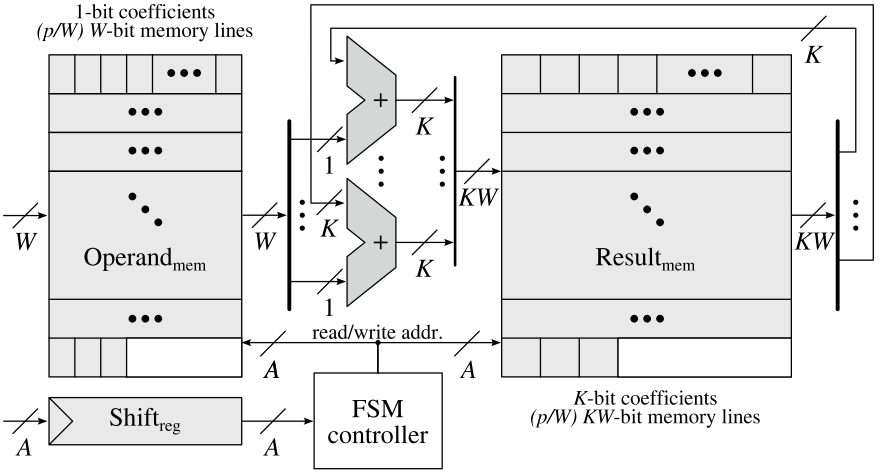


Fig. 3 Baseline architecture of the sparse-dense multiplication components

multiplications are performed by separate components, each dedicated specifically to one of them, but both implement a similar architecture.

The baseline architecture, depicted in Fig. 3, stores in a BRAM memory ($\texttt{Operand}_{\texttt{Mem}}$) the dense operand polynomial and in a flip-flop-based register ($\texttt{Shift}_{\texttt{Reg}}$) the position of a bit set to 1 in the sparse one. The content of $\texttt{Operand}_{\texttt{Mem}}$ is shifted according to the value stored in Shift_{Reg} and accumulated in the result polynomial BRAM memory ($\texttt{Result}_{\texttt{Mem}}$) according to the addition operation specific to the implemented arithmetic. In Fig. 3, W corresponds to the number of polynomial coefficients read and written per clock cycle, K refers to the bit length of the coefficients of the result polynomial, and A refers to the width of read and write addresses.

The computation of the overall sparse-dense multiplication can be parallelized, reducing execution time at the cost of additional area, by instantiating multiple shift-and-accumulate modules. Up to v of such modules can be implemented to perform the shift-and-accumulate operation after feeding them different values of positions of bits set to 1 in the sparse operand. The overall product of the multiplication will finally be obtained as the sum of the result polynomials from each of the instantiated shift-and-accumulate modules.

Sparse-dense binary polynomial multiplication The sparse-dense binary polynomial multiplier [4] is employed within all three KEM primitives of BIKE, i.e., key generation, encapsulation, and decapsulation, and it is designed with the same architecture as the one employed by the binary dense-sparse multiplier instantiated in the BGF decoding module. Its parallelism is similarly configurable by selecting the number of shift-and-accumulate operations to compute concurrently, which can be any value between 1 and v, where v is the Hamming weight of the dense operand polynomial.

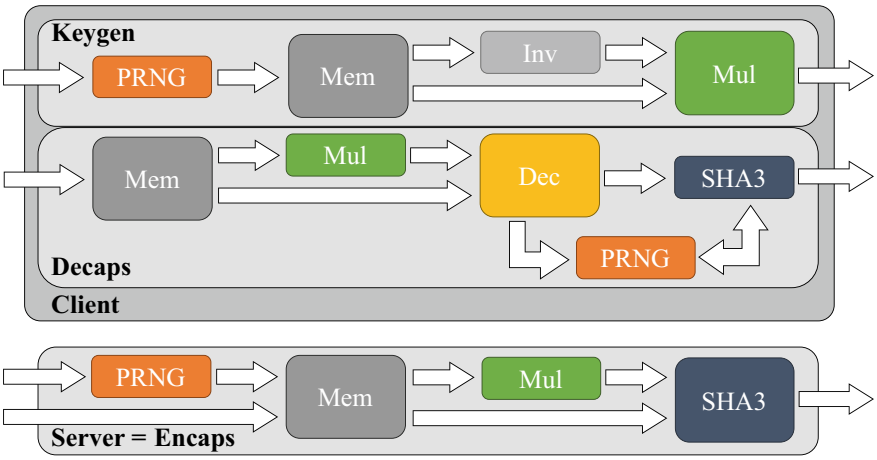


Fig. 4 Top-level architecture of the BIKE client and server cores

Other components The SHA-3 component [14] implements the SHA3-384 cryptographic hash function [12]. It computes the 384-bit digest of the SHA3-384 cryptographic function of the input message according to an architecture similar to the high-speed core detailed in [6], which was modified to support the standard SHA-3 cryptographic hash functions in place of pre-standard Keccak functions.

The pseudorandom number generation (PRNG) component [14] performs the generation of a pseudorandom sequence of bits with fixed Hamming weight by using an internal SHAKE256 module, which implements an architecture similar to the SHA-3 component, albeit producing a variable-length output according to the needs of the surrounding pseudorandom generation logic. The SHAKE256 module expands a seed obtained from a TRNG [18] into a digest output that is broken up into $(\log_2 p)$-bit chunks, each possibly representing the position of a bit set to 1 within a p-bit vector, and the extracted values are evaluated to discard the values which have been generated previously, avoiding cancellations and therefore enabling the generation of a vector with the desired Hamming weight. Moreover, values larger than or equal to p are discarded, providing a uniform distribution of bits set to 1 within the random-generated bit vector.

3 Client-Server BIKE Architecture

Two separate cores target the cryptographic functionality of the client and server nodes of the BIKE key exchange, respectively. The client and server cores, whose architecture is depicted in Fig. 4, make use of the configurable binary polynomial arithmetic and BGF decoding components, the SHA-3 core, and the pseudorandom

number generator that were previously described, and contain additional BRAM-based memories to store the large binary polynomials [15].

The `Client` core is composed of two main modules, `Keygen` and `Decaps`, devoted to the key generation and decapsulation of BIKE, respectively [14]. The `Keygen` module performs three subsequent hardware operations, namely pseudo-random number generation (executed by the `PRNG` component), binary polynomial inversion (`Inv`), and binary polynomial multiplication (`Mul`). Similarly, the `Decaps` module executes a sequence of four hardware operations, namely binary polynomial multiplication (`Mul`), BGF decoding (`Dec`), computation of SHA-3 hash digest (`SHA3`), and pseudorandom number generation (`PRNG`). The `PRNG` and `Mul` components are notably shared between the `Keygen` and `Decaps` modules to minimize duplicate hardware resources.

The `Server` core only includes the `Encaps` module [14], devoted to the encapsulation primitive of BIKE, which requires performing a sequence of three hardware operations, namely pseudorandom number generation (`PRNG`), binary polynomial multiplication (`Mul`), and computation of the SHA-3 hash function (`SHA3`).

The optimal parameterization, which maximizes performance within the available FPGA resources, of the configurable components, i.e., binary polynomial arithmetic and BGF decoding ones, is identified by using a complexity-based heuristic that leverages the knowledge of such parametric components' time and space complexity to steer the design space exploration. The execution time is selected as a proxy for the time complexity, while the space complexity is modeled by the number of occupied BRAM memory blocks since the design is dominated by BRAM usage due to the large polynomials and the exploited parallelism.

4 Experimental Evaluation

The experimental evaluation aims to gauge the performance and resource utilization improvements of the proposed FPGA-based architectures compared to state-of-the-art software, hardware-software, and hardware implementations.

Experimental setup The proposed components were described in SystemVerilog and then implemented in Xilinx Vivado 2020.2 targeting Xilinx Artix-7 FPGAs, which were selected as the target platform since they are the de-facto standard in research, due to their wide availability and best price-performance ratio among FPGAs, and they were chosen as the hardware target by NIST, to avoid differences due to FPGA technologies and ASIC technology nodes. RTL synthesis and implementation were carried out targeting a 91 MHz clock frequency, i.e., an 11ns clock period.

The proposed architectures were validated from the functional point of view, both through post-implementation simulation, on Artix-7 35, Artix-7 50, and Artix-7 200 FPGAs, and through prototype execution on a Digilent Nexys 4 DDR board, which features an Artix-7 100 FPGA. In each case, the results from the executions of 10000

key generations, encapsulations, and decapsulations on the proposed architectures were compared with the corresponding outputs of software execution.

Reference implementations The experimental evaluation was carried out against state-of-the-art software, hardware-software, and hardware implementations of the BIKE post-quantum KEM.

The additional Intel AVX2-optimized software implementation of BIKE [2] was selected as the software reference. It provides a constant-time execution on Intel x86-64 CPUs that support the Intel AVX2 instruction set extension, i.e., CPUs from the Intel Haswell generation and later ones. Within the experimental evaluation, it was executed on an Intel Core i5-10310U CPU, a desktop-class 64-bit processor implementing the x86-64 ISA and providing support for the Intel AVX2 extension, running at a clock frequency up to 4.4 GHz. Moreover, the PC mounting the Intel CPU ran the Ubuntu 20.04.3 LTS operating system.

The solution proposed in [23], which makes use of HLS-generated accelerators, each implementing a BIKE primitive, was selected as the hardware-software reference. Three different combinations of KEM primitives implemented in hardware, depending on the available FPGA resources, with the remaining ones executed instead in software on the CPU, allow targeting three chips from the Xilinx Zynq-7000 heterogeneous SoC family, which feature ARM CPUs coupled with programmable FPGA logic equivalent to the Artix-7 one.

The official FPGA-based hardware implementation [28] was instead selected as the state-of-the-art hardware reference. The proposed design, targeting Xilinx FPGAs and described in SystemVerilog, delivers a unified architecture that implements the whole BIKE KEM and executes it in constant time. The authors provide three instances ranging from a lightweight one that minimizes resource utilization up to mid-range and high-performance ones.

Area results The area of the proposed architecture is evaluated according to its utilization of the FPGA resources available on the target chips. Table 1 details the look-up tables (LUT), flip-flops (FF), and block RAM (BRAM) blocks occupied by the client and server instances. The proposed architecture's smallest client and server

Table 1 Area results, expressed in terms of LUT, FF, and BRAM resources, and execution times, in milliseconds, for the proposed client and server cores

Core	Equivalent security	Lightweight				High-performance			
		Resources			Exec. time	Resources			Exec. time
		LUT	FF	BRAM		LUT	FF	BRAM	
Client	AES-128	31792	17805	43.5	5.71	126510	51492	357	0.58
	AES-192	31411	20181	45.5	19.27	124891	53067	360	1.71
Server	AES-128	19804	11401	30	0.03	91422	46208	275.5	0.03
	AES-192	19979	12282	28	0.08	72725	37795	235.5	0.06

Table 2 Execution times, in milliseconds, for the state-of-the-art and proposed implementations. Legend: **LW** lightweight, **MR** mid-range, **HP** high-performance instances

Equivalent security	Ref. SW [2]	Ref. HW/SW [23]			Ref. HW [28]			Proposed	
	AVX2	LW	MR	HP	LW	MR	HP	LW	HP
AES-128	1.08	617.31	482.48	288.18	11.13	6.36	3.69	5.74	0.61
AES-192	3.51	–	–	–	37.10	19.71	11.69	19.35	1.77

cores fit in Artix-7 50 and 35 FPGAs, respectively, while the largest instances target Artix-7 200 chips, i.e., the highest-end chips of the FPGA family.

The experimental results demonstrate how the proposed cryptographic cores can scale across a range of FPGA chips. Moreover, they show that BRAM memories are the most used resources, relatively to the ones available on the target chip, on the larger Artix-7 200 FPGAs, while instances targeting the smaller chips are bounded by the LUT utilization. The proposed architectures usually employ a large fraction of the available look-up tables while requiring a more limited amount of flip-flops.

Performance results Performance is measured by the execution time of the BIKE KEM primitives on the client and server sides of the key exchange. Table 1 lists the execution times, expressed in milliseconds, for the client and server instances of the proposed architecture, while Table 2 compares the aggregate execution times of BIKE between the state-of-the-art and proposed solutions.

The experimental results highlight significant improvements over the considered state-of-the-art references. The latency of the BIKE KEM can be reduced by almost two times, in the AES-192-equivalent use case, compared to the AVX2-optimized software execution, and the smaller proposed instances outperform even the mid-range state-of-the-art FPGA-based instances. Finally, the best-performing proposed architectures outperform the high-performance state-of-the-art ones by more than six times, as also shown in Fig. 5, which compares the execution time, broken down in the three KEM primitives, between the FPGA-based architectures.

5 Conclusions

This research presented a configurable FPGA-based hardware architecture that implements the BIKE QC-MDPC code-based cryptosystem, aiming to improve performance over the existing state-of-the-art software and hardware solutions.

The proposed architecture provides effective FPGA-based hardware support for QC-MDPC codes suitable to post-quantum cryptography applications. Configurable code and architectural parameters allow using a single design to support different QC-MDPC codes underlying the PQC cryptosystems and to target any FPGA chip from the Xilinx Artix-7 family. Hence, different performance-area trade-offs can be explored through the parametric configurability to satisfy the performance require-

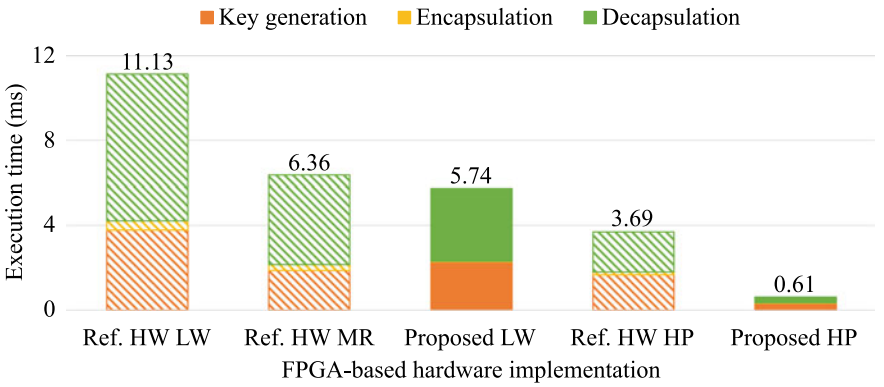


Fig. 5 Execution times of BIKE with AES-128-equivalent security. Legend: **LW** lightweight, **MR** mid-range, **HP** high-performance instances

ments and area constraints set for the overall system that integrates BIKE hardware support. Two modules support the KEM primitives to be executed on the client and server nodes of the key exchange, respectively, and a complexity-based heuristic steers the design space exploration to identify the best parameterization of the configurable hardware components by leveraging the knowledge of their time and space complexity.

The experimental evaluation of the proposed architecture highlighted significant improvements over the state-of-the-art software, hardware-software, and hardware implementations of BIKE from the literature. On the one hand, compared to the reference software implementation, which exploits the Intel AVX2 extension on desktop-class CPUs, AES-128- and AES-192-equivalent security instances of the proposed architecture provide performance speedups of 1.77× and 1.98×, respectively. On the other hand, the proposed FPGA-based BIKE architecture also outperforms the other hardware implementations available from literature, including both HLS-generated and human-designed ones, and provides a speedup over the fastest state-of-the-art FPGA-based instance of more than six times.

References

1. Agosta G, Aldinucci M, Alvarez C, Ammendola R, Arfat Y, Beaumont O, Bernaschi M, Biagioni A, Boccali T, Bramas B, Brandolese C, Cantalupo B, Carrozzo M, Cattaneo D, Celestini A, Celino M, Colonnelli I, Cretaro P, D'Ambra P, Danelutto M, Esposito R, Eyraud-Dubois L, Filgueras A, Fornaciari W, Frezza O, Galimberti A, Giacomini F, Goglin B, Gregori D, Guermouche A, Iannone F, Kulczewski M, Lo Cicero F, Lonardo A, Martinelli AR, Martinelli M, Martorell X, Massari G, Montangero S, Mittone G, Namyst R, Oleksiak A, Palazzari P, Paolucci PS, Reghenzani F, Rossi C, Saponara S, Simula F, Terraneo F, Thibault S, Torquati M, Turisini M, Vicini P, Vidal M, Zoni D, Zummo G (2022) Towards extreme scale technologies and accelerators for eurohpc hw/sw supercomputing applications for exascale: the textarossa approach. Microprocess Microsyst 95:104679. https://doi.org/10.1016/j.micpro.2022.104679. https://www.sciencedirect.com/science/article/pii/S0141933122002095

2. Amazon Web Services - Labs: Additional implementation of bike (bit flipping key encapsulation). https://github.com/awslabs/bike-kem (2020)
3. Aragon, N., Barreto PSLM, Bettaieb S, Bidoux L, Blazy O, Deneuville JC, Gaborit P, Gueron S, Güneysu T, Melchor CA, Misoczki R, Persichetti E, Sendrier N, Tillich JP, Vasseur V, Zémor G (2017) BIKE website. https://www.bikesuite.org/
4. Barenghi A, Fornaciari W, Galimberti A, Pelosi G, Zoni D (2019) Evaluating the trade-offs in the hardware design of the ledacrypt encryption functions. In: 2019 26th IEEE international conference on electronics, circuits and systems (ICECS), pp 739–742. https://doi.org/10.1109/ICECS46596.2019.8964882
5. Bernstein DJ (2006) Curve25519: new diffie-hellman speed records. In: Yung M, Dodis Y, Kiayias A, Malkin T (eds) Public key cryptography–PKC 2006. Springer, Berlin, pp 207–228
6. Bertoni G, Daemen J, Peeters M, Van Assche G, Van Keer R (2011) Keccak implementation overview. https://keccak.team/obsolete/Keccak-implementation-3.1.pdf
7. Chen MS, Chou T, Krausz M (2021) Optimizing bike for the intel haswell and arm cortex-m4. IACR Trans Cryptogr Hardw Embed Syst 2021(3):97–124. https://doi.org/10.46586/tches.v2021.i3.97-124, https://tches.iacr.org/index.php/TCHES/article/view/8969
8. Chen MS, Güneysu T, Krausz M, Thoma JP (2022) Carry-less to bike faster. In: Ateniese G, Venturi D (eds) Applied cryptography and network security. Springer International Publishing, Cham, pp 833–852
9. Comba PG (1990) Exponentiation cryptosystems on the IBM PC. IBM Syst J 29(4):526–538. https://doi.org/10.1147/sj.294.0526
10. Diffie W, Hellman M (1976) New directions in cryptography. IEEE Trans Inf Theory 22(6):644–654. https://doi.org/10.1109/TIT.1976.1055638
11. Drucker N, Gueron S, Kostic D (2020) Qc-mdpc decoders with several shades of gray. In: Ding J, Tillich JP (eds) Post-quantum cryptography. Springer International Publishing, Cham, pp 35–50
12. Dworkin M (2015) Sha-3 standard: permutation-based hash and extendable-output functions. https://doi.org/10.6028/NIST.FIPS.202
13. Fornaciari W, Agosta G, Cattaneo D, Denisov L, Galimberti A, Magnani G, Zoni D (2023) Hardware and software support for mixed precision computing: a roadmap for embedded and hpc systems. In: 2023 design, automation & test in Europe conference & exhibition (DATE), pp 1–6. https://doi.org/10.23919/DATE56975.2023.10137092
14. Galimberti A, Galli D, Montanaro G, Fornaciari W, Zoni D (2022) FPGA implementation of bike for quantum-resistant TLS. In: 2022 25th euromicro conference on digital system design (DSD), pp 539–547. https://doi.org/10.1109/DSD57027.2022.00078
15. Galimberti A, Galli D, Montanaro G, Fornaciari W, Zoni D (2022) On the use of hardware accelerators in qc-mdpc code-based cryptography. In: Proceedings of the 19th ACM international conference on computing frontiers. CF '22, Association for Computing Machinery, New York, NY, USA, pp 193-194. https://doi.org/10.1145/3528416.3530243, https://doi.org/10.1145/3528416.3530243
16. Galimberti A, Montanaro G, Fornaciari W, Zoni D (2023) An evaluation of the state-of-the-art software and hardware implementations of BIKE. In: Bispo Ja, Charles HP, Cherubin S, Massari G (eds) 14th workshop on parallel programming and run-time management techniques for many-core architectures and 12th workshop on design tools and architectures for multicore embedded computing platforms (PARMA-DITAM 2023). Open Access Series in Informatics (OASIcs), vol 107. Schloss Dagstuhl—Leibniz-Zentrum für Informatik, Dagstuhl, Germany, pp 4:1–4:12. 10.4230/OASIcs.PARMA-DITAM.2023.4, https://drops.dagstuhl.de/opus/volltexte/2023/17724
17. Galimberti A, Montanaro G, Zoni D (2022) Efficient and scalable FPGA design of GF(2m) inversion for post-quantum cryptosystems. IEEE Trans Comput 71(12):3295–3307. https://doi.org/10.1109/TC.2022.3149422
18. Galli D, Galimberti A, Fornaciari W, Zoni D (2022) On the effectiveness of true random number generators implemented on FPGAs. In: Orailoglu A, Reichenbach M, Jung M (eds) Embedded computer systems: architectures, modeling, and simulation. Springer International Publishing, Cham, pp 315–326

19. Itoh T, Tsujii S (1988) A fast algorithm for computing multiplicative inverses in GF(2m) using normal bases. Inf Comput 78(3):171–177. https://doi.org/10.1016/0890-5401(88)90024-7. https://www.sciencedirect.com/science/article/pii/S0141933122002095
20. Karatsuba A, Ofman Y (1962) Multiplication of many-digital numbers by automatic computers. Proc USSR Acad Sci 145:293–294
21. McEliece RJ (1978) A public-key cryptosystem based on algebraic coding theory. DSN Progress Report, pp 114–116 (1978)
22. Micciancio D, Regev O (2009) Lattice-based cryptography. In: Post-quantum cryptography, pp 147–191. Springer (2009)
23. Montanaro G, Galimberti A, Colizzi E, Zoni D (2022) Hardware-software co-design of bike with hls-generated accelerators. In: 2022 29th IEEE international conference on electronics, circuits and systems (ICECS), pp 1–4. https://doi.org/10.1109/ICECS202256217.2022.9970992
24. National Institute of Standards and Technology (NIST)—U.S. Department of Commerce: Nistir 8309, status report on the second round of the nist post-quantum cryptography standardization process (2020). https://nvlpubs.nist.gov/nistpubs/ir/2020/NIST.IR.8309.pdf
25. National Institute of Standards and Technology (NIST)—U.S. Department of Commerce: Nistir 8413, status report on the third round of the nist post-quantum cryptography standardization process. https://nvlpubs.nist.gov/nistpubs/ir/2022/NIST.IR.8413.pdf (2022). 10.6028/NIST.IR.8413
26. Nejatollahi H, Dutt N, Ray S, Regazzoni F, Banerjee I, Cammarota R (2019) Post-quantum lattice-based cryptography implementations: a survey. ACM Comput Surv 51(6). https://doi.org/10.1145/3292548, https://doi.org/10.1145/3292548
27. Peikert C (2009) Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In: Proceedings of the forty-first annual ACM symposium on theory of computing. STOC '09, Association for Computing Machinery, New York, NY, USA, pp 333–342. https://doi.org/10.1145/1536414.1536461, https://doi.org/10.1145/1536414.1536461
28. Richter-Brockmann J, Chen MS, Ghosh S, Güneysu (2021) Racing bike: improved polynomial multiplication and inversion in hardware. Cryptology ePrint Archive, Paper 2021/1344. https://eprint.iacr.org/2021/1344
29. Richter-Brockmann J, Mono J, Güneysu T (2021) Folding bike: scalable hardware implementation for reconfigurable devices. IEEE Trans Comput. https://doi.org/10.1109/TC.2021.3078294
30. Rivest RL, Shamir A, Adleman L (1978) A method for obtaining digital signatures and public-key cryptosystems. Commun ACM 21(2):120–126. https://doi.org/10.1145/359340.359342
31. Zoni D, Galimberti A, Fornaciari W (2020) Efficient and scalable FPGA-oriented design of QC-LDPC bit-flipping decoders for post-quantum cryptography. IEEE Access 8:163419–163433. https://doi.org/10.1109/ACCESS.2020.3020262
32. Zoni D, Galimberti A, Fornaciari W (2020) Flexible and scalable FPGA-oriented design of multipliers for large binary polynomials. IEEE Access 8:75809–75821. https://doi.org/10.1109/ACCESS.2020.2989423
33. Zoni D, Galimberti A (2022) Cost-effective fixed-point hardware support for risc-v embedded systems. J Syst Arch 126:102476. https://doi.org/10.1016/j.sysarc.2022.102476, www.sciencedirect.com/science/article/pii/S1383762122000595
34. Zoni D, Galimberti A, Fornaciari W (2021) An FPU design template to optimize the accuracy-efficiency-area trade-off. Sustain Comput: Inform Syst 29:100450. https://doi.org/10.1016/j.suscom.2020.100450, www.sciencedirect.com/science/article/pii/S2210537920301761
35. Zoni D, Galimberti A, Fornaciari W (2023) A survey on run-time power monitors at the edge. ACM Comput Surv. https://doi.org/10.1145/3593044

Model-Driven Development of Formally Verified Human-Robot Interactions

Livia Lestingi

Abstract Introducing service robots into everyday settings entails a significant technological shift for the robotics community. Service settings are characterized by critical sources of uncertainty (mainly due to human behavior) that current software engineering techniques do not handle. This chapter introduces a model-driven framework for developing interactive service robotic scenarios, relying on formal verification to guarantee robustness with respect to unexpected runtime contingencies. Target users specify the characteristics of the scenario under analysis through a custom textual Domain-Specific Language, which is then automatically converted into a network of Stochastic Hybrid Automata. The formal model captures non-traditional physiological (e.g., physical fatigue) and behavioral aspects of the human subjects. Through Statistical Model Checking, it is possible to estimate several quality metrics: if these meet the set dependability requirements, the scenario can be deployed. Specifically, the framework allows for deployment on the field or simulation. Field-collected data are fed to a novel active automata learning algorithm, called L^*_{SHA}, to learn an updated model of human behavior. The formal analysis can then be iterated to update the scenario's design. The overall approach has been assessed in terms of effectiveness and accuracy through realistic scenarios from the healthcare setting.

1 Introduction

Service robots are growingly experimentally deployed to carry out everyday tasks in coordination with humans in different settings, such as healthcare and domestic assistance. Unlike industrial settings, these contexts do not set significant boundaries for human actions. As a result, their behavior is substantially unconstrained and constitutes a critical source of uncertainty. Existing software engineering techniques privilege efficiency-related factors (e.g., the time it takes the robot to complete a task) and, according to practitioners, are not mature enough to handle this degree

L. Lestingi (✉)
Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, MI, Italy
e-mail: livia.lestingi@polimi.it

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_4

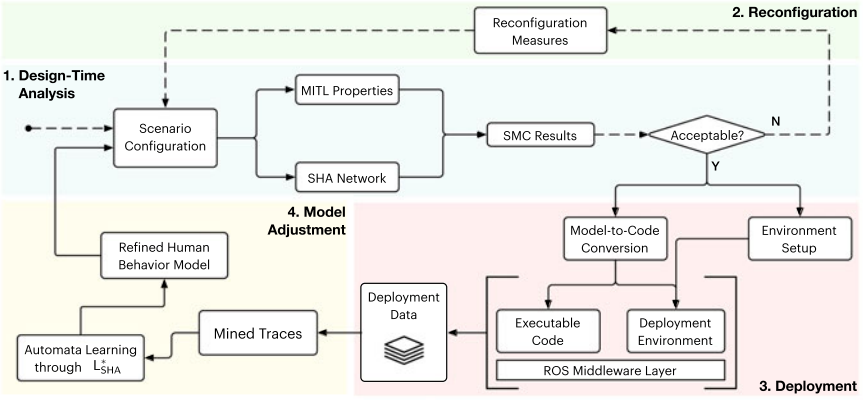


Fig. 1 Model-driven framework's workflow. Macro-phases are shown as colored areas. Blocks represent artifacts, and dashed arrows represent manual tasks

of variability [9]. On the other hand, decisions made at design time impact a hefty percentage of the software lifecycle costs [8], and their validity is questioned if unaccounted-for contingencies emerge at runtime.

This chapter addresses this methodological gap by proposing a model-driven framework for developing service robotic applications where human-robot interaction is a crucial element [14]. Figure 1 gives an overview of the proposed methodology. The framework is devised for practitioners who do not necessarily have prior training in software development; therefore, it supports them throughout all phases, from design to testing and maintenance, while keeping the required manual effort to a minimum degree. The proposed methodology relies, at its core, on formal modeling and verification techniques to develop robotic scenarios with guarantees of robustness to the mentioned sources of uncertainty. Therefore, the framework also puts into question the claim that human behavior modeling falls beyond the limits of formal modeling techniques [17].

Existing works explore the possibility of formalizing human-robot interaction and exploiting the guarantees of formal analysis within the software development process. Previous attempts mostly focus on ensuring that collaborative applications meet safety standards [22] or comply with social norms [19]. The literature also features formalizations of human behavior, for example, by modeling the system as a network of Timed Game Automata [4] or adopting a probabilistic approach, as in the hereby presented work, while focusing on smaller setups [23].

In this work, the robotic scenario is first analyzed offline (macro-phase **1** in Fig. 1) through a custom textual Domain-Specific Language (DSL) [15]. The DSL file is then automatically converted into a formal model capturing the agents involved in the scenario, i.e., the robots and the humans they interact with. As for the latter, in line with the goal of tackling uncertainty, the formal model captures non-traditional aspects of human physiology and their decision-making process. The significance

of physiological aspects primarily relates to the healthcare setting since subjects may be in pain or discomfort, impacting their ability to carry out tasks in coordination with the robots. Incorporating a formalization of the human decision-making process, specifically as a stochastic process, into the model is necessary to have its impacts considered by the formal analysis. Such modeling requirements motivate the choice of Stochastic Hybrid Automata (SHA) as the selected formalism [7]. Given the stochastic nature of the model, the framework then applies Statistical Model Checking (SMC) [1] to estimate quality metrics expressed as Metric Interval Temporal Logic (MITL) properties about the robotic application to be examined by the practitioner.

If the SMC results are not satisfactory, the design must be revised by applying reconfiguration measures (macro-phase **2** in Fig. 1). Otherwise, if such metrics satisfy the practitioner's expectations, the so-obtained design is either deployed on the field or simulated for further investigation (macro-phase **3** in Fig. 1). To this end, the framework introduces a deployment approach with a model-to-code mapping principle to guarantee correspondence between the formal model and the software components deployed at runtime [13]. So-collected data (i.e., either actual sensor logs or simulation logs) are then exploited to learn an updated model of human behavior (macro-phase **4** in Fig. 1) [14]. To this end, a novel active automata learning algorithm, called L^*_{SHA}, targeting SHA is fed with traces (i.e., event sequences) mined from field data. The learned SHA is plugged back into the formal model to either revise the design of the same scenario or in preparation for the development of future ones.

Previous works present learning algorithms for Hybrid (HA) or Probabilistic Automata. Medhat et al. present a framework for HA mining based on clustering [18]. Works focusing on probabilistic systems adopt a frequentist approach [10] or a state merging method [5]. Tappler et al. also propose an extension of L^* to learn Markov Decision Processes based on collected samples [21]. The L^*_{SHA} algorithm contributes to the area as it targets both hybrid and stochastic features.

All the phases of the framework have been experimentally validated on scenarios inspired by the healthcare setting. Experiments aimed at assessing the accuracy (thus, the reliability of the results) of the different artifacts, the flexibility of the framework with respect to realistic service robotic applications, and its capability to mitigate the sources of uncertainties at play. The key results of experimental validation are reported in this chapter, whereas we refer the interested reader to dedicated publications for a detailed report.

The rest of the chapter is structured as follows. Section 2 outlines the main theoretical concepts underlying the work. Each macro-phase is then illustrated in detail, specifically: Sect. 3 describes the design-time analysis phase; Sect. 4 describes the deployment framework; and Sect. 5 describes the model adjustment phase. Finally, Sect. 6 presents future research directions.

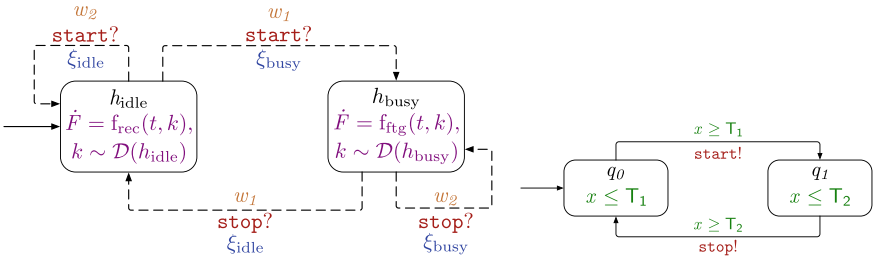
2 Preliminaries

As per Sect. 1, the chosen formalism is SHA, an extension of Timed Automata with hybrid and stochastic features. SHA *locations*, which belong to set L, capture the different operational states of the system under analysis. Figure 2a shows an example of SHA modeling human behavior with locations $L = \{h_{\text{idle}}, h_{\text{busy}}\}$, representing the human standing still and walking, respectively.

In Hybrid Automata (HA), set W contains real-valued variables whose time dynamics are constrained through sets of generic ODEs, called *flow conditions* [2], modeling complex physical behaviors. Given location $l \in L$, function $\mathcal{F}(l)$ assigns flow conditions to l constraining the behavior of real-valued variables in W while in such location. As an example, real-valued variable $F \in W$ in Fig. 2a captures the human's physical fatigue whose time derivative $\dot{F}$ is constrained by functions $\mathrm{f}_{\text{rec}}(t, k)$ and $\mathrm{f}_{\text{ftg}}(t, k)$. In Stochastic HA, given flow condition $f(t, k) \in \mathcal{F}(l)$ where t represents time and k is an independent, randomly distributed parameter, $f(t, k)$ acts as a stochastic process and its domain is $\mathbb{R}_+ \times \mathbb{R}$. For each location $l \in L$, function $\mathcal{D}(l)$ assigns a *probability distribution* to l governing random parameter k (e.g., $\mathcal{D}(h_{\text{idle}})$ and $\mathcal{D}(h_{\text{busy}})$ in Fig. 2a).

SHA *edges* capture transitions between two locations and are labeled with the *event* triggering the transition and, possibly, a *guard condition* and an *update*, expressed in terms of variables in W. Guard conditions (e.g., $x \geq \mathsf{T}_1$ in Fig. 2b) enable the firing of the edge when, given the current value of variables in W, they are verified. Updates are sets of assignments to variables in W that are executed when the edge fires. In SHA, assignments may entail the extraction of a sample from a probability distribution. For example, when the SHA in Fig. 2 switches from h_{busy} to h_{idle}, update ξ_{idle} assigns a sample from $\mathcal{D}(h_{\text{idle}})$ to k.

Given *channel* c, an edge can be labeled either with $\mathrm{c}!$ if the SHA actively triggers an event through c or with $\mathrm{c}?$ if the SHA listens for events on c. Multiple SHA in a *network* (e.g., the human in Fig. 2a and the *orchestrator* in Fig. 2b) synchronize through channels when complementary edges fire simultaneously. For example, the



(a) Example SHA modeling human behavior. (b) Example orchestrator SHA.

Fig. 2 Example SHA network: invariants and flow conditions are in purple, channels in red, probability weights in orange, guards in green, and updates in blue

SHA in Fig. 2 (i.e., the machine) may switch from h_{idle} to h_{busy} when the orchestrator fires an event through channel `start`. In SHA, edges may also be associated with *probability weights* (i.e., dashed arrows in Fig. 2a) determining the bias of the network toward a certain transition: for example, when event `start`! fires, the SHA in Fig. 2a switches to h_{busy} with probability $p = w_1/(w_1 + w_2)$ and stays in h_{idle} with probability $1 - p$.

A location l of an SHA can be endowed with an *invariant*, i.e., a condition over variables in W that must hold as long as the SHA is in l. In Fig. 2b, the combination of invariants and guards on outgoing edges ensures that edges fire exactly when $x = \mathsf{T}_i$, $i \in \{1, 2\}$ holds.

SHA are eligible for SMC, which can be performed, for example, through the Uppaal tool [12]. SMC generates multiple runs of an SHA network M through the Monte-Carlo simulation technique, each simulating the evolution of the system for a given time $\tau \in \mathbb{N}$. These runs are then individually examined to check whether a given MITL property ψ holds, thus constituting a set of Bernoulli trials. The value of expression $\mathbb{P}_M(\psi)$ then corresponds to the confidence interval for the probability of property ψ holding for network M within time τ. By simulating the SHA network, it is also possible to calculate the expected maximum/minimum value of real-valued variables, such as the humans' physical fatigue or the robot's residual charge.

3 Design-Time Analysis and Reconfiguration

The entry point to the model-driven framework is the analysis of the robotic scenario at design time (thus, offline). The goal of this phase is to specify the characteristics of the scenario and subsequently compute quality metrics through formal analysis.

The set of characteristics that can be expressed through the custom DSL constitutes the conceptual model underlying the framework, which is summarized in the following and exemplified through an illustrative use case from the healthcare set-

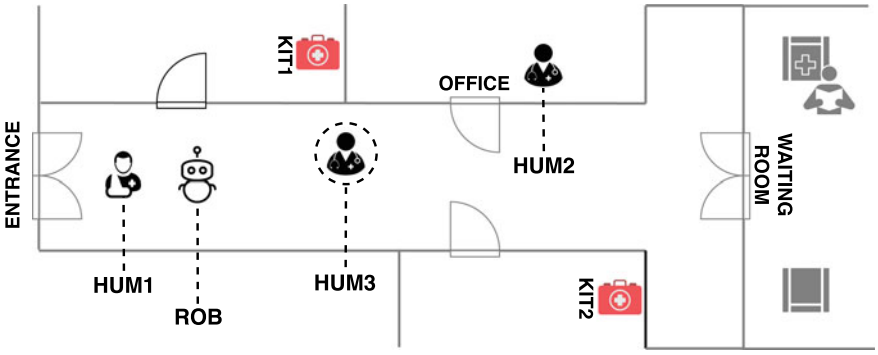


Fig. 3 Layout for the illustrative example representing the agents (in their initial positions) and the POIs. HUM3's initial position is randomized

ting (see Fig. 3). Firstly, the geometrical *layout* where agents will operate has to be defined (e.g., the T-shaped corridor in Fig. 3 with doors to four rooms). The layout can include points of interest (POIs) agents can interact with (e.g., cupboards with medical equipment KIT1 and KIT2 in Fig. 3). Agents are either robots or humans. Mobile robots can be of different commercial models, which determines their technical specifications. Humans have different physiological features (including their age group and health status) and behavioral traits (for example, their level of attentiveness). The example in Fig. 3 features four agents: one robot (ROB) and three humans (HUM1, HUM2, and HUM3).

Having defined the characteristics of the agents, it is necessary to configure the scenario. In this work, a scenario is intended as a composition of robotic *missions*, where each mission is an ordered sequence of *services*. A service represents a task requiring coordination between the human and the robot with a target in space (i.e., a POI), which must conform to a *pattern*. Patterns group recurrent human-robot interaction contingencies, such as the human *following* the robot to a destination or the human and the robot *competing* for the same resource [15]. The mission for the example in Fig. 3 begins with HUM1 *following* the robot to the waiting room until the examination room is appropriately set up. While HUM1 is waiting, the robot and HUM3 *compete* for the same medical kit (KIT1 in Fig. 3): the mission then follows alternative plans depending on the outcome of the competition. If the robot retrieves the resource first, it *delivers* it to HUM2 in the office. Otherwise, HUM2 *leads* the robot to retrieve another medical kit (KIT2 in Fig. 3). Once HUM2 is ready for the visit, the robot *leads* HUM1 to the office and *assists* HUM2 in administering the medication.

The so-obtained DSL file is then automatically converted into an intermediate JSON notation that decouples DSL parsing from the specific verification tool (see Fig. 4). At the current stage of development, the available component that generates the formal model targets SHA as chosen formalism and Uppaal as the verification tool. The generated SHA network is schematically represented in Fig. 4. The network consists of $\mathsf{N_h}$ SHA modeling human behavior for each subject ($\mathcal{A}_{h_i}$ with $i \in \{1, \mathsf{N_h}\}$) and $\mathsf{N_r}$ SHA for each robotic system, made up of the SHA for the robotic platform,

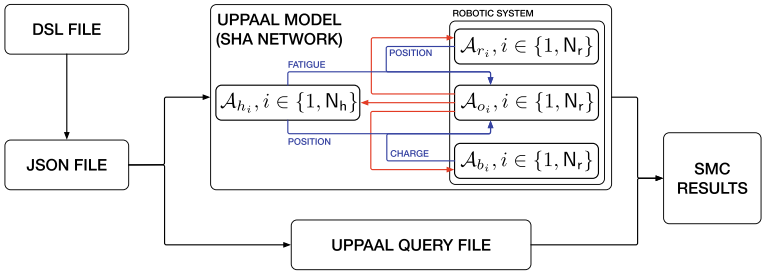


Fig. 4 Design time phase workflow. In the SHA network, blue arrows represent sensor readings shared by the agents with the orchestrator, while red arrows represent decisions made by the orchestrator and communicated to the agents

the battery, and the orchestrator ($\mathcal{A}_{r_i}$, $\mathcal{A}_{b_i}$, and $\mathcal{A}_{o_i}$ with $i \in \{1, \mathsf{N_r}\}$, respectively). The latter acts as the robot controller; specifically, agents periodically share their position within the layout and further data about their current status (i.e., the fatigue level for humans and residual battery charge for robots). The orchestrator examines the latest batch of sensor readings and checks it against its policies to send commands to the agents (e.g., start or stop walking) aiming for the completion of all the services in the mission.

Quality metrics to be computed for the scenario are referred to as *queries*. Possible queries include the probability of completing the mission within a specific time, the probability of critical events (e.g., the human getting fully fatigued or the robot fully discharged) occurring within a specific time, and the estimated value of relevant physical variables captured by the model.

SMC experiments are automatically launched, and their results are provided to the scenario designer. If such results satisfy the designer's expectations, the application can move forward to the deployment macro-phase. Otherwise, as per Fig. 1, the design of the scenario can be revised to improve the desired indicators. Possible reconfiguration measures include selecting different robots from the fleet, changing the order in which services are provided (unless logical dependencies exist), and re-tuning the orchestrator's policies (e.g., a less conservative orchestrator may lead to faster mission completion while requiring more effort on the humans' side).

Realistic service robot scenarios from the literature or industrial use cases have been collected to assess the coverage of the design-time analysis phase (i.e., whether they would be analyzable through the framework or fall out of its scope) [15]. Results show that 24 out of 27 scenarios are analyzable through the framework, leading to a coverage rate of 88%. The accuracy of the formal model (specifically, the SHA modeling the robotic system) has been assessed with respect to field-collected data on 6 scenarios, resulting in estimation errors up to 6.7% for the probability of success, 0.61% for the robot's charge, and 8.6% for human fatigue.

4 Application Deployment

Once the design of the robotic scenario satisfies the set of requirements, the application can be deployed on the field or simulated in a virtual environment. At this stage, it is paramount to guarantee to a degree the correspondence between the formal model and the behavior of the system at runtime. To this end, the deployment phase entails the mapping of SHA features to *deployment units* constituting the deployment framework [13]. Units either consist of simulator scripts governing the behavior of agents in the virtual scene or low-level components controlling the robotic device and the sensors worn by human subjects. The application of the mapping principle to recurring modeling patterns results in recurring code patterns capturing, for example, the periodic refresh of sensor readings and consequent sharing with the orchestrator deployment unit.

The resulting deployment infrastructure features a deployment unit for each agent and a standalone unit for each orchestrator (one for each robot in the fleet). Orchestrators communicate with agents over a middleware layer based on ROS publisher/subscriber nodes [20]. Each sensor associated with an agent corresponds to a ROS publisher node that periodically shares over dedicated topics the latest reading, to whom the orchestrator subscribes. Correspondingly, the orchestrator's commands are transmitted over dedicated ROS topics with the agents. The ROS-based middleware layer decouples the orchestrators from the specific technology exploited for the agents' deployment unit, constituting a standard communication interface. As a result, the deployment framework flexibly supports physical agents, simulated agents, and a hybrid setting (e.g., physical robots synchronizing with human subjects in the simulation scene).

The deployment framework has also been tested in terms of accuracy, specifically regarding the correspondence between formal analysis results and the behavior observed at runtime (thus, the accuracy of the model-to-code mapping principle) [13]. Results show a deviation of physical variables values obtained through SMC and simulation of up to 5.35%: given that, at the time of writing, no standardization exists for acceptable thresholds in service robot applications, whether such values meet the facility's requirements is up to the stakeholder.

5 Model Adjustment

For the first design-time analysis iteration, it can be assumed that the SHA modeling human behavior within the formal model is an *underapproximation* of real human behavior. Therefore, upon deploying the mission with real human subjects who perform a broader range of actions, it is plausible that results obtained at design time are found to be no longer accurate. To address this issue, sensor data can be exploited as part of a data-driven learning technique of a model of human behavior that is up-to-date with the knowledge accumulated through deployment.

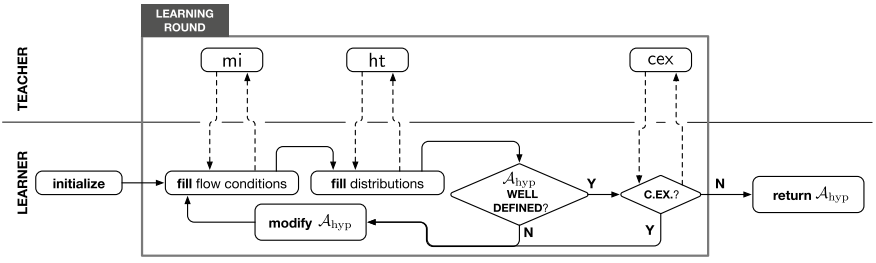


Fig. 5 High-level workflow of L^*_{SHA} split into the teacher's and the learner's lanes. Dashed arrows represent the submission of a query and the retrieval of the teacher's answer

To this end, an active automata learning algorithm targeting SHA has been developed (see Fig. 5 for the algorithm's workflow) [14]. The algorithm extends the well-known algorithm for Deterministic Finite-state Automata (DFA) learning L^* [3], hence the name $\mathsf{L}^*_{\mathrm{SHA}}$. $\mathsf{L}^*_{\mathrm{SHA}}$, like L^*, relies on the interaction between a *learner* and a *teacher* (or *oracle*). The learner is in charge of maintaining the hypothesis automaton $\mathcal{A}_{\mathrm{hyp}}$, while the teacher stores the available knowledge about the System Under Learning (SUL). The learner submits *queries* to the teacher and refines the hypothesis based on the teacher's answers.

Knowledge stored by the teacher is in the form of signals collected by sensors. While $\mathsf{L}^*_{\mathrm{SHA}}$ is domain-agnostic, in its application to human behavior learning such signals consist of the agents' positions, human physical fatigue, and data concerning the environment (e.g., humidity and temperature). As per the example in Fig. 2, human behavioral states differ based on how fatigue evolves (e.g., it increases while walking and decreases while resting). Therefore, to identify SHA locations correctly, in this use case, the fatigue signal is split into segments based on *events* that occurred during the mission (e.g., human velocity switching from 0 to a value greater than 0 indicates that the human *started walking*). A sequence of events constitutes a *trace*. In $\mathsf{L}^*_{\mathrm{SHA}}$, the teacher stores all collected traces and the associated signals.

As per Fig. 5, learning occurs in *rounds*. At the beginning of each round, since the algorithm targets SHA, each location $l \in L$ of $\mathcal{A}_{\mathrm{hyp}}$ that has already been identified needs to be labeled with flow conditions and probability distributions (i.e., functions $\mathcal{F}(l)$ and $\mathcal{D}(l)$, respectively). For flow conditions, the learner submits mi queries, which exploit the Derivative Dynamic Time Warping (DDTW) technique [11] to identify the function out of a set of candidates that best fits a specific signal segment. Concerning probability distributions, the learner submits ht queries for the teacher to determine through a Kolmogorov-Smirnov two-sample test whether the samples of a random parameter observed in the aftermath of a specific trace constitute a new population or they are not statistically different from previously identified populations [16].

After assigning all flow conditions and probability distributions, the learner checks whether $\mathcal{A}_{\mathrm{hyp}}$ is well-defined, that is whether it is *closed* and *consistent*. The hypothesis is closed if all edges reach existing locations (in other words, if a location is defined for each identified operational state). The hypothesis is consistent if no location has more than one outgoing edge with the same event label (in other words, if the SHA is deterministic with respect to edge outputs). If either of the two conditions is not verified, the learner modifies $\mathcal{A}_{\mathrm{hyp}}$ to make it closed and consistent.

Once $\mathcal{A}_{\mathrm{hyp}}$ is well-defined, the current learning round ends with the learner submitting a cex query to the teacher, that is asking whether, given the teacher's knowledge, a counterexample to $\mathcal{A}_{\mathrm{hyp}}$ exists. A counterexample is a trace known to the teacher but which is not captured by $\mathcal{A}_{\mathrm{hyp}}$ or is not compatible (i.e., is a source of non-closedness or non-consistency). If a counterexample exists, a new round of learning is necessary; otherwise, $\mathsf{L}^*_{\mathrm{SHA}}$ terminates returning $\mathcal{A}_{\mathrm{hyp}}$.

Within the model-driven framework, the learned SHA constitutes a refinement of the model of human behavior, which is plugged back into the SHA network to iterate the design-time analysis for the same or different scenarios. Experiments have been

carried out by simulating a broader range of human actions (e.g., running, sitting, and walking while carrying a load) through the simulated deployment environment to assess the gain in accuracy with the refined model [14]. The latter amounts to an average 18.1% accuracy gain for the estimation of the probability of success and 7.7% for the estimation of fatigue. Naturally, better accuracy comes at the cost of the time necessary to complete the learning (approximately 35min for the largest model) and the increase in complexity of the resulting SHA network (thus, in verification time).

6 Future Research Outlook

The framework is open to several future extensions. Ongoing work involves a refinement of the SHA network with cognitive and psychological models of the human decision-making process to be accounted for by the formal analysis and the deployed orchestrator.

As for the model adjustment phase, $\mathsf{L}^*_{\mathrm{SHA}}$ works under a set of simplifying assumptions, which may not be realistic with real CPSs. Therefore, further work is necessary to extend the applicability domain of $\mathsf{L}^*_{\mathrm{SHA}}$. In more general terms, the degree of applicability of active automata learning techniques (which usually rely on the availability of an omniscient oracle to drive the learning) to real systems is an open research question.

Finally, the reconfiguration phase of the framework is now performed entirely manually, which is not aligned with the initial goal of keeping the automation level as high as possible. An automated strategy synthesis procedure (for example, exploiting the Uppaal Stratego tool [6]) could be developed to compute alternative mission plans that optimize the quality metrics, thus at least partially automating the mission redesign task.

References

1. Agha G, Palmskog K (2018) A survey of statistical model checking. TOMACS 28(1):1–39
2. Alur R, Courcoubetis C, Halbwachs N, Henzinger TA, Ho PH, Nicollin X, Olivero A, Sifakis J, Yovine S (1995) The algorithmic analysis of hybrid systems. TCS 138(1):3–34
3. Angluin D (1987) Learning regular sets from queries and counterexamples. Inf Comput 75(2):87–106
4. Bersani MM, Soldo M, Menghi C, Pelliccione P, Rossi M (2020) PuRSUE-from specification of robotic environments to synthesis of controllers. Form Asp Comput 32(2):187–227
5. Carrasco RC, Oncina J (1994) Learning stochastic regular grammars by means of a state merging method. In: International colloquium on grammatical inference. Springer, pp 139–152
6. David A, Jensen PG, Larsen KG, Mikučionis M, Taankvist JH (2015) Uppaal stratego. In: International conference on tools and algorithms for the construction and analysis of systems. Springer, pp 206–211

7. David A, Larsen KG, Legay A, Mikučionis M, Poulsen DB, Van Vliet J, Wang Z: Statistical model checking for networks of priced timed automata. In: International conferences on formal modeling and analysis of timed systems. Springer, pp 80–96
8. Fraunhofer Institute for Manufacturing Engineering and Automation: EFFIROB: economic feasibility studies on innovative service robot applications (2010). https://www.ipa.fraunhofer.de/en/reference_projects/EFFIROB.html
9. García S, Strüber D, Brugali D, Berger T, Pelliccione P (2020) Robotics software engineering: a perspective from the service robotics domain. In: ESEC/FSE. ACM, USA, pp 593–604
10. Ghezzi C, Pezzè M, Sama M, Tamburrelli G (2014) Mining behavior models from user-intensive web applications. In: International conferences on software engineering, pp 277–287
11. Keogh EJ, Pazzani MJ (2001) Derivative dynamic time warping. In: International conferences on data mining. SIAM, pp 1–11
12. Larsen KG, Pettersson P, Yi W (1997) UPPAAL in a nutshell. Int J Softw Tools Tech Transf 1(1–2):134–152
13. Lestingi L, Askarpour M, Bersani MM, Rossi M (2021) A deployment framework for formally verified human-robot interactions. IEEE Access 9:136616–136635
14. Lestingi L, Bersani MM, Rossi M (2022) Model-driven development of service robot applications dealing with uncertain human behavior. IEEE Intell Syst
15. Lestingi L, Zerla D, Bersani MM, Rossi M (2023) Specification, stochastic modeling and analysis of interactive service robotic applications. Robot Auton Syst 104387
16. Lilliefors HW (1967) On the Kolmogorov-Smirnov test for normality with mean and variance unknown. J Am Stat Assoc 62(318):399–402
17. Luckcuck M, Farrell M, Dennis LA, Dixon C, Fisher M (2019) Formal specification and verification of autonomous robotic systems: a survey. ACM Comput Surv 52(5):100:1–100:41
18. Medhat R, Ramesh S, Bonakdarpour B, Fischmeister S (2015) A framework for mining hybrid automata from input/output traces. In: EMSOFT. IEEE, pp 177–186
19. Porfirio D, Sauppé, A., Albarghouthi A, Mutlu B (2018) Authoring and verifying human-robot interactions. In: Proceedings of the 31st annual ACM symposium on user interface software and technology, pp 75–86
20. Quigley M, Conley K, Gerkey B, Faust J, Foote T, Leibs J, Wheeler R, Ng AY (2009) ROS: an open-source robot operating system. In: ICRA workshop on open source software, vol 3, p 5
21. Tappler M, Aichernig BK, Bacci G, Eichlseder M, Larsen KG (2019) L*-based learning of Markov decision processes. In: International symposium on formal methods. Springer, pp 651–669
22. Vicentini F, Askarpour M, Rossi MG, Mandrioli D (2019) Safety assessment of collaborative robotics through automated formal verification. IEEE Trans Robot. https://doi.org/10.1109/TRO.2019.2937471
23. Webster M, Western D, Araiza-Illan D, Dixon C, Eder K, Fisher M, Pipe AG (2020) A corroborative approach to verification and validation of human-robot teams. Int J Robot Res 39(1):73–99

Electronics

Electronic Bio-Reconfigurable Impedance Platform for High Sensitivity Detection of Target Analytes

Paola Piedimonte

Abstract The present research presents a portable bioelectronic platform for multiplex detection to read biosensor chips with several sensing sites for real-time analyte capture. The technique is based on Differential Impedance Sensing (DIS) of the target through functionalized nanoparticle amplification. Gold-interdigitated microelectrodes are the core of the biosensing system. They are designed in a differential configuration, reference and active sensor, to counteract all possible mismatches such as temperature fluctuations and variations in the ion content of the solution. The surface of the sensor is biochemically functionalized with a synthetic probe specifically developed for the selected target. The successful combination of all of these elements allowed the system to detect IgG antibodies spiked in buffer with a limit of detection of below 100 pg/mL. In a real case study for viral infection diagnosis, the system has been challenged with infected human serum samples for digital counts of anti-dengue virus antibodies, achieving the detection of clinically relevant target concentrations. Also, the bio-reconfigurability of the system has been successfully tested with oligonucleotide detection down to pM target concentration. To allow the portability of the entire measurement setup, the setup has been equipped with a custom electronic board based on an FPGA module allowing a multiplexing approach for the parallel reading of several electrodes. The final system provides simple and effective bio-reconfigurability, exploiting advances in bio-recognition through proper probe selection and boosting the possible use of multiplex sensing to a broad spectrum of needs.

1 Introduction

Detection of viruses is essential for the control and prevention of viral infections. For the diagnosis, it is possible to directly detect the whole virus [1] or to determine the antibodies produced against virus proteins during and/or after the virus incubation period [2, 3]. The analytes as viral nucleic acids (DNA and RNA), viral proteins,

P. Piedimonte (✉)
Department of Electronics Information and Bioengineering, Politecnico di Milano, Piazza L. da Vinci 32, 20133 Milan, Italy
e-mail: paola.piedimonte@polimi.it

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_5

intact viral particles and antibodies can all be used for the diagnosis [4]. Conventionally, they are detected using traditional methods like polymerase chain reaction (PCR) [5], virus culture, and enzyme-linked immunosorbent assay (ELISA) [6]. Still, these tests are usually time and reagent-consuming and do not have multiplex capability, thus not allowing the detection of several targets simultaneously [7]. In recent years, there has been a focus on simpler and faster detection methods with biosensors solutions. Fundamental to these systems is the biotransducer that converts the concentration of the targeted analyte into a proportional signal as a combination of a biological recognition layer and a physical transducer. Among the transduction mechanisms, biosensors can rely on dyes or fluorescent labels [8, 9] or innovative approaches such as functionalized tapered optical fiber [10], silicon microring resonators for active opto-magnetic biosensing [11] or Surface Plasmon Resonance-based technologies (SPR) [12]. These techniques have further improved the quality of the measurement but still require complex components and can be hardly miniaturized into a multipurpose platform. To overcome these limitations, an electrical approach using impedimetric biosensors can be convenient. As a device, planar interdigitated electrodes (IDEs), which consist of a pair of comb-shaped micro-band array electrodes and a pad at the bottom, are widely applied to the field of sensors due to their simple manufacturing process. They can be miniaturized, integrated in a microfluidic system and fabricated with bio-compatible materials. The currents/voltages between the IDEs can be evaluated to directly detect binding [13] or indirectly detect the change of the electrical properties (complex impedance, resistance or capacitance) of the medium around the electrodes [14]. Between non-faradaic [15] and faradaic [16] approach, the latter application requires a wide-range frequency sweep and the use of potentially hazardous redox mechanisms for current induction resulting in complex experimental protocol and electronic reading system. Non-faradaic impedance systems can be used in the simpler fixed frequency to provide transient information [17] about rapid changes in impedance thanks to increasing sampling [18, 19]. As the main result of this thesis, it has been demonstrated for the detection of Dengue Virus infection a non-faradaic based biosensor system that combines a specific biochemistry recognition mechanism with an electronic platform based on Differential Impedance Sensing [20]. Due to the target features, the impedance signal is amplified by hybridizing the probe-target with a functionalized polystyrene nanoparticle. The increased signal-to-noise ratio helps to overcome drawbacks such as laborious surface preparation, the usually limited storage time of the sensor and surface biofouling that occurs in unpurified samples. Moreover, the single-frequency measurements and data analysis on the enhanced signal simplify the platform hardware and software, enabling portable analysis.

2 The Differential Impedance Sensing Approach

2.1 Platform Overview and Experimental Protocol

The sensors based on IDEs have been designed in a differential configuration in order to evaluate the total value of the impedance and the beads contribution. In this structure, the selected probe is spotted only over the active sensor, while the reference sensor operates as a negative control without surface functionalization. The resulting measurement is unaffected by macroscopic temperature fluctuations and variations in the liquid medium (i.e., ion concentration) that determine the assay's conductance. A schematic representation of the biosensing system is shown in Fig. 1. The gold IDEs have fingers of 3-μm width, 90-μm length and 3-μm spacing. The active area of each sensor (90 μm × 90 μm) and the comb dimensions of 3 μm have been optimized considering the resolution of the spotting machine for the functionalization of the sensor surface with the probe, the full coverage of the active area and the beads dimension (800 nm).

The IDEs detection properties have been optimized and validated using finite element method simulations (COMSOL Multiphysics) to reach the maximum impedance variation for one-bead detection. The resulting sensing area of the IDE leads to a dynamic range of 100 ppm. A single bead in the sensing volume over the electrodes would give a signal variation of about 15 mΩ (over 935 Ω of total impedance), allowing a direct correlation between the electrical signal, the number of beads and the number of biological targets captured by the probes. The fabrication process of the electrodes was carried out at PoliFAB (the micro and nanotechnology center of the Politecnico di Milano) on 3-inch borosilicate wafers using standard microfabrica-

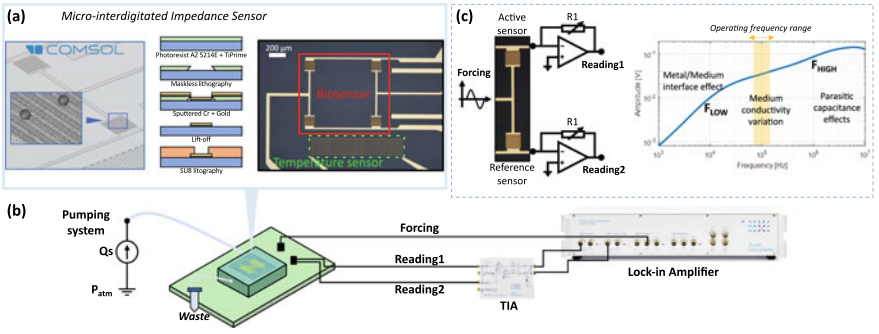


Fig. 1 The Biosensing system. **a** The workflow for electrode realization: COMSOL simulation of the differential IDE structure, process steps for microfabrication of gold IDE and photographs of the final biosensor structure. **b** A schematic view of the microfluidic system of the differential electronic impedance measurement setup. **c** shows the configuration to perform the impedance measurement in a digital differential mode and the full spectrum of impedance versus frequency in PBS solution. The selected working frequency around 1 MHz ensures that we are working in a region sensitive to variations in the liquid conductivity

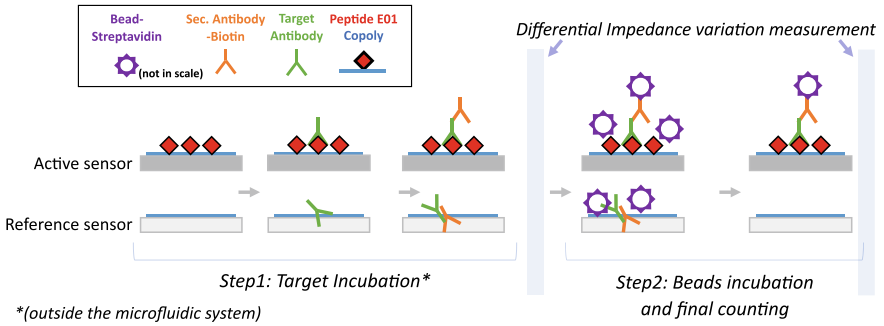


Fig. 2 Scheme of the protocol of the assay. The impedance measurement to take into account for the differential detection are the ones before and after the beads injection in buffer condition

tion techniques. The interdigitated gold electrodes are exposed to liquid with micro windows patterned on a SU8 layer (thickness of 5 μm) used as a passivation layer over metal connections to limit leaky current paths.

The experimental protocol, reported in Fig. 2, consists of an *incubation phase* targeting specific primary antibody binding and labeling by secondary antibody, followed by a *beads counting phase* targeting nanobeads binding, impedance measurement and beads count. While this second phase is always performed in the microfluidic chamber, the incubation phase can also be conducted outside the microfluidic system.

The limit of detection (LOD) of the system has been investigated and the same protocol has been replicated for different concentrations of primary antibody, from 50 μg/ml down to 100 pg/ml. The LOD was extrapolated from the impedance value of blank samples plus 3 standard deviations (3σ) using the linear range of the calibration curves.

$$LOD = 3.3 \cdot \frac{\sigma}{S} \tag{1}$$

where S is the calibration curve in Fig. 3 and σ is the standard deviation of impedance background in the control sample.

Figure 3 summarizes these results. For antibody concentrations above 50 μg/mL, no significant change was observed due to the saturation of the binding sites with anti-IgG antibodies. On the other extreme, the system demonstrates to operate down to a concentration of 100 pg/mL with signals (700 mΩ) well above the noise ground limit of 150 mΩ as present at the output of the lock-in amplifier.

For the negative controls, the same protocol has been applied with zero concentration of primary antibody, resulting in a mean value of 372 ± 90 mΩ, well below the signal measured at the minimum concentration of 100 pg/mL. Figure 3 also shows the photographs of the active sensor surface taken at the end of the measurement after drying the chip. They clearly exhibit the corresponding decrease in beads density on the active area of the sensors as the primary antibody concentration decreases. The

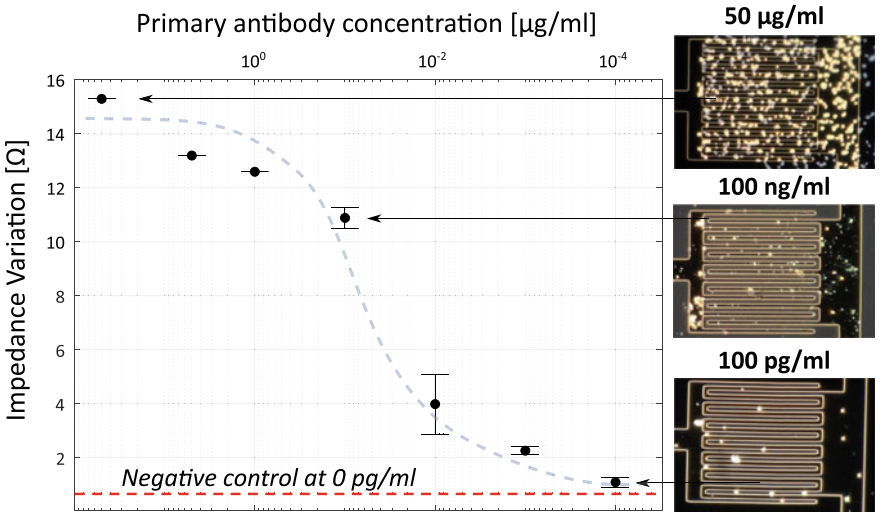


Fig. 3 Impedance variation as a function of serial dilutions of anti-E01 IgG primary antibody at decreasing concentrations to assess the platform sensitivity in terms of LOD. Error bars show standard error. The figures show the coverage of the active sensor by beads. All measurements have been performed with a 50 mV amplitude signal at 1 MHz

platform tracks the number of beads that can be counted by impedance measurement down to a few tens of beads, certifying the immunosensor concept and operation.

2.2 Detection of IgG Antibodies in Human Serum for Dengue Virus Diagnosis

To validate the clinical effectiveness and robustness of the adopted technique, the biosensing system has been tested for Dengue Virus detection. The target biomarkers for this experiment are the IgG antibodies anti-DNGV in human sterilized serum samples of infected patients. As negative control in order to avoid false-positive results, human serum samples from healthy patients negative to IgG antibodies anti-DNGV have been tested as well. The protocol used is the standard one described in the previous section and provides in addition all the stages of incubation of the serum for probe-target hybridization inside the microfluidic chamber. Figure 4 summarizes the measurements at different dilutions of the serum sample positive to anti-Dengue antibody (from 1:25 to 1:100000). Healthy serum samples were included in the analysis as negative controls. Negative samples show a ground noise that we consider as a cut-off signal, with a differential impedance value below 1.23 Ω. As a consequence, the 1:100000 serum dilution provides the minimum detectable signal. The photographs of the active sensor surface visually certify the effectiveness of the

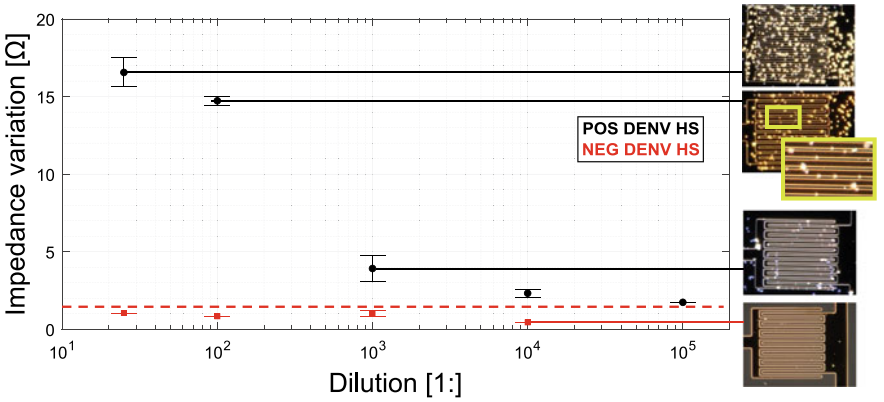


Fig. 4 Differential Impedance sensing variation versus dilution in a buffer of human serum positive to anti-Dengue Virus antibody. On the right, photographs of the active sensors after the final wash for four exemplary cases. Error bars show standard error. All measurements have been performed with 50 mV amplitude signal at 1 MHz

detection mechanism for few exemplary dilutions and the number of beads over the active sensor decreases with sample dilution down to about one hundred. As can be seen in the magnification from Fig. 4 also clusters of beads are formed during washing and/or PBS evaporation contributing to the single concentrations error bar.

As reported in [20], the detection with the DIS platform has achieved results that outperform the one obtained over the same positive and negative serum samples with a standard ELISA test.

3 Multiplex Differential Impedance Sensing

3.1 Electronic Multichannel Reading Board

A compact and portable electronic board has been designed to operate a lock-in processing on up to 7 sensors in parallel in a differential architecture (1 input of the 8 available is used for the reference sensor). As shown in Fig. 5, the analog sections of the board have been developed around an XEM7310 FPGA module by Opal Kelly. The FPGA controls in real-time the generation of the signals to stimulate the sensors, the acquisition of the responses from the sensors and their processing in parallel by implementing dual-phase digital lock-in and transmission to an external PC with a USB port. The *Sensor forcing* section comprises two counter-phase sinusoidal voltage generators (frequency at 1 MHz and tunable amplitude up to 1V), generated with a fast DAC (AD9706, 12-bit, 175 Mbps). They can be operated independently or can be simultaneously applied to the active electrode and to the control electrode of a physical differential sensor pair when local on-chip differential sensing is desired.

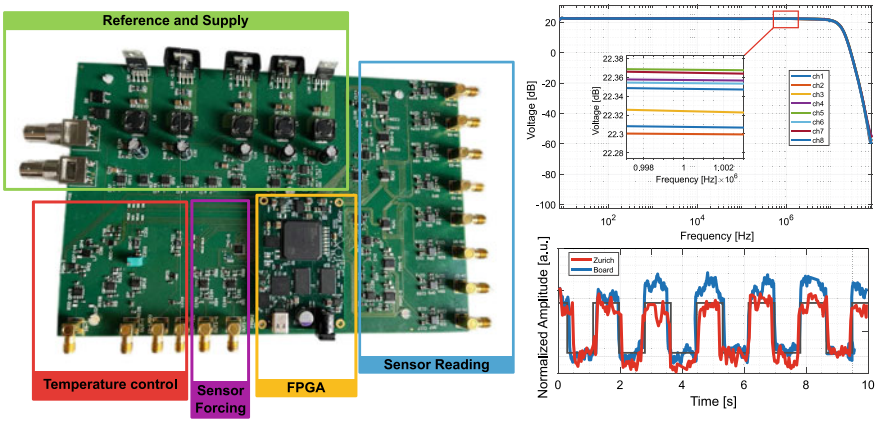


Fig. 5 Photograph of the electronic board for multiplex experiments. All the different sections highlighted are: generation and reading of the sensor, temperature control, power supplies, references and the FPGA module. On the right, a comparison between the commercial system and the electronic reading board in detecting a signal variation of 100 ppm and the transfer function for all the 8-channels for the sensor reading, with a zoom in the region around the working frequency of 1 MHz

The 8 transimpedance amplifiers (TIA) read the currents from 8 sensors. The TIA uses capacitive feedback to combine wide bandwidth and low noise. The output is further processed by a gain amplifier digitally selectable from 0.1 to 1000 and converted in the digital domain by 12-bit ADC (LTC2250) at a sampling rate of 100 Msps. The board also houses an additional lock-in channel for the readout of the on-chip temperature sensor. This *Temperature control* module on the board allows a precise measurement of the local temperature of the biosensor liquid by tracking the resistance (around 1 kΩ) of a serpentine made of the same gold thin-film of the IDEs.

To characterize the control platform realized, the performances in reading the impedance are compared with the commercial lock-in system (HF2LI, Zurich Instruments). An extensive characterization is reported in [20]. Figure 5 reports the results for the 100 ppm variation comparison and the evaluation of the eight transfer functions in order to certify the correct operation and the interchangeability of the eight reading channels.

3.2 Selective DNA Detection

The system with the electronic reading board has been challenged with a multiplexing acquisition of impedance measurement of a chip realized with multiple IDEs. For this experiment, a new chip with a multi-sensor IDE configuration has been design. The layout allows to have 7 sensors to be dedicated between reference, target DNA

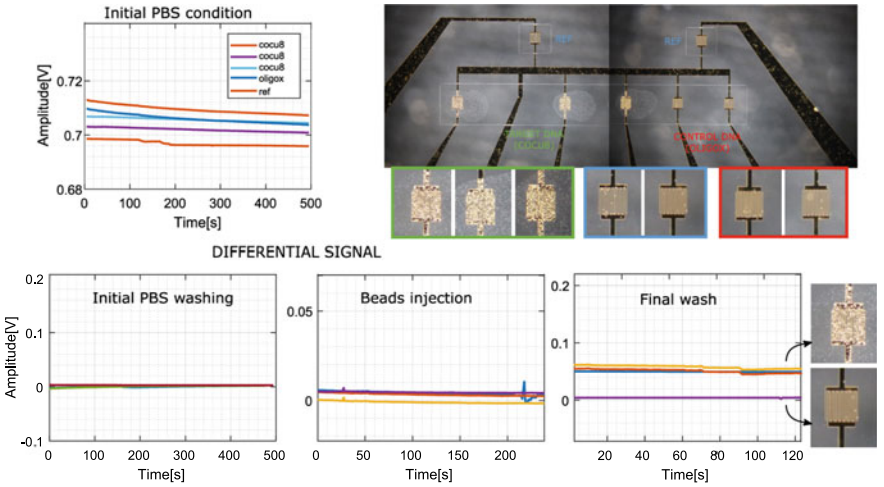


Fig. 6 Signals from the multiplexing experiment. On the top, the value of electrodes in PBS solution (Rsol = 1 kΩ) and on the right the device under test. On the bottom, the signals from the final counting phase. The final washing step reveals the beads bounded only over the target sensors. Picture of the electrodes after the final wash. The beads are specifically bounded only over the active sensor with the probe for the COCU8. Reference and the OLIGOX electrodes are perfectly clean

and negative control on the same area inside the microfluidic chamber. In this case, 2 electrodes will be used as a reference electrode, 3 for target detection (specific probe) and 2 as active control (not specific probe). In the following experiment, the COCU8 is the actual probe for the target DNA COCU10-BIO, while the OLIGOX has the function of negative control and REF as reference sensor for the differential measurement configuration approach. The protocol adopted for the experiment consists as usual in an incubation phase for target hybridization and the steps inside the microfluidic system (Beads incubation and final wash). The results are summarized in Fig. 6.

Figure 6 shows the picture of the electrodes after the final wash. It is possible to notice the optical confirmation of the beads bounded only over the active sensors functionalized with the probe for the target. These results confirm the possibility to operate in a multiplexing acquisition configuration with the electronic board as an acquisition system.

4 Conclusions and Future Perspective

The developed biosensor system is based on the impedance variation between microelectrodes upon the capture of the target analyte, grafted over the biosensor surface through specific probe immobilization. A properly functionalized nanobead has been

used to enhance the electronic signal since the dimensions and the structure of the target moiety would not allow direct label-free detection. The biosensor core, made of a borosilicate chip with microelectrodes, is integrated into a microfluidic path and electronically accessed to perform impedance detection by custom electronic circuits featuring high portability and multichannel operation. In this way, multiple sensing sites in parallel can be addressed, extremely important from a diagnostic point of view since they will allow performing multiplex analysis starting from a single clinical sample. The preparation of the biosensor chip by updating the antibody and the oligonucleotide linker makes this platform concept of Impedance Sensing of durable application and very practical industrial interest, yet reaching clinically breakthrough results. The experiment results show that beads count by a truly differential sensors architecture operated in a lock-in scheme is very effective in monitoring specific IgG antibodies in human serum and buffer down to a few single counts resolution, i.e. a LOD of 88 pg/mL. The sensitivity obtained by the system reaches and possibly outperforms other methods yet operating in a simple and clear protocol as demonstrated in the comparison of the proposed platform response to human serum positive to DENV with a commercial Dengue virus IgG kit. For future work, the ongoing SARS-CoV-2 pandemic represents a starting bench for the development and implementation of such a new biosensor. SARS-CoV-2 will be targeted by direct capture of the entire virus in solution, using DNA-labelled antibodies directed against the SARS-CoV-2 spike protein. This strategy will bring advantages in terms of reduced sample handling and processing (meaning less contamination and no loss of viral components) and no need for harsh chemicals nor for sample purification or amplification, resulting in a reduction of time and cost of the analysis.

References

1. Mavrikou S, Moschopoulou G, Tsekouras V, Kintzios S (2020) Development of a portable, ultra-rapid and ultra-sensitive cell-based biosensor for the direct detection of the SARS-CoV-2 s1 spike protein antigen. Sensors 20(11):3121 May
2. Ferrari D, Clementi N, Mancini N, Locatelli M (2022) SARS-CoV-2 infection despite high levels of vaccine-induced anti-receptor-binding-domain antibodies: a study on 1110 health-care professionals from a northern Italian university hospital. Clin Microbiol Infect 28(2):305–307 February
3. Khan MZH, Hasan MR, Hossain SI, Ahommed MS, Daizy M (2020) Ultrasensitive detection of pathogenic viruses with electrochemical biosensor: state of the art. Biosens Bioelectron 166:112431 October
4. Bhalla N, Jolly P, Formisano N, Estrela P (2016) Introduction to biosensors. Essays Biochem 60(1):1–8 June
5. Yang S, Rothman RE (2004) PCR-based diagnostics for infectious diseases: uses, limitations, and future applications in acute-care settings. Lancet Infect Dis 4(6):337–348
6. Lequin RM (2005) Enzyme immunoassay (EIA)/enzyme-linked immunosorbent assay (ELISA). Clin Chem 51
7. Desmet C, Blum LJ, Marquette CA (2013) Multiplex microarray ELISA versus classical ELISA, a comparison study of pollutant sensing for environmental analysis. Environ Sci: Process Impacts 15(10):1876

8. MacNevin CJ, Watanabe T, Weitzman M, Gulyani A, Fuehrer S, Pinkin NK, Tian X, Liu F, Jin J, Hahn KM (2019) Membrane-permeant, environment-sensitive dyes generate biosensors within living cells. J Am Chem Soc 141(18):7275–7282
9. Guo Y, Lijun X, Hong S, Sun Q, Yao W, Pei R (2016) Label-free DNA-based biosensors using structure-selective light-up dyes. Analyst 141(24):6481–6489 November
10. Zhao Y, Tong R-J, Xia F, Peng Y (2019) Current status of optical fiber biosensor based on surface plasmon resonance. Biosens Bioelectron 142(111505):111505 October
11. Borga P, Milesi F, Peserico N, Groppi C, Damin F, Sola L, Piedimonte P, Fincato A, Sampietro M, Chiari M, Melloni A, Bertacco R (2022) Active opto-magnetic biosensing with silicon microring resonators. Sensors 22(9)
12. Suthanthiraraj PPA, Sen AK (2019) Localized surface plasmon resonance (LSPR) biosensor based on thermally annealed silver nanostructures with on-chip blood-plasma separation for the detection of dengue non-structural protein NS1 antigen. Biosens Bioelectron 132:38–46
13. Nadzirah S, Azizah N, Hashim U, Gopinath SCB, Kashif M (2015) Titanium dioxide nanoparticle-based interdigitated electrodes: a novel current to voltage DNA biosensor recognizes E. coli O157:H7. PLoS One 10(10):e0139766
14. Qureshi A, Niazi JH, Kallempudi S, Gurbuz Y (2010) Label-free capacitive biosensor for sensitive detection of multiple biomarkers using gold interdigitated capacitor arrays. Biosens Bioelectron 25(10):2318–2323
15. Berggren C, Bjarnason B, Johansson G (2001). Capacitive biosensors. Electroanalysis 13
16. Ohno R, Ohnuki H, Wang H, Yokoyama T, Endo H, Tsuya D, Izumi M (2013) Electrochemical impedance spectroscopy biosensor with interdigitated electrode for detection of human immunoglobulin A. Biosens Bioelectron 40
17. Kirchhain A, Bonini A, Vivaldi F, Poma N, Di Francesco F (2020) Latest developments in non-faradic impedimetric biosensors: towards clinical applications. TrAC-Trends Anal Chem 133:116073
18. Sathya S, Muruganand S, Manikandan N, Karuppasamy K (2019) Design of capacitance based on interdigitated electrode for biomems sensor application. Mater Sci Semicond Process 101:206–213
19. Gajdosova VP, Lorencova L, Blsakova A, Kasak P, Bertok T, Tkac J (2021) Challenges for impedimetric affinity sensors targeting protein detection. Curr Opin Electrochem 28(100717):100717
20. Piedimonte P, Sola L, Cretich M, Gori A, Chiari M, Marchisio E, Borga P, Bertacco R, Melloni A, Ferrari G, Sampietro M (2022) Differential impedance sensing platform for high selectivity antibody detection down to few counts: a case study on dengue virus. Biosens Bioelectron 202(113996):113996 April

Development of Crosspoint Memory Arrays for Neuromorphic Computing

Saverio Ricci, Piergiulio Mannocci, Matteo Farronato, Alessandro Milozzi, and Daniele Ielmini

Abstract Memristor-based hardware accelerators play a crucial role in achieving energy-efficient big data processing and artificial intelligence, overcoming the limitations of traditional von Neumann architectures. Resistive-switching memories (RRAMs) combine a simple two-terminal structure with the possibility of tuning the device conductance. This Chapter revolves around the topic of emerging memristor-related technologies, starting from their fabrication, through the characterization of single devices up to the development of proof-of-concept experiments in the field of in-memory computing, hardware accelerators, and brain-inspired architecture. Non-volatile devices are optimized for large-size crossbars where the devices' conductance encodes mathematical coefficients of matrices. By exploiting Kirchhoff's and Ohm's law the matrix–vector-multiplication between the conductance matrix and a voltage vector is computed in one step. Eigenvalues/eigenvectors are experimentally calculated according to the power-iteration algorithm, with a fast convergence within about 10 iterations to the correct solution and Principal Component Analysis of the Wine and Iris datasets, showing up to 98% accuracy comparable to a floating-point implementation. Volatile memories instead present a spontaneous change of device conductance with a unique similarity to biological neuron behavior. This characteristic is exploited to demonstrate a simple fully-memristive architecture of five volatile RRAMs able to learn, store, and distinguish up to 10 different items with a memory capability of a few seconds. The architecture is thus tested in terms of robustness under many experimental conditions and it is compared with the real brain, disclosing interesting mechanisms which resemble the biological brain.

S. Ricci (✉) · P. Mannocci · M. Farronato · A. Milozzi · D. Ielmini
Dipartimento Di Elettronica, Informazione E Bioingegneria (DEIB), Politecnico Di Milano, Piazza L. da Vinci 32, 20133 Milano, Italy
e-mail: saverio.ricci@polimi.it

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_6

1 Introduction

With the advent of the Internet-Of-Things and with the ever-growing number of people gaining the possibility to purchase smartphones and tablets capable to store a large amount of photo, video, music and applications in a single portable device, the global amount of data has increased exponentially, which raises strong requirements in terms of energy efficiency and processing speed for data analysis [1–3]. To satisfy these requirements, the computing performance of modern computers has increased steadily in the past few decades thanks to the scaling down of the transistor dimensions and the consequent higher density of information being stored in the same area, as predicted by Moore's law. The downscaling is now approaching its natural end mainly due to the increasing leakage of the complementary metal–oxide–semiconductor (CMOS) transistors due to their extreme miniaturization [2]. The operating frequency of each transistor has already reached an upper limit set by the maximum acceptable power dissipation, preventing further speed improvement at the device level to avoid an excessive temperature increase of the chip.

If on one side we have reached a limit on data transport speed due to the transistors, on the other side we have to consider that there is an additional limit imposed by the fact that conventional computing systems are based on the von Neumann architecture [4, 5], where memory and processing units are physically separated, which leads to an additional inevitable bottleneck due to the necessary data movement between the two separated units, which causes significant latency and energy consumption. This latency becomes significant when operation must be repeated thousands or millions of times, as it happens to tensor products and matrix multiplications, where the operation between the elements of the matrices cannot be done in parallel but only one operation after the other, finally collecting all the results.

Alternative in-memory computing approaches are becoming increasingly attractive to develop novel logics and neuromorphic computations to overcome Von Neumann bottleneck issues [4–6]. Indeed, typical operations like image learning, pattern recognition and decision exhibit high computational cost for boolean CMOS processors, while, for human brain, they represent elementary processes. In this scenario, the development of new devices designed specifically for neuromorphic computing could enable high density and low power networks to properly operate learning and recognition tasks. Among the various emerging memories, also known as memristors, resistive switching memories appear as one of the most promising technologies for in-memory computing, thanks to the CMOS-compatible fabrication process, the small area and the analog programming.

Differently from conventional memories based on transistors, which are able to store binary values only, specifically "1" (transistor in pass mode) and "0" (transistor switched off), memristors can store information in their electrical properties, like the resistance (or conductance) for example, in an analog way. Moreover, by organizing these memories in a matrix configuration, also known as crosspoint architecture, the matrix–vector multiplication is performed in one step only, carrying out all the single elements multiplications simultaneously exploiting the Kirchhoff's law [5, 6, 8].

Because of the novelty of this technology, problems of reliability and integration with existing technologies affect the emerging memories and further studies are required to overcome the limits by optimizing the materials and their responses and developing architecture designs and algorithms to exploit the innovative features and the strong parallelism of the physical multiplication [4, 5]. In this scenario, this Chapter focuses on the topic of RRAMs for high-density crosspoint arrays, starting from their fabrication, through the characterization of single devices up to the development of proof-of-concept experiments in the field of in-memory computing, hardware accelerators and brain-inspired architecture.

2 Non-volatile RRAMs

A resistive switching memory is a two-terminal device where the conductance can be manipulated by externally applied voltage pulses. The main structure is composed by an oxide layer sandwiched between two metals, in the so-called Metal–Insulator-Metal.

(MIM) structure. The RRAM switching mechanism refers to the possibility of creating and disrupting a conductive path across the oxide, creating a conductive bridge between the metals, by locally changing the oxygen vacancy concentration and for this reason they are also known as RedOx RRAMs (ReRAM). By applying a positive voltage to the top electrode (TE), the oxygen vacancies can migrate and reallocate inside the oxide layer with a consequent change of the electrical properties, where the formed oxygen vacancy-based conductive channel dictates a low resistance state (LRS), as depicted in Fig. 1a. The application of a negative voltage to the TE, instead, induces a vacancy dispersion into the oxide, the conductive path is dissolved and the resistivity rises-up, bringing the device in a high resistive state (HRS).

The typical electrical response of a RRAM is reported in Fig. 1b, where the hysteresis of the I-V curves changes according to the maximum current [1, 3, 6, 7], called compliance current (I_C). The dependence of the conductance as a function of the I_C is clearly visible in Fig. 1c, with a linear dependence linked to the possibility of enlarging the conductive channel diameter by increasing the current [3, 7]. Inversely, with the increase of the reset amplitude the conductive state is brought back to the HRS and the larger the voltage, the less conductive the device is, as seen in Fig. 1c. The exponential behavior is explained as the presence of an activation energy required to move the vacancies and the defects, resulting in an Arrhenius-like process.

The tunability of the conductance is the key point of the RRAM technology and the advantage is clear when the devices are organized in a matrix configuration, with the TEs and the BEs placed orthogonally. By exploiting Kirchhoff's and Ohm's law the matrix–vector-multiplication between the conductance matrix and a voltage vector is computed in one step only [7–9]. Each element in the matrix must be programmed properly to a desired value, by using multiple set and reset operations, as seen in Fig. 2a, where a device is programmed passing from 0 μS to 82 μS using set operation and then reset till the target of 73 μS. Figure 2b and c report the before

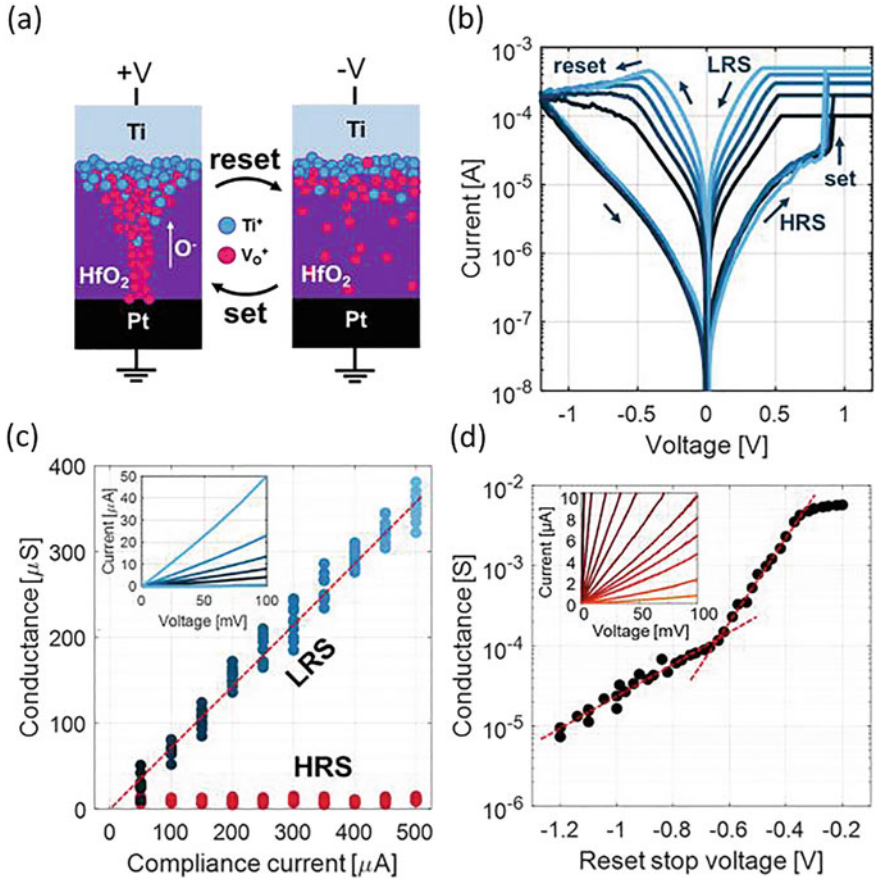


Fig. 1 Physical mechanism and quasi static characterization of Pt/HfO2/Ti non-volatile RRAMs. **a** Sketch of the switching mechanism with the formation and dissolution of the conductive channel. **b** I-V curves at different compliance currents in logarithmic scale. The device passes from the HRS to different LRS states through set transitions and then reset [7]. **c** Conductance levels as a function of I_C. **d** Conductance levels associated with the reset amplitudes. The values spread in a range between 5 mS and 10 μS

and after programming of an 8 × 8 crossbar (visible in Fig. 2d). The final matrix encodes the coefficients of the covariance matrix of the Wine dataset [9], with an acceptable maximum error of ±3 μS.

The power iteration is an algorithm able to extract the eigenvector components of a matrix by computing vector matrix multiplication between the matrix and the vector obtained in the previous step [8, 10]. After some iterations the values converge to asymptotic values, which are proportional to the mathematical eigenvector (the factor is linked to the one to convert the matrix to a conductance matrix) [10]. Figure 2e sketches the equivalent circuits which implements the power iteration algorithm: the current coming from a first MVM product is converted in voltage, which feed again

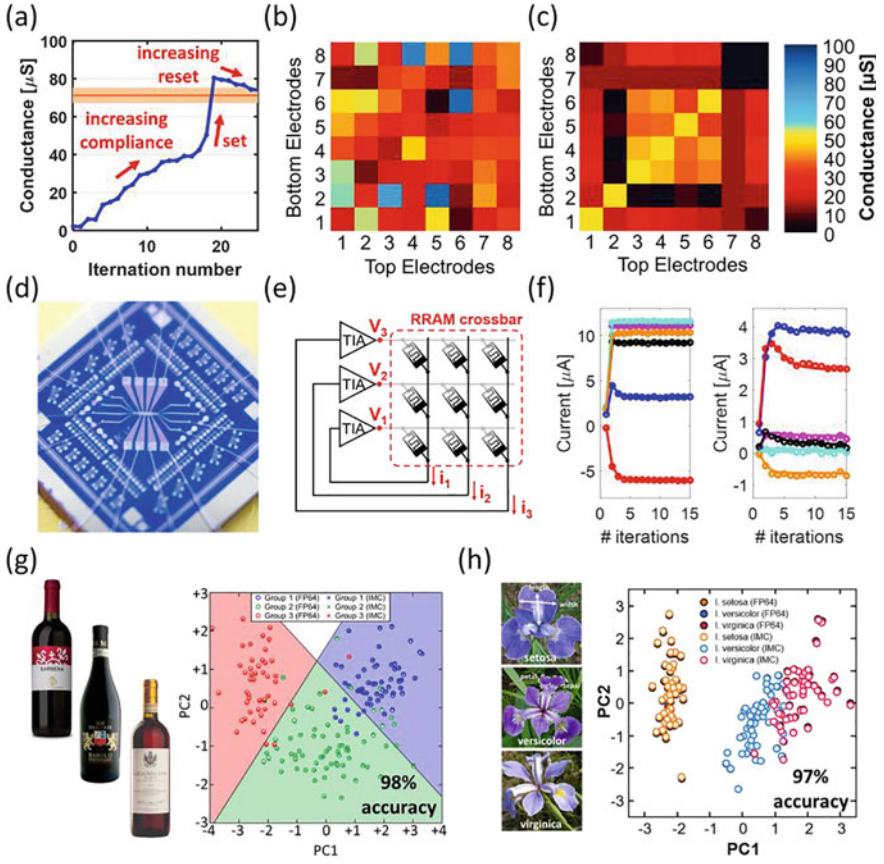


Fig. 2 Program and verify operation for In-Memory Computing and PCA. **a** Tuning of the conductive state using set and reset operations. **b** Initial state of an 8 × 8 crossbar after fabrication. **c** Conductance matrix programmed through the program and verify algorithm. The matrix encodes the Wine dataset covariance matrix [9]. **d** Optical image of an 8 × 8 crossbar bonded. **e** Conceptual circuit to implement the power iteration algorithm. **f** Eigenvector computation for PC1 (on the left) and PC2 (on the right). The curves stand for the evolution of the current values. **g** Wine dataset projection along the first two PCs [9]. **h** Iris dataset projection [7]

the MVM. The extraction of the first and second greater eigenvectors, also called principal components (PC), can be followed in Fig. 2f. Within 10 iterations, the currents converge to the asymptotic values [9]. Finally, the extracted PCs are used to project the dataset along the components and to group the different wines, as seen in Fig. 2g, with an accuracy of 98%, comparable with the floating point 64 software-based computation [9]. The Wine dataset contains 3 different wines classified with 6 properties, like chemical values, color and sugar content. The eigenvectors are proportional to these values and all the wines can be written as a combination of such components. To validate the approach, the same experiment is repeated in Fig. 2h

by looking at the Iris dataset [7], containing three different Iris flowers labelled with petal and sepal length and width. These results support RRAM crosspoint arrays for accelerating advanced machine learning with IMC.

3 Volatile RRAMs

Filamentary switching memories are a different class of RRAMs which rely on a metallic filament to change the electrical properties, where high mobility metal ions migrate from one electrode to the other creating a conductive bridge [11, 12]. Silver-based RRAMs exhibit spontaneous disruption of the metallic conductive filament with a lifetime ranging from few microseconds to several seconds, thus by controlling and predicting the filament lifetime, devices can be engineered for a wide range of applications. When a positive bias is applied to the Ag electrode, the electric field leads the Ag ions to migrate across the oxide and the resistivity drops down, creating a conductive path made of nanoclusters [11]. Reducing the voltage, the filament spontaneously disrupts, the resistivity rises-up and a gap occurs, which is responsible for the absence of conductance. Figure 3a reports the electrical response associated to the mechanisms described.

Because of the spontaneous disruption of the filament [11–13], it is important to study the temporal evolution of the devices, by switching on the memory and then monitoring the state until it switches off. The time window in which the filament remains stable is called retention time. Figure 3b collects the cumulative distribution curves of the retention time as a function of the maximum current reached during the switching [12], current which is limited by exploiting the saturation region of transistors. The larger is the current and the longer is the average retention time,

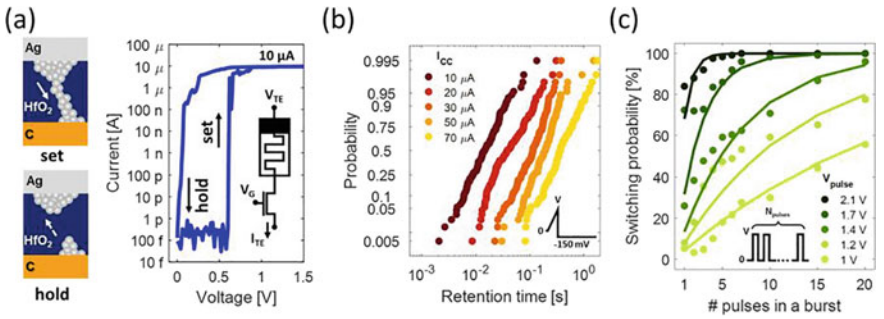


Fig. 3 Ag-based volatile 1T1R RRAM electrical characterization. **a** Quasi-static I–V sweep. **b** Retention time distributions for different maximum current. 3 V and 1 ms triangular pulse are applied to set the device, and a constant −150 mV bias voltage is applied to monitor the status of the device. **c** Impact of the number of pulses applied. Considering a group of pulses, the probability of finding the device in the ON state increases with the number of applied pulses

according to the fact that the filament has a greater diameter and thus it is naturally more robust.

Moreover, the devices result to be sensitive also to the pulse amplitude [12, 13], meaning that for small amplitudes the devices do not switch on while for large values (>3 V for example) the devices always switch on. The trends of the probability that the device switches on after a pulse as a function of the pulse amplitude and the number of pulses [13] are shown in Fig. 3c. By increasing the pulse amplitude (V_{pulse}), the probability increases and it starts saturating at 100% after few pulses (like in the case of 2.1 V, the darker curve). For low voltages (1 V pulses, the lighter green curve) the probability weakly increases with the number of pulses.

The fact that stochastic properties, retention time and switching probability, are tunable with the voltage and the temporal dynamic is adaptable according to the compliance current are explored in a simple neuromorphic circuit. Short-term memory is a primary concept in human life, since it is responsible of the storing of acquired information in the meantime that it is processed and evaluated. The proposed system has two main features: storing the information in the memory and later recognizing it, as depicted in.

Figure 4a: the memory has an item stored inside, for example an advertisement spot (marked with a specific color), which is linked to specific areas that are activated. When the true item arrives (the orange in the example), the system recognizes it, and a trigger signal is generated. When other items arrive, the system does not recognize them and thus it is not triggered. Being a short-term memory, if the system is not refreshed somehow, providing for example the true item, it forgets the stored information.

The circuit is implemented using 5 different devices (in Fig. 4b) where the total current is summed together, and the transistor share the same gate to have similar time responses. At each device is sent a signal which is calibrated to have a specific switching probability (P_{ON}), considering the device-to-device variability. This switching probability can be seen as the volume, thus the higher the volume the higher the relevance we give to the spot. Figure 4c shows the evolution of the system when a pattern is applied multiple times as soon it is impressed in the system (3 devices are switched on) and then random patterns are applied. When the right pattern arrives (marked with a dot) the system is triggered and recognizes it, otherwise no. Different experimental parameters, in terms of pattern rate, delay and amplitude, are tested, finding the best condition when the switching probability is low while the refresh is high (in Fig. 4d). This condition is in good agreement with what happens to the human brain during the advertisement: all the spots have a small relevance, but when the right one is on the tv the attention is high because the spot is recognized, thus we can distinguish what we like from the other spots. Differently, when the spot is less broadcasted (small spike rate, in Fig. 4e) the information is lost, and it is more difficult to recognize it. On the other hand, the volume plays a crucial role, because our attention changes drastically. For great P_{ON} (so large volume) the system easily changes the information stored and thus is not able anymore to recognize the first one.

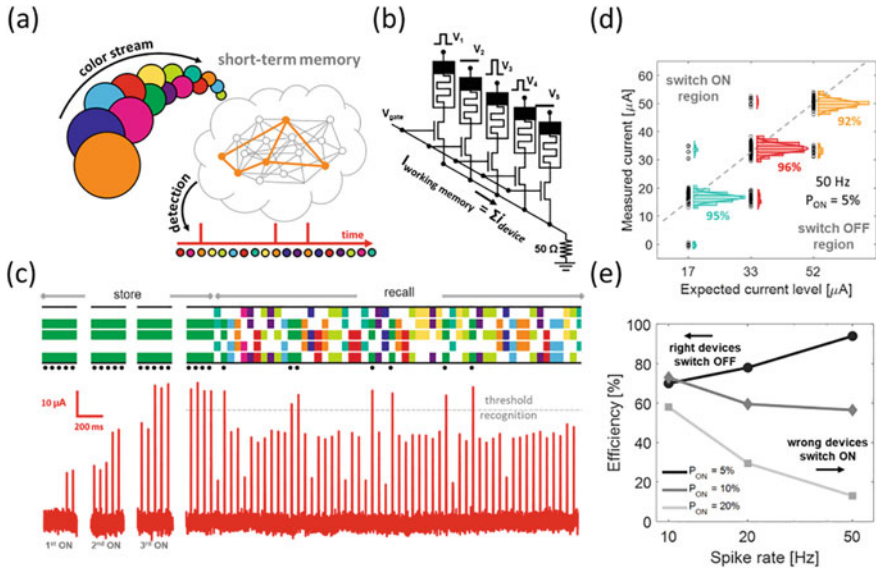


Fig. 4 Sketch of the memristive implementation of a working memory circuit emulator. **a** After an object (a color for example orange) is stored in the mnemonic architecture, a stream of objects is sent to the system. When the true object arrives, the system is refreshed and triggered. **b** Real implementation using 5 Ag-based volatile 1T1R RRAMs. The devices are connected in parallel to sum the currents and share the same gate voltage to have the similar electrical responses. **c** Example of an experimental trace. In the store phase the same pattern is sent multiple times to switch ON the right devices, until all the 3 RRAMs are in the ON state. In the recall phase random pattern are sent. The current is discretized in four levels, according to the number of ON devices. A suitable current threshold is used to discriminate when the true pattern arrives. **d** Correlation plot of the best experimental parameters to check the accuracy of the system. **e** Behavior of the memristive architecture by changing experimental parameters. The best results are achieved when the system is frequently refreshed

4 Conclusions

This Chapter aimed to give an overview on emerging memories and on the potentialities of resistive switching devices in the field of in-memory computing, hardware accelerators and brain-inspired architecture. In RRAMs with Pt/HfO_2/Ti stack the conductance value of the memory can be tuned in an analog way according to external parameters, such as the current flowing through the device. This gives the possibility to directly map mathematical weights into conductance values and, in a suitable crossbar configuration, to operate the MVM operation in one step. The eigenvalue/eigenvector calculation is experimentally demonstrated, and the extracted component is used in the PCA of a large dataset, showing not only a fast convergence but also an accuracy comparable to the FP64 software-based solution, with a value up to 98%. On the other side, by changing one of the electrodes, silver-based devices feature a spontaneous change of device conductance with a retention ranging from 1 ms to

several seconds, as it happens in the biological systems. This similarity is crucial to implement biological functions and tasks, as the short-term-memory typical of living animals. The simple structure combined with a wide flexibility in terms of electrical responses and properties, supports the RRAM technology as interesting candidate for accurate acceleration of machine learning, in-memory computing, and neuromorphic systems.

Acknowledgements The authors would like to thank M. Asa, A. Scaccabarozzi, C. Somaschini, C. Nava, S. Fasoli, S. Bigoni, E. Sogne and G. Cannetti for help in the fabrication process. This work was partially performed in Polifab, the micro and nanofabrication facility of Politecnico di Milano. This article received funding from the European Union's Horizon 2020 research and innovation program (grant agreement no. 824164).

References

1. Xia Q, Yang JJ (2019) Memristive crossbar arrays for brain-inspired computing. Nat Mater 18:309–323
2. Yang JJ, Strukov DB, Stewart DR (2013) Memristive devices for computing. Nat Nanotechnol 8(1):13–24
3. Kim H, Mahmoodi MR, Nili H, Strukov DB (2021) 4K-memristor analog-grade passive crossbar circuit. Nat Commun 12(1):5198
4. Milano G, Pedretti G, Montano K, Ricci S, Hashemkhani S, Boarino L, Ielmini D, Ricciardi C (2022) In materia reservoir computing with a fully memristive architecture based on self-organizing nanowire networks. Nat Mater 21(2):195–202
5. Pedretti G, Mannocci P, Li C, Sun Z, Strachan JP, Ielmini D (2021) Redundancy and analog slicing for precise in-memory machine learning—part II: applications and benchmark. IEEE Trans Electron Devices 68(9):4379–4383
6. Wang R, Shi T, Zhang X, Wei J, Lu J, Zhu J, Wu W, Liu Q, Liu M (2022) Implementing in-situ self-organizing maps with memristor crossbar arrays for data mining and optimization. Nat Commun 13:2289
7. Ricci S, Mannocci P, Farronato M, Hashemkhani S, Ielmini D (2022) Forming-free resistive switching memory crosspoint arrays for in-memory machine learning. Adv Intell Syst 4:2200053
8. Mannocci P, Baroni A, Melacarne E, Zambelli C, Olivo P, Pérez E, Wenger C, Ielmini D (2022) In-memory principal component analysis by crosspoint array of resistive switching memory: a new hardware approach for energy-efficient data analysis in edge computing. IEEE Nanatechnol Mag 16(2):4–13
9. Ricci S, Mannocci P, Farronato M, Ielmini D (2023) In-memory computing with crosspoint resistive memory arrays for machine learning. In: Proceedings of SIE 2022. SIE 2022. Lecture notes in electrical engineering, vol 1005. Springer
10. Jolliffe I (2005) Principal component analysis. In: Everitt BS, Howell DC (eds) Encyclopedia of statistics in behavioral science. Wiley, Chichester, UK, p bsa501
11. Covi E, Wang W, Lin Y, Farronato M, Ambrosi E, Ielmini D (2021) Switching dynamics of Ag-based filamentary volatile resistive switching devices—part I: experimental characterization. IEEE TED 2021, vol 68, No. 8
12. Wang W, Covi E, Milozzi A, Farronato M, Ricci S, Sbandati C, Pedretti G, Ielmini D (2021) Neuromorphic motion detection and orientation selectivity by volatile resistive switching memories. Adv Intell Syst 3:2000224

13. Ricci S, Kappel D, Tetzlaff C, Ielmini D, Covi E (2022) Decision making by a neuromorphic network of volatile resistive switching memories. In: 2022 29th IEEE international conference on electronics, circuits and systems (ICECS), Glasgow, United Kingdom, 2022, pp 1–4

Systems and Control

Reconciling Deep Learning and Control Theory: Recurrent Neural Networks for Indirect Data-Driven Control

Fabio Bonassi

Abstract This Brief aims to discuss the potential of Recurrent Neural Networks (RNNs) for indirect data-driven control. Indeed, while RNNs have long been known to be universal approximators of dynamical systems, their adoption for system identification and control has been limited by the lack of solid theoretical foundations. We here intend to summarize a novel approach to address this gap, which is structured in two contributions. First, a framework for learning safe and robust RNN models is devised, relying on the Incremental Input-to-State Stability (δISS) notion. Then, after a δISS black-box model of the plant is identified, its use for the design of model-based control laws (such as Nonlinear MPC) with closed-loop performance guarantees is illustrated. Finally, the main open problems and future research directions are outlined.

1 Introduction

In recent decades, the control systems community has devoted an increasing research interest to data-driven control. This term relates to the approaches in which the control system is directly synthesized based on the data collected from the physical plant to be controlled (direct approaches), or designed relying on a dynamical model identified from such data (indirect approaches).

The common rationale behind these methods is that retrieving first-principle models of physical systems is generally a time-consuming task, and these models are often valid only in a neighborhood of the nominal operating conditions. Such limitations

This work has been partially supported by the European Union's Horizon 2020 programme under the Marie Skłodowska-Curie grant No. 953348.

F. Bonassi (✉)
Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Via Ponzio 34/5, 20133 Milano, Italy
e-mail: fabio.bonassi@polimi.it

Department of Information Technology, Uppsala University, Lägerhyddsvägen 1, 75237 Uppsala, Sweden

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_7

arise, e.g., from assumptions needed to obtain reasonable analytical models, or from the estimation of their unknown parameters, which is typically carried out by locally perturbing the operating conditions. Data-driven control aims to address these limitations, exploiting the information embedded in the measured data to synthesize control systems that are as accurate and global as possible, while minimizing the need for human intervention in the design phase.

In this context, researchers and engineers soon realized that many of the tools and methodologies developed within the deep learning community could find relevant applications for data-driven control. Neural Networks (NNs), and in particular Recurrent Neural Networks (RNNs), have been the object of many research efforts and engineering applications, see [21], owing to their advanced capabilities of modeling dynamical systems, for which they are known universal approximators. Although these modeling capabilities have been known for decades, the use of RNNs for learning dynamic systems has only recently been made effective by the increased availability of data, the development of RNN architectures less prone to vanishing and exploding gradient problems [20], and the development of open source software platforms for training them.

In light of their flexibility, there exists a multitude of different data-driven control strategies making use of NNs and RNNs [9]. In this work, we focus on (*i*) the use of RNNs for black-box nonlinear system identification and (*ii*) the design of theoretically-sound control laws based on the learned RNN models. By resorting to this indirect data-driven control paradigm, one can exploit RNNs' modeling capabilities while relying on the vast literature of nonlinear model-based control strategies, such as Nonlinear Model Predictive Control (NMPC).

Although RNN-based NMPC has been successful in numerous applications, such a strategy has often garnered criticism by the control systems community, due to the lack of solid theoretical foundations guaranteeing the accuracy of the learned models, let alone the closed-loop stability and performances. Despite these clear goals, only limited theoretical results, mainly related to the simplest RNN architecture (i.e. "vanilla" RNNs), had been obtained [25]. Researchers have indeed struggled to build a theoretical framework for the adoption of more advanced RNN architectures, such as Gated Recurrent Units (GRU) and Long Short Term-Memory (LSTM) networks, due to their structural complexity.

Contributions

This work is intended to outline some contributions given in [6] that aim to fill the above-mentioned methodological and theoretical gaps, establishing a novel and theoretically-sound framework for RNN-based indirect data-driven control. The approach is structured in two contributions.

Learning stable RNN models—A methodology for learning "safe" and "robust" RNN models is devised, by resorting to classical nonlinear stability notions such as the Incremental Input-to-State Stability (δISS, [1]). To this end, novel sufficient conditions on the weights of several RNN architectures, such as Neural NARXs (NNARXs), as well as more advanced RNNs like GRUs and LSTMs, have been pro-

posed in [7, 8, 24], respectively. These conditions are leveraged to devise a training procedure for learning RNNs with δISS certification [9].

Control design—Based on the identified δISS RNN models, several control architectures with closed-loop guarantees are devised. One approach relies on the design of a state observer with exponential convergence guarantees to synthesize an NMPC law [6, 10]. Relying on the model's δISS, this control strategy can guarantee nominal closed-loop stability and recursively feasibility provided that the cost function is designed according to the proposed criterion, which makes the design procedure fairly easy. For the sake of compactness, only this control architecture is reported here, see Sect. 3.2. More involved NMPC architectures, able to attain asymptotic offset-free tracking of piecewise-constant reference signals, can also be synthesized, as shown in [12, 14]. These architectures rely on the enlargement of the RNN model with integral action and on the design of a state observer for the resulting enlarged system, which is provenly enabled by the δISS of the RNN model itself.

Remarkably, the proposed framework for learning δISS RNN models has also been shown to enable the design of a variety of other control architectures with closed-loop guarantees. To mention a few, in [22, 23] a disturbance estimation-based NMPC has been devised for LSTM models, whereas in [11] an Internal Model Control (IMC) architecture with local stability guarantees has been proposed. This latter control strategy has been shown to attain closed-loop performances close to those of NMPC laws at a fraction of their online computational burden, making it suitable for implementation on embedded control boards with limited computational resources.

The following notation is adopted here. Given a vector $v \in \mathbb{R}^n$, we denote by v' its transpose and by $\|v\|_p$ its p-norm. The Hadamard (element-wise) product between two vectors v and w of the same dimensions is denoted as $v \circ w$. Given a matrix A, $\|A\|_p$ is used to indicate its induced p-norm. Time-varying vectors are denoted by the time index k as a subscript. Sequences of vectors spanning from time k_1 to $k_2 \geq k_1$ are denoted as $v_{k_1:k_2} = \{v_{k_1}, v_{k_1+1}, ..., v_{k_2}\}$.

2 Learning Stable RNN Models

Let us consider an RNN in the state-space form

$$\Sigma(\Phi) : \begin{cases} x_{k+1} = f(x_k, u_k; \Phi) \\ y_k = g(x_k; \Phi) \end{cases}, \tag{1}$$

where $x_k \in \mathbb{R}^{n_x}$, $u_k \in \mathcal{U} \subseteq \mathbb{R}^{n_u}$, and $y_k \in \mathbb{R}^{n_y}$ denote the state, input, and output vectors, respectively, and $n_u = n_y$ for simplicity. The exact expression of the state transition function $f(\cdot)$ and of the output function $g(\cdot)$ depends on the specific RNN under consideration; see [6]. These functions are parametrized by the weights Φ,

which are learned during the training procedure. In the following, let us compactly denote the state evolution of the system by $x_k(x_0, u_{0:k-1}; \Phi)$, obtained initializing (1) in x_0 and feeding it with the input sequence $u_{0:k-1}$.

Recalling the definitions of $\mathcal{K}_\infty$ and $\mathcal{KL}$ functions from [6, 9], we can now formulate the considered stability notion. Note that we here restrict the analysis to the δISS property, as it is the strongest and most useful property for the control synthesis. Other less strict stability notions are available in [6].

Definition 1 (δISS [1]) System (1) is regionally δISS in its invariant set $\mathcal{X}$ if there exist a $\mathcal{KL}$ function β and a $\mathcal{K}_\infty$ function γ such that, for any pair of initial states $x_{a,0} \in \mathcal{X}$ and $x_{b,0} \in \mathcal{X}$, and any pair of input sequences $u_{a,0:k} \in \mathcal{U}$ and $u_{b,0:k} \in \mathcal{U}$, at any time step $k \in \mathbb{Z}_{\geq 0}$ it holds that

$$\|x_{a,k} - x_{b,k}\|_p \leq \beta(\|x_{a,0} - x_{b,0}\|_p, k) + \gamma\Big(\max_{\tau \in \{0,\dots,k-1\}} \|u_{a,\tau} - u_{b,\tau}\|_p\Big), \tag{2}$$

where $x_{*,k}$ is short for $x_k(x_{*,0}, u_{*,0:k-1}; \Phi)$.

Note that the δISS implies, among other desirable properties, that (*i*) the effect of the initial conditions asymptotically vanishes, meaning that the modeling performances of the RNN are asymptotically independent of the random initial conditions; (*ii*) the RNN is robust against input perturbations, since closer input sequences imply a tighter asymptotic bound on the distance between the resulting state trajectories; (*iii*) the RNN is bounded-input bounded-state stable; (*iv*) the RNN admits exactly one equilibrium for any constant input $\bar{u} \in \mathcal{U}$ [22].

Theorem 1 (δISS sufficient conditions [6, 9]) *For each of the considered RNN architectures there exist sufficient conditions, in the form of nonlinear non-convex inequalities on the network's weights, compactly denoted as*

$$\nu(\Phi) < 0, \tag{3}$$

that guarantee the δISS in the sense specified by Definition 1.

Of course, each architecture has different expressions for condition (3). The interested reader is addressed to [7, 8, 13] for the exact expression in the case of NNARXs, LSTMs, and GRUs, respectively. Moreover, let us notice that for these architecture the δISS property is *exponential*, i.e., there exist $\mu > 0$ and $\lambda \in (0, 1)$ such that β can be expressed as $\beta(s, k) = \mu\lambda^k s$.

2.1 Training Procedure

Being a known function of the weights, the δISS condition (3) can be used not only to assess a-posteriori the stability of a trained model but can also be enforced during the

training procedure, allowing one to learn RNN models with δISS certification. In the following, a training algorithm based on the Truncated Back-Propagation Through Time (TBPTT, [3]) is therefore outlined. A more detailed version of the algorithm can be found in [6].

Assume that N_{tr} pairs of input-output training subsequences of length T_s are available, and let them be denoted by $(u^{\{i\}}_{0:T_s}, y^{\{i\}}_{0:T_s})$, with $i \in \mathcal{I} = \{0, ..., N_{tr}\}$. Such subsequences are randomly extracted from the normalized[1] input-output sequences recorded from the plant during the experiment campaign. Note that N_{tr} and T_s are designed so that the subsequences are partially overlapping, which allows to mitigate the vanishing gradient phenomenon [20].

The training procedure is iterative, where at each iteration (known as epoch) the set $\mathcal{I}$ is randomly partitioned in B batches, denoted by $\mathcal{I}^{\{b\}}$. For each batch $b \in \{1, ..., B\}$, the training loss function is defined as

$$\mathcal{L}(\mathcal{I}^{\{b\}}; \Phi) = \sum_{i \in \mathcal{I}^{\{b\}}} \mathrm{MSE}\Big(y_{\tau_w:k}\big(x_0, u^{\{i\}}_{0:k}; \Phi\big), y^{\{i\}}_{\tau_w:k}\Big) + \rho\big(\nu(\Phi)\big), \tag{4}$$

where the first term penalizes the Mean Square Error (MSE) between the measured output sequence $y^{\{i\}}_{\tau_w:k}$ and the free-run simulation of the RNN (1) (starting from random initial conditions and fed by the input sequence $u^{\{i\}}_{0:k}$) after a *washout* period $\tau_w > 0$, which accommodates the initial transient. The second term is a regularizer that penalizes the violation of the δISS condition. The loss function gradient $\nabla_\Phi \mathcal{L}(\mathcal{I}^{\{b\}}; \Phi)$ is then backpropagated via gradient descent, or by accelerated gradient descent methods like ADAM and RMSProp [2].

At the end of each epoch, the performance metrics of the RNN on a validation dataset are computed. The training procedure is halted when the stability condition (3) is satisfied and the validation performance metrics stop improving, yielding the trained weights $\Phi^\star$. Finally, the modeling performances of the trained network are assessed on an independent test dataset.

3 Control Design

3.1 Definition of the Control Problem

At this stage, let us assume that an RNN model of the system, $\Sigma(\Phi^\star)$, has been trained and that it satisfies the δISS conditions described in Theorem 1. It is reminded that $\mathcal{X}$ denotes the invariant set with respect to the input set $\mathcal{U}$.

Under the Certainty Equivalence Principle (CEP), the control problem consists in synthesizing a control law that steers the model's output y_k to a piecewise-constant setpoint $\bar{y}$, while fulfilling the input constraint $u_k \in \mathcal{U}$. Letting $\mathrm{Int}(\mathcal{S})$ be the interior

[1] Input and output vectors are henceforth assumed to have zero mean and unity scale.

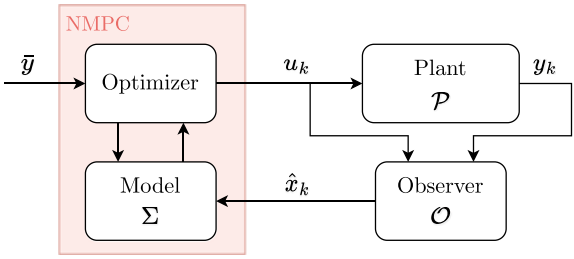


Fig. 1 Schematic of an RNN-based NMPC

part of set $\mathcal{S}$, the following assumption can be introduced to state the control problem more formally.

Assumption 1 Given the output setpoint $\bar{y}$, there exist $\bar{x} \in \text{Int}(\mathcal{X})$ and $\bar{u} \in \text{Int}(\mathcal{U})$ such that the triplet $\bar{z} = \bar{z}(\bar{y}) = (\bar{x}, \bar{u}, \bar{y})$ constitutes a feasible equilibrium of the RNN model (1), that is, $\bar{x} = f(\bar{x}, \bar{u}; \Phi^\star)$ and $\bar{y} = g(\bar{x}; \Phi^\star)$.

The control problem can now be formally stated.

Problem 1 Given the δISS RNN model $\Sigma(\Phi^\star)$ and the output setpoint $\bar{y}$, steer the system to the feasible equilibrium $\bar{z}(\bar{y})$ by means of a control action that satisfies the input constraint $\mathcal{U}$.

Leveraging the model's δISS, Problem 1 has been addressed with a variety of approaches [6], such as internal model control [11] and nonlinear model predictive control, see e.g. [23, 24], and [14]. In the following, one of the possible NMPC approaches is outlined for illustrative reasons.

3.2 NMPC Design

In this section, the synthesis procedure of the scheme depicted in Fig. 1 is summarized. Note that, since RNNs are generally black-box models and NMPC is a state-feedback control law, the model states need to be estimated by a suitably-designed state observer. The synthesis of the proposed control architecture is therefore structured in two steps, i.e., (*i*) the design of a state observer for the RNN model and (*ii*) the formulation of NMPC's underlying Finite Horizon Optimal Control Problem (FHOCP).

Weak Detector Design—In order to estimate the states of the black-box models from the plant's input and output data, a state observer with convergence guarantees should be designed. While nonlinear state observers can be designed with several different approaches, such as moving horizon estimators, we here consider Luenberger-like observers. Such observers are generally synthesized by including in the model dynamics a suitably designed innovation term.[2] In the following, we

[2] See [23, 24] for the design of observers for LSTMs, and [6] for GRUs.

denote such observer by

$$\mathcal{O}(\Phi_o): \quad \hat{x}_{k+1} = f_o(\hat{x}_k, u_k, y_k; \Phi_o), \tag{5}$$

parametrized by $\Phi_o = \Phi^\star \cup \Phi_L$, where Φ_L collects the observer's innovation gains.

Definition 2 (*Weak detector*) System (5) is said to be a weak detector of model (1) if there exist $\mu_o > 0$ and $\lambda_o \in (0, 1)$ such that, for any initial condition of the model $x_0 \in \mathcal{X}$, any initial guess $\hat{x}_0 \in \mathcal{X}$, and any input sequence $u_{0:k}$, it holds that $\|\hat{x}_k - x_k\|_2 \leq \mu_o \lambda_o^k \|\hat{x}_0 - x_0\|_2$.

Relying on the δISS property of the trained RNN model, in [6, 23, 24] sufficient conditions on the innovation gains Φ_L which guarantee the state observer to be a weak detector have been devised. A notable case is that of GRU models, where the devised conditions can be leveraged to formulate the observer design problem as a convex optimization program [6, Proposition 6.1].

Formulation of the FHOCP—According to the MPC paradigm, the control law is retrieved by solving, at every time-step k, the underlying FHOCP. Such an optimization problem relies on the RNN predictive model of the system, i.e. (1), to predict the future state trajectories throughout the prediction horizon N, given the current state estimate $\hat{x}_k$ yielded by the observer (5) and the applied control sequence. Let therefore $u_{k:k+N-1|k}$ be the control sequence applied throughout the prediction horizon, and let $x_{k:k+N|k}$ indicate the resulting state trajectories, where, of course, $x_{k|k} = \hat{x}_k$. Under this notation, letting $\mathcal{N} = \{0, \ldots, N-1\}$, the considered FHOCP can be stated as follows.

$$\min_{u_{k:k+N-1|k}} \quad \sum_{\tau=0}^{N-1} \left(\|x_{k+\tau|k} - \bar{x}\|_Q^2 + \|u_{k+\tau|k} - \bar{u}\|_R^2 \right) + V_{\bar{z}}(x_{k+N|k}) \tag{6a}$$

$$s.t. \quad x_{k|k} = \hat{x}_k \tag{6b}$$

$$x_{k+\tau+1|k} = f(x_{k+\tau|k}, u_{k+\tau|k}; \Phi^\star) \qquad \forall \tau \in \mathcal{N} \tag{6c}$$

$$u_{k+\tau|k} \in \mathcal{U} \qquad \forall \tau \in \mathcal{N} \tag{6d}$$

Note that the predictive model is initialized at the observer state estimate in (6b), whereas its dynamics are embedded by means of constraint (6c). Input constraint satisfaction is ensured by (6d), while $x_{k+\tau|k} \in \mathcal{X}$ is guaranteed by the invariance of $\mathcal{X}$. The cost function (6a) is composed by two terms. The first term penalizes states' and inputs' deviations from their equilibrium values $\bar{x}$ and $\bar{u}$, respectively, by the weights $Q \succ 0$ and $R \succ 0$. The term $V_{\bar{z}}(x_{k+N|k})$ represents a terminal cost approximating the cost-to-go from the terminal state $x_{k+N|k}$ to the equilibrium $\bar{x}$ under the constant input $\bar{u}$. That is,

$$V_{\bar{z}}(x_{k+N|k}) = \sum_{h=0}^{M} \|x_{k+N+h|k} - \bar{x}\|_S^2, \tag{6e}$$

where $x_{k+N+h+1|k} = f(x_{k+N+h|k}, \bar{u}; \Phi^\star)$ for any $h \in \{0, ..., M-1\}$, $S \succ 0$ is the weight, and $M > 0$ the simulation horizon. According to the receding horizon principle, at every step k the FHOCP (6) is solved, and the first optimal control action is applied, i.e., $u_k = u^\star_{k|k}$. Then, at the next time step, the entire procedure is repeated, yielding an implicit state-feedback control law $u_k = \kappa_{\text{MPC}}(\hat{x}_k)$.

In this framework, the model's δISS property and the state observer's exponential convergence have been leveraged to propose conditions on the NMPC design parameters (Q, R, S, N, and M) that allow attaining nominal closed-loop stability and recursive feasibility [6, Theorem 6.2]. Such conditions boil down to inequalities on the singular values of the weight matrices Q and R, and to an explicit minimum value for the simulation horizon M [10]. Seen through these lenses, the devised NMPC scheme can be regarded as a constrained quasi-infinite horizon NMPC [5], where however a minimum prediction horizon is known explicitly.

3.3 Offset-Free NMPC

Albeit attaining desirable nominal closed-loop guarantees, applying the control scheme proposed in Sect. 3.2 to the plant may result in non-ideal tracking performances. The model may indeed be affected by plant-model mismatch, in which case zero-error output regulation might not be achieved. For the main RNN architectures, this problem has been addressed by resorting to the two traditional approaches for offset-free NMPC, namely

- *Integral action-based approaches*—In the spirit of [17], integrators can be placed on the output tracking error so that, as long as the closed-loop stability is preserved, robust asymptotic zero-error output regulation is achieved in virtue of the Internal Model Principle [16]. Applications of such strategy to RNN architectures are available in [12, 14, 19].
- *Disturbance estimation-based approaches*—Along the lines of [18], an approach guaranteeing offset-free static performances relies on the enlargement of the system model with the disturbance dynamics. This allows the disturbance to be estimated by means of a state observer and to be accounted for and compensated for in the NMPC formulation. Applications of this strategy to LSTM models are available in [22, 23].

4 Open Problems and Future Research Directions

Despite the great potential that RNNs have shown in the context of data-driven control, there are several open issues that have been only partially addressed, and whose resolution would lead to improved applicability of these strategies, even in

safety-critical contexts. Below, these issues are briefly outlined, while for more details the interested reader is addressed to [6].

i. *Safety verification*—The problem of safety verification consists in assessing that the RNN model's output reachable set lies within a "safe" set, e.g., the set of physically-meaningful outputs. Safety verification hence allows certifying that the model does not generate unsafe or unexpected outputs. While this procedure is notoriously involved for nonlinear systems, especially for RNNs, their δISS certification allows for retrieval of an analytical expression of the output reachable set. Such an expression is however conservative in general, calling for numerical procedures to approximate the output reachable set and hence for RNNs' probabilistic safety verification algorithms [13].
ii. *Lifelong learning*—A common problem in the context of indirect data-driven control is ensuring that the identified model remains an accurate approximation of the plant throughout its lifespan. While in the case of plant's dramatic variations (e.g., due to faults) the common practice is to collect new data and learn a new RNN model of the system, in the case of moderate and slow variations it would be advisable to exploit the online data to adapt the model to these changes. This practice is commonly referred to as lifelong learning and should be conducted by averting the catastrophic forgetting phenomenon, i.e., the overfitting of the most recent data and the consequent forgetting of the past data. For black-box RNN models, this issue represents an open research topic, whereas for NNARX architectures preliminary results have been reported in [15], based on a moving horizon estimation approach.
iii. *Physics-based machine learning*—One of the research directions that have been recently considered to be most promising by the scientific community is that of physics-based machine learning. In summary, it consists in exploiting the available qualitative knowledge of the physical laws governing the plant in order to improve the consistency, interpretability, generalizability, and ultimately the accuracy of the model. As discussed in [9] and references therein, physical consistency can be achieved via a suitable design of the training loss function and of the NN architecture, so as to ensure—for example—a known dynamical structure or the satisfaction of known relationships between variables [4].
iv. *Robust control design*—A control architecture capable of ensuring robustness properties with respect to disturbances and plant-model mismatch while satisfying input and state constraints is one of the most challenging research directions when adopting RNN models, due to their structural complexity. A preliminary approach in this direction has been proposed in [26] for NNARX models and, more recently, in [22] for LSTM models.

5 Conclusions

In this Brief, we summarized a novel framework towards the training of black-box Recurrent Neural Network (RNN) models with Incremental Input-to-State Stability (δISS) certification. The proposed method thus allows learning RNNs that are safe and robust against input perturbations and mismatches in initial conditions and applies to a variety of RNN architectures, such as Neural NARXs, Gated Recurrent Units, and Long Short-Term Memory networks. Relying on the model's δISS, theoretically sound model-based control strategies can be synthesized. In particular, in this Brief the design of a nonlinear model predictive control law with nominal closed-loop stability guarantees has been outlined, discussing the extension of the scheme to also achieve asymptotic zero-error setpoint tracking. Finally, the main open problems and promising future research directions, such as safety verification of RNN models and physics-based machine learning, have been reported.

References

1. Bayer F, Bürger M, Allgöwer F (2013) Discrete-time incremental ISS: a framework for robust NMPC. In: 2013 European control conference (ECC), pp 2068–2073. IEEE (2013)
2. Bengio, Y., Goodfellow, I., Courville, A.: Deep learning, vol. 1. MIT Press Massachusetts, USA (2017)
3. Bianchi, F.M., Maiorino, E., Kampffmeyer, M.C., Rizzi, A., Jenssen, R.: Recurrent neural networks for short-term load forecasting: an overview and comparative analysis. Springer (2017)
4. Boca de Giuli L, La Bella A, Scattolini R (2023) Physics-informed neural network modelling and predictive control of district heating systems. arXiv e-prints, arXiv-2310
5. Boccia, A., Grüne, L., Worthmann, K.: Stability and feasibility of state constrained MPC without stabilizing terminal constraints. Syst Control Lett **72**, 14–21 (2014)
6. Bonassi F (2023) Reconciling deep learning and control theory: recurrent neural networks for model-based control design. Doctoral dissertation, Politecnico di Milano, Advisor: R. Scattolini
7. Bonassi, F., Farina, M., Scattolini, R.: On the stability properties of gated recurrent units neural networks. Syst Control Lett **157**, 105049 (2021)
8. Bonassi, F., Farina, M., Scattolini, R.: Stability of discrete-time feed-forward neural networks in NARX configuration. IFAC-PapersOnLine **54**(7), 547–552 (2021)
9. Bonassi, F., Farina, M., Xie, J., Scattolini, R.: On recurrent neural networks for learning-based control: recent results and ideas for future developments. J Process Control **114**, 92–104 (2022)
10. Bonassi, F., La Bella, A., Farina, M., Scattolini, R.: Nonlinear MPC design for incrementally ISS systems with application to GRU networks. Automatica **159**, 111381 (2024)
11. Bonassi F, Scattolini R (2022) Recurrent neural network-based internal model control of unknown nonlinear stable systems. Eur J Control 100632
12. Bonassi, F., Oliveira da Silva, C.F., Scattolini, R.: Nonlinear MPC for offset-free tracking of systems learned by GRU neural networks. IFAC-PapersOnLine **54**(14), 54–59 (2021)
13. Bonassi F, Terzi E, Farina M, Scattolini R (2020) LSTM neural networks: input to state stability and probabilistic safety verification. In: Learning for dynamics and control. PMLR, pp 85–94
14. Bonassi F, Xie J, Farina M, Scattolini R (2022) An offset-free nonlinear MPC scheme for systems learned by Neural NARX models. In: 2022 IEEE 61st conference on decision and control (CDC), pp 2123–2128

15. Bonassi F, Xie J, Farina M, Scattolini R (2022) Towards lifelong learning of recurrent neural networks for control design. In: 2022 European control conference (ECC), pp 2018–2023
16. Francis, B.A., Wonham, W.M.: The internal model principle of control theory. Automatica **12**(5), 457–465 (1976)
17. Magni, L., De Nicolao, G., Scattolini, R.: Output feedback and tracking of nonlinear systems with model predictive control. Automatica **37**(10), 1601–1607 (2001)
18. Morari, M., Maeder, U.: Nonlinear offset-free model predictive control. Automatica **48**(9), 2059–2067 (2012)
19. da Silva Oliveira CF (2021) Offset-free nonlinear MPC for systems learned by LSTM networks. Master thesis. Politecnico di Milano, Italy
20. Pascanu R, et al (2013) On the difficulty of training recurrent neural networks. In: International conference on machine learning. PMLR, pp 1310–1318
21. Pillonetto G, Aravkin A, Gedon D, Ljung L, Ribeiro AH, Schön TB (2023) Deep networks for system identification: a survey. arXiv:2301.12832
22. Schimperna I, Magni L (2023) Robust offset-free constrained model predictive control with long short-term memory networks–extended version. arXiv:2303.17304
23. Schimperna, I., Toffanin, C., Magni, L.: On offset-free model predictive control with long short-term memory networks. IFAC-PapersOnLine **56**(1), 156–161 (2023)
24. Terzi, E., Bonassi, F., Farina, M., Scattolini, R.: Learning model predictive control with long short-term memory networks. Int J Robust Nonlinear Control **31**(18), 8877–8896 (2021)
25. Wu, Z., Luo, J., Rincon, D., Christofides, P.D.: Machine learning-based predictive control using noisy data: evaluating performance and robustness via a large-scale process simulator. Chem Eng Res Design **168**, 275–287 (2021)
26. Xie, J., Bonassi, F., Farina, M., Scattolini, R.: Robust offset-free nonlinear model predictive control for systems learned by neural nonlinear autoregressive exogenous models. Int J Robust Nonlinear Control **33**(16), 9992–10009 (2023)

On Data-Driven Optimization Methods in the Design and Control of Autonomous Systems

Lorenzo Sabug Jr.

1 Introduction

Context. In various applications in engineering, science, and other domains, we are faced with various difficult optimization problems, entailing the tuning or design of certain variables to minimize an objective function, subject to the satisfaction of constraints. Such problems are characterized by a non-trivial relation between the input variables (referred in the literature as tuning or design variables) and the objective and/or constraints. Examples of applications involve design of systems with different physical mechanisms, e.g. mechanical, electronic, hydrodynamic, etc., and where the fitness depends on their complex interaction with the environment. In many cases, hard-to-model factors like integration of digital and analog design, usage patterns, and environmental fluctuations, make the optimization problem even more challenging. Figure 1 illustrates the factors affecting the fitness of a complex system with respect to the design variables. Hence, closed-form mathematical expressions relating the design variables to the objective and constraints are not available, or otherwise highly difficult to extract or solve. Instead, the only sensible approach to evaluate the fitness is through simulations and/or experiments, i.e., by sampling the fitness at individual sampling points. This problem class, where we only have access to the objective and constraint values via sampling, is called *black-box* (or *global*) optimization.

Global optimization has attracted the interest of engineering practitioners and applied mathematicians for several decades, and in fact, it is still an open question for research. With the recent rise of machine learning and hybrid systems design,

The research that culminated in this contribution was supported by the Philippine Department of Science and Technology–Science Education Institute (DOST-SEI). Furthermore, the author acknowledges Prof. Lorenzo Fagiano and Prof. Fredy Ruiz for their research supervision and guidance.

L. Sabug Jr. (✉)
Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milan, Italy
e-mail: lorenzojr.sabug@polimi.it

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_8

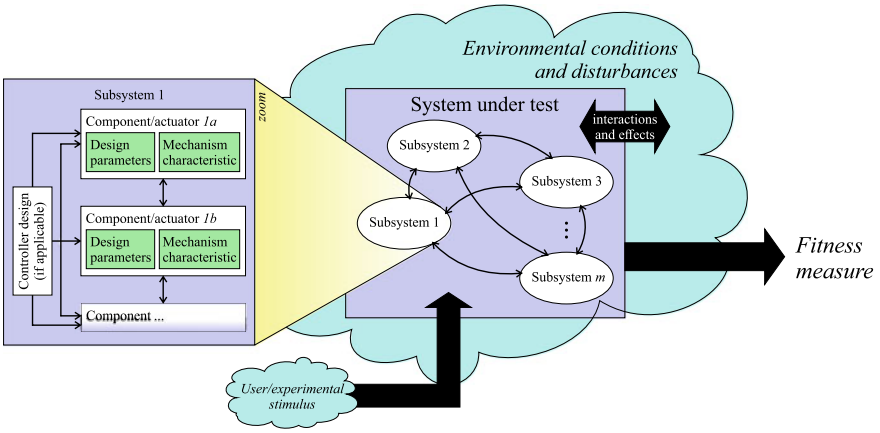


Fig. 1 Factors affecting the fitness of a complex system

technical applications call for more effective techniques to tune parameters in higher dimensions and in the presence of black-box constraints and evaluation noise.

Previous works. Most approaches to global optimization address a trade off between two conflicting goals: *exploitation*, where one attempts to sample around the current best point to improve it, and *exploration*, where distant or high-uncertainty regions are sampled to discover more the underlying functions. Previously developed methods for global optimization are plenty, however, most of them can be grouped in four conceptual categories:

1. *Population-based methods*: this category includes methods which involve a group ("population") of agents scattered throughout the optimization space, evaluating the objective/constraints at their respective locations. With succeeding iterations, these agents move their respective locations based on heuristics [7], usually based on animal behavior. "Generation-based" methods [4] are also included, whose agents evolve by a mix of combination and random mutation. While these have been highly popular due to their empirical performance and low computational burden, they need numerous function evaluations due to their batch-based paradigm. Hence, these algorithms are limited to "cheap" objectives and constraint, i.e., those which are easy to evaluate.
2. *Direct search techniques*: they entail evaluating points in a chosen set of search directions, which can be randomly-generated or along the basic directions, and comparing these points with the current best one ("incumbent"). Example methods in this category are Compass Search [8], Generalized Pattern Search [17], and the Mesh Adaptive Direct Search [1]. While they have negligible computational burden due to their simplicity, they perform well in practice, and are becoming more popular. However, convergence properties to the global optimum is not guaranteed at least when such methods are used without modifications.

3. *Model-based methods*: they involve iteratively refining a surrogate model from the data acquired so far. The surrogate model is then used for a "cheap" optimization, to select the next point for sampling. Examples for this category include kriging-based methods [9], and the popular Bayesian optimization [6]. While these techniques receive wide attention especially in the machine learning community, the surrogate modelling part brings high computational burden. As a result, these methods are mostly limited to lower dimensionality and lower evaluation budgets.
4. *Lipschitz-based methods*: their mechanism rests on the assumption that the underlying functions are Lipschitz continuous. These functions usually exploit the information regarding the lowest possible bound of the functions in the unsampled regions, to select the most promising point for sampling in the next iteration. In this category, we count the Piyavskii-Schubert method from the 1970s [12, 16], the Dividing Rectangles (DIRECT) method [5], and recently-proposed LiPO/AdaLIPO algorithms [10]. However, using the lower bounds as prediction is an optimistic one, which the sample after evaluation does not necessarily follow.

Contributions. This chapter describes the first time that the Set Membership approach is used for building a black-box optimization method, to address the shortcomings of previous works. While Set Membership approaches [11] have been used for system and function identification, filtering, and data-driven control, it has not yet been used for global optimization. With attractive properties like simple model-building technique, non-parametric modelling, and uncertainty quantification, it is a promising candidate for surrogate modelling, around which we can build a global optimization method. In the proposed Set Membership Global Optimization (SMGO) [14, 15], we build the Set Membership models of the objective and constraints from data (assuming their Lipschitz continuity). These models are then used to intelligently trade off between exploitation and exploration to select the next point for sampling. The resulting method is shown to have theoretical convergence, low computational burden, and reproducibility of results. We describe the problem setup, mechanism behind SMGO, followed by a discussion of its properties. Lastly, we summarize two case studies that successfully used SMGO in different engineering design problems.

2 Problem Setup

We consider the problem of finding a point that minimizes the scalar objective function $f(\boldsymbol{x})$, subject to the satisfaction of one or more constraints $g_s(\boldsymbol{x})$, $s = 1, \ldots, S$. The point $\boldsymbol{x} \doteq [x_1\ x_2\ \ldots\ x_D]$ is also referred as the *decision variable vector*, where D is the *dimensionality* of the problem. Furthermore, we are considering a search set $\mathcal{X} \subset \mathbb{R}^D$ that is convex and compact. This is a very common assumption in practical applications; in fact, most engineering design problems simply define respective search ranges $[\underline{x}_d, \overline{x}_d]$ for the decision variables $x_d,\ d = 1, \ldots, D$, which makes $\mathcal{X}$ a hyperrectangle.

We assume to have no access to the closed-form analytical expressions, nor any derivative/gradient information for the objective f and all constraints g_s. Instead, we only have access to their function values by sampling individual test points $\boldsymbol{x}$, using experiments, simulations, or a combination of both. Hence, f and g_s are what we call *hidden* or *black-box* functions, as they are referred in the literature. In addition, do not have assumptions on their convexity, nor even on the number of distinct local/global minima in the search space. Nevertheless, we do take an assumption regarding their regularity:

Assumption 1 The objective function f, and all constraint functions g_s are locally Lipschitz continuous throughout the considered search space $\mathcal{X}$, with their respective finite (but unknown) Lipschitz constants $\gamma, \rho_1, \ldots, \rho_S$:

$$f \in \mathcal{F}(\gamma),\ g_1 \in \mathcal{F}(\rho_1), \ldots,\ g_S \in \mathcal{F}(\rho_S)$$

where

$$\mathcal{F}(\eta) \doteq \{h : |h(\boldsymbol{x}_1) - h(\boldsymbol{x}_2)| \ \leq \eta \left\|\boldsymbol{x}_1 - \boldsymbol{x}_2\right\|, \forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}\}.$$

The above assumption is reasonable, and in fact one that is taken in most problems that involve physical systems, where the rates of change are finite. Furthermore, we take an assumption on the acquired values of f and g_s:

Assumption 2 The values of the objective function f and all constraints g_s can be evaluated at any point $\boldsymbol{x}^{(i)} \in \mathcal{X}$ without noise, in a setup referred to as an exact evaluation:

$$z^{(i)} = f(\boldsymbol{x}^{(i)}),\ c_1^{(i)} = g_1(\boldsymbol{x}^{(i)}), \ldots,\ c_S^{(i)} = g_S(\boldsymbol{x}^{(i)}).$$

At any chosen point $\boldsymbol{x}^{(i)}$, we assume that each evaluation gives access to both the values of $f(\boldsymbol{x}^{(i)})$ and all $g_s(\boldsymbol{x}^{(i)})$, in what we call a "synchronous evaluation".

When considering constraints, we adopt the convention that g_s is satisfied at $\boldsymbol{x}$ if $g_s(\boldsymbol{x}) \geq 0$. We consider that the feasible set $\mathcal{G}$, which is the intersection of the respective satisfaction sets of g_s, exists and has a finite measure:

Assumption 3 Consider the feasible set $\mathcal{G} \doteq \mathcal{X} \cap \left\{\cap_{s=1}^{S} \{\boldsymbol{x} : g_s(\boldsymbol{x}) \geq 0\}\right\}$. We assume that

$$\mathcal{L}(\mathcal{G}) > 0,$$

where $\mathcal{L}$ is the operator for the Lebesgue measure.

Due to the above assumptions, we can then declare that at least one global minimizer $\boldsymbol{x}^*$ exists, defined as

$$\boldsymbol{x}^* \in \mathcal{X}^* \doteq \left\{ \boldsymbol{x} \in \mathcal{G} \;\middle|\; \forall \boldsymbol{x}' \in \mathcal{G},\, f(\boldsymbol{x}') \geq f(\boldsymbol{x}) \right\}. \tag{1}$$

and with the corresponding minimum objective $z^* = f(\boldsymbol{x}^*)$.

3 Set Membership Global Optimization (SMGO)

3.1 Algorithm

Overview The Set Membership Global Optimization (SMGO), discussed in [14, 15], is a new global optimization technique, that uses the Set Membership approach to strategically trade off between exploitation (sampling around the current best evaluated point, to improve on the current best objective) and exploration (sampling around undiscovered regions of the search space, to learn more about the function). A general flow of the algorithm is shown in Fig. 2.

Data set and model update Let us denote a new sample at iteration n by a tuple $\mathring{\boldsymbol{x}}^{(n)} \doteq (\boldsymbol{x}^{(n)}, z^{(n)}, \boldsymbol{c}^{(n)})$, composed of the sampled point $\boldsymbol{x}^{(n)}$, the corresponding objective value $z^{(n)}$, and the vector of constraint values $\boldsymbol{c}^{(n)} \doteq [c_1^{(n)}, \ldots, c_S^{(n)}]$. In this step, we iteratively introduce the new entry to the data set $\boldsymbol{X}^{\langle n-1 \rangle}$, building $\boldsymbol{X}^{\langle n \rangle}$:

$$\boldsymbol{X}^{\langle n \rangle} = \boldsymbol{X}^{\langle n-1 \rangle} \cup \mathring{\boldsymbol{x}}^{(n)}.$$

Given the data set $\boldsymbol{X}^{\langle n \rangle}$, we identify the best tuple $\mathring{\boldsymbol{x}}^{*\langle n \rangle}$ as follows:

$$\mathring{\boldsymbol{x}}^{*\langle n \rangle} = (\boldsymbol{x}^{*\langle n \rangle}, z^{*\langle n \rangle}, \boldsymbol{c}^{*\langle n \rangle}) \doteq \arg \min_{\mathring{\boldsymbol{x}}^{(i)} \in \boldsymbol{X}^{\langle n \rangle}} z^{(i)}, \text{ s.t. } \boldsymbol{c}^{(i)} \geq 0.$$

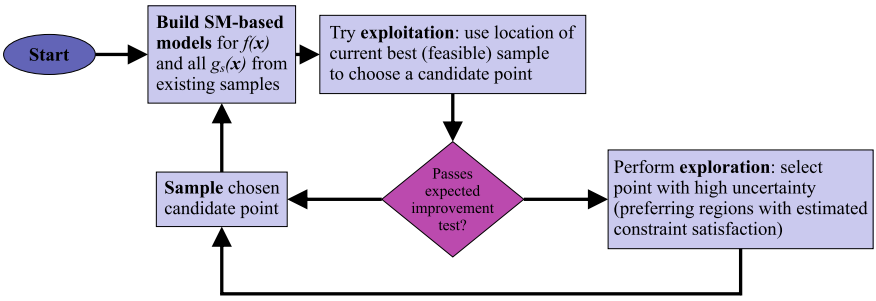


Fig. 2 SMGO algorithm logic

Furthermore, the estimates $\tilde{\gamma}^{\langle n \rangle}, \tilde{\rho}_1^{\langle n \rangle}, \ldots, \tilde{\rho}_S^{\langle n \rangle}$ of the Lipschitz constant $\gamma, \rho_1, \ldots, \rho_S$ are then updated, which refines the SM model (further details on calculating these estimates are discussed in [15]). From these information, we build the SM upper- and lower bounds for f, denoted as $\overline{f}^{\langle n \rangle}(\boldsymbol{x})$ and $\underline{f}^{\langle n \rangle}(\boldsymbol{x})$ as illustrated in Fig. 3. We also define the central estimate $\tilde{f}^{\langle n \rangle}(\boldsymbol{x}) \doteq \frac{1}{2}\left(\overline{f}^{\langle n \rangle}(\boldsymbol{x}) + \underline{f}^{\langle n \rangle}(\boldsymbol{x})\right)$, and the uncertainty $\lambda^{\langle n \rangle}(\boldsymbol{x}) \doteq \overline{f}^{\langle n \rangle}(\boldsymbol{x}) - \underline{f}^{\langle n \rangle}(\boldsymbol{x})$. Analogously, we denote the upper- and lower bounds, central estimate, and uncertainty for a constraint g_s as $\overline{g}_s^{\langle n \rangle}(\boldsymbol{x})$, $\underline{g}_s^{\langle n \rangle}(\boldsymbol{x})$, $\tilde{g}_s^{\langle n \rangle}(\boldsymbol{x})$, and $\pi_s^{\langle n \rangle}(\boldsymbol{x})$, respectively.

From the SM-based models we can estimate the regions $\tilde{\mathcal{G}}_s$ which satisfy the corresponding constraint g_s, shaded in the second and third rows of Fig. 3. Using a weighting factor $\Delta \in [0, 1]$ that we refer as the *risk parameter* [15], we define the satisfaction region estimate $\tilde{\mathcal{G}}_s$ as

$$\tilde{\mathcal{G}}_s \doteq \left\{ \boldsymbol{x} \in \mathcal{X} : \Delta \tilde{g}_s(\boldsymbol{x}) + (1 - \Delta)\underline{g}_s(\boldsymbol{x}) \geq 0 \right\}.$$

A setting of $\Delta = 0$ uses the most cautious estimate of satisfaction regions, using the SM lower bounds $\underline{g}_s$ as a worst-case estimate. On the other hand, $\Delta = 1$ leads to the most lenient satisfaction estimate using $\tilde{g}_s$, and a much larger $\tilde{\mathcal{G}}_s$. The satisfaction region estimates from different Δ is shown in Fig. 4. Given all $\tilde{\mathcal{G}}_s$, we define the feasible region as

$$\tilde{\mathcal{G}} \doteq \cap_{s=1}^{S} \tilde{\mathcal{G}}_s.$$

However, we note that even with the most cautious setting $\Delta = 0$, constraint violations *might* still occur in the setting that we treat because we do not have a knowledge of the Lipschitz constants for f and all g_s (see Assumption 1).

Generation of Candidate Points From the samples, we methodically generate a set $\boldsymbol{E}^{\langle n \rangle}$ of candidate points, from which we select the next sampling point $\boldsymbol{x}^{(n+1)}$ via exploitation or exploration. The candidate points generation method is highly flexible, and can be adjusted according to the needs of the user. Moreover, this subroutine can be skipped entirely, and the exploitation/exploration routines can be treated as continuous-space optimization routines.

A suggested method in [15] to generate candidate points is iterative: for every incoming sampled point $\boldsymbol{x}^{(n)}$, it generates candidate points in the positive and negative cardinal directions up to the boundaries of $\mathcal{X}$, and further ones in the direction to all existing sampled points in $\boldsymbol{X}^{\langle n \rangle}$. This candidate point generation approach is illustrated in Fig. 5 for a two-dimensional example.

Exploitation We now attempt to select a sampling point from candidate points lying in a small region around the best sampled point, referred to as the *trust region*. Furthermore, we only choose from candidate points also estimated to be feasible, i.e., those also belonging to $\tilde{\mathcal{G}}$. The metric to choose the exploitation candidate point

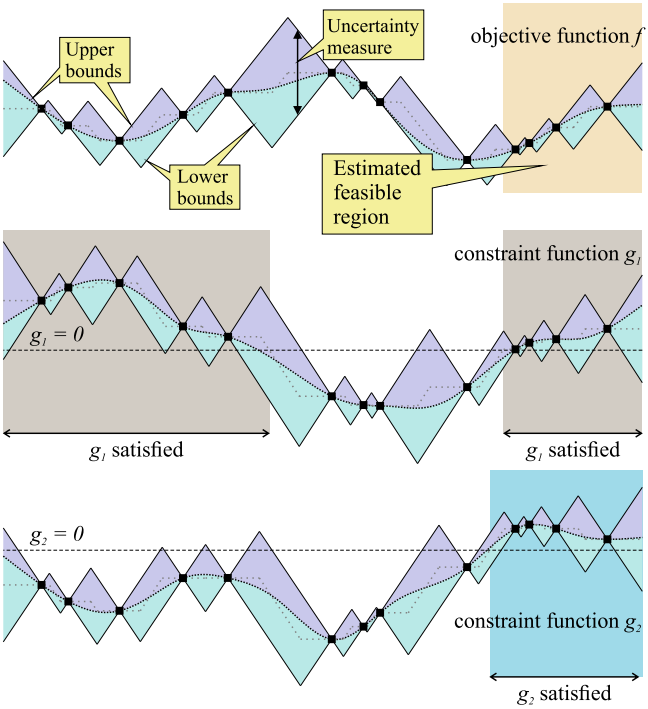


Fig. 3 Set Membership models of objective and constraints from samples

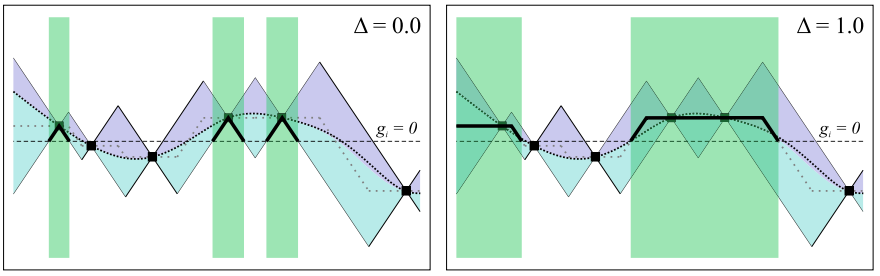


Fig. 4 Satisfaction region estimates from different Δ values

$\boldsymbol{x}_\theta^{\langle n \rangle}$ is based on its promise to improve on the current best objective value, which prioritizes lower central estimate, and, to a small degree, a higher uncertainty.

The chosen exploitation point $\boldsymbol{x}_\theta^{\langle n \rangle}$, if it exists, is subjected to an *expected improvement test* (EIC) [14, 15], to evaluate if it is worthy to be evaluated using an expensive experiment or simulation. This condition is checked using the SM bounds; in particular, we test if the lower bound $\underline{f}(\boldsymbol{x}_\theta^{\langle n \rangle})$ improves on the best sampled objective value by at least a set threshold η, as in Fig. 6. If the EIC is passed, $\boldsymbol{x}_\theta^{\langle n \rangle}$ is assigned as the next sampling point:

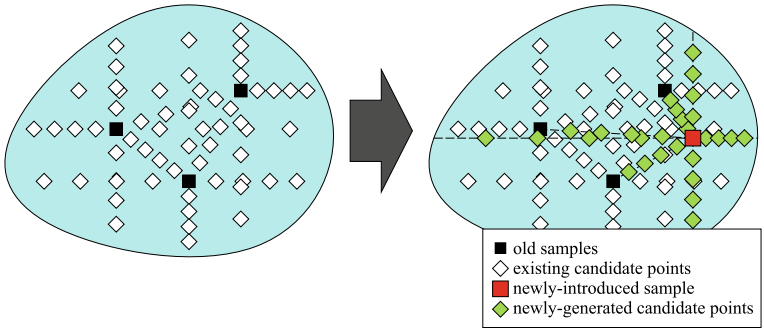
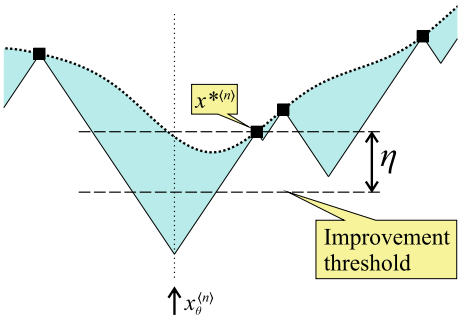


Fig. 5 Generation of candidate points, with the method used in [15]

Fig. 6 Expected improvement test



$$\boldsymbol{x}^{(n+1)} \leftarrow \boldsymbol{x}_{\theta}^{\langle n\rangle},$$

otherwise, we skip it, and choose $\boldsymbol{x}^{(n+1)}$ by the exploration routine, as discussed next.

Exploration The exploration subroutine of SMGO attempts to discover the shape of the function by sampling a point in the high-uncertainty regions. In contrast to exploitation when we restrict ourselves to feasible candidate points within the trust region, we now choose a candidate point from throughout $\mathcal{X}$. A merit function is designed to pick a point with the highest uncertainty with respect to the objective and constraints, and prioritizing points with higher number of (estimated) satisfied constraints. The chosen exploration point $\boldsymbol{x}_{\phi}^{\langle n\rangle}$ is then directly assigned as the sampling point for the next iteration $n + 1$:

$$\boldsymbol{x}^{(n+1)} \leftarrow \boldsymbol{x}_{\phi}^{\langle n\rangle}.$$

4 Algorithm Properties

Convergence In [15], the proposed SMGO-Δ is proven as convergent to a feasible point whose objective is within a finite precision $\varepsilon > 0$ from the absolute minimum z^*, assuming only a Lipschitz continuity of the underlying black-box functions f and all g_s. We have shown that consecutive exploitation routines will fail the expected improvement test, which, due to the algorithm logic (see Fig. 2), allows for exploration samplings to be done infinitely often. Furthermore, the design of the exploration routine, through the candidate points generation technique and the exploration merit function, causes a progressively dense distribution of points to be sampled throughout the search space $\mathcal{X}$. As a result, we can approach the optimal point $\boldsymbol{x}^*$ up to any finite radius within a finite number of samplings, and correspondingly, the best sampled objective will be ε-optimal with respect to z^*. More details on the convergence proofs are provided in [15].

Computational complexity The practical implementation of SMGO is based on keeping a database of candidate points, storing their respective SM-based bounds. As this database is used as a look-up table for the exploitation and exploration routines, most of the computations are devoted to updating this database at every iteration. The computational complexity of SMGO is mostly due to the number of candidate points generated, which, for the candidate points generation mechanism described in [15], results in $\mathcal{O}(Dn + n^2)$. More discussions regarding the SMGO computational complexity, and iterative implementations can be found in [14, 15].

Implementation aspects There are important concerns that arise in most practical implementations, that we need to address in building SMGO. The most apparent concern is on the presence of noise and/or disturbance of unknown bounds. In this case, assuming that these bounds are finite (but we do not assume anything regarding its distribution), we can estimate the noise bounds by utilizing the method proposed in [2], and integrate this information in the construction of the SM bounds. The exploitation and exploration routines are performed as usual.

As SMGO proposes a methodical approach to generating candidate points, the results are completely reproducible from one run to another, i.e., given the same starting point, SMGO will produce the same result and the same sampling history, assuming the absence of noise. However, this same methodical generation of candidate points severely limit the possible search directions, especially during early iterations, SMGO allocates a fixed number of candidate points at the start of the algorithm, scattered around $\mathcal{X}$ according to a pseudo-random distribution. This ensures that even during the initial iterations, SMGO can have more options on sampling locations, while still maintaining reproducibility (because the pseudo-random distribution can be duplicated between runs).

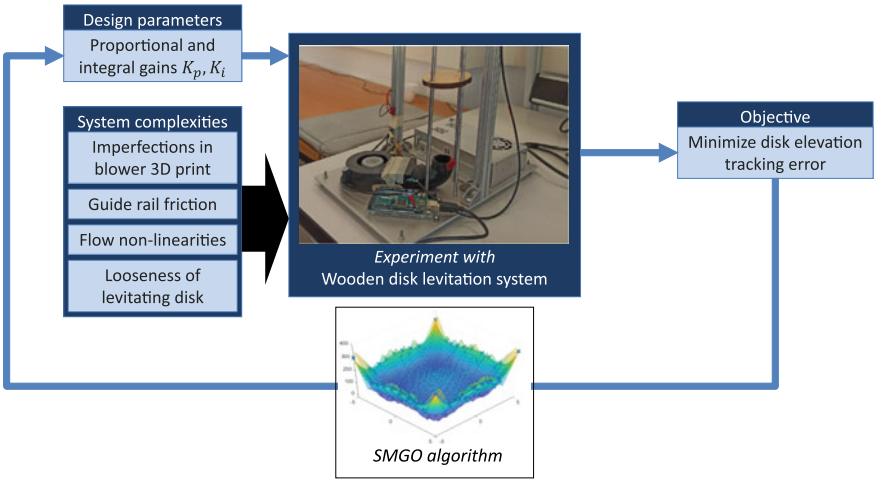


Fig. 7 Control design for a wooden disk elevation system

5 Sample Applications

5.1 Experiment-Based Controller Tuning

In this first application, the design of a proportional-integral (PI) controller for a tabletop wooden disk elevation system. As shown in Fig. 7, it has the objective of achieving the best disk elevation tracking performance, i.e., minimizing the elevation tracking error with respect to the a set reference height. Even in this seemingly simple system, there are already several non-trivial mechanisms at play, including imperfections in the blower nozzle 3D printing, non-linear aerodynamic response of the blower, friction of the guide rails, and looseness of the wooden disk.

We applied SMGO in tuning the PI controller, and results show that the transient response of a controller tuned via black-box optimization (SMGO) outperforms one which was tuned from an estimated system model. More information regarding the system model, particularities in the optimization setup, and the results can be found in [3].

5.2 Plant-Controller Co-Design

Another application that involves non-convex optimization is plant-controller co-design. Consider a CubeSat to be designed for optical missions, i.e., taking images of ground targets. In this case, our objective is to minimize the attitude (pointing) error of the CubeSat, to satisfy its optical mission. In addition, we are constrained

by a minimum percentage of communications time per flyby over the ground station (GS), located in Kiruna, Sweden, and also the maximum average power consumption per camera task. We designed for this case the following: sliding mode controller tuning for reaction wheels (RWs), sizing of magnetic rods, and sizing of hysteretic materials. This design problem is highly difficult because of the interactions between the passives design (magnetic rods and hysteretic materials) and the RW controller, and the non-trivial effects with the environment, in particular the Earth's magnetic field. A diagram illustrating the complexity of the optimization problem is shown in Fig. 8, and more information regarding the system description and non-trivial interactions of the design variables and the objective/constraints can be found in [13].

The SMGO algorithm was used for the design process, interfaced with a MATLAB/Simulink-based CubeSat model. Simulations of image acquisition tasks and GS communication scenarios are run with the CubeSat with different passive magnet, hysteretic material, and RW controller tunings. For comparison, other commonly-used design strategies like independent design, sequential design, and Latin hypercube-based sampling are tested as well. As can also be seen in detailed results in [13], SMGO-based design was found to have the best attitude tracking performance, while satisfying the operational constraints on GS communication and power consumption.

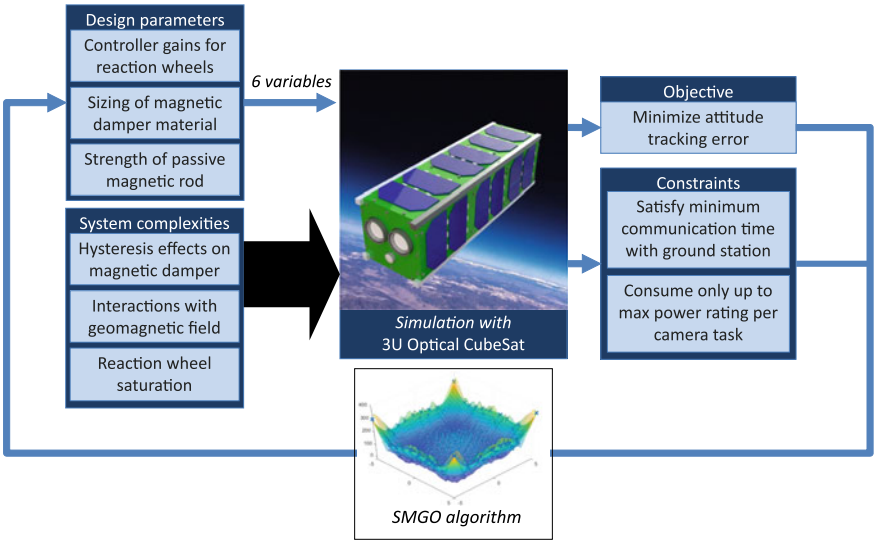


Fig. 8 Plant-controller co-design for an optical CubeSat

6 Conclusions

In this contribution, the Set Membership Global Optimization (SMGO) is introduced, which is a new approach for non-convex optimization based on the Set Membership framework. The resulting approach is discussed, using the data-derived Set Membership model bounds and uncertainties to intelligently trade off between exploitation and exploration to decide on the next sampling point. We provide an overview of its theoretical properties, in particular its convergence to the global optimal value up to any finite precision, as well as several implementation concerns. We have also provided overview information on two test cases on which SMGO was used on controller design of a disk levitation system, and a simulations-based plant-controller co-design for an optical spacecraft.

References

1. Audet C, Dennis JE (2006) Mesh adaptive direct search algorithms for constrained optimization. SIAM J Optim 17(1):188–217. https://doi.org/10.1137/040603371
2. Fagiano L, Novara C (2016) Learning a nonlinear controller from data: theory, computation, and experimental results. IEEE Trans Autom Control 61(7):1854–1868. https://doi.org/10.1109/TAC.2015.2479520
3. Galbiati R, Sabug L, Ruiz F, Fagiano L (2022) Direct control design using a Set Membership-based black-box optimization approach. In: 2022 IEEE conference on control technology and applications, CCTA 2022, pp 1259–1264. 10.1109/CCTA49430.2022.9966147
4. Holland JH (1992) Genetic algorithms. Sci Am 267(1):66–73. http://www.jstor.org/stable/24939139
5. Jones DR, Perttunen CD, Stuckman BE (1993) Lipschitzian optimization without the Lipschitz constant. J Optim Theory Appl 79(1):157–181. https://doi.org/10.1007/BF00941892
6. Jones DR, Schonlau M, Welch WJ (1998) Efficient global optimization of expensive black-box functions. J Glob Optim 13:455–492. https://doi.org/10.1023/A:1008306431147
7. Kennedy J, Eberhart R (2011) Particle swarm optimization. In: Proceedings of ICNN'95—international conference on neural networks, vol 4. IEEE, pp 1942–1948. 10.1109/ICNN.1995.488968
8. Kolda TG, Lewis RM, Torczon V (2003) Optimization by direct search: new perspectives on some classical and modern methods. SIAM Rev 45(3):385–482. https://doi.org/10.1137/S003614450242889
9. Li Y, Wu Y, Zhao J, Chen L (2017) A Kriging-based constrained global optimization algorithm for expensive black-box functions with infeasible initial points. J Glob Optim 67(1–2):343–366. https://doi.org/10.1007/s10898-016-0455-z
10. Malherbe C, Vayatis N (2017) Global optimization of Lipschitz functions. In: 34th international conference on machine learning, ICML 2017, vol 5, pp 3592–3601
11. Milanese M, Novara C (2004) Set Membership identification of nonlinear systems. Automatica 40(6):957–975. https://doi.org/10.1016/j.automatica.2004.02.002
12. Piyavskii S (1972) An algorithm for finding the absolute extremum of a function. USSR Comput Math Math Phys 12(4):57–67. https://doi.org/10.1016/0041-5553(72)90115-2
13. Sabug L, Incremona GP, Tanelli M, Ruiz F, Fagiano L (2023) Simultaneous design of passive and active spacecraft attitude control using black-box optimization. Control Eng Pract 135:105516. https://doi.org/10.1016/j.conengprac.2023.105516
14. Sabug L, Ruiz F, Fagiano L (2021) SMGO: a set membership approach to data-driven global optimization. Automatica 133:109890. https://doi.org/10.1016/j.automatica.2021.109890

15. Sabug L, Ruiz F, Fagiano L (2022) SMGO-DeltaΔ: balancing caution and reward in global optimization with black-box constraints. Inf Sci 605:15–42. https://doi.org/10.1016/j.ins.2022.05.017
16. Shubert BO (1972) A sequential method seeking the global maximum of a function. SIAM J Numer Anal 9(3):379–388. https://doi.org/10.1137/0709036
17. Torczon V (1997) On the convergence of pattern search algorithms. SIAM J Optim 7(1):1–25. https://doi.org/10.1137/S1052623493250780

Model Predictive Control for Constrained Navigation of Autonomous Vehicles

Danilo Saccani

Abstract As autonomous vehicles become increasingly prevalent in our daily lives, new control challenges arise to ensure their safety and the safety of their surroundings. This work addresses these challenges by developing a suitable regulator that strikes a balance between different objectives. The first one is 'safety', which involves satisfying constraints and consistently avoiding obstacles. The second objective is 'exploitation', which aims to optimize the utilization of existing knowledge about the environment, reducing the overly cautious behaviour of guaranteed collision-free approaches. The third objective is 'exploration', which pertains to the ability to discover potential unknown areas while avoiding getting stuck in blocked regions. The design of motion planning algorithms for such systems requires carefully managing the trade-off between these requirements. Among the various approaches to dynamic path planning, discrete optimization methods such as Model Predictive Control (MPC) have gained significant attention. MPC excels in handling state and input constraints to ensure safety while minimizing a cost function defined by the user, enabling both exploitation and exploration aspects. By developing a suitable regulator and leveraging MPC approaches, this work aims to address the complex control challenges faced by autonomous vehicles and other safety-critical applications, ensuring a balance between safety, exploitation, and exploration.

1 Introduction

Over the past decade, advancements in technology, coupled with their decreasing costs, have led to the widespread integration of autonomous systems into our daily lives [1]. These systems have had a significant impact on various fields, including

This research has been supported by the Italian Ministry of University and Research (MIUR) under the PRIN 2017 grant n. 201732RS94 "Systems of Tethered Multicopters", and by the European Union's Horizon 2020 research.

D. Saccani (✉)
Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Via Ponzio 34/5, 20133 Milano, Italy
e-mail: danilo.saccani@polimi.it

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_9

technology, science, and society, making them valuable tools for a range of applications, from military and commercial uses to hobbies. Currently, autonomous systems are primarily utilized in known and closed environments like industrial production plants, where the system behavior can be assured by detailed models of the system and its environment. However, there is a growing demand to employ autonomous systems, particularly autonomous vehicles, in heterogeneous and unknown environments, such as search and rescue robots [12], environmental monitoring drones [3], and self-driving cars on highways [5], among others. This trend has sparked extensive research in multiple domains, demonstrating that autonomous vehicles can carry out complex missions without human intervention. In this doctoral dissertation, we try to answer the following question to achieve autonomous navigation:

How can I reach a particular location without colliding with the surrounding environment?

Trying to answer this question, we can identify the following requirements:
Safety: Vehicle navigation approaches have to guarantee classical requirements, such as the design of a dynamically feasible trajectory able to satisfy actuators and state variables limits while at the same time guaranteeing the satisfaction of constraints arising from the perception of the environment. To guarantee an obstacle-free motion is also required to the controller to ensure persistent constraints satisfaction despite different sources of uncertainty that can lead to a failure or crash of the system.
Exploitation: The main task we aim to solve is reaching a given location. This objective must be achieved guaranteeing the above mentioned safety requirement. These objectives, however, are often conflicting and can potentially lead to performance degradation during the navigation and to a too-conservative behaviour of the vehicle if the trade-off between exploitation and safety is not considered.
Exploration: During this phase, it is crucial to account for obstacles and other agents in the environment that may hinder direct access to a desired location. To ensure successful navigation, it becomes necessary to gather and store information about the surroundings. This information is then utilized to devise a strategy that avoids the system from becoming trapped in a local minimum, even when the desired location is ultimately reachable.

This doctoral thesis aims to design multiple predictive control architectures that address the trade-off between the identified requirements for autonomous vehicle navigation. This summary brief is structured as follows: In Sect. 3, the design of MPC schemes that can satisfy safety requirements under time-varying constraints, which shift with the system state, is presented. Then, Sect. 4 introduces a novel MPC formulation named multi-trajectory MPC, which allows for better exploitation of current information about the environment. In Sect. 5, a high-level receding horizon strategy for environment exploration is presented, which utilizes a graph-based map of the environment constructed online. Section 6 showcases real-world applications and simulations, demonstrating the proposed approach's effectiveness in addressing various problems within realistic scenarios. Finally, Sect. 7 concludes this brief with some final remarks and outlines possible future research directions.

2 Constrained Autonomous Navigation Problem

As mentioned in Sect. 1, for autonomous vehicles navigation, ensuring safe and efficient path planning is of paramount importance. To solve this problem, this doctoral thesis proposes Model Predictive Control (MPC) strategies to drive the system to the final target. MPC is a optimization-based control strategy that combines real-time feedback with predictive models to calculate and optimize the trajectory of the vehicle [11]. By leveraging mathematical models and optimization algorithms, MPC empowers autonomous vehicles to make intelligent decisions, navigate complex environments, and avoid obstacles effectively. At its core, MPC operates by continuously predicting the future behavior of the vehicle and optimizing its trajectory based on a defined objective and set of constraints. This predictive nature enables the vehicle to anticipate potential obstacles and navigate complex scenarios with agility. By considering both current and future states of the vehicle, MPC provides a reliable approach to path planning, facilitating smoother and safer autonomous driving experiences. The predictive modeling aspect of MPC involves creating a mathematical representation of the vehicle's dynamics and its interaction with the surrounding environment. This model captures essential parameters such as the vehicle's position, velocity, acceleration, and other relevant variables. The accuracy and fidelity of the predictive model play a crucial role in the effectiveness of MPC, as it directly influences the quality of trajectory predictions and subsequent decision-making. To leverage the complexity of the model and achieve faster implementation MPC can be employed in a hierarchical manner. By dividing the control problem into multiple levels, each addressing different time scales and aspects of the autonomous vehicle's behavior, MPC allows for efficient decision-making and real-time responsiveness. In this doctoral thesis, we are interested in analyzing high-intermediate levels, where we design a high-level plan (Sect. 5) and use a trajectory planner to refine the plan and generate a feasible trajectory considering the vehicle's dynamic constraints and environmental factors (Sects. 3 and 4). Here, MPC can be employed to optimize the trajectory based on real-time sensor data and accurately predict the vehicle's future behavior, accounting for factors like vehicle dynamics and environment conditions. To this end, we consider an autonomous vehicle described by a generic nonlinear discrete-time invariant model with a state vector $\boldsymbol{x}(k) \in \mathbb{R}^{n_x}$ that encompasses essential vehicle dynamics, such as position and velocity, and an input vector $\boldsymbol{u}(k) \in \mathbb{R}^{n_u}$. The system is characterized by a state dynamics function $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$. The system is subject to time-invariant input and state constraints that impose physical limitations or boundaries on these variables, given by:

$$\boldsymbol{u}(k) \in \mathcal{U}, \quad \boldsymbol{x}(k) \in \bar{\mathcal{X}} \tag{1}$$

Here, $\mathcal{U} \subset \mathbb{R}^{n_u}$ and $\bar{\mathcal{X}} \subset \mathbb{R}^{n_x}$ represent non-empty closed convex sets. The system is also subject to a non-empty time-varying convex set of constraints $\mathcal{X}(k) \subset \mathbb{R}^{n_x}$ arising from the perception of the environment:

$$\boldsymbol{x}(k) \in \mathcal{X}(k). \tag{2}$$

We are now in a position to formally define the problem we aim to solve. Given a state reference $\bar{\boldsymbol{r}} \in \mathbb{R}^{n_x}$ and the system state $\boldsymbol{x}(k)$ at each time step, our objective is to design a state feedback control law $\boldsymbol{u}(k) = \kappa(\boldsymbol{x}(k), \bar{\boldsymbol{r}}, \mathcal{X}(k))$ that drives the system's state as close as possible to the reference $\bar{\boldsymbol{r}}$, while satisfying the constraints (1) and (2) for all $k \geq 0$.

3 Environment Aware MPC for Autonomous Navigation

When considering autonomous navigation problems, it is crucial for the system to perceive its surroundings and avoid collisions with the environment or other agents. This perception can be translated into a set of system state constraints that may change over time. An example of this is a vehicle that is equipped with an exteroceptive sensor capable of sensing its surrounding unknown environment. At each time step, sensor measurements can be used to derive a safe set, which evolves during navigation. While theoretical guarantees such as recursive feasibility and stability have been studied extensively for MPC schemes with time-invariant constraints, the problem of time-varying state constraints has received relatively less attention in the literature, despite its broad impact in various applications. The work presented in [10] is one of the few papers that addresses this problem. The authors analyze two interesting cases: modeled constraint variation and constraints with bounded changes. Ensuring persistent constraint satisfaction becomes crucial to meet the safety requirement, given the time-varying nature of the constraints. Recursive feasibility is a crucial property of MPC as it guarantees the ability to find a feasible solution at each sampling time. It ensures that the optimization problem can be successfully solved, resulting in a trajectory that satisfies all constraints while optimizing the defined objectives. When time-varying constraints are considered in MPC, the challenge lies in maintaining feasibility throughout different time steps despite the potential evolution of the constraints, which could be influenced by environmental or sensor factors. In this section, our focus is on achieving persistent constraint satisfaction under two specific types of time-varying constraints that arise when an autonomous vehicle is equipped with an on-board local sensor for perceiving the surroundings.

3.1 State Shifting Constraint

When the vehicle is equipped with an on-board sensor or a communication device, the information provided by the sensor can be interpreted as a set of constraints centered at the vehicle's position. In [14], we considered a constant polytopic set centered at the vehicle position to describe a communication set around the agent. Despite the constant nature of the constraint set, the shifting feature can cause a loss of feasi-

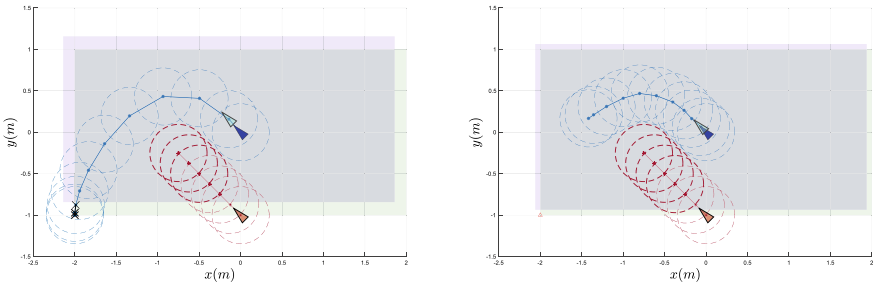


Fig. 1 Example showing the loss of recursive feasibility for a position shifting state constraint. On the left, the traditional implementation without recursive feasibility guarantees, while on the right the proposed implementation. We consider the trajectory of the blue agent, while the red agent can be seen as a fixed obstacle. Blue line with '*' represents the predicted state trajectory at time k. Due to the presence of a terminal state constraint, the tail of this trajectory represent a candidate trajectory at time $k+1$. Light green polytope represents the state constraint at time k, while light blue polytope represents the state constraint shifted at the first predicted state. In both examples, an obstacle avoidance constraint is imposed, and the dashed circles around the trajectories represent the size of the agents

bility, as depicted in Fig. 1 (left). To guarantee recursive feasibility, we proposed an implementation that ensures, by construction, that the tail of the predicted trajectory lies within the safe sets generated by shifting the set along the trajectory, as shown in Fig. 1 (right). The second type of shifting constraints considered in this doctoral thesis is a time-varying state constraint represented by a sequence of time-varying and unpredictable state constraints that shift with the vehicle state. Unlike the previous case, the shape of the constraint changes at each time step. Due to the unpredictable and time-varying nature of the constraints, in this case, it is not possible to guarantee recursive feasibility anymore, however, under suitable assumptions, it is possible to ensure persistent constraints satisfaction. In [13], these constraint sets were used to represent a sequence of unpredictable obstacle-free regions determined by a drone's sensor during navigation in an unknown environment. Under the assumption of a static environment, it can be proven that persistent constraint satisfaction is guaranteed if the predicted trajectory remains within one of the constraint sets encountered up to the current time step.

4 Multi-trajectory MPC

As shown in Sect. 3 and illustrated in Fig. 1, ensuring recursive feasibility with time-varying constraints requires striking a delicate balance between safety and system performance optimization. To manage this trade-off, we propose the multi-trajectory MPC (mt-MPC) formulation. This approach predicts two trajectories: a "safe" trajectory that remains within the safe set and reaches a safe state, and an "exploiting" trajectory that allows violations of current constraints if they are overly conserva-

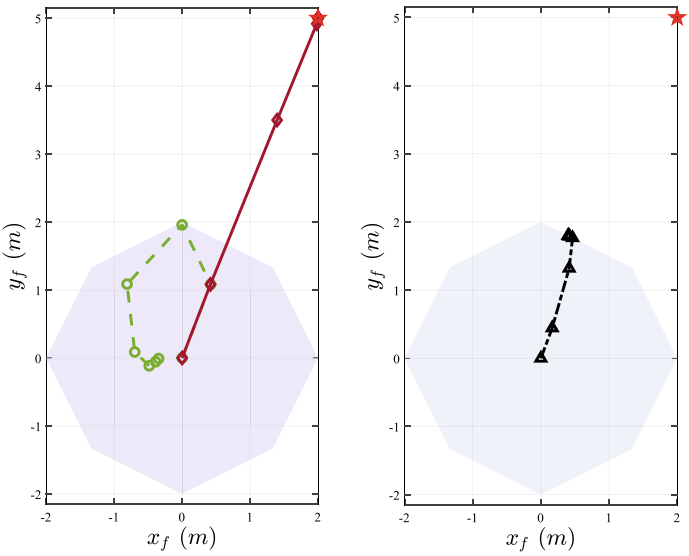


Fig. 2 Example illustrating the predicted state evolution with the multi-trajectory approach (left) versus a single-trajectory one (right). Green line with '$\circ$' represents the safe trajectory; Red line with '$\diamond$' is the exploitation trajectory; black dash-dotted line with '$\diamond$' is the trajectory of the single-trajectory MPC. Red '$\star$': target. The time-varying constraints are represented with colored polytopes

tive. The two trajectories share a common control action at the current step and diverge subsequently in the prediction. To illustrate this concept, consider the two-dimensional example shown in Fig. 2, where the right side demonstrates a single-trajectory approach and the left side presents a multi-trajectory approach. The trajectories are calculated by solving a Finite Horizon Optimal Control Problem (FHOCP) with the goal of reaching a target beyond the safe set. The single-trajectory MPC constrains the entire predicted trajectory to lie within the safe set, minimizing the average deviation from the target. However, it neglects the possibility of an improved safe set in the future. In contrast, the multi-trajectory strategy plans a significantly better (yet currently unfeasible) exploiting trajectory while maintaining a backup safe trajectory in case the constraints' set does not expand towards the target. By comparing the positions reached by the two approaches at the first predicted time step, implemented in a receding horizon fashion, the potential advantage of the multi-trajectory approach becomes evident. Thus, the resulting FHOCP is divided into two predictions: the exploitation trajectory, which is considered in the cost function to drive the system towards the desired reference, and the safe trajectory, which satisfies the time-varying state constraints and ensures reaching a steady state within the current set of state constraints. The problem is generally a nonlinear program (NLP), and under regularity assumptions, a numerical solver can compute a (possibly local) minimum. To reduce the complexity, a linear model of the system and a quadratic cost function can be considered, resulting in a quadratic programming (QP) problem,

as shown in [13]. If feasible, the QP problem can be efficiently solved for a global minimizer. While the ideal formulation guarantees the recursive feasibility of the approach by applying one of the methods presented in Sect. 3 to the safe trajectory, it may not ensure the convergence of the MPC scheme. To guarantee convergence, we propose a modified version of the FHOCP inspired by the works presented in [8, 9]. This modified version includes a convergence constraint that enforces a decreasing cost function associated with the safe trajectory over time, along with an offset cost that drives the terminal state towards the final reference.

5 Navigation Around Obstacles

The possibility that the system gets stuck in front of an obstacle that obstructs the path between the current vehicle position and the target is a common challenge in optimization-based autonomous navigation. To tackle this issue, we propose a mapping and exploration strategy called G-BEAM (Graph-Based Exploration and Mapping). The G-BEAM strategy leverages the under-approximation of the free space derived from exteroceptive sensor measurements to construct a graph representation of the environment. This graph is subsequently used to compute an optimal path for either exploring the environment or reaching a predetermined target. Before delving into the details of the approach, let us provide a description of the employed sensor and how we can effectively utilize the sensor measurements.

5.1 Exteroceptive Sensor and Convex Under-Approximation of the Free Space

We assume that, at each time step, we have access to the vehicle's position and readings from an exteroceptive sensor, which is oriented parallel to the horizontal plane and capable of detecting the surrounding environment. The vehicle's position can be measured using a Global Positioning System (GPS) sensor, while the exteroceptive sensor can be a Light Detection and Ranging (LiDAR) sensor and/or stereo cameras. The exteroceptive sensor provides a point cloud representation of the environment surrounding the vehicle at each time step. We assume that the sensor measurements are evenly spaced in all directions on a unit sphere centered around the vehicle's position. These sensor measurements yield a non-convex region representing the obstacle-free area around the vehicle. However, for the sake of simplicity and efficiency, we aim to obtain a convex polytopic under-approximation of this obstacle-free region. In the literature, there have been numerous studies on the inner approximation of non-convex regions using convex sets. Various optimization-based approaches have been proposed (e.g., [2, 6]), where the positions of known obstacles are utilized to obtain the largest convex set within the free area. However, these meth-

ods typically require a priori knowledge of the environment, such as a set of convex polytopes or a map, and do not guarantee that the current vehicle position is included in the obtained set. In contrast, we propose an algorithm that relies solely on local sensor measurements and does not require a preexisting map of the environment. Given a set of sensor measurements, we begin by constructing the smallest convex polytope defined by a predetermined number of vertices centered at the vehicle's position and within the measurements. We then iteratively expand these vertices until they reach the sensor's maximum detection range or until another vertex's expansion includes a sensor measurement.

5.2 Graph-Based Exploration and Mapping

The proposed solution entails the addition of a further hierarchical layer above the MPC controller proposed in the previous section. The choice of this structure is motivated by the difference in complexity and speed requirements of the involved algorithms. The main idea is to build a reachability graph that describes the environment, which can be used seamlessly for both navigation and exploration tasks. Nodes and collision-free arcs are acquired from the convex polytope and used to update the graph together with an exploration gain representing the expected amount of information gained by reaching that node. Then, a receding horizon navigation logic selects the next target node to be provided at the lower level controller. See Fig. 3 (left) for an example of the obtained graph. The navigation approach in G-BEAM is an event-based receding horizon one with similarities with MPC: each time the current reference node is reached, the graph is updated and a new temporary target is computed by planning a path in the graph culminating at a target node. Thus, comparing the approach with an MPC scheme, we exploit the graph-based model of the environment, to find the path on the graph that minimizes a user-defined cost-function. Indeed, the target node is computed by maximizing a reward function (minimizing the negative reward function) that trades off between environment exploration and reaching the external target $\bar{\boldsymbol{r}}$ (exploitation), if provided.

6 Applications to Autonomous Navigation Problems

We evaluated the performances of the proposed approaches on realworld and simulated problems.

The first practical application considers the navigation out-of-the-lab of a drone prototype built with off-the-shelf components (see [13] for further details). The drone, shown in Fig. 3 (in the middle), is equipped with a 2-dimensional LiDAR (Light Detection and Ranging) sensor able to detect the surrounding environment and it is controlled at a lower level by a commercial flight controller. The higher-level control strategies presented in Sects. 3, 4 and 5 are implemented on a low-cost onboard

Fig. 3 In the middle the DJI drone used for the experiments. On the right the experimental test of the mt-MPC approach. Closed-loop trajectory obtained with mt-MPC (blue line with '◊'). Safe set $\mathcal{X}(k)$ at different time step k in red. On the left the reachability graph obtained with G-BEAM without providing an external target

computer and adopted to safely navigate and explore the environment. The experiments shown that the approach effectively allows to reach a target or explore the environment while avoiding a-priori unknown obstacles and that the approaches can be efficiently implemented in real-time on low cost hardware.

The proposed framework has been applied also to multi-agents problems. In [4], has been exploited to navigate a System of TEthered Multicopters (STEM) in a partially known environment. The system consists of multiple aircraft interconnected by power supply tethers (see [7] for further details on the system) and equipped with various exteroceptive sensors for detecting the surroundings. Initially, an optimal mission planner is developed to determine the position references for the system's configuration in the nominal environment. Subsequently, a reactive path planner utilizes the sensor readings and time-varying state constraints derived from the sensor measurements, following the presented framework. This path planner ensures the system navigates to the desired configuration, avoiding collisions with obstacles, even if they differ from the nominal ones. In [14] the multi-trajectory formulation has been extended to consider multi-agent systems with time-varying network topology. Every agent is equipped with a communication device, enabling bidirectional communication with other agents when their communication sets overlap. The communication sets depend on the system's position, resulting in a time-varying communication topology. Re-configuring the controller in control system networks, involving agents joining or leaving the network, remains an open challenge. To address this, the multi-trajectory MPC scheme is utilized to guarantee automatic plug-and-play operations that cannot be denied.

7 Conclusions

This brief provided an overview of a framework utilizing MPC for the safe navigation of autonomous vehicles towards a target reference. The theoretical results

presented demonstrate that MPC serves as a suitable tool to accomplish this task, striking a balance between exploiting the vehicle's capabilities and ensuring constraint satisfaction for enhanced safety. The control strategies discussed in Sects. 3, 4 and 5 have demonstrated their remarkable effectiveness in real-world experiments. Future research directions include expanding the multi-trajectory MPC approach to diverse scenarios, such as reconfigurable systems, and incorporating additional learning components into the problem. Furthermore, exploring its application in economic contexts could be intriguing, where unpredictable time-varying constraints can be viewed as evolving resources. In conclusion, this work highlighted the significance of studying time-varying constraints to embrace MPC as a promising approach for the constrained navigation of autonomous vehicles holds great potential for advancing the future of transportation.

References

1. Bagloee SA, Tavana M, Asadi M, Oliver T (2016) Autonomous vehicles: challenges, opportunities, and future implications for transportation policies. J Mod Transp 24:284–303
2. Bemporad A, Rocchi C (2011) Decentralized linear time-varying model predictive control of a formation of unmanned aerial vehicles. In: 2011 50th IEEE conference on decision and control and European control conference. IEEE, pp 7488–7493
3. Bolognini M, Izzo G, Marchisotti D, Fagiano L, Limongelli MP, Zappa E (2022) Vision-based modal analysis of built environment structures with multiple drones. Autom Constr 143:104550
4. Bolognini M, Saccani D, Cirillo F, Fagiano L (2022) Autonomous navigation of interconnected tethered drones in a partially known environment with obstacles. In: 2022 IEEE 61st conference on decision and control (CDC). IEEE, pp 3315–3320
5. Claussmann L, Revilloud M, Gruyer D, Glaser S (2019) A review of motion planning for highway autonomous driving. IEEE Trans Intell Transp Syst 21(5):1826–1848
6. Deits R, Tedrake R (2015) Computing large convex regions of obstacle-free space through semidefinite programming. In: Algorithmic foundations of robotics XI. Springer, pp 109–124
7. Fagiano L (2017) Systems of tethered multicopters: modeling and control design. IFAC-PapersOnLine 50(1):4610–4615
8. Fagiano L, Teel AR (2013) Generalized terminal state constraint for model predictive control. Automatica 49(9):2622–2631
9. Limon D, Ferramosca A, Alvarado I, Alamo T (2018) Nonlinear MPC for tracking piece-wise constant reference signals. IEEE Trans Autom Control 63(11):3735–3750
10. Liu Z, Stursberg O (2019) Recursive feasibility and stability of mpc with time-varying and uncertain state constraints. In: 2019 18th European control conference (ECC). IEEE, pp 1766–1771
11. Mayne DQ, Rawlings JB, Rao CV, Scokaert PO (2000) Constrained model predictive control: stability and optimality. Automatica 36(6):789–814
12. Murphy RR, Kravitz J, Stover SL, Shoureshi R (2009) Mobile robots in mine rescue and recovery. IEEE Robot Autom Mag 16(2):91–103
13. Saccani D, Cecchin L, Fagiano L (2022) Multitrajectory model predictive control for safe uav navigation in an unknown environment. IEEE Trans Control Syst Technol (2022)
14. Saccani D, Fagiano L, Zeilinger MN, Carron A (2023) Model predictive control for multi-agent systems under limited communication and time-varying network topology. arXiv:2304.01649

Telecommunications

Cooperative Processing and Learning Methods for High-Resolution Environmental Perception

Luca Barbieri

Abstract Cooperative positioning approaches enable interconnected agents to share information across the network, thereby improving accuracy, reliability, and safety compared to conventional single-agent localization methods. This chapter presents novel cooperative localization and learning strategies to provide precise positioning in harsh propagating environments as well as reliable environmental mapping for highly-dynamic scenarios. At first, positioning and environmental perception tasks are addressed separately. More specifically, augmentation strategies are proposed to improve positioning accuracy in complex environments by exploiting prior information on the tracking environment. Next, decentralized Federated Learning (FL) policies are developed to obtain accurate environmental sensing at the agents in a privacy-preserving and communication-efficient manner. Then, the localization and environmental perception problems are solved via a unified solution by designing a data-driven cooperative strategy where agents collaborate to enhance their environmental awareness and their positioning capabilities concurrently. Finally, Bayesian FL tools are developed so that the agents are able to incorporate uncertainty in their decisions and consequently provide trustworthy environmental perception. The achieved results show how the proposed techniques can enable accurate, communication-efficient, and trustworthy localization and sensing.

1 Introduction

Next-generation wireless networks will facilitate the development of connected automated industrial systems by exploiting disruptive technologies, such as THz frequencies, Reconfigurable Intelligent Surfaces (RIS) as well as Integrated Sensing and Communication (ISAC) systems [15]. Thanks to these new technological developments, distributed computing tools will replace energy-hungry cloud processing functions by pushing the intelligence directly into edge devices or agents [31]. The formation of self-sustained, cooperative networks is beneficial for advanced mobil-

L. Barbieri (✉)
Politecnico di Milano, Milan, Italy
e-mail: luca1.barbieri@polimi.it

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_10

ity services as they allow merging (partial) information acquired from spatially-distributed agents and consequently improve the sensing/localization capabilities of the agents themselves. Machine Learning (ML) tools are also paramount in these contexts to extract useful relationships from the data collected by the agents, allowing further positioning/sensing performance enhancements.

Driven by all these key elements, this chapter presents novel cooperative localization and sensing strategies for future mobility systems comprising indoor/outdoor scenarios characterized by complex propagating conditions and/or highly dynamic interactions among the agents. These conditions may arise due to harsh environments in which the agents are deployed (e.g., industrial facilities) or due to the agents' mobility (e.g., vehicular contexts). To solve such challenges, wireless networks are exploited to enable cooperative schemes where networked devices collaborate in sharing information with the goal of estimating their position, perceiving the surrounding environment, or both. Besides, data-driven approaches are tightly integrated into the proposed algorithms to enable efficient and trustworthy sensing/positioning functionalities.

The chapter initially addresses the problem of high-precision localization and environmental perception as two separate tasks. Individual solutions are proposed for both tasks that aim at augmenting the positioning/sensing performance by exploiting side information from the surrounding environment. Next, a combined approach is proposed where the localization of the mobile agents is integrated with the perception of the environment at each agent by means of a cooperative approach. Finally, careful attention is given so that the proposed methods provide not only accurate positioning/sensing functionalities but also trustworthy responses. This aspect is treated in the last part of the chapter where a trustworthy, yet accurate perception system is developed.

The rest of the chapter presents in more detail the main research activities included in my Ph.D. thesis [3] and in the publications [2, 4–11, 32]. Further results of the research conducted during the Ph.D. can be found in [13, 14, 20, 21, 29, 30], which are omitted here as not fitting the main scope of the Ph.D. thesis. The organization of the chapter is as follows. Sect. 2 addresses the problem of high-accuracy localization in complex propagating environments, while Sect. 3 focuses on providing a solution for accurate environmental perception integrating distributed learning tools. Then, localization and sensing tasks are combined via a unified cooperative approach in Sect. 4, whereas Sect. 5 studies how to enhance not only the accuracy but also the trustworthiness of perception systems. Finally, Sect. 6 draws some conclusions.

2 Localization of Mobile Agents in Complex Environments

Accurate location information has become a fundamental requirement in many of today's services. Positioning estimates can be typically obtained by harnessing the radio signals exchanged over a wireless network. However, complex environments, such as industrial plants, pose a major problem in this respect, as they heavily affect

the quality of the location-dependent information that can be extracted from wireless signals. Therefore, such environments call for augmentation/mitigation strategies able to amend Non Line of Sight (NLOS)-corrupted radio signals and profitably use them to enhance localization accuracy.

Throughout the years, several approaches have been developed to mitigate the impact of complex propagating conditions and improve positioning performance. Statistical characterization of the Channel Impulse Response (CIR) can be used to detect NLOS propagation and consequently correct localization measurements [19]. Another popular approach is to rely on Bayesian tracking filters to integrate information on the propagation environment [27]. More recently, ML methods have been also proposed to provide accurate localization in harsh environments [26]. They mostly rely on supervised techniques to augment the localization accuracy under NLOS propagation.

2.1 A Bayesian Tracking Framework for NLOS Compensation

To address the aforementioned challenges, the Ph.D. thesis proposes a novel NLOS mitigation approach that incorporates multiple (hybrid) localization measurements, namely Time Difference of Arrivals (TDoAs) and Angle of Arrivals (AoAs), as well as an efficient tracking algorithm to estimate the position of the agents accurately. The technique is specifically formulated for Ultra WideBand (UWB) systems. Still, it is general enough to be applied to any wide bandwidth and multi-antenna system, such as the ones foreseen for Beyond 5G networks.

More specifically, the method aims at embedding the propagation information of the environment in which the localization task has to be carried out while mitigating the NLOS and multipath impairments. It does so by jointly tracking the agents' position and the Line of Sight (LOS)/NLOS conditions, referred to as sight conditions, experienced at the reference stations or Access Points (APs). The sight conditions evolution is modeled as a first-order Markov chain with transition probabilities describing the change of state from LOS to NLOS and from LOS to LOS, and calibrated according to the available layout information of the tracking area. Based on the current estimated value of the APs sight variables, the measurements are statistically described to take into account the actual propagating conditions and compensate for the measurements affected by NLOS. The developed statistical framework is integrated with a Jump Markov System (JMS) that enables the description of the relationship between sight conditions and the position of the agents, allowing the overall problem to be solved via a Bayesian filtering approach. In this respect, a Particle Filter (PF) implementation is considered to track the joint position-sight state efficiently across time. The overall methodology is summarized in Fig. 1.

The proposed NLOS compensation tool has been evaluated considering real raw UWB data collected inside a fully-functional industrial facility and compared against

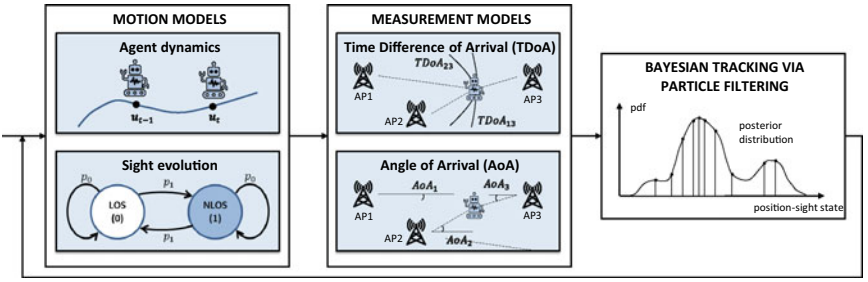


Fig. 1 Bayesian NLOS mitigation methodology: the aim is to jointly track the agents' positions as well as the visibility conditions of the APs so that highly corrupted localization measurements can be compensated and used for improving the positioning performances

a conventional Bayesian filter that does not compensate for the NLOS as well as state-of-the-art NLOS mitigation methods [5]. Experimental results showed the superiority of the proposal in providing more accurate localization compared to all other methods considered, especially in areas highly affected by non-ideal propagating conditions. Besides, the proper integration of hybrid positioning measurements is beneficial for further improving the position estimate of the agents. We refer the interested reader to [2, 5] for more details about the methodology and additional analyses.

3 Federated Learning for Enhanced Perception

Future mobility systems will require highly-accurate localization and environmental awareness capabilities to detect possible hazardous situations and act accordingly. This section moves in this direction and complements the previous one by proposing a sensing system for accurate environmental perception in high-mobility scenarios, i.e., road vehicles. A wireless network connecting the vehicles is exploited to implement cooperative perception strategies, where a set of networked vehicles equipped with imaging sensors aim at obtaining enhanced perception capabilities.

Conventional cooperative sensing methods rely on data-sharing procedures where raw or partially processed data are exchanged over the network [28]. However, the introduction of regulations restricting the access and distribution of data among multiple parties makes such techniques unfeasible. On the other hand, Federated Learning (FL) procedures can be used to learn a ML model able to provide the same sensing functionalities as standard cooperative perception approaches. FL [32] resorts to the exchange of locally trained instances of a shared ML model without requiring any data exchange. Even though FL represents a promising privacy-preserving solution, communication-efficient designs are required to make FL platforms more sustainable, especially when large models need to be exchanged over the network.

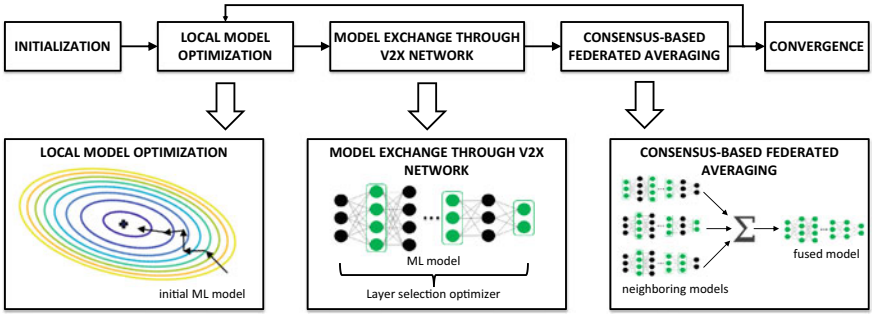


Fig. 2 Communication-efficient decentralized FL policy: vehicles exchange fractions of their local models and implement an average consensus policy for fusing the model updates received from their neighbors

3.1 Communication-Efficient FL Policy

This section discusses how to improve the communication efficiency of decentralized learning policies so as to obtain more sustainable FL-based perception systems without penalizing sensing performances. A communication-efficient design is introduced where the vehicles participating in the FL process are able to intelligently select a subset of the parameters of the ML model to be exchanged via Vehicle-to-Everything (V2X) networking.

As depicted in Fig. 2, the proposed communication-efficient strategy tries to reduce the communication overhead by choosing the layers of the Neural Network (NN) according to the local data quality observed at the vehicles. We develop a *layer selection optimizer* that dynamically selects the layer parameters according to the normalized squared gradients observed during the local optimization step performed by the vehicles. The gradients are firstly sorted in a descending manner and only the layers associated with the strongest gradient magnitudes are selected and propagated to the neighbors. Intuitively, higher magnitude gradients convey more informative updates; therefore, the corresponding layers should be transmitted more frequently. Additionally, a randomized policy is integrated with the layer selection optimizer that chooses the layers in a independent and identically distributed (i.i.d.) fashion. Besides providing a more fair layer exchange process during the FL process, the combination of gradient-based and randomized selection strategies has been found to provide higher-quality models with improved generalization abilities [9].

The performances of the communication-efficient design have been evaluated considering a challenging automotive vertical, where vehicles are required to optimize a large NN for accurately classifying road users/objects present in the driving environment via lidar point clouds [6]. The assessment was focused on characterizing the impact of the layer selection process on the final accuracy of the trained models while also comparing the achieved results against conventional centralized and decentralized learning strategies.

Numerical results showed that the developed design provides substantial communication overhead reduction (up to 80%) while approaching the performances of conventional (uncompressed) FL tools. Additionally, balancing gradient-based and randomized selection policies is beneficial for heavily limiting communication resource consumption without introducing accuracy penalties. Interestingly, layers possessing the least number of trainable parameters should be selected more frequently as they heavily impact the learned models' quality [9]. The interested reader can find additional layer selection strategies and further numerical evaluations in [6–9, 32].

4 Cooperative Localization and Sensing in Connected Vehicle Scenarios

The previous sections treated localization and sensing as two separate tasks. However, in next-generation communication systems, such as 6G, these two functionalities are expected to be integrated into the same infrastructure so as to exploit even more their synergy. In line with this trend, this section introduces a more complete system compared to Sects. 2 and 3 integrating cooperative localization and sensing into a unified solution where the goal is to augment the Global Navigation Satellite System (GNSS) performances under complex urban environments.

Perception sensors, particularly Lidar devices, have been increasingly adopted in mobility systems to provide detailed and rich information on the surrounding environment [24]. Cooperative methods have also been studied in such systems [22, 33] to improve environmental awareness by fusing information across multiple interconnected agents. However, considering the sheer amount of data generated by these sensors, conventional signal processing tools may be inadequate as they might introduce large delays. On the other hand, data-driven methods enable the efficient processing of large data volumes while also extracting useful information beneficial for jointly carrying out positioning and sensing tasks [17].

4.1 Data-Driven Joint Cooperative Localization and Perception

Based on the above discussion, this section develops a data-driven cooperative positioning and environmental sensing solution to increase the vehicles' localization performance compared to conventional GNSS-based systems. The proposal's main idea is to make the vehicles align their (limited) view of the surrounding environment with other vehicles to improve the detection of the objects along the road and implicitly refine the vehicle positioning as well.

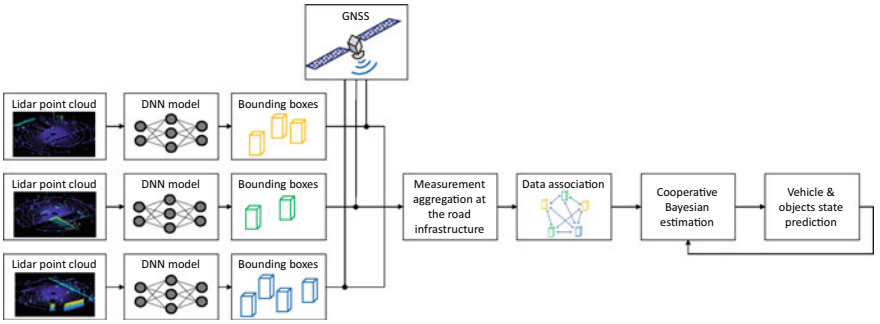


Fig. 3 Data-driven cooperative localization and sensing: vehicles employ a ML model to localize static objects in the driving environment via lidar sensors. The individual detections are collected at a centralized infrastructure which is able to refine both the objects and the vehicle positions

The developed method extends the Implicit Cooperative Positioning (ICP) framework introduced in [12] by integrating a realistic lidar sensing platform. In particular, a Deep Neural Network (DNN)-assisted sensing framework is designed to recognize and localize road objects (e.g., poles) from lidar sensors available at moving vehicles. The DNN-based detector learns how to recognize static objects as their use has been acknowledged to provide better benefits in ICP [12]. In particular, the detection process focuses on recognizing poles since they are largely present in the driving environment, easily recognizable through the lidar point cloud, and do not require new installations and/or calibrations. Once the vehicles have estimated the position of the poles present in the driving environment, the aggregated information is collected by a centralized road infrastructure which is tasked to cooperatively localize both objects and vehicles employing a Bayesian tracking tool. By doing so, multiple poles estimated at different vehicles can be coherently fused and exploited to improve the vehicles' positioning accuracy. A block scheme summarizing the main operations required to run the developed approach is shown in Fig. 3.

The evaluation of the proposed approach considers a highly-realistic vehicular scenario simulated using the CARLA software [16], an advanced, high-fidelity autonomous driving simulator that allows defining complex driving conditions as well as generating accurate sensors readings. Numerical results have shown that the developed cooperative localization and sensing approach outperforms a conventional GNSS-based tracking tool while providing similar results to a cooperative oracle system where vehicles always detect all possible poles within the lidar sensing range regardless of actual visibility conditions [4]. The interested reader can look at [4, 11] for the complete description of the methodology.

5 Bayesian Federated Learning for Trustworthy Environmental Perception

Throughout the years, ML tools have been demonstrated to provide excellent performances in solving complex tasks, particularly in big-data regimes where large collections of data are available. However, when data are scarce or limited, NNs trained under the conventional, frequentist, learning paradigm, tend to provide overconfident and often incorrect predictions while also suffering from overfitting. This is further exacerbated when considering FL-based sensing platforms as vehicles may converge to the same unreliable ML model, thereby posing major safety concerns.

Most of the solutions proposed to address the aforementioned challenges rely on Bayesian learning strategies, where the goal is to learn the posterior distribution of the ML model parameters in place of finding a single model parameters' value that fits well the training data [23]. Some Bayesian FL systems have been recently proposed based on the Partitioned Variational Inference (PVI) framework developed in [1] or on Markov Chain Monte Carlo (MCMC)-based sampling schemes [18]. Still, implementations of Bayesian FL systems over cooperative wireless networks typically assume noiseless communications and, thus, are hardly applicable in real-world scenarios.

5.1 Channel-Driven Bayesian FL Strategy

This section presents a fully decentralized Bayesian FL framework for trustworthy environmental perception in vehicular networks. Compared to the previously-analyzed FL system introduced in Sect. 3, here, the proposal extends the frequentist tools to embrace Bayesian learning strategies. The aim is to obtain ML models that concurrently provide accurate perception capabilities and reliably quantify the uncertainty associated with their predictions. Besides, the proposed method exploits the noise introduced by the propagation in a novel fashion to wirelessly implement the Bayesian FL process.

To obtain an approximation of the global posterior distribution shared by all vehicles, a Bayesian FL system is proposed extending the Decentralized Stochastic Gradient Langevin Dynamics (DSGLD) [18] scheme. The proposal builds on the concept of *channel-driven sampling* [25], whereby the Bayesian FL strategy is implemented over wireless networks, and the channel noise introduced by the propagation is repurposed for obtaining the final posterior distribution. Indeed, under DSGLD, vehicles update their local posterior samples using Stochastic Gradient Descent (SGD), combine the samples received from their neighbors using a consensus strategy and, finally, add Gaussian noise to obtain a new sample approximating the (global) posterior distribution. Therefore, channel-driven sampling allows each vehicle to directly use the channel noise for the sampling process of DSGLD. An over-the-air computing policy is also proposed to wirelessly aggregate the samples

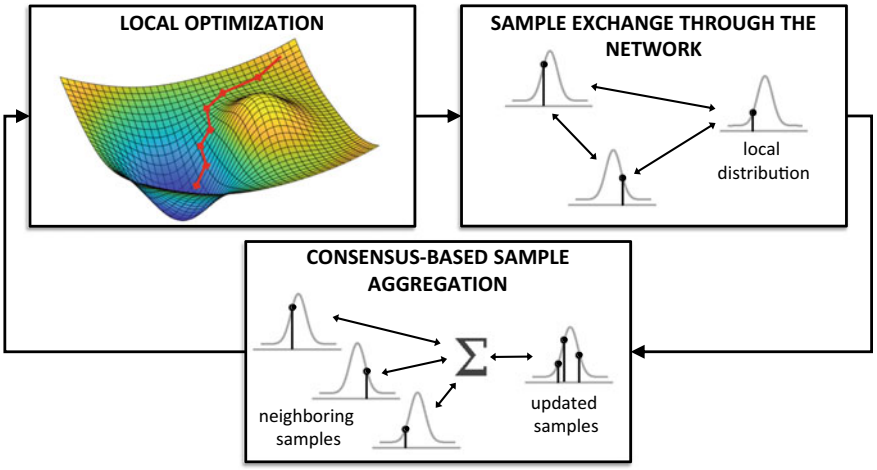


Fig. 4 Decentralized Bayesian FL system: vehicles exchange samples drawn from their local posterior and fuse those received from their neighbors following a consensus-based aggregation strategy

produced at the vehicles in an analog fashion so as to reduce the training latency associated with the cooperative learning process. The block scheme summarizing the proposed strategy is depicted in Fig. 4. For more details on the methodology, the interested reader can refer to [10].

The developed Bayesian FL tool is evaluated considering the same cooperative sensing task as in Sect. 3 and is compared against a standard frequentist FL tool. Numerical results show that the proposed strategy provides highly-accurate perception models that reliably quantify the uncertainty of their predictions, while the conventional FL strategy lacks such uncertainty quantification and consequently provides unreliable ML models [10].

6 Concluding Remarks

This chapter presented several methodological advancements aimed at enhancing localization accuracy, environmental awareness, or both in multi-agent networks. Cooperative systems underpin the proposed algorithms in order to provide enhanced environmental awareness or augmented localization performances, thanks to collaborative functions implemented by interconnected agents, devices, or vehicles. ML methods are also instrumental in achieving highly-accurate results especially when conventional methods fail to provide any reasonable outcome. Shifting from standard signal processing tools to ML strategies is also often required, if not mandatory, as model-driven approaches may be too complex to implement or require too much time

to be formulated. Moreover, they enable efficient and timely processing of massive amounts of data that may also be required for latency-critical services.

The techniques discussed in this chapter represent fundamental building blocks that can be combined for developing a larger, more refined localization and sensing system, where accuracy is not the only performance metric to be considered. We believe the proposed framework is a starting point that could be extended to embrace novel technologies, ad-hoc implementations, or better integration possibly looking at future technological developments.

References

1. Ashman M, Bui TD, Nguyen CV, et al (2022) Partitioned variational inference: a framework for probabilistic federated learning. CoRR
2. Barbieri L, Brambilla M, Pitic R, Trabattoni A, Mervic S, Nicoli M (2020) UWB real-time location systems for smart factory: augmentation methods and experiments. In: 2020 IEEE 31st annual international symposium on personal, indoor and mobile radio communications, pp 1–7
3. Barbieri L (2023) Cooperative processing and learning methods for high-resolution environmental perception. Ph.D. thesis, Politecnico di Milano
4. Barbieri L, Brambilla M, Nicoli M (2023) Deep neural networks for cooperative lidar localization in vehicular networks. In: 2023 IEEE international conference on communications (ICC), pp 1–6
5. Barbieri L, Brambilla M, Trabattoni A, Mervic S, Nicoli M (2021) UWB localization in a smart factory: augmentation methods and experimental assessment. IEEE Trans Instrum Meas 70:1–18
6. Barbieri L, Savazzi S, Brambilla M, Nicoli M (2021) Decentralized federated learning for extended sensing in 6G connected vehicles. Veh Commun 100396
7. Barbieri L, Savazzi S, Nicoli M (2021) Decentralized federated learning for road user classification in enhanced V2X networks. In: 2021 IEEE international conference on communications workshops (ICC Workshops), pp 1–6
8. Barbieri L, Savazzi S, Nicoli M (2022) Communication-efficient distributed learning in V2X networks: Parameter selection and quantization. In: 2022 IEEE global communications conference (GLOBECOM), pp 1–6
9. Barbieri L, Savazzi S, Nicoli M (2023) A layer selection optimizer for communication-efficient decentralized federated deep learning. IEEE Access 11:22155–22173
10. Barbieri L, Simeone O, Nicoli M (2023) Channel-driven decentralized Bayesian federated learning for trustworthy decision making in D2D networks. In: ICASSP 2023–2023 IEEE international conference on acoustics, speech and signal processing (ICASSP), pp 1–5
11. Barbieri L, Tedeschini Camajori B, Brambilla M, Nicoli M (2023) Implicit vehicle positioning with cooperative lidar sensing. In: ICASSP 2023–2023 IEEE international conference on acoustics, speech and signal processing (ICASSP), pp 1–5
12. Brambilla M, Nicoli M, Soatti G, Deflorio F (2020) Augmenting vehicle localization by cooperative sensing of the driving environment: insight on data association in urban traffic scenarios. IEEE Trans Intell Transp Syst 21(4):1646–1663
13. Camajori Tedeschini B, Brambilla M, Barbieri L, Nicoli M (2022) Addressing data association by message passing over graph neural networks. In: 2022 25th international conference on information fusion (FUSION), pp 01–07
14. Camajori Tedeschini B, Savazzi S, Stoklasa R, Barbieri L, Stathopoulos I, Nicoli M, Serio L (2022) Decentralized federated learning for healthcare networks: A case study on tumor segmentation. IEEE Access 10:8693–8708

15. De Lima C, Belot D, Berkvens R, Bourdoux A, Dardari D, Guillaud M et al (2021) Convergent communication, sensing and localization in 6G systems: An overview of technologies, opportunities and challenges. IEEE Access 9:26902–26925
16. Dosovitskiy A, Ros G, Codevilla F, Lopez A, Koltun V (2017) CARLA: an open urban driving simulator. In: Conference on robot learning. PMLR, pp 1–16
17. Fayyad J, Jaradat MA, Gruyer D, Najjaran H (2020) Deep learning sensor fusion for autonomous vehicle perception and localization: a review. Sensors 20(15)
18. Garbazbalaban M, Gao X, Hu Y, Zhu L (2021) Decentralized stochastic gradient Langevin dynamics and Hamiltonian monte carlo. J Mach Learn Res 22(239):1–69
19. Guvenc I, Chong CC, Watanabe F: NLOS identification and mitigation for UWB localization systems. In: 2007 IEEE wireless communications and networking conference, pp 1571–1576 (2007)
20. Haghshenas M, D'Adda M, Linsalata F, Barbieri L, Nicoli M, Magarini M (2021) On the performance of zero-forcing beamforming in a real I2V scenario at millimiter wave. In: 2021 International Balkan conference on communications and networking (BalkanCom), pp 56–60
21. Haghshenas M, Linsalata F, Barbieri L, Brambilla M, Nicoli M, Magarini M (2022) Analysis of spatial scheduling in downlink vehicular communications: sub-6 GHz vs mmWave. ITU J Futur Evol Technol 3:523–534
22. Héry E, Xu P, Bonnifait P (2021) Consistent decentralized cooperative localization for autonomous vehicles using LiDAR, GNSS, and HD maps. J Field Robot 38(4):552–571
23. Jospin LV, Laga H, Boussaid F et al (2022) Hands-on Bayesian neural networks-a tutorial for deep learning users. IEEE Comput Intell Mag 17(2):29–48
24. Li Y, Ibanez-Guzman J (2020) Lidar for autonomous driving: The principles, challenges, and trends for automotive lidar and perception systems. IEEE Signal Process Mag 37(4):50–61
25. Liu D, Simeone O (2022) Wireless federated Langevin monte carlo: repurposing channel noise for Bayesian sampling and privacy. IEEE Trans Wirel Commun 1–1
26. Maranó S, Gifford WM, Wymeersch H, Win MZ (2010) NLOS identification and mitigation for localization based on UWB experimental data. IEEE J Sel Areas Commun 28(7):1026–1035
27. Nicoli M, Morelli C, Rampa V (2008) A jump Markov particle filter for localization of moving terminals in multipath indoor scenarios. IEEE Trans Signal Process 56(8):3801–3809
28. Patwari N, Ash J, Kyperountas S, Hero A, Moses R, Correal N (2005) Locating the nodes: cooperative localization in wireless sensor networks. IEEE Signal Process Mag 22(4):54–69
29. Piavanini M, Barbieri L, Brambilla M, Cerutti M, Ercoli S, Agili A, Nicoli M (2022) A calibration method for antenna delay estimation and anchor self-localization in UWB systems. In: 2022 IEEE international workshop on metrology for industry 4.0 & IoT (MetroInd4.0&IoT), pp 173–177
30. Piavanini M, Barbieri L, Brambilla M, Cerutti M, Ercoli S, Agili A, Nicoli M (2022) A self-calibrating localization solution for sport applications with UWB technology. Sensors 22(23)
31. Saad W, Bennis M, Chen M (2020) A vision of 6G wireless systems: applications, trends, technologies, and open research problems. IEEE Netw 34(3):134–142
32. Savazzi S, Nicoli M, Bennis M, Kianoush S, Barbieri L (2021) Opportunities of federated learning in connected, cooperative, and automated industrial systems. IEEE Commun Mag 59(2):16–21
33. Zhang Y, Chen L, XuanYuan Z, Tian W (2020) Three-dimensional cooperative mapping for connected and automated vehicles. IEEE Trans Ind Electron 67(8):6649–6658

Synthesis of Filters and Filtering Antennas for Micro and Millimeter Waves Applications

Steven Caicedo Mejillones, Matteo Oldoni, and Michele D'Amico

Abstract Filters and antennas are the closest building blocks to the air interface in modern wireless communications systems. Filters allow the transmission of signals in a target frequency range and attenuate those that operate in the unwanted range. Antennas help radiate signals within their operating range. This article is a summary of the author's doctoral thesis and focuses on the development of new synthesis-based methods for the design of these two building blocks and the integration between them, in other words, filtennas. The main advantage of the synthesis-based design is that the expected frequency response of the final prototype is approximated in advance. Therefore, the selection of the best solution that satisfies the given requirements can be done through fast but accurate circuit simulations. When the best solution is found, then the actual prototype is designed according to the synthesized circuit.

1 Synthesis-Based Filter Design

This section presents an overview of advanced filter synthesis techniques described in [4–6]. All the techniques described here follows the same principle: they start with the well-known coupling-matrix synthesis, suitable circuit or matrix transformations are then applied to get the required topology.

1.1 Accurate Synthesis of Extracted-Pole Filters

This method relies in the well-known accuracy of the coupling matrix synthesis method. To understand the method, let us work with synthesis of a filter with 20 dB of return loss and transmission zeros placed at $[\infty, -1.889, 1.1512, 1.702]$ rad/s.

This work was supported by the European Union's Horizon 2020 Research Program 5G STEP FWD under Grant Agreement No. 722429.

S. Caicedo Mejillones (✉) · M. Oldoni · M. D'Amico
Politecnico Di Milano, 20133 Milan, Italy
e-mail: stevenkleber.caicedo@polimi.it

S. Caicedo Mejillones
SIAE Microelettronica, 20093 Cologno Monzese, Italy

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_11

Then, the first step is to synthesize a coupling matrix using any of the methods already available in the literature [7]. Then, the circuit is transformed into the arrow form [7] as shown in Fig. 1a (without the blue coupling). After that, each transmission is extracted by means of matrix rotations. The first rotation create a coupling (dashed blue line in Fig. 1a) in such a way that the last three resonators form a triplet that contains the first zero to be extracted. Then, successive rotations push this triplet towards the position where it is required to locate the extracted pole (Fig. 1b and c). After that, a delta-to-star circuit transformation is performed, obtaining the circuit in Fig. 1d. This must be done with each of the transmit zeros, the output of this algorithm is shown in Fig. 1e. Note that the scattering parameters shown in Fig. 1f are preserved throughout the transformation. Figure 2 shows the synthesis of a fully canonical stopband filter, it is evident that the synthesis with the old method [1] produces a response destroyed by round-off errors. Instead, the new method produces a response that matches the ideal equiripple response.

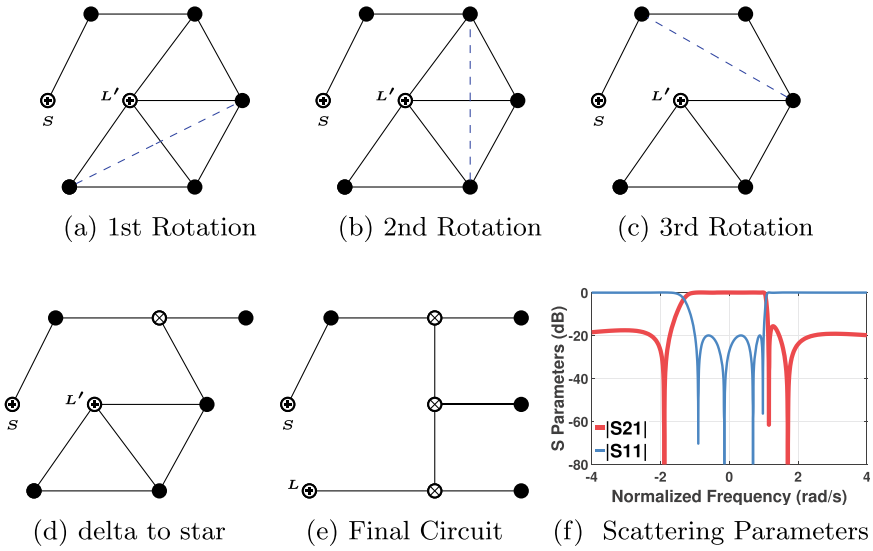


Fig. 1 Synthesis of extracted-pole filters by matrix rotations. Black circles: unit capacitance in parallel to frequency-independent (FI) susceptances (resonating nodes). Circles with (×) are FI susceptances (non-resonant nodes), and with (+) are unit conductance (source, load). Lines are admittance inverters (couplings)

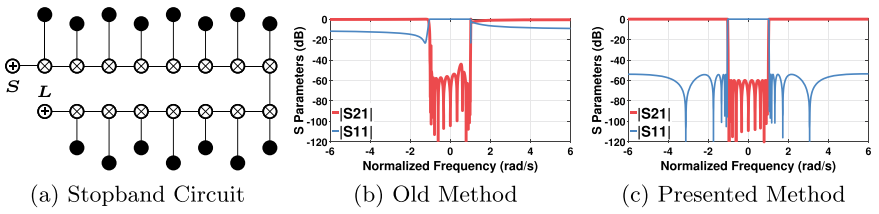
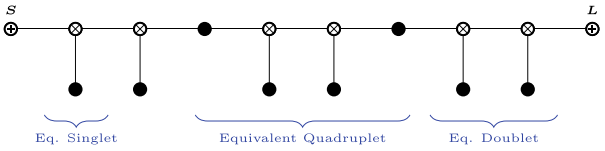


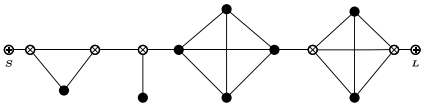
Fig. 2 Fully canonical stop-band filter synthesis using old and new method

1.2 Synthesis of Cascade-Block Filters

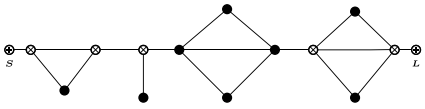
This section presents the overview of the unified analytical method for the synthesis of cascaded *n-tuplets* prototype filters including non-resonating nodes (NRNs) and extracted pole blocks. This method helps to overcome the issues of accuracy, computation time, and uncertainty of optimization methods used to synthesize some topologies, particularly those that include singlets, doublets, or mixed topologies. To understand the method, let us work with synthesis of a filter with 20 dB of return loss, and withe transmission zeros located at -3, 2, ∞, $-0.1 + 0.79i$, $-0.1 - 0.79i$, ∞, 3 and -2 rad/s. Complex transmission zeros helps to group delay equalization. The method begins with previously described extracted-pole synthesis arbitrarily defining the transmission zeros, as shown in Fig. 3a. Then, a filter topology transformation is applied by grouped node blocks to obtain the desired topology, as shown in Fig. 3b. Finally, some matrix rotations are applied to remove redundant couplings obtaining the circuit in 3c. The scattering parameters and group delay shown in Figs. 3d and 3e are preserved throughout the topology transformation. Note that this is a circuit of mixed topology: singlet—extracted-pole—quadruplet—doublet, which was not possible to synthesize analytically with the synthesis techniques available before the publication of the papers [4, 6].



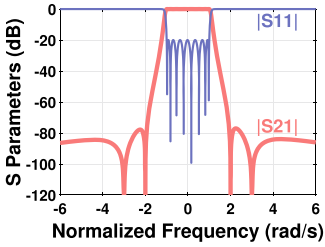
(a) synthesized Extracted-Pole.



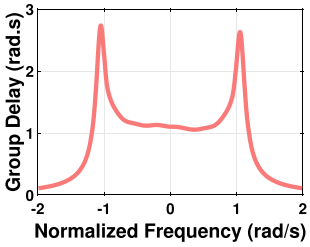
(b) After Matrix Transformation.



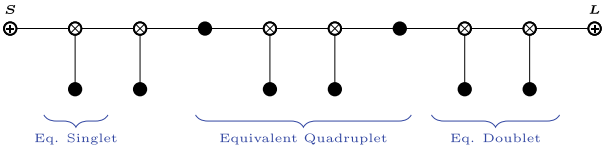
(c) After Matrix Rotation.



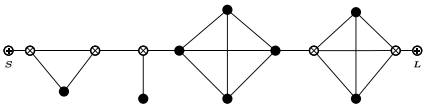
(d) Scattering Parameters.



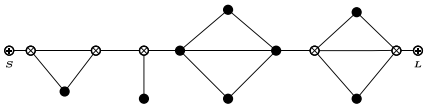
(e) Group Delay.



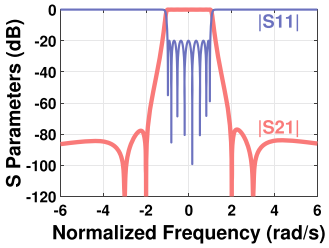
(a) synthesized Extracted-Pole.



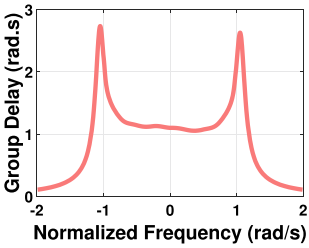
(b) After Matrix Transformation.



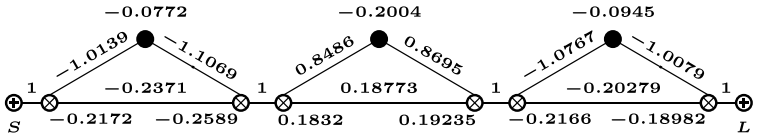
(c) After Matrix Rotation.

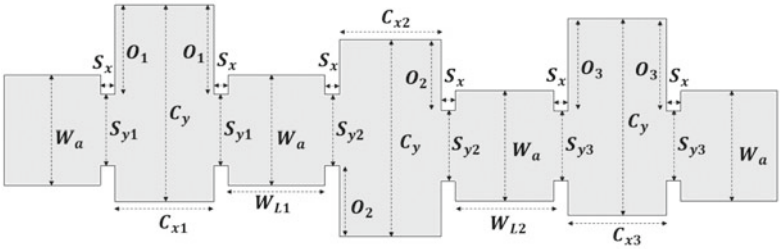


(d) Scattering Parameters.



(e) Group Delay.





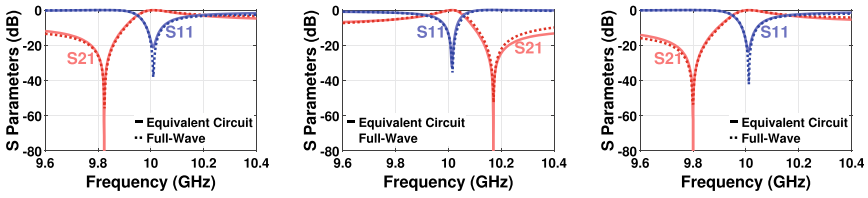


Fig. 3 Synthesized circuit: singlet (x3). Physical Filter: TE_{201} cavity (x3). Unitary inverters are 90° phase shift. Black circles are resonators denormalized with $C = 1/(2\pi\ BW)$, $L = 1/(C * (2\,\pi\ F_c)^2)$. BW is bandwidth. Dimensions in mm: $W_a = 22.86$, $h = 10.16$, $C_y = 40.806$, $C_{x1} = 20.484$, $C_{x2} = 21.047$, $C_{x3} = 20.372$, $S_x = 3$, $S_{y1} = 14.699$, $S_{y2} = 14.654$, $S_{y3} = 14.294$, $O_1 = 18.6255$, $S_2 = 14.723$, $O_3 = 19.298$, $W_{L1} = 20.0738$, $W_{L2} = 20.4047$

1.3 Synthesis-Based Filter Design

This subsection intends to show the flexibility of the previously presented methods to synthesize different topologies that actually implements the same filtering function. These topologies are implemented into waveguide technology.

The filtering function for this example requires a passband from 9.966 to 10.045 GHz with return loss of 16 dB. The required attenuation is 75 dB between 9.8 and 9.82 GHz and more than 30 dB between 10.16 and 10.18 GHz. To fulfill these specifications a fully-canonical function with transmission zeros are placed at 9.823, 10.17 and 9.8 GHz. As state before, there are several synthesizable circuits that could implement this filtering function. One of them is the singlet-singlet-singlet topology. To proceed with the design, first the circuit shown in the top part of Fig. 3 is synthesized. Then each singlet is implemented with a TE_{201} cavity by optimizing it according to the corresponding circuit block. This figure also shows the scattering parameters of the designed TE_{201} cavities together with those of the corresponding equivalent singlets. Once all the blocks are designed, they are joined by 90° waveguide lines as shown in Fig. 3.

A second synthesizable topology for this filtering function is the extracted-pole—singlet extracted-poles shown in the top part of Fig. 4. The singlet is implemented as before with a TE_{201} cavity. The extracted-poles are implemented as stubs since there is no non-resonant nodes in the block. As in the previous topology, all the full-wave blocks are optimized to have the scattering parameters equal to those of the corresponding circuit blocks. Then, they are assembled using waveguides whose electrical length is defined by the circuit.

Finally, Fig. 5 shows instead the scattering parameters of both designs: singlet-singlet-singlet and extracted-pole -singlet- extracted-pole. It can be seen that the full-wave simulations are in good agreement with the circuit simulations. Also both topologies implement the same filtering function.

2 Synthesis-Based Antenna Design

This section present a general overview of the synthesis and design methodology of a series-fed proximity-coupled antenna array. Further details can be found in [3]. This type of antenna was first presented in [11] and then further exploited in different works [10]. These works provide some guidelines to get an initial prototype, but then the design procedure is mainly based on full-wave optimization. This optimization could be time consuming, particularly for relative high number n of elements (e.g. n = 8).

That is why, I have proposed a synthesis-based design method. The method starts with the synthesis of an equivalent circuit of the antenna array. This circuit allows to estimate the antenna return loss and antenna radiation pattern. At this point, a screening can be done to verify which circuit best fit the antenna requirements.

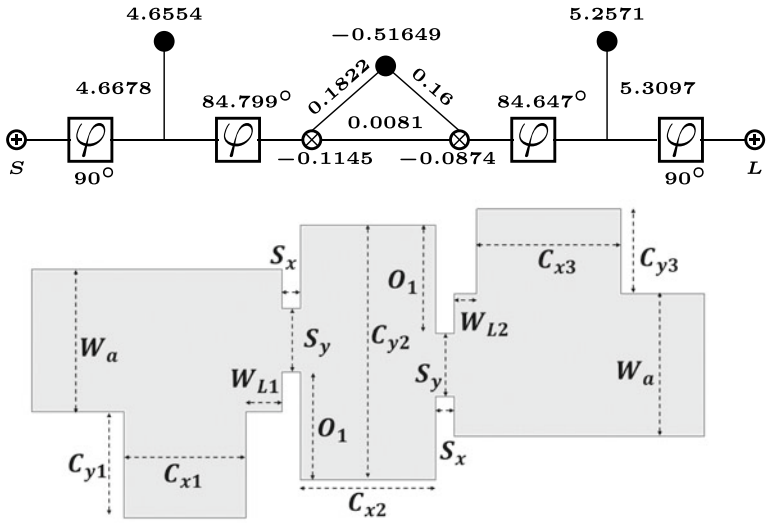


Fig. 4 Synthesized circuit: extracted-pole—singlet—extracted-pole. Physical Filter: stub—TE_{210} cavity—stub. Dimensions in mm: $W_a = 22.86$, $h = 10.16$, $C_{x1} = 19.54$, $C_{y1} = 17$, $C_{x2} = 21.622$, $C_{y2} = 40.806$, $C_{x3} = 23.04$, $C_{y3} = 13.59$, $S_x = 2.98$, $S_y = 10.166$, $O_1 = 17.304$, $W_{L1} = 5.754$, $W_{L2} = 3.603$

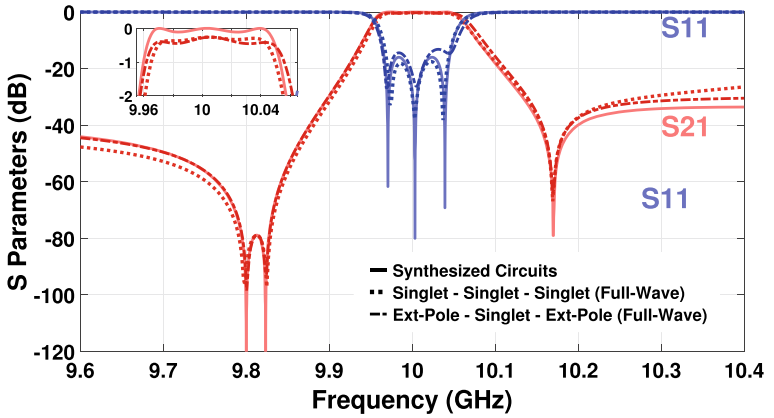


Fig. 5 Scattering parameters of synthesized circuits (solid line) and of the designed physical filters (dotted, dotted-dashed lines)

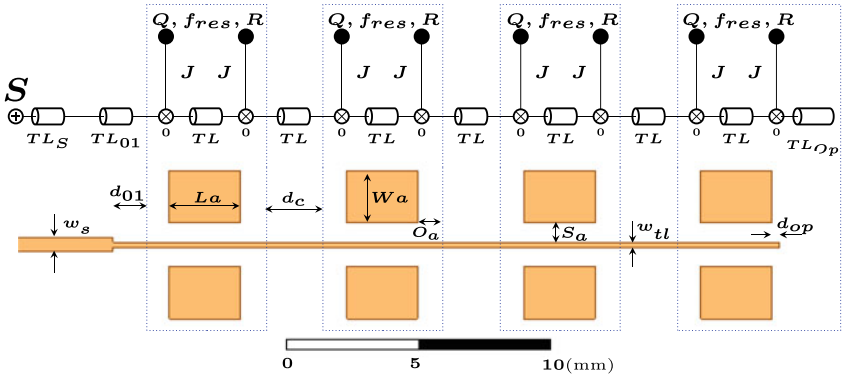
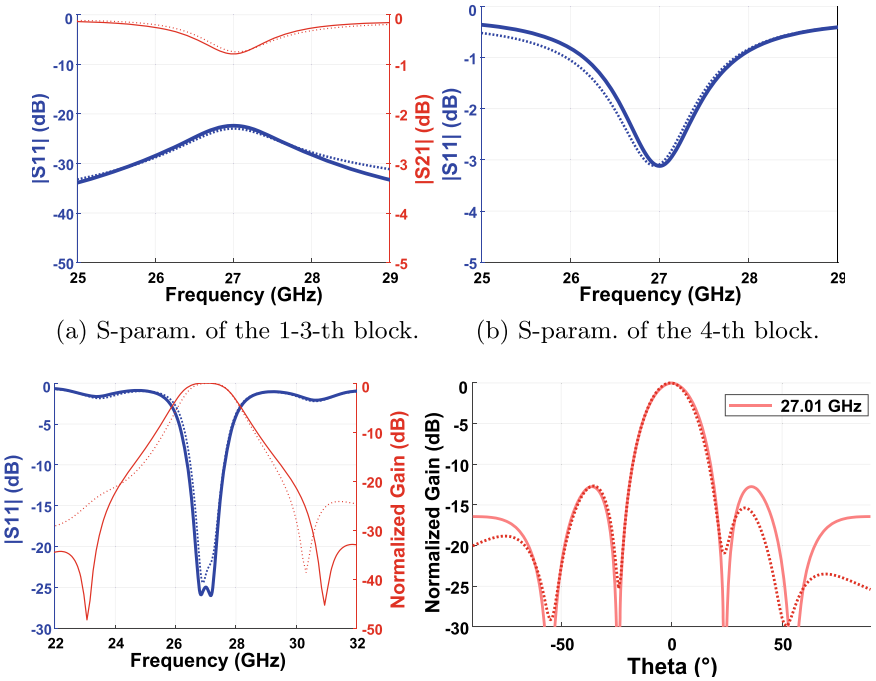


Fig. 6 Equivalences between synthesized circuit and designed antenna array. Extracted-poles: $(Q,\ J,\ f_r,\ R) = (25,\ 0.3536\,\Omega^{-1},\ 27\,\text{GHz}, 0.0086\,\Omega)$. Transmission lines: $TL_{01}/\ TL\ /TL_{Op} = \left(T_\sigma,\ T_\delta,\ Z_{ref},\ f_{ref},\ \theta_{ref}\right) = \left(100\sqrt{\text{Hz}},\ 0.002,\ 78\,\Omega,\ 27\,\text{GHz},\ \frac{\pi}{2}/\pi/0\,\text{rad}\right)$. Source S and TL_S impedance: $Z_S = 50\,\Omega$. PCB substrate: $\epsilon_r = 3.66$, $\tan\delta = 0.004$, Physical antenna dimensions in mm: $h_{cond} = 0.04$, $h_{diel} = 0.254$, $w_s = 0.52$, $w_{tl} = 0.2$,$d_{01} = 1.86$, $L_a = 2.74$, $d_c = 3.46$, $W_a = 2$, $O_a = 0.32$, $S_a = 0.73$, $d_{op} = 0.28$

Once, the best synthesized circuit is chosen, the actual antenna can be designed by using the divide-to-conquer approach. That is, each base element of the antenna array is designed according to the corresponding block of the synthesized circuit. Once all the blocks are designed, they are putting together by means of transmission lines according to the synthesized circuit.

For a better understanding let us analyze the example shown in Fig. 6. This figure shows the circuit synthesized by means of the method described in [3]. Let us call the block highlighted by the dotted rectangles the base antenna/circuit block. For this example, the first three blocks are the same, the last one differs only due to the open-ended line. Therefore, for the actual design of the antenna, the physical parameters L_a, W_a and S_a of the base antenna block are optimized for the best fit in magnitude to the S parameters of the basic circuit block. Then, O_a is optimized for the best fit in phase. Since the last block has an open line, d_{op} is optimized to match the S-parameters of the corresponding circuit block in magnitude and phase. Figure 7a shows the two-port scattering parameters for the first three circuit blocks and those for the optimized antenna blocks. Figure 7b shows instead the one-port scattering parameter for the last circuit/antenna block.

Once all the blocks have been designed, the complete antenna array is built by joining all the antenna blocks by means of suitable microstrip transmission lines. To verify the accuracy of the design, Fig. 7c shows the S_{11} parameter and the broadside antenna gain G of the whole synthesized circuit and those of the designed antenna array. Figure 7d shows instead the antenna array radiation pattern computed using both the circuit and the designed array. All these figures show a very good fit the circuit and full-wave simulations. It is noticeable that the synthesized circuit not only



(a) S-param. of the 1-3-th block. (b) S-param. of the 4-th block.

(c) S-param. of the whole Circuit. (d) Norm. radiation pattern at 27 GHz.

Fig. 7 Scattering parameters of the Synthesized Circuit blocks in solid lines, corresponding full-wave response of the Antenna blocks in dotted lines

estimate accurately the scattering parameters but also the far-field behavior of the actual antenna array. This result was possible without any full-wave optimization of the entire circuit, which makes the design procedure easier and less time-consuming.

3 Synthesis-Based Filtenna Design

This section present a general overview of the synthesis and design methodology of a circularly polarized coaxial horn filtering antenna (filtenna). Further details can be found in [2, 8, 9]. This solution tries to help to the interference mitigation between units on a space system. The filtenna must allow the transmission with 19 dB of gain and 15 dB of return loss in the portion of the Ka-band reserved for the communication with the earth (25.5–27 GHz). The filtenna must also attenuate 20 dB the band reserved for improved sensing of snow and ice thickness (35.5–36 GHz).

The proposed filtering antenna, shown in Fig. 8, consist of integrating a filtering function into the flare of the horn antenna. So that, there is no need for more space for the filter part. The design methodology consist of (a) first design a standard horn

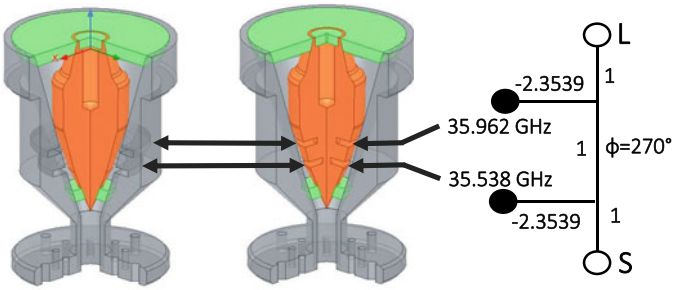
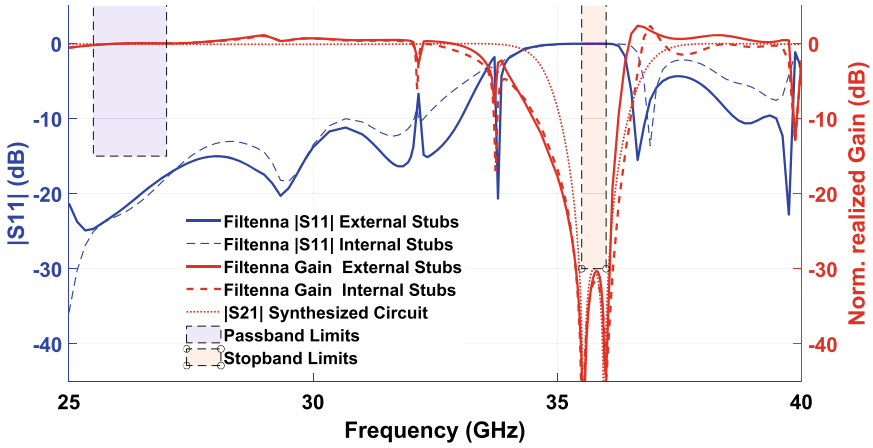


Fig. 8 Coaxial Horn Filtering Antenna with stubs in the horn body (left), and in the coaxial core (middle) . Gray: horn body. Orange: coaxial core. Green: Teflon supports. On the right is the synthesized stop-band circuit

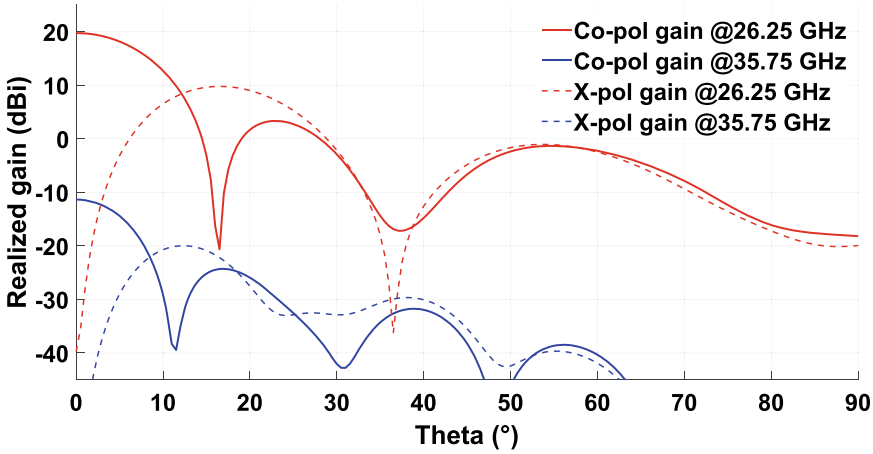
antenna that fulfills the specified gain and return loss; (b) synthesize a stop-band circuit that fulfill the specified attenuation; (c) integrate this filtering function into the horn flare of the designed horn antenna.

In order to integrate the filtering function into the flare horn, we first tried using standard circular stubs. However, it was found that the filtering behavior of these stubs had a bad performance in the stop band because of high-order modes are exited in the horn flare (TM_{11} specifically). To overcome this problem, a coaxial core is inserted such that the TM_{11} cutoff frequency is located as far as possible from the stopband at the position where the coaxial stubs will be placed. In other words, a coaxial core is inserted into the horn in order to have a suitable area to implement the filtering function with stubs. This coaxial core should also be designed to have good return loss (>15 dB) and antenna gain comparable to the antenna gain without it. Finally, the stubs are placed into the suitable area and tuned according to the synthesized circuit.

Two options were envisioned for the stubs: (a) the stubs carved into the horn antenna body (external stubs), (b) the stubs excavated into the coaxial core (internal stubs) as shown in Fig. 8. The last one has the advantage that the horn body is standard, and all the complexity of the filtering function relies only on the coaxial core. This allows also to tune the stopband by just changing the coaxial core. Figure 9a shows the full-wave simulations of both designs as well as the transmission parameter of the synthesized circuit. It can be seen a very good match between the normalized broadside gain of the designed filtering antennas and the $|S_{21}|$ parameters of the synthesized circuit. Also, the return loss is higher than 15 dB as required. Figure 9b shows instead the radiation pattern in the centers of the passband and stopband. This verifies the 30 dB of difference between the passband and stopband co-polarization and cross-polarization gain in all the directions (except in the radiation nulls) and not only at the broadside. This figure also shows more than 19 dB of gain in the passband as requested.



(a) $|S_{11}|$ and broadside gain normalized to 26.25 GHz.



(b) Radiation pattern at 26.25 GHz (passband) and at 35.75 GHz (stopband).

Fig. 9 Frequency Response of the designed Coaxial Horn Filtering Antenna.

4 Conclusion

This paper presents first a general overview of novel filter synthesis techniques that are more precise and completely analytical for extracted-pole and cascaded-block filters including non-resonating nodes. All these synthesis methods follows the same principle: starting with the well-known coupling-matrix synthesis, suitable circuit or matrix transformations are then applied to obtain the required topology. Then, the paper provides a summary of novel synthesis-based designs of filters, antennas, and

filtering antennas. These designs starts with the synthesis of an equivalent circuit that estimate accurately the full-wave behavior, then the actual full-wave prototype is designed by block using the divide-to-conquer technique. Finally, the prototype are assembled by means of transmission lines. The results have shown a very good agreement between the synthesized circuits and the full-wave simulations of the actual prototypes.

References

1. Amari S, Macchiarella G (2005) Synthesis of inline filters with arbitrarily placed attenuation poles by using nonresonating nodes. IEEE Trans Microw Theory Tech 53(10):3075–3081
2. Caicedo Mejillones S, Oldoni M, Moscato S, D'Amico M, Gentili G (2023) Circularly polarized coaxial horn filtenna for electromagnetic interference mitigation. IEEE Trans Antennas Propag 71(12):9487–9496. https://doi.org/10.1109/TAP.2023.3321422
3. Caicedo Mejillones S (2023) Synthesis of filters and filtering antennas for micro and millimeter wave applications. Doctoral Dissertation, Politecnico di Milano
4. Caicedo Mejillones S, Oldoni M, Moscato S, Macchiarella G (2020) Analytical synthesis of fully canonical cascaded-doublet prototype filters. IEEE Microw Wirel Compon Lett 30(11):1017–1020
5. Caicedo Mejillones S, Oldoni M, Moscato S, Macchiarella G, D'Amico M, Gentili G (2021) Accurate synthesis of extracted-pole filters by topology transformations. IEEE Microw Wirel Compon Lett 31(1):13–16
6. Caicedo Mejillones S, Oldoni M, Moscato S, Macchiarella G, D'Amico M, Gentili GG, Biscevic G (2021) Unified analytical synthesis of cascaded n-tuplets filters including nonresonant nodes. IEEE Trans Microw Theory Tech 69(7):3275–3286
7. Cameron R, Mansour R, Kudsia C (2007) Microwave filters for communication systems: fundamentals. Wiley, Design and Applications
8. Oldoni M, Caicedo Mejillones S, Moscato S, Giannini A (2021) Ka-band coaxial horn filtenna for enhanced electromagnetic compatibility on spacecraft. In: 2021 IEEE MTT-S international microwave filter workshop (IMFW), pp 269–271
9. Oldoni M, Caicedo Mejillones S, Moscato S, Giannini A (2022) Filtennas in space: a novel approach for radio-frequency interference mitigation. In: 2022 ESA workshop on aerospace EMC (Aerospace EMC), pp 1–6
10. Tian H, Liu C (2019) Gu X (2019) Proximity-coupled feed patch antenna array for 79 ghz automotive radar. J Eng 19:6244–6246
11. Zhang Y, Zhang XY, Pan YM (2017) Compact single- and dual-band filtering patch antenna arrays using novel feeding scheme. IEEE Trans Antennas Propag 65(8):4057–4066

Innovative Cross-Layer Optimization Techniques for the Design of Optical Networks

Mëmëdhe Ibrahimi

Abstract Optical networks have become indispensable in the era of *5G-and-beyond* communications, supporting applications that require unprecedented capacity, reliability, and high Quality-of-Transmission (QoT) of lightpaths. To meet these requirements, network operators strive to provide innovative solutions while managing network costs effectively. This work summarizes the main findings of my Ph.D. thesis *Innovative Cross-Layer Optimization Techniques for the Design of Filterless and Wavelength-Switched Optical Networks*, that has been conducted in partnership with an industrial partner, *SM-Optics*. The main objective of the Ph.D. thesis is to investigate solutions to reduce network costs while enabling network expandability through novel network architectures. To ensure cost savings and scalability, (1) we optimize the deployment of Optical Amplifiers (OA) while accurately modeling physical layer impairments in filterless networks, (2) we propose a modular node architecture relying on pluggable devices and a scalable add/drop section at the node level for traffic grooming and capacity increase, and (3) we investigate the application of Machine Learning (ML)—regression approaches to estimate lightpaths' QoT as they allow to make informed decisions about how conservative or aggressive a network operator can be when taking network planning choices, i.e., deploying a new lightpath. Numerical evaluations show that our proposed approaches achieve significant cost savings compared to benchmark approaches: (1) ∼50% savings in OA cost, (2) ∼50% savings in node architecture equipment cost, (3) ∼70% in penalty costs for deploying wrong lightpath configurations.

1 Introduction

To cope with high capacity demands and reliable connectivity due to *5G-and-beyond* applications while keeping network costs to minimum, novel network architectures must be adopted. Filterless Optical Networks (FONs) are emerging as a cost-effective and scalable solution, and currently being deployed by network operators. In partic-

M. Ibrahimi (✉)
Politecnico di Milano, Department of Electronics, Information and Bioengineering, Via Giuseppe Ponzio, 34, 20133 Milano, Italy
e-mail: memedhe.ibrahimi@polimi.it

F. Amigoni (ed.), *Special Topics in Information Technology*,
PoliMI SpringerBriefs, https://doi.org/10.1007/978-3-031-51500-2_12

ular, horseshoe filterless networks have been proven as a practical solution [1, 8, 12]. The main benefits brought by the deployment of FON are *(i)* low CapEx costs as costly Wavelength Selective Switches (WSS) are replaced by passive splitters and combiners; *(ii)* a compact shelf configuration is possible due to the low power requirements of passive splitters and combiners, and a modular architecture relying on pluggable devices, i.e., optical amplifiers; and *(iii)* a modular, i.e., scalable, add/drop section at node level and deployment of equipment supporting traffic-grooming ensure a capacity increase to cope with new traffic requirements. To this end, the objective is to investigate and develop novel optimization techniques to minimize the cost of Optical Amplifiers and the cost Optical Transport Network (OTN) traffic-grooming boards in FONs.

Moreover, the continuous growth in network design complexity has brought the need for new enabling methodologies such as Machine Learning (ML) to address the challenges in network control, design and management [4, 13]. A continuous challenge for network designers remains the accurate estimation of physical layer impairments. Compared to traditional approaches, e.g., closed-form formulas, that lead to an under-utilization of spectral resources due to design margins, ML has been shown to be an efficient solution capable of capturing the complex non-linear nature of signal propagation and estimating uncertainties introduced by time-varying penalties.

To this end, the objective is to develop novel ML-regression approaches that estimate the probability distribution of unestablished lightpaths' Signal-to-Noise Ratio (SNR), i.e., quantifying if and how far the SNR is from the threshold (which is pivotal in presence of uncertainties introduced by fast time-varying penalties) [7].

The following sections are organized as follows: Sect. 2 describes the placement of Optical Amplifiers in horseshoe FONs. Section 3 describes the problem of minimizing OTN traffic-grooming equipment cost in mixed 10G/100G/200G FONs. Section 4 describes the application of ML-regression to estimate the SNR distribution of unestablished lightpaths.

2 Optical Amplifier Placement in Metro Networks

2.1 Introduction and Problem Description

Fiber-To-The-Home technology has shown to be crucial in coping with high capacity demands due to an emerging use of applications related to remote working, teleconferencing, video on demand, gaming and online education. As a result, the management of metro networks has been transformed, leading operators to redefine the design process with a main objective to minimize network costs. An alternative to reduce costs is to utilize the short link distances and high number of nodes to optimize the number, location and type of Optical Amplifiers (OAs) in egress of network nodes, i.e., booster amplifiers (*boosters*), in ingress of network nodes, i.e.,

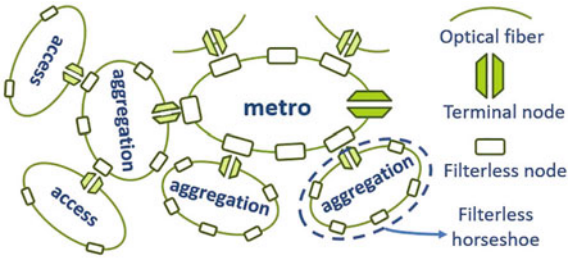


Fig. 1 Metro network composed of interconnected horseshoe topologies

pre-amplifiers (*pre-amps*), and those located along fiber links, i.e., in-line amplifiers (*ILAs*), while guaranteeing lightpath feasibility in terms of Signal-to-Noise Ratio (SNR) [9].

FONs are emerging as a promising technological direction to reduce cost in optical networks as costly WSS are replaced by broadcast-and-select switching architectures composed by passive splitters and combiners. However, due to channels' broadcast beyond lightpath termination, FONs suffer a reduced spectral efficiency. Moreover, when deploying OAs in FONs, the absence of WSSs causes propagation of Amplified Spontaneous Emission (ASE) noise beyond lightpath termination, and accumulation of ASE generated before a lightpath is initiated. Hence, FON architectures may become more sensitive to lightpath degradation due to higher ASE noise compared to WSS-based node architectures (WSS serves to block unintended ASE accumulation).

We consider a practical case of a metro network composed by several interconnected filterless branches, where each branch is constituted by a horseshoe as shown in Fig. 1. These horseshoe topologies typically contain two types of nodes: Terminal (T) nodes that interconnected to the rest of the metro network and are equipped with filters that block ASE noise propagation, and Filterless (F) nodes that are placed along the optical line and are equipped with passive splitters, combiners and variable optical attenuators.

Figure 2 shows an example of signal propagation and ASE noise accumulation in a filterless horseshoe. Lightpath-1 (L_1) is generated at *T-1* on wavelength λ_1 (red colored) and destined to node *F1* and lightpath-2 (L_2) is generated at node *F1* on wavelength λ_2 (blue colored) and terminated at node *F2*. Due to the broadcast feature of FON, λ_1 propagates beyond *F1* and λ_2 propagates beyond *F2*. As a result, ASE noise generated by *OA-A*, i.e., *ASE-A*, propagates beyond destination and accumu-

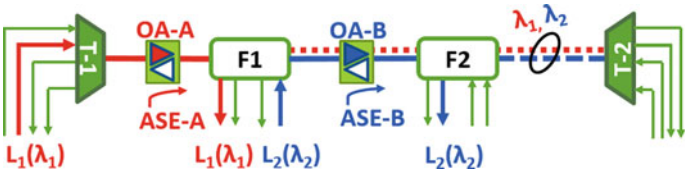


Fig. 2 Example of ASE noise accumulation in a filterless horseshoe topology

lates with ASE noise generated by *OA-B*, i.e., *ASE-B*, and impact the SNR of L_2. Given the additive nature of the ASE noise generated by amplifiers [9], the SNR due to ASE contribution for L_2, i.e., SNR^{ASE}_{LP-2}, can be expressed as:

$$\frac{1}{SNR^{ASE}_{LP-2}} = \frac{1}{SNR^{ASE}_{A}} + \frac{1}{SNR^{ASE}_{B}} \tag{1}$$

where SNR^{ASE}_{A} is the accumulated SNR_{ASE} contribution due to *OA-A* amplifier, and SNR^{ASE}_{B} is the SNR_{ASE} contribution due to *OA-B* amplifier. Note that, the same can be generalized for any lightpath i that traverses M optical amplifiers from source to destination and is impacted by the accumulated ASE noise of N optical amplifiers located before the source node [1].

Problem statement. The problem of OA placement in filterless metro networks can be stated as follows: **Given** a horseshoe FON, a set of traffic demands, and a set of candidate locations to place OAs, **decide** the OA placement (location, i.e., booster, pre-amplifier and ILA) and decide the Route and Spectrum Allocation (RSA) for each traffic demand, **constrained by** a required QoT for each lightpath (SNR and received power thresholds), spectrum continuity and contiguity constraint, network capacity constraint, with the **objective** of minimizing the overall cost, constituted by the deployed OAs.

2.2 Genetic Algorithm for OA Placement

Due to high combinatorial complexity of the problem (considering x OA candidate locations, there are 2^x combinations of OA placement) and, its non-linear nature which requires to consider several cross-layer design parameters, we have developed a Genetic Algorithm (GA) to solve this problem. The evolutionary process of the GA is driven by competition among members (solutions) of the population and genetic operations, such as mutation and crossover. Each solution of the population is encoded as a string of binary values, i.e., the genes, which represent the candidate locations for OA placement, and assume value "1" if an OA is placed and "0" if the OA is not placed.

Each solution is characterized by its fitness value and feasibility status. The fitness of a member of the population is the cost accounting for the type and location of the placed OAs. The objective is to minimize total OAs cost, therefore a lower fitness value is preferred compared to a higher fitness value. Feasibility represents the extent to which a solution satisfies the constraints, and it is defined as the ratio between the total number of feasible lightpaths and the total number of lightpaths routed. A lightpath i is feasible if its SNR (SNR_i) and received power ($P_{rec,i}$) are greater than a pre-defined threshold [9].

We propose two versions of the GA, *minCostGA* and *ConstrainedGA*. The objective of *minCostGA* is to minimize cost, however, this may lead to a drawback on

SNR performance. *ConstrainedGA* overcomes the drawbacks of *minCostGA* and minimizes OA cost while guaranteeing the SNR performance provided by the benchmark approaches.

2.3 Illustrative Numerical Results

We evaluate the GA performance on realistic 6-node horseshoe topologies. A candidate OA location is assumed every 10 km of fiber and booster and pre-amplifier candidate locations at each node. The length of the horseshoe is varied, simulating small metro (100 km) and regional metro (900 km) networks. Results are averaged for each horseshoe length by considering 20 topologies with random link lengths with a variability of ±50% from average link length. For example, a horseshoe of length 300 km with 5 links, the average link length is 60 km so each link's length is equal to a random value between 30 km and 90 km.

Five approaches are compared: *minCostGA-FON*, *minCostGA-WSON*, *ConstrainedGA-FON*, *ConstrainedGA-WSON*, and Baseline. As a benchmark strategy, i.e. Baseline, we consider an OA placement working as follows: *(i)* all nodes are equipped with pre-amplifiers and boosters (booster OA gain is set to compensate for the node loss and pre-amp gain set to compensate for the span it terminates), and *(ii)* inline amplifiers are placed approximately every 60 km (corresponding to an OA gain of 15 dB).

Figure 3 shows the numerical results in terms of total cost of deployed amplifiers in cost units (*cu*) in case of FON and WSON network architectures. Compared to Baseline approach, *minCostGA-FON* and *minCostGA-WSON* achieve OA cost savings up to 60% and 52%, respectively. However, despite the significant cost savings, *minCostGA* has lower minimal SNR (minSNR) and average SNR (avgSNR) of around 5 dB compared to Baseline.

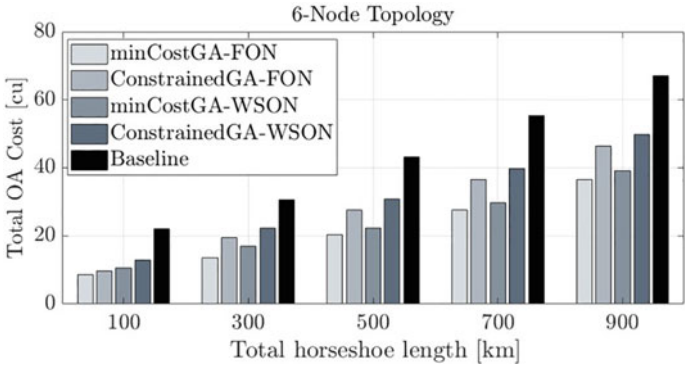


Fig. 3 OAs total cost for FON, WSON and Baseline for varying total horseshoe length

To overcome the lower SNR performance, we developed *ConstrainedGA* that minimizes overall OA cost while meeting Baseline SNR. Figure 3 shows that *ConstrainedGA* achieves cost savings up to 56% (in FON) and up to 41% (in WSON) compared to Baseline. Comparing OA placement in FON vs WSON, we observe that OA placement in FON allows to save up to 25% total OA cost savings compared to WSON.

3 Minimizing Equipment Cost in Mixed 10G/100G/200G Filterless Horseshoes with Hierarchical OTN Boards

3.1 Introduction and Problem Description

An alternative to further reduce costs in metro FON networks is to optimize the deployment of traffic-grooming boards and interfaces deployed in OTN equipment. The practical need to solve this problem comes from the fact that real-world metro networks still employ legacy 10G technology, hence a gradual upgrade that mixes coherent (100G/200G) and non-coherent (10G) transmission technologies is required for cost-efficient short/mid-term network planning. Additionally, we consider real filterless horseshoe networks, that currently represent a prominent candidate for cost-effective optical-network deployment.

However, accounting for a hierarchy of different OTN grooming boards while employing mixed coherent and non-coherent transmission technologies makes the problem extremely complex, as it accounts for the inter-dependency between the deployment of various types of OTN boards and the establishment of lightpaths at different rates. In fact, to the best of our knowledge, no previous works have tackled the grooming problem considering: *(i)* the hierarchical grooming-node structure consisting of various *stacked* OTN boards, *(ii)* the *co-existence of coherent and non-coherent transmission technologies* (100G/200G and 10G lightpaths), and *(iii)* filterless node architecture that adds significant complexity to the problem as it impacts wavelength allocation and lightpath establishment.

Problem statement. The problem of minimizing equipment cost in filterless horseshoe networks with hierarchical OTN boards (*minOTN*) can be summarized as follows: **Given** a filterless horseshoe topology, a set of traffic requests between node pairs, a set of candidate OTN boards and interfaces to be placed at each node, **decide** jointly: *(i)* the deployment of OTN boards and interfaces (including location and type), DCM modules and filters for non-coherent traffic, and *(ii)* the route and wavelength assignment (RWA) and traffic-grooming of traffic requests, **constrained by** *(i)* traffic-processing capacity of each OTN board and interface type, *(ii)* maximum number of client interfaces given for each board, *(iii)* wavelength capacity, *(iv)* filterless networks constraints on wavelength assignment and *(v)* ensuring dedicated path protection for traffic requests, with the **objective** of minimizing equipment cost

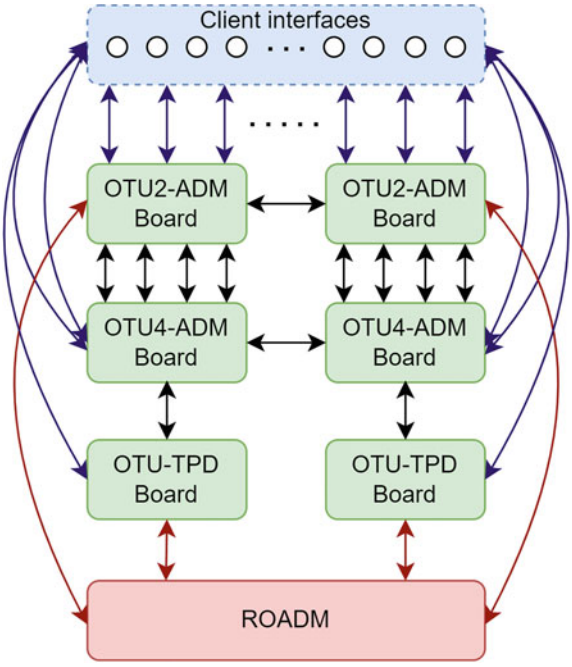


Fig. 4 Structure of the node with OTN traffic-grooming boards

of deployed equipment (OTN boards and interfaces, transponders, DCM modules and filters).

Problem modeling. Figure 4 shows the hierarchical structure of the node, and it is composed of up to three stacked OTN boards: OTU2-ADM, OTU4-ADM, OTU-TPD. Each node may be equipped with a pair of OTU2-ADM, OTU4-ADM, and OTU-TPD boards. Note that, at the optical layer, ROADM is not equipped with WSS, but rather with passive splitters and combiners [6]. Each OTU2-ADM supports add/drop and performs multiplexing, and has ten access interfaces, each with a maximum capacity of 10 Gbit/s, allowing clients to connect directly to the board. The access interfaces support various types of client signals such as SDH (i.e., STM1, STM4, STM16, STM64), Ethernet (i.e., 1GbE, 10GbE), and OTN (i.e., OTU1). Each OTU2-ADM board has four Small Form-factor Pluggable (SFP) interfaces that can be used as a 10 Gbit/s OTN point-to-point (p2p) line interface connecting to OTU4-ADM or a 10 Gbit/s OTN optical interface connecting to ROADM, capable of establishing a non-coherent 10G lightpath. OTU4-ADM performs add/drop and multiplexing, and clients can connect to OTU4-ADM boards through ten 10 Gbit/s access interfaces. Moreover, each OTU4-ADM board can receive/send traffic from/to OTU2-ADM via its four 10 Gbit/s line interfaces. OTU4-ADM can groom traffic of connected clients and OTU2-ADM into an OTU4 signal at the p2p line interface connected to an OTU-TPD board. An OTU-TPD board provides an interface for at most two OTU4-ADM boards and has a colored output interface connected to a ROADM and establishes a coherent 100G/200G lightpath. Optimizing the architecture of such

hierarchical OTN boards enables a cost-efficient traffic-grooming, leading to overall network cost reduction.

3.2 Strategies to Solve MinOTN

To solve *minOTN* we have developed an ILP model and, to scale with larger problem instances, we developed a Genetic Algorithm (GA). We benchmark ILP and GA to a state-of-the-art approach, referred to as Omnibus (OB) [6].

ILP model. The objective function ($\min \sum_{j \in N} \delta_j * \mu_j + \sum_{i \in N_3, w \in W} \tau_{i,w} * \mu_i$) is to minimize the cost (μ_j) of logical nodes (δ), i.e., OTN boards and interfaces, and cost (μ_i) of transponders ($\tau_{i,w}$) used to establish lightpaths on wavelength w.

Main constraints of the problem refer to the capacity of OTN boards and their interfaces, transponder interfaces, wavelength capacity, filterless constraints, and other constraints referring to the physical constraints of OTN boards.

Genetic Algorithm (GA). The encoding of *minOTN* is done such that OTN board/interface placement and routing of demands are encoded in gene clusters and genes: a gene cluster is defined for each demand and, in each gene cluster, the genes represent candidate paths to route the demand.

Fitness value is defined as the total cost of deployed equipment (we minimize cost, so a lower *fitness value* is desirable). *Feasibility* refers to constraints violation: a solution has feasibility equal to one, if it satisfies all constraints of the problem.

Omnibus (OB). OB is considered as a benchmark scenario as it represents the real-world OTN boards deployment approach. OTN boards are placed considering 100G lightpaths between all neighboring nodes, hence traffic-grooming is performed at every node. In case traffic cannot be served by an OB with single 100G lightpaths, an upgrade is adopted at nodes generating/handling more traffic by doubling the deployed equipment.

3.3 Illustrative Numerical Results

Experimental evaluations are performed on a real 5-node filterless horseshoe topology for three traffic matrices with incremental traffic, i.e., TM1, TM2 (TM1+45% additional traffic), and TM3 (TM1+60% additional traffic), and compare ILP, GA and OB. The cost model is provided by the industrial partner (*SM-Optics*) [6]. Figure 5 shows the total cost (and its breakdown) of deployed equipment in terms of equipment type for three traffic matrices (TM1, TM2 and TM3) considering ILP, GA and OB. ILP and GA reach the same equipment cost in case of TM1 and TM2, while in case of TM3, GA reaches a 4% higher equipment cost compared to ILP. In terms of execution time, GA finds a solution in under 5 min while ILP takes up to 9 h.

Compared to OB, ILP and GA achieve cost savings of 51% and 30% in case of TM1 and TM3, respectively. ILP and GA allow significant cost savings due to a

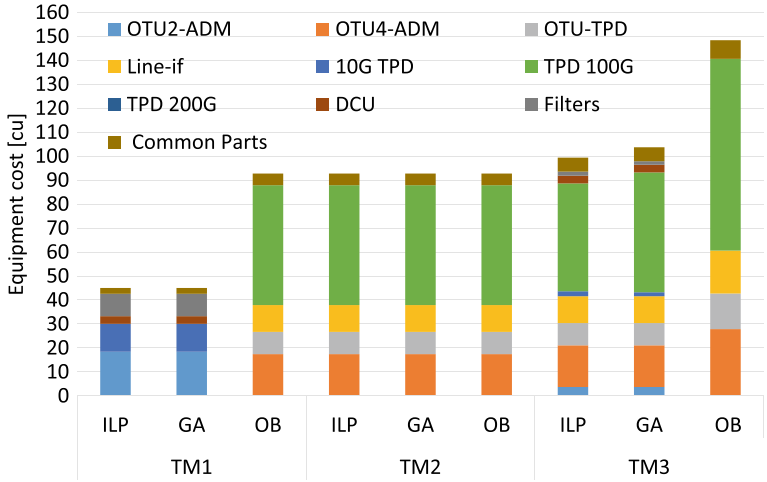


Fig. 5 Equipment cost and each equipment contribution in case of ILP, GA and OB

predominant deployment of 10G transponders instead of coherent 100G transponders as in OB. In case of TM2, ILP and GA reach the same solution as OB. The reason is that in case of TM1, 100G lightpaths are populated around 50%, so there is sufficient residual capacity to serve the added traffic for TM2 without additional equipment. In case of TM1 and TM3, equipment for establishing non-coherent lightpaths (transponders, OTU2-ADM boards, filters and DCM modules) deployed by ILP and GA compose 67% and 12%, respectively. Compared to ILP and GA, equipment deployed by OB are only for coherent lightpath establishment.

4 Machine Learning for Quality-of-Transmission Estimation of Unestablished Lightpaths in Wavelength Switched Optical Networks

4.1 Introduction and Problem Description

To ensure effective and optimized design and planning of optical networks, accurate prediction of lightpath QoT prior to deployment is imperative. Traditionally, QoT estimation in optical networks has been addressed using margined formulas (e.g., the GN-model [14]) that are computationally-fast but lead to under-utilization of spectral resources [15]. Alternatively, ML uses historical data and overcomes the short-comings of margined formulas and estimates lightpaths' QoT in reasonable time by modelling uncertainties not captured by physical layer models [3, 10]. Several studies has tackled the ML-based QoT estimation as a classification problem [11, 16].

However, a classification-based approach has three main drawbacks: (1) it does not convey how close to the system threshold the predicted SNR is; (2) it does not return the predicted distribution of the SNR value; and (3) during training, no distinction is made between a training sample with a SNR slightly above the threshold and a training sample that is way above the threshold, which leads to loss of information. To overcome these drawbacks of ML-based classification, we investigate ML-based *regression* approaches to estimate lightpaths' SNR, under the assumption that the SNR measured at the receiver can be modeled as a random variable [5, 7].

We assume that a lightpath is characterized by a number of features, e.g., amount of traffic, modulation format, total lightpath length, number of links traversed by the lightpath, length of longest link. However, for a given lightpath configuration, the SNR may still exhibit variations, as it depends on several other factors not captured by the considered features as, e.g., fast time-varying penalties due to polarization-dependent losses. It follows that the SNR associated to a lightpath configuration can be modeled as a random variable and thus be characterized by a Probability Distribution Function (PDF). We propose three regression approaches that estimate the parameters characterizing the distribution of the random variable, i.e., SNR:

1. *Matched Gaussian Distribution Regressor (MD-R)*, returns the mean and variance of a Gaussian distribution modeling the SNR value.
2. *Moments Estimation Regressor (ME-R)*, enhances *MD-R* by predicting the four moments of the probability distribution, i.e., mean, variance, skew, and kurtosis.
3. *Quantile Estimation Regressor (QE-R)*, removes the assumption of an underlying Gaussian distribution and estimates the quantiles of the PDF.

To better explain our approach, let us consider a given lightpath configuration as testing instance. *MD-R*, *ME-R* and *QE-R* are used to estimate the parameters of the SNR distribution and serve to answer the question: *what is the probability that the SNR for this testing instance is below the system threshold?*

Figure 6 shows an illustrative example of three SNR estimation approaches. Given an unestablished lightpath described by a set of features, the associated SNR may exhibit different values due to varying network conditions, e.g., noise figure of optical amplifiers, fast time-varying penalties. Such values could be measured only a-posteriori (i.e., after the lightpath establishment) by means of optical performance monitors (OPM) [2] at the receiver node, and constitute the *ground truth* empirical PDF (see Fig. 6a). A standard ML regression model may be leveraged to estimate a scalar value for the SNR, given the features (Fig. 6b). However, this does not capture the uncertainties in the SNR value, e.g., uncertainty introduced due to time-varying penalties. Conversely, if a regressor is used to predict the parameters of the SNR distribution (e.g., mean and variance of a Gaussian distribution as in Fig. 6c or the first four moments of a Gamma distribution as in Fig. 6d), it is then possible to assess how well the estimated PDF fits the observed ground truth samples. This way, a network operator is allowed to set how *conservative* or *aggressive* its planning choices should be, i.e., when deploying a new lightpath. In other words, with the proposed approaches, an operator seeking *low-margin operation* of its network can more discerningly decide how aggressive such margin reduction should be.

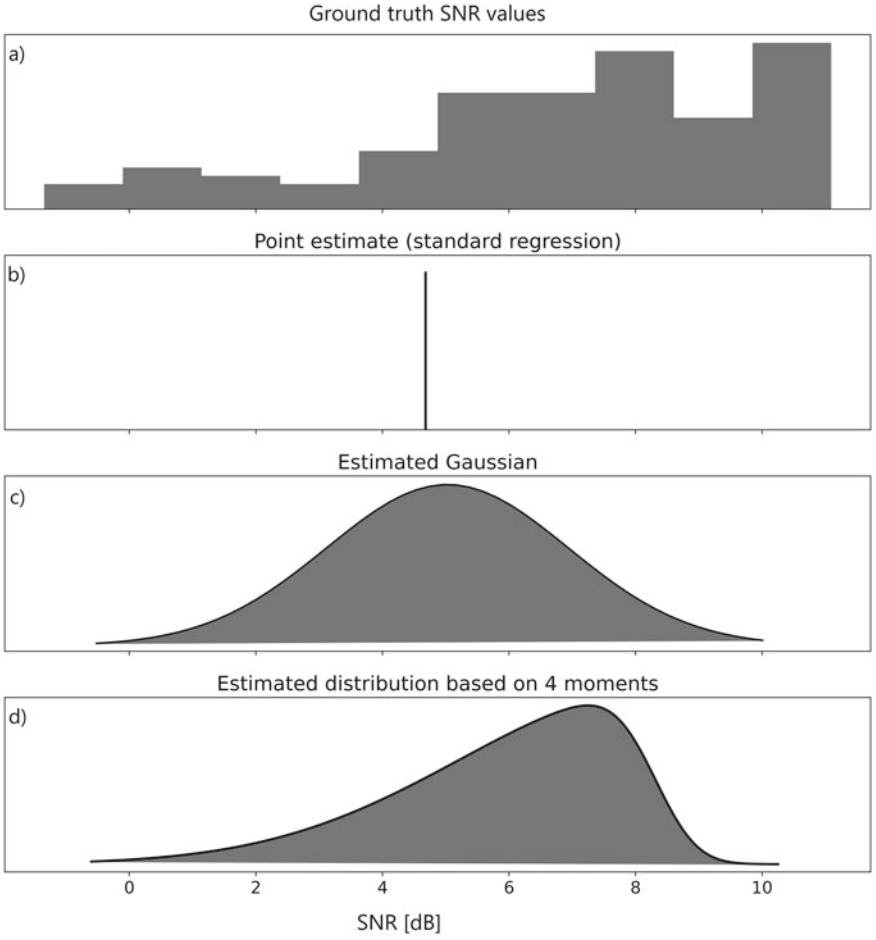


Fig. 6 Example of SNR estimation approaches: **a** SNR ground truth values; **b** Point estimate of SNR; **c** Gaussian distribution, i.e., mean, variance, estimation of SNR; and **d** SNR distribution estimation based on mean, variance, skew and kurtosis

4.2 Illustrative Numerical Result

Synthetic traces of realistic SNR values are generated via E-Tool [16] that assumes frequency slice units of 12.5 GHz, a total 4 THz link capacity and transceivers operating at 28 GBaud with a 37.5 GHz channel bandwidth. We assume a per-link random penalty parameter that accounts for fast time-varying impairments (e.g., polarization effects), according to an exponential distribution (according to the principle of maximum entropy) with a 1 dB average. A dataset of $N = 1000$ lightpaths is generated by randomly choosing a bitrate in [50, 500] Gbps range with 50 Gbps granularity and one of the $r \cdot M$ possible combinations ($r = 3$ routes $\cdot$ $M = 6$ modulation formats,

i.e., (DP)-BPSK, DP-QPSK and DP-n-QAM, with n = 8, 16, 32, 64). For each lightpath, the SNR calculation is repeated $k = 100$ times under different random penalty samples. For the train/test split, a 80/20 ratio is considered. Standard metrics used in ML, e.g., Root Mean Square Error (RMSE), are difficult to interpret from a network operation point of view. Therefore, we provide a cost analysis which quantifies the penalties in the context of wrong lightpath deployment decisions.

Let us consider a lightpath j which belongs to the set J of candidate lightpaths, characterized by the set of features V. The question to be answered is: is SNR_j lower than a system defined threshold SNR_T?

Given the set of features V, let F_{G_i} be the estimated CDF of the random variable G_i that models the SNR, according to estimator i, where $i \in \{MD-R, QE-R, ME-R\}$. The probability that the SNR is below the threshold SNR_T, according to estimator i, can be computed as $p_i = F_{G_i}(SNR_T)$. Different estimators will estimate different probabilities.

One should then make a decision on the basis of this probability. We consider two different penalty costs associated to the two ways that a decision can be wrong: an underestimation cost (C_u), that is paid when SNR_j is estimated to be lower (`Below`) than SNR_T, but is in fact higher (`Above`) than SNR_T; and an overestimation cost (C_o), that is paid when SNR_j is estimated to be higher (`Above`) than SNR_T, but is in fact lower (`Below`) than SNR_T.

The expected penalty for a deployment decision (`Above` or `Below`) is the probability that such decision is wrong, times the cost of taking such a wrong decision. Therefore, if the decision is that SNR_j is *below* SNR_T, the estimated probability of being wrong is equal to $(1 - p_i)$. It follows that the expected cost of deciding that $SNR_j < SNR_T$, according to estimator i is: $C_{i,\texttt{Below}} = (1 - p_i) \cdot C_u$ while if the decision is that SNR_j is *above* SNR_T, the probability of being wrong is equal to p_i. Therefore, the expected cost of deciding that $SNR_j > SNR_T$, according to estimator i is: $C_{i,\texttt{Above}} = p_i \cdot C_o$ For each estimator i, we make a decision D_i according to the following rules: D_i is `Below` if $C_{i,\texttt{Below}} < C_{i,\texttt{Above}}$, and `Above` otherwise. The decision D_i is compared to the decision that would be made by leveraging the ground truth (D_{GT}), i.e., the one computed based on actual SNR measurements. The ground truth decision is defined as follows: D_{GT} is `Below` if $SNR < SNR_T$, and `Above` otherwise. *SNR* is the actual sampled value. If $D_i \neq D_{GT}$, then the respective cost associated to the decision is added to the total penalty cost of estimator i is: $PC_i = (\sum_{j \in J} min(C_{i,\texttt{Below}}, C_{i,\texttt{Above}}))/|J|$, where $|J|$ is the total number of lightpaths. Note that we add the $min(C_{i,\texttt{Below}}, C_{i,\texttt{Above}})$ to the PC_i since the estimator makes the decision based on the comparison between $C_{i,\texttt{Below}}$ and $C_{i,\texttt{Above}}$, so the minimum cost reflects the cost of the wrong decision of the estimator.

We compare the performance of the three proposed estimation approaches in terms of decision penalties against the following four baselines: (1) always decide `Below`, *(ADB)*; (2) always decide `Above`, *(ADA)*; (3) random decision, *(RD)*; and (4) cost-insensitive decision *(CI)*, i.e., make the decision `Below` (`Above`) if the SNR mean value estimated by the Gaussian regressor, i.e., *MD-R*, is below (above) the threshold. We also consider a lower bound of the obtainable cost which reflects

Table 1 Penalty cost (PC) in cost units (cu) for each estimator

	IE	ME-R	QE-R	MD-R	ADB	CIB	RDB	ADA
PC_i [cu]	0.059	0.069	0.074	0.087	0.275	1.527	2.532	7.248

the penalty cost incurred by an "ideal" estimator (noted as *IE*), which always returns as output an estimated CDF identical to GT.

We assign the following numerical value to each cost type: $C_u = 1$ cu (cost unit) and $C_o = 10$ cu. These values can be interpreted as follows. If we underestimate the lightpath's SNR we may erroneously consider as infeasible a lightpath configuration that was in fact feasible. Hence, a lightpath with a less spectrally-efficient modulation format will be deployed, leading to the unnecessary occupation of some spectral resources, yet no service disruption will be incurred. Conversely, in the case of overestimation, we erroneously decide to deploy a lightpath with a modulation format which will lead to a below-threshold SNR, eventually resulting in service disruption.

From a network operator's point of view, the penalty in case of service disruption is higher compared to the penalty of deploying a lightpath that does not adopt the most spectrally-efficient modulation format. This disparity is captured by our cost values, though they may not exactly reflect the actual economic losses experienced by a network operator.

We perform a set of 100 sequences of deployment decisions, each one including 500 candidate lightpaths. Therefore, the total penalty cost for each estimator is averaged over 100 simulations. Table 1 reports the average penalty per instance for each estimator, in the order of performance from best to worst.

We observe that our proposed estimators all perform much better compared to baseline approaches (as for IE, it provides the lowest cost penalty, but it only represents an ideal lower bound for this cost analysis). Results confirm the importance of estimating the probability distribution instead of using a point estimate. In fact, the cost penalty of *MD-R* (which assumes a Gaussian distribution of the estimated PDF) is significantly lower compared to CIB (the standard regressor that only estimates the SNR mean value). Utilizing more sophisticated estimators, as, e.g., *ME-R*, we achieve an even better performance in terms of cost penalty ($0.069 < 0.087$). Note that *ADB* baseline provides a better cost result compared to *ADA* baseline, because we use a 70/30% ratio between the lightpath configurations in the dataset having a SNR value below/above threshold, and hence the estimator is more likely to correctly predict a SNR value being below threshold.

References

1. Ayoub O, Karandin O, Ibrahimi M, Castoldi A, Musumeci F, Tornatore M (2022) Tutorial on filterless optical networks [invited]. J Opt Commun Netw 14(3):1–15
2. Christodoulopoulos K, Kokkinos P, Di Giglio A, Pagano A, Argyris N, Spatharakis C, Dris S, Avramopoulos H, Antona J, Delezoide C, et al (2015) ORCHESTRA-optical performance monitoring enabling flexible networking. In: 2015 17th international conference on transparent optical networks (ICTON). IEEE, pp 1–4
3. Di Cicco N, Ibrahimi M, Rottondi C, Tornatore M (2022) Calibrated probabilistic qot regression for unestablished lightpaths in optical networks. In: 2022 international Balkan conference on communications and networking (BalkanCom), pp 21–25. 10.1109/BalkanCom55633.2022.9900791
4. Di Cicco N, Talpini J, Ibrahimi M, Savi M, Tornatore M (2023) Uncertainty-Aware QoT forecasting in optical networks with bayesian recurrent neural networks. In: IEEE ICC'23—ONS symposium. Rome, Italy
5. Ibrahimi M, Abdollahi H, Rottondi C, Giusti A, Tornatore M (2020) Machine learning regression vs. classification for qot estimation of unestablished lightpaths. In: 2020 advanced photonic congress (APC), pp 1–2
6. Ibrahimi M, Ayoub O, Attarpour A, Musumeci F, Castoldi A, Ragni M, Tornatore M (2022) Minimizing equipment and energy cost in mixed 10g and 100g/200g filterless horseshoe networks with hierarchical otn boards. In: annales des télécommunications, pp 1–15
7. Ibrahimi M, Abdollahi H, Rottondi C, Giusti A, Ferrari A, Curri V, Tornatore M (2021) Machine learning regression for qot estimation of unestablished lightpaths. J Opt Commun Netw 13(4):B92–B101. https://doi.org/10.1364/JOCN.410694
8. Ibrahimi M, Ayoub O, Karandin O, Musumeci F, Castoldi A, Pastorelli R, Tornatore M (2021) Qot-aware optical amplifier placement in filterless metro networks. IEEE Commun Lett 25(3):931–935. https://doi.org/10.1109/LCOMM.2020.3034736
9. Ibrahimi M, Ayoub O, Musumeci F, Karandin O, Castoldi A, Pastorelli R, Tornatore M (2020) Minimum-cost optical amplifier placement in metro networks. J Light Technol 38(12):3221–3228. https://doi.org/10.1109/JLT.2020.2991374
10. Ibrahimi M, Rottondi C, Tornatore M (2022) Machine learning methods for quality-of-transmission estimation. In: Machine learning for future fiber-optic communication systems. Elsevier, pp 189–224
11. Jimenez T, Aguado JC, de Miguel I, Duran RJ, Angelou M, Merayo N, Fernandez P, Lorenzo RM, Tomkos I, Abril EJ (2013) A cognitive quality of transmission estimator for core optical networks. J Light Technol 31(6):942–951
12. Karandin O, Ayoub O, Ibrahimi M, Musumeci F, Castoldi A, Pastorelli R, Tornatore M (2021) Optical metro network design with low cost of equipment. In: 2021 international conference on optical network design and modeling (ONDM), pp 1–4
13. Musumeci F, Rottondi C, Nag A, Macaluso I, Zibar D, Ruffini M, Tornatore M (2019) An overview on application of machine learning techniques in optical networks. IEEE Commun Surv Tutor 21(2):1383–1408
14. Poggiolini P, Bosco G, Carena A, Curri V, Jiang Y, Forghieri F (2014) The GN-model of fiber non-linear propagation and its applications. J Light Technol 32(4):694–721
15. Pointurier Y (2017) Design of low-margin optical networks. J Opt Commun Netw 9(1):A9–A17
16. Rottondi C, Barletta L, Giusti A, Tornatore M (2018) Machine-learning method for quality of transmission prediction of unestablished lightpaths. IEEE/OSA J Opt Commun Netw 10(2):A286–A297