

München 2024

Technische
Universität
München



Tim Lüth, Maik Berger, Burkhard Corves (Hrsg.)

GETRIEBETAGUNG 2024

26. – 27. September 2024



λoγoς

Tagungsband
Getriebetagung 2024

Tagungsband

Getriebetagung 2024

München

26.-27. September 2024

Herausgegeben von

Tim Lüth

Maik Berger

Burkhard Corves

Herausgeber:

Prof. Dr. rer. nat. Dipl.-Ing. Tim Lüth
Technische Universität München
Lehrstuhl für Mikrotechnik und Medizingerätetechnik
Boltzmannstraße 15
85748 Garching

Prof. Dr.-Ing. Maik Berger
Technische Universität Chemnitz
Professur Montage- und Handhabungstechnik
Reichenhainer Straße 70
09126 Chemnitz

Prof. Dr.-Ing. Dr. h. c. Burkhard Corves
RWTH Aachen
Institut für Getriebetechnik, Maschinendynamik und Robotik
Eilfschornsteinstraße 18
52062 Aachen

Für den Inhalt der Beiträge sind deren Autoren selbst verantwortlich

Bibliografische Information der Deutschen Nationalbibliothek:

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.dnb.de> abrufbar.

Dieses Werk ist lizenziert unter der Creative Commons Attribution 4.0 Lizenz CC BY-NC-ND (<https://creativecommons.org/licenses/by-nc-nd/4.0/>). Die Bedingungen der Creative-Commons-Lizenz gelten nur für Originalmaterial. Die Wiederverwendung von Material aus anderen Quellen (gekennzeichnet mit Quellenangabe) wie z.B. Schaubilder, Abbildungen, Fotos und Textauszüge erfordert ggf. weitere Nutzungsgenehmigungen durch den jeweiligen Rechteinhaber.

Titelbild: © Astrid Eckert / TUM

ISBN 978-3-8325-5866-6



Logos Verlag Berlin GmbH
Georg-Knorr-Str. 4, Geb. 10
D-12681 Berlin
Tel.: +49(0)30 / 42 85 10 90
<https://www.logos-verlag.de>

Vorwort

Die Getriebetechnik war lange eine der Hauptfachrichtungen, welche jungen Maschinenbauingenieuren in Ihrem Studium beigebracht wurde. Jedoch schrumpfte der Stellenwert der Getriebetechnik, als Motoren und deren Ansteuerung kleiner und günstiger wurden. Zunehmend wurden die klassischen Getriebe durch vermeintliche Universallösungen wie Roboterarme mit vielen Freiheitsgraden ersetzt.

Heute sind wir an einem Punkt, an dem die Getriebetechnik wieder an Bedeutung gewinnt. Dies lässt sich sowohl aus den Anforderungen der Industrie nach schnelleren und günstigeren Lösungen als auch an den Anmeldezahlen der Studierenden für die Lehrveranstaltungen, die das Wissen um die Getriebetechnik vermitteln, erkennen.

Dieses zwischenzeitliche Schwinden des Interesses an der Getriebelehre und der normalerweise damit eingehende Wissensverlust konnten jedoch verhindert werden durch die Motivation und das Engagement vieler einzelner Personen.

Eine dieser Personen ist Dr. Franz Irlinger. Über 35 Jahre lang hat er nicht nur in Industrie- und Forschungsprojekten die Anwendung der Getriebelehre fortgeführt, sondern auch über diese Zeit unzähligen Studierenden dieses Fach nähergebracht, sein Wissen an diese weitergegeben und in diesen eine Begeisterung für den Fachbereich entzündet.

Ende September geht Dr. Franz Irlinger in den wohlverdienten Ruhestand. Diesen Anlass möchten wir deshalb nutzen, um uns bei ihm für die vielen Jahre des Engagements für die Getriebetechnik zu bedanken.

Im Namen des Programmausschusses, der Ausrichter und Organisatoren

Prof. Dr. rer. nat. Dipl.-Ing. Tim C. Lüth

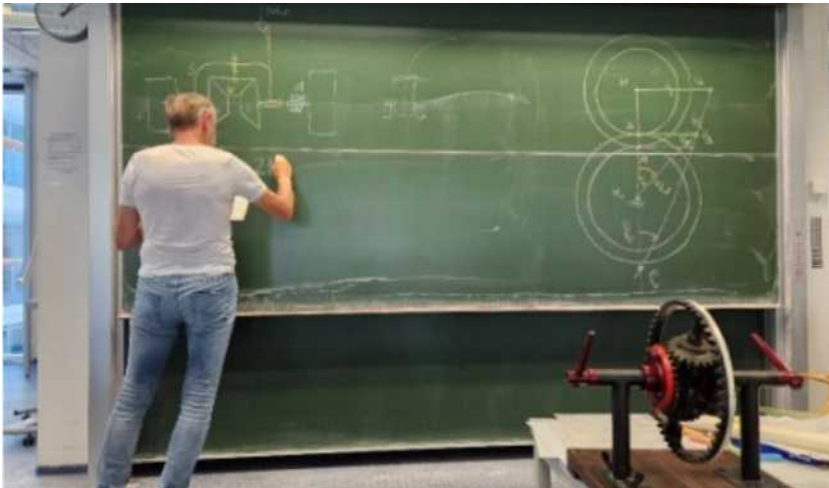


Abbildung 1: Dr. Franz Irlinger in der Getriebelehre Vorlesung. Oben 1989. Unten 2024.

Inhaltsverzeichnis

Digitale Werkzeugkette zur leichtbauoptimierten Auslegung von schnelllaufenden, ebenen Mechanismen	1
Nils Brückmann; Burkhard Corves; Mathias Hüsing; Kai-Uwe Schröder; Raphael Cleven	
Auslegung eines Wellgetriebes	5
Simeon Wolf; Vincent Brünjes; Mathias Hüsing; Burkhard Corves	
Nützt uns die KI beim Mechanismenentwurf?	17
Franz Irlinger	
Automatische Konstruktion eines Viergelenkgetriebes für ein Klappmesser mithilfe der SG-Library	29
Markus Huber; Franz Irlinger; Tim Lüth	
Automatisches Folgen von Personen und Gestensteuerung für einen mobilen Roboter mithilfe von bildbasierter menschlicher Posenerkennung	41
Dingzhi Zhang; Franz Irlinger, Tim C. Lüth	
Vier Wege zum Ball'schen Punkt	51
Stefan Gössner	
Untersuchung von Methoden zur Synthese von Pick-and-Place Bewegungen	65
Georg Steinert; Jens-Peter Majschak	
Normierte Doppel-S-Profile für die Bewegungsaufgaben der VDI-Richtlinie 2143	79
Deborah Dietterle; Burkhard Alpers	
Parameterstudie zur gezielten Rekonfiguration nachgiebiger Mechanismen	91
Vivien Plat; Lena Zentner	

Die Software Mechanism Developer - nun verfügbar für Lehre und Industrie	93
Vincent Brünjes; Mathias Hüsing; Burkhard Corves	
Transversalsymmetrische Gelenke im Berechnungstool “CoMSys”	97
Hannes Jahn; Lena Zentner	
Einordnung der IKBT und Anwendung mit Blender – eine Methode zur Generierung der analytischen IK serieller Roboter	99
Jascha Paris	
Entwurf eines Koppelgetriebes für die stufenlose Höhenverstellung einer landwirtschaftlichen Stelzradzugmaschine	115
Robert Konradt, Stephan Schmidt, Sebastian Frasch, Andreas Scholz	
Entwicklung eines Drahtfördersystems für Drähte mit sehr kleinem Durchmesser	127
Pascal Schnabel; Holger Letsch; Maik Berger	
Erzeugung von Polygonprofilen für Welle-Nabe-Verbindungen	141
Uwe Bäsel	
Auslegung eines monolithischen Positioniermechanismus als Fünfgelenk mit Abwurfmechanismus	153
Lucas Artmann; Vitus Hensgens; Tim C. Lüth	
Entwurf von topologieoptimierten nachgiebigen Beinen für vierbeinige Roboter	167
Yilun Sun, Tim C. Lüth	
Validierung einer energieeffizienten Antriebsmethode von Kurvengetrieben an einem Prüfstand	169
Thomas Knobloch; Lukas Kaboth; Rufat Asgarzade; Mathias Hüsing; Burkhard Corves	
VR-gestützte interaktive Ansteuerung einer Montagezelle mit Delta-Endeffektor	181
Akhilraj Anilkumar; Claas Duarte Nanninga; Mike Männel; Maik Berger	
2D “TenSens”-Sensor zur simultanen Messung von Kräften und Momenten	193
Lukas Merker; Thomas Kissinger; Valter Böhm; David Herrmann; Lena Zentner	

Digitale Werkzeugkette zur leichtbauoptimierten Auslegung von schnelllaufenden, ebenen Mechanismen

Nils Brückmann*; Burkhard Corves*; Mathias Hüsing*; Kai-Uwe Schröder**; Raphael Cleven*

* RWTH Aachen University, Institut für Getriebetechnik, Maschinendynamik und Robotik
brueckmann@igmr.rwth-aachen.de

** RWTH Aachen University, Institut für Strukturmechanik und Leichtbau

Kurzfassung

Im Rahmen dieses Beitrags soll die Idee einer digitalen Werkzeugkette vorgestellt werden, die die Methoden der Mechanismensynthese, Belastungsanalyse und Strukturoptimierung modellbasiert miteinander verbindet und es somit ermöglicht, in einer signifikant kürzeren Entwicklungszeit dynamisch leistungsfähigere bzw. ressourceneffizientere ebene Mechanismen zu entwickeln.

Title / Abstract

Digital Tool Chain for Lightweight-optimized Design of High-speed, Planar Mechanisms.

This contribution presents the idea of a digital tool chain that combines the methods of mechanics synthesis, load analysis and structural optimization in a model-based approach and thus enables the development of dynamically more powerful and resource-efficient planar mechanisms in a significantly shorter development time.

1 Erweiterte Kurzfassung

Ebene Mechanismen finden im Maschinen- und Anlagenbau als Führungs- und Übertragungsgetriebe für wiederkehrende, komplexe und hochdynamische Bewegungsabläufe weit verbreitete Anwendung. Im Vergleich zu Robotern sind sie im Fall einer gleichbleibenden, zyklischen Bewegung durch die geringere Anzahl an Antrieben robuster, kostengünstiger, energieeffizienter und leistungsfähiger. Die Maschinen sind in der Regel produktspezifisch und aufgrund des hohen Entwicklungsaufwands investitionsintensiv. Für solche Bewegungseinrichtungen hat sich ein überwiegend manueller Entwicklungsprozess etabliert, der das Problem in Teilprobleme zerlegt, diese nacheinander löst und die Wechselbeziehungen zwischen den Lösungen durch manuelle Iterationen berücksichtigt. Im Rahmen dieses Beitrags soll die Idee einer digitalen Werkzeugkette vorgestellt werden, die die Methoden der Mechanismensynthese, Belastungsanalyse und Strukturoptimierung modellbasiert miteinander verbindet und es somit ermöglicht, in einer signifikant kürzeren Entwicklungszeit dynamisch leistungsfähigere bzw. ressourceneffizientere ebene Mechanismen zu entwickeln.

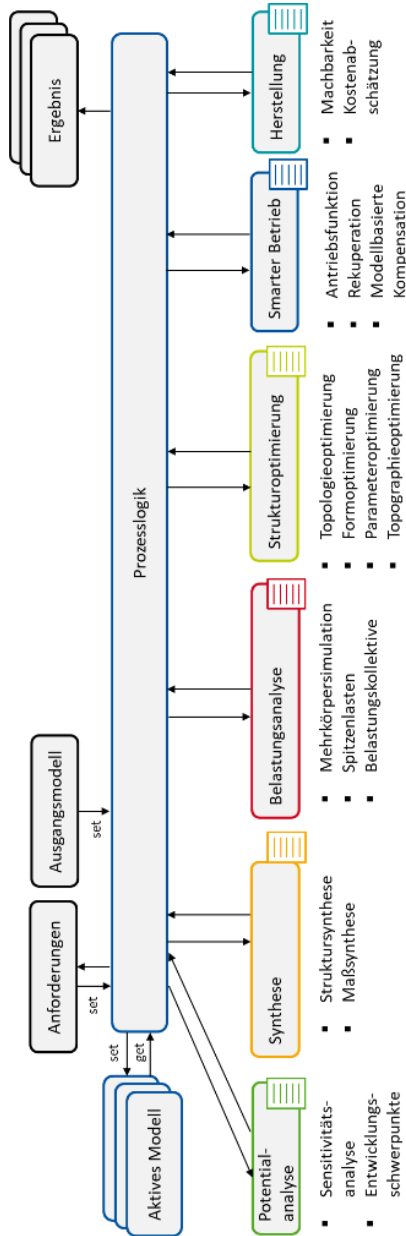


Abb. 1: Digitale Werkzeugkette mit Funktionsbausteinen

Auf Basis der in Abb. 1 dargestellten Prozesskette ergeben sich folgende Funktionsbausteine:

1. Der *Digitale Prozessmanager* ist in der Lage, durch die iterative, meta-heuristische Orchestrierung der Funktionsbausteine automatisch die Komponentengeometrie für verschiedene Leichtbauvarianten eines ebenen Mechanismus zu erzeugen.
2. Die *Potentialanalyse* identifiziert auf Basis der individuellen Eigenschaften des Mechanismus die sinnvollen Kombinationen von Methoden und Komponenten für den Leichtbau.
3. Die *Synthese* maximiert das Leichtbaupotential unter Einhaltung der Anforderungen durch Festlegung der Struktur und der kinematischen Abmessungen des Mechanismus.
4. Die *Belastungsanalyse* ermittelt durch Mehrkörpersimulation (MKS) automatisch und allgemeingültig die Betriebslasten sowie die Beanspruchungen aller Komponenten.
5. Die *Strukturoptimierung* minimiert unter Einhaltung der Anforderungen das Komponentengewicht durch den adaptiven Einsatz von Topologie-, Topographie-, Parameter- und Formoptimierungsmethoden durch Anpassung der Komponentengeometrie.
6. Der *Smarte Betrieb* optimiert die Antriebsfunktion auf die energieminimale bzw. leistungsmaximale Funktionserfüllung mit den Optionen der Rekuperation bzw. der regelungstechnischen Kompensation von Nachgiebigkeiten.
7. Die *Herstellung* trifft eine Kostenabschätzung auf Basis der unternehmensspezifischen Kosten für eingesetzte Materialien und Fertigungsverfahren.

Auslegung eines Wellgetriebes

Simeon Wolf*; Vincent Brünjes*; Mathias Hüsing*; Burkhard Corves*

*RWTH Aachen University, Institut für Getriebetechnik, Maschinendynamik und Robotik
wolf@igmr.rwth-aachen.de

Kurzfassung

Wellgetriebe (auch Harmonic Drives genannt) sind gleichförmig hochübersetzende, koaxiale Zahnradgetriebe in denen ein flexibles, im elastischen Bereich verformtes, Zahnrad verbaut ist. Sie finden ihre Anwendung in Bereichen, die hohe Genauigkeit und Steifigkeit bei gleichzeitig geringem Gewicht und kleinen Bauraum erfordern. Der folgende Beitrag beschreibt eine softwaregestützte Auslegung dieser Getriebe. Basierend auf der Getriebekinematik wird dabei die konjugierte Verzahnung der Getriebebauteile generiert.

Title / Abstract

Design of a strain wave gearing

Harmonic Drives are uniformly high-reduction, coaxial gearboxes in which a flexible, elastically deformed gear is installed. They are used in applications where gears with high accuracy and stiffness, combined with low weight and installation space, are required. The following article outlines a software-based approach for designing these gearboxes, which involves generating the conjugate gearing of the gear components based on their kinematics.

1 Einführung

Wellgetriebe bestehen aus den drei Bauteilen *Wavegenerator* (WG), *Circular Spline* (CS) und *Flexspline* (FS), siehe Abb. 1. Sie lassen sich als einstufiges, offenes Umlaufräder- bzw. Planetengetriebe einordnen, wobei der Wavegenerator dem Steg, der Flexspline dem Planeten und der Circular Spline dem Hohlrad entspricht [1, S. 85]. Die Drehzahl des Flexsplines wird mechanisch auf eine Zentralwelle FS_Z zurückgeführt. Dies kann entweder durch einen nach innen oder außen gewölbten Flansch am Flexspline (Becher- oder Hut-Bauform) oder durch ein zweites Hohlrad, den *Dynamic Spline* (DS), erfolgen (Ring-Bauform).

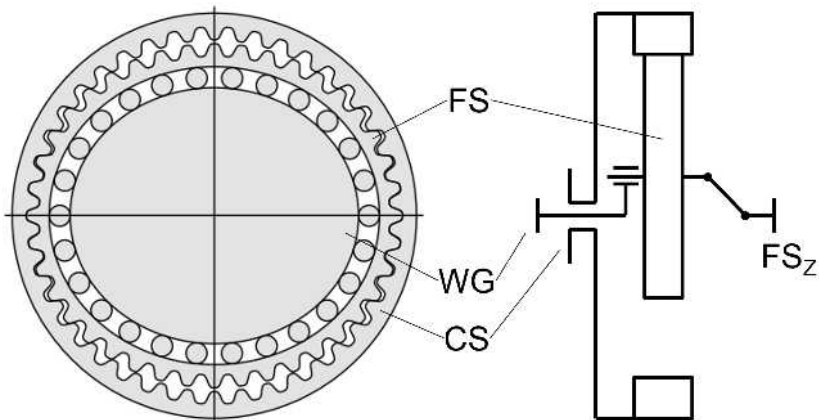


Abb. 1: Funktionsprinzip des Wellgetriebes

Der Antrieb des Getriebes erfolgt standardmäßig am Wavegenerator, einem ovaler Körper, um den der Flexspline, ein dünner flexibler außenverzahnter Ring, umläuft. Der Flexspline wird ständig durch die ovale Kontur des Wavegenerators elastisch verformt und verzahnt an den zwei gegenüberliegenden Scheitelpunkten mit dem Circular Spline, einem starren Hohlrad, das standardmäßig zwei Zähne mehr als der Flexspline besitzt. Bei Drehung des Wavegenerators wandern die Zahneingriffsbereiche mit, sodass der Flexspline pro Umdrehung des Wavegenerators um zwei Zähne im Circular

Spline weiterwandert. Das Übersetzungsverhältnis i (vom Wavegenerator zum Flexspline) entspricht dabei der halben Zahnzahl des Flexsplines z_{FS} :

$$i_{WG \rightarrow FS} = \frac{z_{FS}}{\Delta z} = \frac{z_{FS}}{2} < 0 \quad (1)$$

2 Kinematik

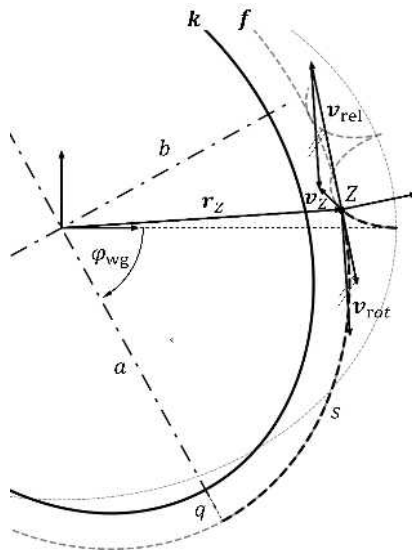


Abb. 2: Kinematik

Bei der Bewegung des Wellgetriebes läuft der Flexring entgegengesetzt um den sich drehenden Wavegenerator, siehe Abb. 2. Zur Beschreibung dieser Bewegung werden zunächst einige Vereinfachungen und Annahmen getroffen [2], [3, S. 139]:

- Die Verformung und Kinematik des Flexsplines wird vereinfacht nur zweidimensional beschrieben.
- Ein einzelner Zahnkörper des Flexsplines wird als Starrkörper betrachtet.

- In der Mitte des Zahnkranzes des Flexsplines wird von einer neutralen Faser f ohne Zug- und Druckspannungen mit konstantem Umfang U_f ausgegangen.

Im idealen spielfreien Fall hat jeder Punkt der neutralen Faser den gleichen Abstand q zur Kontur k des Wavegenerators [4, S. 4]. Für den Umfang des Wavegenerators U_k gilt somit:

$$U_k = U_f - 2\pi q \quad (2)$$

Die Ortskurve eines Punkts Z der neutralen Faser r_z in Abhängigkeit zum Antriebswinkel φ_{wg} des Wavegenerators wird durch die Integration der Absolutgeschwindigkeit v_z erhalten, die sich aus zwei Geschwindigkeitskomponenten zusammensetzt:

1. Rotationsgeschwindigkeit v_{rot} durch Drehung des Wavegenerators:

$$v_{rot} = \omega_{wg} \times r_z \quad (3)$$

2. Relativgeschwindigkeit v_{rel} durch Umlaufen des Flexsplines um den Wavegenerator:

$$v_{rel} = v_{rel} \mathbf{t} \quad (4)$$

Die Relativgeschwindigkeit zeigt in Richtung der Einheitstangente \mathbf{t} der neutralen Faser mit dem konstanten Betrag $v_{rel} = \omega_{wg}(a + q)$.

$r_z(\varphi_{wg})$ wird durch numerische Integration der Differenzialgleichung

$$dr_z = \left[\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} r_z(\varphi_{wg}) + (a + q) \mathbf{t}(\varphi_{wg}) \right] d\varphi_{wg} \quad (5)$$

mit der Anfangsbedingung

$$r_z(\varphi_{wg} = 0) = \begin{pmatrix} a + q \\ 0 \end{pmatrix} \quad (6)$$

ermittelt. Die Einheitstangente der über den Kurvenparameter τ parametrisierten Kurve \mathbf{f} der neutralen Faser

$$\mathbf{t}(\varphi_{\text{wg}}) = \frac{d\mathbf{f}(\tau(\varphi_{\text{wg}}))}{d\tau} \quad (7)$$

wird über das Gleichsetzen des zurückgelegten Wegs s des Punkts auf der neutralen Faser mit der Bogenlänge der Faser ermittelt, sodass über

$$s(\varphi_{\text{wg}}) = (a + q)\varphi_{\text{wg}} = \int_0^\tau \left| \frac{d\mathbf{f}(\tau')}{d\tau'} \right| d\tau' \quad (8)$$

eine Beziehung für $\tau = \tau(\varphi_{\text{wg}})$ gegeben ist.

Die Definition des Übersetzungsverhältnisses besagt, dass ein Punkt der neutralen Faser nach i -Umdrehungen des Wavegenerators einmal im Getriebe umgelaufen sein muss. Der Punkt legt somit den Relativweg

$$s = (a + q)2\pi i \quad (9)$$

um den Wavegenerator zurück, wobei er sich dabei um $(i + 1)$ -mal um den Wavegenerator bewegt:

$$(a + q)2\pi i = (i + 1)U_{\text{nf}} \quad (10)$$

Die große Halbachse a des Wavegenerators muss somit passend zum Übersetzungsverhältnis i gewählt werden.

3 Generierung der konjugierten Zahnflanken

Da es sich bei der Wellgetriebeverzahnung nicht um eine klassisch wälzende Verzahnung zweier Starrkörper handelt, können gängige Verzahnungen (bspw. mit evolventischem Profil) inkl. deren Auslegungsmethoden nicht angewendet werden.

Die Grundlage einer gleichförmigen Drehübertragung stellt das *allgemeine Verzahnungsgesetz* dar [5, S. 11]. [6] leitet eine allgemeine Version des Verzahnungsgesetzes (notwendige Bedingung) her: Seien \mathbf{n}_{x_i} die äußeren Normalenvektoren zweier Zahnflanken in den Punkten \mathbf{x}_1 und \mathbf{x}_2 . Dann ist eine notwendige Bedingung für das Kontaktieren beider Zahnflanken, dass der Vektor der Relativbewegung beider Punkte $\dot{\mathbf{x}}_1 - \dot{\mathbf{x}}_2$ senkrecht zu \mathbf{n}_{x_1} und \mathbf{n}_{x_2} liegt. Für kontaktierende Punkte $\mathbf{x}_1 = \mathbf{x}_2$ gilt somit:

$$\mathbf{n}_{x_1} \cdot (\dot{\mathbf{x}}_1 - \dot{\mathbf{x}}_2) = 0 \quad (11)$$

Bei vorgegebener Flexspline-Zahnflanke und stillstehendem Circular Spline kann vereinfacht werden:

$$\mathbf{n}_{x_{fs}} \cdot \dot{\mathbf{x}}_{fs} = 0 \quad (12)$$

Über diese Bedingung lassen sich die Punkte der konjugierten Zahnflanke erzeugen, siehe Abb. 3.

In Abb. 4 sind die vollständigen, nach dem obigen Verfahren erzeugten, konjugierten Zahnflanken für eine vorgegebene Flanken­geometrie des Flexsplines dargestellt - links für die Zahnflanke des Circular Splines, rechts für die Flanke des Dynamic Splines. Außerdem ist der Verlauf der Flexspline Verzahnung für eine halbe Umdrehung des Wavegenerators dargestellt. Die Einhüllende dieser Kurvenschar entspricht der konjugierte Zahnflanke. Im ortsfesten Gehäuse-Koordinatensystem erhält man die konjugierte Zahnflanke des Circular Splines und der Flexspline-Zahn wandert zum nächsten Zahn. Im sich drehenden Abtriebs-Koordinatensystem zirkuliert der Flexspline-Zahn und man erhält die konjugierte Zahnflanke des Dynamic Splines.

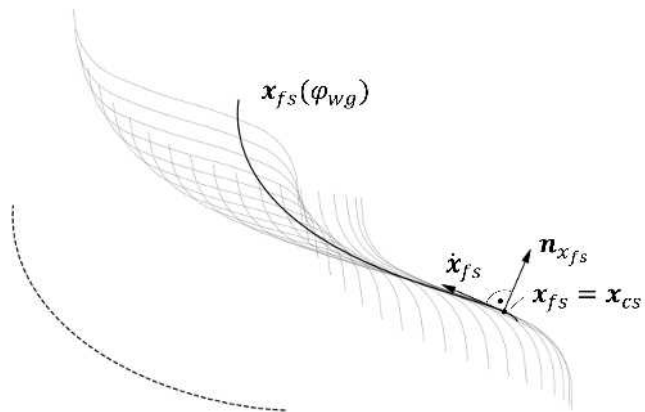


Abb. 3: Generierung der konjugierten Zahnflanke

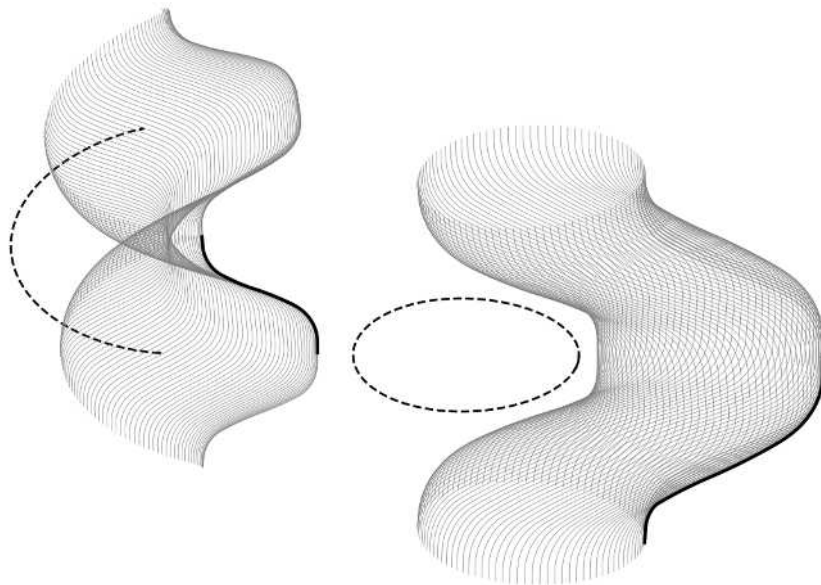


Abb. 4: Konjugierte Zahnflanken des Circular- und Dynamic Splines

4 Ergebnis und Ausblick

Aus den konjugierten Zahnflanken werden die entsprechenden Hohlräder des Wellgetriebes generiert. In Abb. 5 ist das Ergebnis der Auslegung für ein Wellgetriebe mit dem Übersetzungsverhältnis $i = 20$ dargestellt. Der Dynamic Spline ist mit gestrichelter Linie gezeichnet.

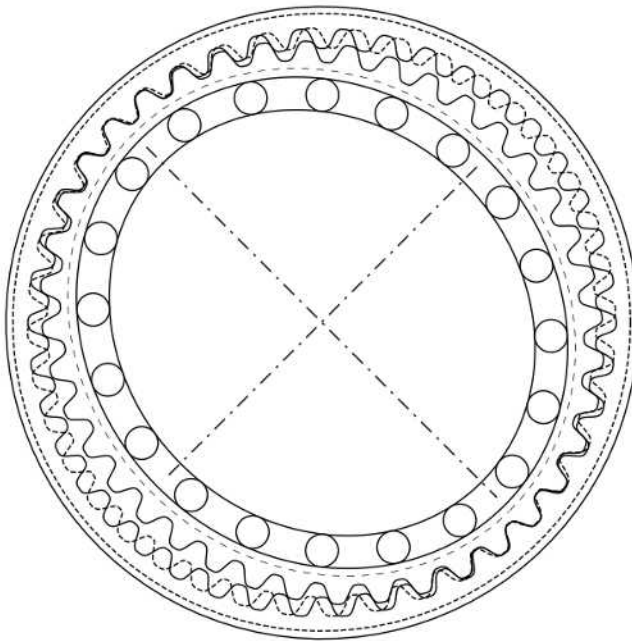


Abb. 5: Ergebnis der Getriebeauslegung

Das Programm ermöglicht die Getriebeparameter zu variieren und die Auswirkungen in Echtzeit graphisch darzustellen. In Abb. 6 wurde beispielhaft der Eingriffswinkel von 20° (links) zu 30° (rechts) variiert. Dies kann sinn-

voll sein, um die Kraftübertragung im Zahnkontakt, bspw. hin zu einem verbesserten Wirkungsgrad, zu optimieren.

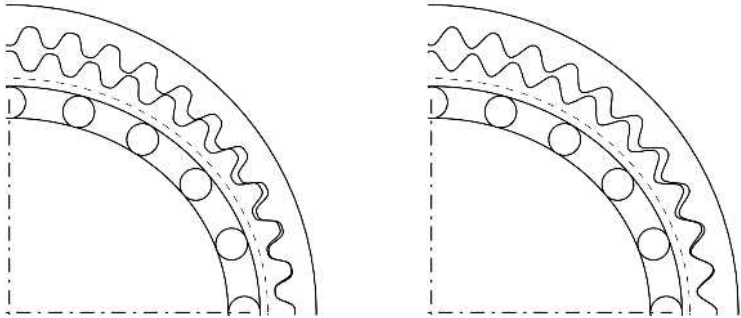


Abb. 6: Variation des Eingriffswinkels

In Abb. 7 wurde die Beschreibungsform der Kontur des Wavegenerators von einer Ellipse (links) zu einer, aus einem Kreisbogen im Scheitelpunkt und einem Polynom 5. Ordnung zusammengesetzten, Kontur geändert (rechts). Dadurch werden einerseits im Zahneingriffsbereich mehr Zähne in den Eingriff gebracht aber andererseits die Biegespannungen im Flexspline erhöht. Des weiteren ist bei der zusammengesetzten Kontur die kleine Halbachse des Ovals kleiner als bei der elliptischen Kontur, wodurch größere Zahnhöhen möglich sind.

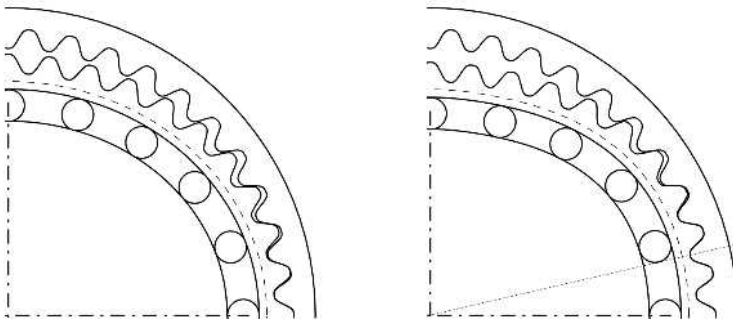


Abb. 7: Veränderung der Kontur des Wavegenerators

Das vorgestellte Berechnungsprogramm kann in den iterativen Auslegungsprozess eines Wellgetriebes eingebunden werden. Im Rahmen einer ersten Konzeptionierung lassen sich unterschiedliche Zahnflanken- und Wavegenerator-Profile vorgegeben und miteinander vergleichen. Diese sind dann prototypisch oder simulativ bspw. mit Finite Elemente Analysen hinsichtlich getriebespezifischer Optimierungsparameter, wie besonders der Torsionssteifigkeit sowie dem Anregungsverhalten bzw. dem Übersetzungsfehler, zu bewerten. Um eine Bewertung der Reibung im Zahnkontakt vorzunehmen kann das Berechnungsprogramm um die Ausgabe der Gleitgeschwindigkeit im Zahnkontakt erweitert werden. Des weiteren können Methoden entwickelt werden, die den Getriebebauteilen Toleranzen und Spiel, insbesondere an den Zahnflanken, hinzufügen.

Literatur

- [1] Müller, H. W. *Die Umlaufgetriebe. Auslegung und vielseitige Anwendungen*. Zweite neubearbeitete und erweiterte Auflage. Springer eBook Collection. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998. 1261 S. isbn: 9783642587252. doi: 10.1007/978-3-642-58725-2.

- [2] Gravagno, F., Mucino, V. H. und Pennestri, E. „Influence of wave generator profile on the pure kinematic error and centrodes of harmonic drive“. In: *Mechanism and Machine Theory* 104 (Okt. 2016), S. 100–117. issn: 0094-114X. doi: 10.1016/j.mechmachtheory.2016.05.005.
- [3] Dong, H., Zhang, J. und Wang, D. „A Tooth Profile Design Method for Harmonic Drive Without Tip Interference“. In: *Mechanism Design for Robotics*. Springer International Publishing, Aug. 2018, S. 138–146. isbn: 9783030003654. doi: 10.1007/978-3-030-00365-4_17.
- [4] Song, C. et al. „Parameter design of double-circular-arc tooth profile and its influence on meshing characteristics of harmonic drive“. In: *Mechanism and Machine Theory* 167 (2022), S. 104567. issn: 0094-114X. doi: <https://doi.org/10.1016/j.mechmachtheory.2021.104567>. url: <https://www.sciencedirect.com/science/article/pii/S0094114X21003116>.
- [5] Klocke, F. *Zahnrad- und Getriebetechnik. Auslegung - Herstellung - Untersuchung - Simulation*. Hrsg. von Christian Brecher. München: Carl Hanser Verlag, 2017. 668 S. isbn: 9783446430686.
- [6] Johann, A. und Scheurle, J. „On the generation of conjugate flanks for arbitrary gear geometries“. In: *GAMM-Mitteilungen* 32.1 (Mai 2009), S. 61–79. issn: 1522-2608. doi: 10.1002/gamm.200910005.

Nützt uns die KI beim Mechanismenentwurf?

Franz Irlinger

TU München, Lehrstuhl für Mikrotechnik und Medizingerätetechnik
irlinger@tum.de

Kurzfassung

Jahrzehnte lang geschah die Speicherung von Wissen mittels Bücher, Artikel, Anschauungsmodellen und durch Weitergabe in Form von Vorlesungen und Schulungen durch Experten. Zunehmend werden nun auch elektronische Medien genutzt. Aber neben der reinen Speicherung ist natürlich auch die Abrufbarkeit, Aufbereitung und Vermittlung immer wichtiger. Der Artikel versucht einen Überblick über die Aufgaben, Methoden und den aktuellen Stand der Wissensrepräsentation im Bereich des Entwurfs von Mechanismen zu geben und zu identifizieren in welchen Bereichen der Ingenieur einen Beitrag liefern könnte.

Does AI help us with mechanism design? / Abstract

For decades, knowledge was stored in books, articles, illustrative models and passed on in the form of lectures and training courses by experts. Electronic media are now increasingly being used. However, in addition to pure storage, retrievability, processing and communication are also becoming increasingly important. This article attempts to provide an overview of the tasks, methods and current status of knowledge representation in the field of mechanism design and to identify the areas in which engineers could make a contribution.

1 Umfeld

Automatisierte Handhabungsvorgänge spielen eine immer größer werdende Rolle auch außerhalb der klassischen, bisherigen Einsatzbereiche. So können im Bereich Alten- und Krankenpflege, Serviceaufgaben oder bei Katastropheneinsätzen aufgabenangepasste schnell zu fertigende Mechanismen an der Schnittstelle zur Robotik wertvolle Beiträge leisten. (Klauber et al. 2023)

2 Stand der Technik

Der Entwurf solcher Mechanismen setzt dabei eine umfangreiche Kenntnis kinematischer Grundlagen voraus, die sich aber durchaus in der Literatur, in Firmen und an Universitäten finden lassen. Die Zahl der auf diesem Gebiet ausgebildeten Ingenieure, welche in der Industrie erfolgreich mit derartigen konstruktiven Aufgaben betreut sind, ist dennoch gering und nimmt teilweise sogar ab. Zeitgleich stehen aber durch AI-Werkzeuge, basierend auf maschinellen Lerntechnologien, Versprechungen im Raum, dass dieses, bis in das vorletzte Jahrhundert zurückreichende Wissen, dem Konstrukteur aufbereitet zur Verfügung steht. Ist diese Euphorie begründet, oder naht auf diesem Gebiet der 3. Winter? (Lanquillon 2023)

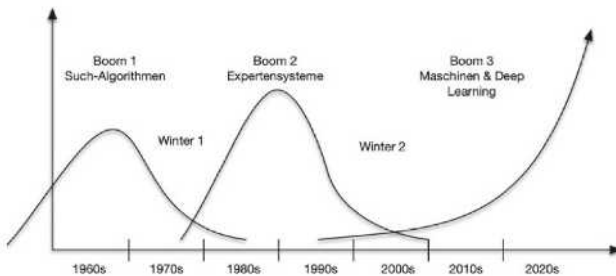


Abbildung 1: Zyklen in der KI (Kreutzer 2023)

Nachdem bereits in den 80er Jahren sehr viele erfolgversprechende Ansätze vorgestellt wurden (VDI Gesellschaft Entwicklung, Konstruktion, Vertrieb 1989) ist die Euphorie im Zusammenhang mit Expertensystemen in den letzten Jahren deutlich zurück gegangen (Braune 2021; Winkel). Hoffnungen werden zurzeit wieder durch AI-Werkzeuge wie ChatGPT von der Firma OpenAI geweckt. Aber auch die Nutzung von

ChatGPT erzeugt sehr unterschiedliche Antwortqualitäten. Dies wird deutlich, wenn man etwa eine Frage zu Ludwig Burmester stellt.

Ich:

Was waren die größten wissenschaftlichen Leistungen von Ludwig Burmester?

ChatGPT:

Ludwig Burmester war ein herausragender Wissenschaftler, dessen größte Beiträge im Bereich der kinematischen Geometrie liegen. Hier sind einige seiner wichtigsten wissenschaftlichen Leistungen:

Burmester-Theorie: Seine bekannteste wissenschaftliche Leistung ist die Entwicklung der Burmester-Theorie, die ein zentraler Bestandteil der kinematischen Synthese ist. Diese Theorie beschäftigt sich mit der Erstellung von Kopplungskurven, die durch die Bewegung von Verbindungsgliedern in Mechanismen entstehen. Burmester entwickelte Methoden zur genauen Bestimmung der geometrischen Orte (Burmester-Kurven), die von den Gelenkpunkten dieser Mechanismen durchlaufen werden können, um bestimmte gewünschte Bewegungsabläufe zu erreichen.

Beiträge zur Konstruktionsgeometrie: Burmester trug maßgeblich zur Entwicklung von Techniken bei, die in der technischen Zeichnung und im Maschinenbau verwendet werden, um komplexe Mechanismen zu konstruieren und zu analysieren. Seine Arbeiten erleichtern das Design von Mechanismen, die präzise Bewegungsabläufe ausführen müssen, wie sie beispielsweise in der Automobilindustrie und in der Robotik vorkommen.

Lehrbuch und Bildungseinfluss

Abbildung 2: Chat mit ChatGPT – beachtenswerte Aussagen grau markiert

Wie leicht zu erkennen ist, sind einige Aussagen hierbei definitiv falsch. Die sehr eloquente Formulierung des Sachverhalts täuscht dabei einen fachlich unerfahrenen Leser darüber hinweg, und vermittelt ein Gefühl von Wahrheit. Daher ist es zu dem jetzigen Zeitpunkt keine gute Idee den Aussagen von ChatCPT zu blind vertrauen.

Inspiziert durch die Möglichkeiten wurde am Lehrstuhl MiMed der Test unternommen, Prüfungen von ChatGPT lösen zu lassen. Dabei wurden beispielhaft drei Klausuren zugrunde gelegt. In der Klausur Mathematische Tools (Bachelor) erzielte ChatGPT eine 1,7. Die Aufgaben sind dabei meist so gehalten, dass zu technischen Problemstellungen eine Lösung in Form eines kleinen Matlab-Skripts gefunden werden soll. Diese Code-Schnipsel können auch im Internet gefunden werden, sodass eine eigentliche Problemlösung nicht notwendig ist.

Aufgabe 3 Numerische Methoden: Fast-Fourier-Transformation (2 Punkte)

Folgendes Matlab-Skript berechnet die Funktion $y = \cos(2 \cdot x) + \cos(x) + 0.5$ im Wertebereich von 0 bis 2π an 10 äquidistanten Stützstellen. Auf die sich ergebenden 10 Werte für y wird die Fast-Fourier-Transformation angewendet. Welche Ausgabe erwarten Sie? (Nur eine Antwort ist richtig)

```
x=0:2*pi/10:2*pi-1e-14;
y=cos(2*x)+cos(x)+0.5;
f=abs(fft(y))
```

f = [5 0 0 0 0 0 0 0 0 0]
 f = [5 0 0 0 0 0 0 0 0 5]
 f = [5 5 0 0 0 0 0 0 0 5]

f = [5 5 0 0 0 0 0 0 0 0]
 f = [5 5 0 0 0 0 0 5 5]
 f = [5 5 5 0 0 0 0 5 5]

f = [5 5 5 0 0 0 0 0 0]
 f = [5 0 0 0 0 0 0 0 0]

Abbildung 3a: Beispiel einer von ChatGPT gelösten Aufgabe aus der Klausur Mathematische Tools

Im Fach Kinematische Auslegung von Gelenkstrukturen mit Matlab und CAD fiel die Leistung auf 4,0 ab. So konnte ChatGPT nur bei bei Fragen mit reinem mathematischem Hintergrund punkten. Daher wurden in diesem Zusammenhang etwa Fragen zu den Eigenschaften von Rotationsmatrizen richtig beantwortet.

Aufgabe 1 Drehmatrizen (7 Punkte)

In dieser Aufgabe sind keine Begründungen oder Beweise erforderlich.

Gegeben sei eine räumliche Drehmatrix $R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \in SO(3)$

Das Ergebnis von $R^{-1} \cdot R^T$ ist:

$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$
 $\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$

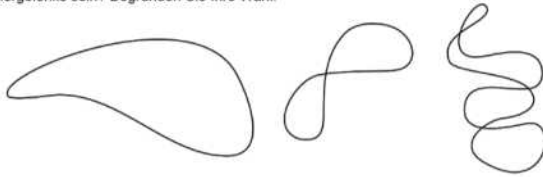
R
 $\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

Abbildung 3b: Beispiel einer von ChatGPT gelösten Aufgabe aus der Klausur Kinematische Auslegung von Gelenkstrukturen mit Matlab und CAD

Die Interpretation von Abbildungen ist für einen Chat-Bot zwar möglich aber die Zuordnung von kinematischen Konzepten oder Objekten gelingt nur sehr bedingt. Vor diesem Hintergrund war klar, dass ChatGPT in der Prüfung Bewegungstechnik nicht über die 5,0 hinauskommt. Hier sind sowohl die Angaben wie auch die geforderten Lösungen fast ausschließlich grafischer Natur.

Aufgabe 4:

- c) Welche der unten skizzierten Kurven kann keinesfalls die Koppelkurve eines Viergelenks sein? Begründen Sie Ihre Wahl!

**ChatGPT:**

Analyse der Kurvenform (rechte Kurve): Die Kurve zeigt eine komplexe, mehrschleifige Form mit geschwungenen Übergängen. Solche Kurven sind charakteristisch für die Koppelkurven von Viergelenkgetrieben, wenn der Punkt auf der Koppel, der die Kurve beschreibt, optimal positioniert ist. Allerdings sind bei Viergelenken bestimmte Aspekte zu beachten:

1. **Kontinuität und Geschlossenheit:** Viergelenkkurven sind meist kontinuierlich und geschlossen. Die hier gezeigte Kurve erscheint als kontinuierliche Bahn ohne sichtbare Diskontinuitäten oder abrupte Richtungsänderungen.
2. **Komplexität der Bewegung:** Die Kurve weist mehrere Schleifen und Windungen auf, was darauf hinweist, dass der Koppelpunkt eine relativ komplexe Bahn zurücklegt. Solche Formen sind möglich, wenn das Viergelenk speziell angeordnet ist, um komplexe Bewegungen zu erzeugen. Diese können durch unterschiedliche Längen und Anordnungen der Gelenke erreicht werden.
3. **Glattheit der Kurve:** Die Kurve zeigt einen glatten Verlauf ohne sichtbare Knicke oder Ecken, was typisch für Koppelkurven ist, die durch kontinuierliche Bewegungen eines mechanischen Getriebes erzeugt werden.

Schlussfolgerung:

Basierend auf der Form der Kurve und den Eigenschaften, die typischerweise von Koppelkurven eines Viergelenks erwartet werden, ist es wahrscheinlich, dass diese spezifische Kurve durch ein Viergelenk erzeugt werden kann. Es gibt keine offensichtlichen Merkmale in der Kurve, die darauf hindeuten würden, dass sie nicht von einem Viergelenk stammen könnte. Die Komplexität der Kurve deutet vielmehr darauf hin, dass das Viergelenk speziell für eine solche Bewegung konfiguriert sein könnte.

Falls weitere Kurvenbilder vorliegen und verglichen werden sollen, um festzustellen, welche definitiv nicht von einem Viergelenk stammen könnte, wäre das hilfreich für eine umfassendere Analyse.

Abbildung 3c: Beispiel einer von ChatGPT zu lösenden Aufgabe aus der Klausur Bewegungstechnik

Bei der Lösung von ChatGPT wird ein weiterer Punkt im Umgang mit Chat-Bots klar. Wenn dieser die Lösung nicht unmittelbar angeben kann, antwortet er in Allgemeinplätzen, die nicht falsch sind, sondern den Nutzer dazu bringt seine Frage zu präzisieren. Dies können natürlich nur Experten, die in diesem Fall Chat-GPT an die Lösung heranzuführen. Hier lernt nicht der Fragende, sondern der Bot. Die Antwort von ChatGPT ist erst nach längerer, hier nicht abgedruckter Diskussion entstanden, und trifft, wie wir alle wissen nicht ins Schwarze. (Koppelkurven sind Kurven 6. Ordnung und können mit einer Geraden nur maximal 6 Schnittpunkte aufweisen)

3 Nachteile des Stands der Technik

Obwohl auf dem Gebiet der Kinematik ein Jahrhunderte altes Wissen in Form von Büchern (Burmester 1888), Aufsätzen, existiert, ist die Verfügbarkeit und ein zuverlässiger Abruf bisher nicht möglich. Auch die digitalisierten Varianten von Lehrbüchern z.B. (Kerle et al. 2015) oder hervorragende Wissenssammlungen wie die dmg-Lib (DMG-Lib: Startseite 2024) sind für die Verarbeitung durch AI-Tools nicht ausreichend vorbereitet.

4 Eigener Ansatz

Der Autor will sich in diesem Zusammenhang nicht anmaßen hier einen tragfähigen Ansatz vorzustellen, diese Aufgabenstellung, nämlich das Auffinden und Verarbeiten von Wissen zu lösen. Dazu gibt es eine Vielzahl von Firmen und Forschungseinrichtungen, die auf diesem Gebiet seit Jahren forschen, aber auch Werkzeuge entwickeln. Vielmehr soll versucht werden das Problem einzuordnen und für das unbestreitbar vorhandene Wissen eine Repräsentation vorzuschlagen, die der automatischen Verarbeitung besser zugänglich ist. Dabei müssen aber die Konzepte auf denen die Generation neuerster KI-Werkzeuge basieren, bekannt sein. Der Vorschlag sollte dabei lediglich als Grundlage für eine jetzt dringend zu startende Entwicklung gesehen werden, um das Gebiet des Mechanismenentwurfs besser für den Zugriff von wissensverarbeitenden Werkzeugen vorzubereiten. Auch wenn Kinematikexperten keinen Beitrag zur Verbesserung von speziellen Problemen der KI-Entwicklung liefern können, so ist es aber doch Ihre Aufgabe, ihr

Wissensgebiet auf diesen Einsatz hin vorzubereiten und in die Diskussion mit Entwicklern zu gehen.

5 Erwartete Vorteile

Bereits in der Vergangenheit hat sich gezeigt, dass es unabdingbar ist Entwicklungen auf dem Gebiet der IT zusammen mit Wissenschaftlern der entsprechenden Domänen durchzuführen. So wird etwa durch die Erstellung einer eindeutigen Ontologie im Bereich der Kinematik mit zugeordneter grafischer Repräsentation von beteiligten Objekten das Wissen für Text- und Data-Mining besser zugänglich gemacht werden. Durch eine zu integrierende Regelbasis wird darüber hinaus die Wissensverarbeitung erleichtert.

6 Lösungsstruktur

Um Vorschläge in diesem Zusammenhang sinnvoll zu entwerfen, müssen zuerst die Begriffe im Zusammenhang mit AI-Projekten geklärt und hinreichend verstanden sein. Dazu existiert eine Unzahl an Büchern und Artikel z.B. (Russell and Norvig 2023). Aber alleine der Besitz der Bücher und die pdfs auf der Festplatte machen einem nicht zum Experten, aber: Wer gar nichts weiß, muss alles glauben! Daher hier erst mal ein kleiner Einblick in die Konzepte der AI. Aber am Anfang die Frage: Was ist KI und welche Aufgaben hat Sie, und wie löst sie diese. KI sollte intelligentes Verhalten zeigen, Das bedeutet Wissen zu erfassen, aufbereiten, abzuspeichern, wieder aufzufinden und aus diesem Wissen Schlüsse zu ziehen, die idealerweise jenseits der bereits vorgedachten Lösungen liegen. Gelingt dies, wird und darf es als Künstliche Intelligenz bezeichnet werden. Für den Benutzer einer solchen Intelligenz sind aber natürlich erst mal andere Formen der Intelligenz wie Kommunikationsfähigkeit augenfällig, dürfen aber von den eigentlichen Aufgaben nicht ablenken.

Arten von Intelligenz	Komplexität der Ansätze	Arten von Wissen	Lösungsvorgehen
Sprachliche/linguistische Logisch-mathematische Bildlich-räumliche Körperlich-kinästhetische Inter- und Intrapersonelle Kreative/schöpferische Existenzielle/spirituelle	Denken und Deduktion Lernen und Induktion Kreativität und Kognition Wahrnehmung Bewusstsein Selbstbewusstsein	Faktenwissen Methodenwissen Objektwissen Metawissen, Wissen über Modelle Wissen über Konzepte	Suchen Regelbasiert Nutzerinteraktion Einsatz von Agenten Neuronale Netze Machine Learning

Abbildung 4: Wordcloud im Zusammenhang mit AI

7 Lösungsprozess

Der zu durchlaufende Prozess beginnt mit der Bereitstellung des Wissens. Hier ist auf dem Bereich der Kinematik natürlich bereits ‚alles‘ vorhanden. Dennoch könnte neues Wissen bereits im Hinblick auf dessen maschinelle Verarbeitung entsprechend aufbereitet werden. Der folgende Schritt des Text Minings gefolgt vom Data Mining ist dann eine Domäne in der die Informatiker liefern müssen. Aber im nächsten Schritt nämlich dem Labeling der Daten und dem Extrahieren von Preloading Rules kann und sollte der Kinematiker wertvolle Beiträge liefern. Essentiell aber ist die Bereitstellung einer umfassenden Ontologie die ein Modell der Kinematischen Konzepte beschreibt. Darin sind beteiligte Objekte, Regeln, Methoden, Abläufe und deren Repräsentationen in grafischer Form enthalten. Diese Beschreibung hilft darüber hinaus eine Aufgabe in eine automatisiert lösbare Beschreibung zu überführen. In Bild 5 sind die Bereiche mit Beteiligung von Kinematikexperten grün hinterlegt. Die Lösung der Problemstellung ist dann in Form von Simulationsskripts, Konstruktionen oder bereits in Form von druckbaren stl-Dateien auszugeben. (Lueth and Irlinger 2022)

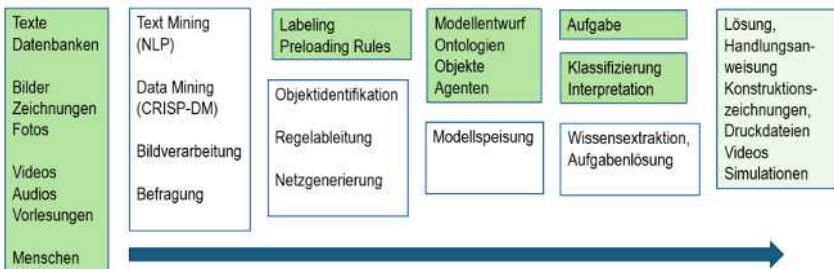


Abbildung 5: Von dem Wissen zur Lösung

8 Realisierung

Die Sammlung von verschiedenen Wissensformen ist auf dem Gebiet der Kinematik bereits weit vorgeschritten. Die wohl herausforderndste Aufgabe ist ein Gerüst für das gefundene Wissen zu erarbeiten. Ein erster sehr unvollständiger Ansatz wurde bereits in der Arbeit des Autors von 1985 vorgestellt. In diesem regelbasierten Ansatz wurden einige wenige Objektklassen definiert, die automatisiert Analyseaufgaben lösen konnten. (Irlinger 1993)

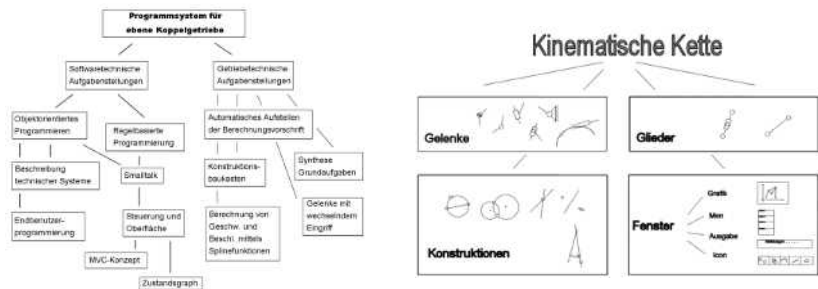


Abbildung 6: Aufgaben eines interaktiven ‚Expertensystems‘ mit einem ersten Versuch einer Ontologie auf dem Gebiet der Kinematik. (Irlinger 1993)

Wie so oft in der Vergangenheit wurde das System nicht weiterentwickelt und gepflegt. Zum damaligen Zeitpunkt war es immerhin der Nachweis, dass technische Sachverhalte und Zusammenhänge in einer objektorientierten Modellierung durchaus abbildbar sind und es sich für das Speichern von Expertenwissen eignet. Die damals gewählte Programmiersprache ‚Smalltalk‘ zwang zwar zu einer konsequent objektorientierten Modellierung und Programmierung, Lösungen und auch deren Konzepte konnten nur schwer in andere Programmiersprachen übernommen werden. Im Vordergrund standen aber auch die Realisierung weiterer Konzepte. So eignet sich diese Programmiersprache zum Aufbau einer Fachsprache für die Nennung von Zusammenhängen und vor allem für Lösungskonzepte. Dieser Ansatz mittels Fachsprachen erlangt gerade im Umfeld von KI eine zentrale Bedeutung.

Aber wie könnte eine Form der Wissensspeicherung im Zusammenhang mit dem Entwurf von Mechanismen aussehen? Am besten man fragt einfach ChatGPT was sie sich wünscht.

Ich:

Wie kann ich Informationen über Mechanismen abspeichern, sodass Du Sie optimal für die Weiterverarbeitung nutzen kannst?

ChatGPT:

..... Du kannst diese Informationen in einem Textdokument (.txt), einer Tabelle (z.B. .xlsx oder .csv) oder in einem strukturierten Dokumentenformat wie JSON oder XML abspeichern.

..... Ja, das Speichern von Ergebnissen objektorientierter Modellierung in einem strukturierten Format wie JSON ist sehr hilfreich, um Fragen zu beantworten. Solche strukturierten Daten können leicht analysiert und verarbeitet werden, was die Grundlage für präzise und schnelle Antworten bildet. Hier sind einige spezifische Vorteile und wie dies für verschiedene Fragen nützlich sein kann:

Vorteile von JSON für die Beantwortung von Fragen

Klar strukturierte Informationen: JSON ermöglicht es, komplexe Daten hierarchisch und organisiert darzustellen, was die Suche und Extraktion spezifischer Informationen erleichtert.

Maschinenlesbarkeit: JSON kann leicht von vielen Programmiersprachen und Tools gelesen und verarbeitet werden, was eine schnelle Antwortzeit ermöglicht.

Flexibilität: JSON erlaubt die Speicherung unterschiedlicher Datentypen und -strukturen, was die Darstellung verschiedener Modellierungsergebnisse unterstützt.

Hier ein Beispiel für JSON:

```
{
  "Titel": "Elektromagnetische Induktion",
  "Beschreibung": "Elektromagnetische Induktion beschreibt das Phänomen, bei dem ein veränderliches Magnetfeld eine elektrische Spannung in einem Leiter induziert.",
  "Kategorie": "Physik",
  "Tags": ["Faraday'sches Gesetz", "Magnetfeld", "Stromerzeugung"],
  "Funktionsweise": [ "Ein Magnetfeld ändert sich in der Nähe eines Leiters.", "Diese Änderung erzeugt eine elektrische Spannung im Leiter gemäß dem Faraday'schen Induktionsgesetz." ],
  "Formel": "\\( \\mathcal{E} = -\\frac{d\\Phi}{dt} \\)",
  "Beispiel": "Ein Generator wandelt mechanische Energie in elektrische Energie um durch Drehung einer Spule in einem Magnetfeld." ],
  "Diagramme und Abbildungen": [ "Diagramm 1: Darstellung eines Generators", "Abbildung 2: Magnetfeld und Induktionsspule" ],
  "Literaturverweise": [ "Introduction to Electrodynamics von David J. Griffiths", "Fundamentals of Physics von Halliday, Resnick und Walker" ]
}
```

Abbildung 7: Wissensrepräsentation in Form einer JSON-Datei

Wie so oft sollte man die Lösung von ChatGPT aber dennoch hinterfragen, oder zumindest nicht sofort beginnen mit diesem Ansatz zu arbeiten. Dieses Beispiel zeigt nämlich einen Ansatz zur Speicherung funktionsorientierter Zusammenhänge. Vielversprechender wäre eine ähnliche klare Struktur für die Speicherung von objektorientierten

Ansätzen zu verfolgen. Dabei müsste man sich nur auf die Eigenschaften von Klassen und den daraus abgeleiteten Objekten stützen.

9 Zusammenfassung

In Zeiten der Wissensverarbeitung durch Chatbots zeigt sich ein deutlicher Unterschied der Leistungsfähigkeit solcher KI-Werkzeuge je nach Domäne. Im Bereich Faktenwissen und mathematischen Methoden erreichen Chatbots beeindruckende Leistungen. Grund ist die formalisierte Beschreibung der notwendigen Gesetzmäßigkeiten und das hohe Sprachverständnis der Tools. Nicht zu vergessen ist aber auch, dass viele der Antworten nicht das Ergebnis logischer Schlüsse der KI sind, sondern Fundstellen welche im Internet verfügbar sind.

Der Siegeszug solcher Werkzeuge sollte aber nicht alleine den Informatikern überlassen werden. Die Kinematiker können und müssen dafür sorgen, dass auch Ihr Wissensgebiet formalisiert aufgearbeitet wird und genormte Beschreibungen in Form von Sprache, mathematischen Modellen und Skizzen entstehen. Das kann und wird dafür sorgen, dass kinematische Konzepte wieder mehr Einzug in technischen Lösungen finden.

Das Wissen ist da, es muss nur für eine größere Zahl von Ingenieuren nutzbar gemacht werden. Lassen Sie uns auf einen zweiten Frühling bei den Expertensystemen hoffen!

10 Literatur

Braune, Reinhard (2021): Genaulagen-Synthese von ebenen Koppelgetrieben mit aufgabenspezifisch konzipierten Bearbeitungsstrategien. Berlin, Heidelberg: Springer Berlin Heidelberg.

Burmester, Ludwig (1888): Lehrbuch der Kinematik. Die ebene Bewegung. Leipzig: Verlag von Arthur Felix.

DMG-Lib: Startseite (2024). Available online at <https://www.dmg-lib.org/dmglib/main/portal.jsp>, updated on 4/30/2024, checked on 4/30/2024.

Irlinger, Franz (1993): Objektorientierte Programmierung in der Getriebetechnik. Franz Irlinger. Als Ms. gedr. Düsseldorf: VDI-Verl.

Kerle, Hanfried; Corves, Burkhard; Hüsing, Mathias (2015): Getriebe-technik. Grundlagen, Entwicklung und Anwendung ungleichmäßig über- setzender Getriebe. 5., überarbeitete und erweiterte Auflage. Wies- baden: Springer Vieweg (Lehrbuch).

Klauber, Jürgen; Wasem, Jürgen; Beivers, Andreas; Mostert, Carina (Eds.) (2023): Krankenhaus-Report 2023. Personal. Springer-Verlag GmbH. 1. Auflage 2023. Berlin: Springer Berlin; Springer (Krankenhaus- Report, 2023).

Kreutzer, Ralf T. (Ed.) (2023): Künstliche Intelligenz Verstehen. Grund- lagen - Use-Cases - Unternehmenseigene KI-Journey. 2nd ed. Wies- baden: Springer Fachmedien Wiesbaden GmbH.

LANQUILLON, CARSTEN. SCHACHT SIGURD (2023): Knowledge Science - Grundlagen. Methoden der kunstlichen intelligenz fur die Wissens Extraktion aus Texten. [S.I.]: Springer Vieweg.

Lueth, Tim; Irlinger, Franz (2022): A Method for the Automatic Design of Mechanisms on the Example of a Four-Bar Linkage. In Andrés Kecskeméthy, Vincenzo Parenti-Castelli (Eds.): ROMANSY 24 - Robot Design, Dynamics and Control. Proceedings of the 24th CISM IFToMM Symposium. Symposium on Robot Design, Dynamics and Control. 1st ed. 2022. Cham: Springer International Publishing; Springer (CISM International Centre for Mechanical Sciences, Courses and Lectures, 606), pp. 12–36.

Russell, Stuart J.; Norvig, Peter (2023): Künstliche Intelligenz. Ein moderner Ansatz. 4., aktualisierte Auflage. München: Pearson Studium ein Imprint von Pearson Deutschland; Pearson Studium (Pearson Studium - IT).

VDI Gesellschaft Entwicklung, Konstruktion, Vertrieb (1989): Experten- systeme in Entwicklung und Konstruktion. Bestandsaufnahme und Ent- wicklungsrichtungen. Düsseldorf: VDI Verlag (VDI Berichte, 775).

Winkel, Olaf: Brauchen wir einen konzeptionellen Neuaufbruch im Wissensmanagement?, checked on 4/28/2024.

Automatische Konstruktion eines Viergelenkgetriebes für ein Klappmesser mithilfe der SG-Library

Markus Huber; Franz Irlinger; Tim Lüth

Technische Universität München
Lehrstuhl für Mikrotechnik und Medizingerätetechnik
markus.m.huber@tum.de

Kurzfassung

Im folgenden Beitrag soll gezeigt werden, wie die automatischen Konstruktionsfunktionen der Matlab-Bibliothek SG-Library genutzt werden können, um ein Viergelenkgetriebe zu Entwerfen und zu Konstruieren. Als Anwendungsbeispiel wird hierfür ein Klapptaschenmesser verwendet. Ausgangslage für die automatische Konstruktion ist die gewünschte 2D-Kontur der Griffschale und die 2D-Klingenkotur. Diese beiden Konturen sind notwendig, um passende Gestellpunkte auf der Griffschale und Anlenkpunkte auf der Klinge zu finden. Ebenfalls werden für die Klinge zwei Posen vorgegeben, welche das Viergelenk exakt erreichen soll. Als letzte Vorgabe wird der Funktion ein Bereich übergeben, welchen die Klingenkotur nicht berühren darf, um die Sicherheit der Finger des Benutzers sicherzustellen und den Lösungsraum einzuschränken. Die Software rastert nun den Lösungsraum, berechnet alle möglichen Kombinationen und filtert diese auf ihre Tauglichkeit für den Anwendungsfall. Anschließend kann der Nutzer sich Animationen der Lösungen ansehen und seine präferierte Lösung auswählen. Da nicht alle Komponenten des Viergelenks in der selben Ebene liegen können, wird im nächsten Schritt die Schichtung der Bauteile festgelegt. Nun können alle Bauteile

kollisionsfrei auskonstruiert werden und die Bauteile als STL-Dateien abgespeichert werden. Abschließend kann diese mechanische Konstruktion für den Anwendungsfall und das Fertigungsverfahren optimiert werden und in die Griffschalen des Messers eingebaut werden.

Title / Abstract

Automatic Construction of a Four-Bar Linkage for a Folding Knife using the SG-Library

The following article will show how the automatic design functions of the Matlab library SG-Library can be used to design and construct a four-bar linkage. A folding pocket knife is used as an application example. The starting point for the automatic design is the desired 2D contour of the handle shell and the 2D blade contour. These two contours are necessary to find suitable frame points on the handle scales and pivot points on the blade. Two poses are also specified for the blade, which the four-bar linkage should achieve exactly. Finally, the function is given an area that the blade contour must not touch in order to limit the solution space and ensure the safety of the user's fingers. The software now grids the solution space and calculates all possible combinations and filters them for their suitability for the application. The user can then view animations of the solutions and select their preferred solution. As not all components of the four-bar linkage can work in the same plane, the layering of the components is defined in the next step. Now all components can be designed collision-free and the components can be saved as STL files. Finally, this mechanical design can be optimized for the application and the manufacturing process and installed in the handle scales of the knife.

1 Anwendung

Koppelgetriebe waren früher aus der Ausbildung zum Maschinenbauingenieur nicht wegzudenken. Gerade in Zeiten, in welchen eine Maschine noch eine Königswelle besaß, welche alle anderen Komponenten versorgt hat. Durch sinkende Preise von Servomotoren und vereinfachter

Ansteuerung ist die Verbreitung von Koppelgetrieben deutlich zurückgegangen. Im Zuge dessen hat auch die Getriebetechnik einen kleineren Stellenwert in der Ausbildung neuer Ingenieure eingenommen. Heutzutage werden jedoch wieder vermehrt Mechanismen und Koppelgetriebe benötigt, da hohe Taktraten, Anforderungen an die Energieeffizienz und Motorkosten die Verwendung von vielen einzelnen Motoren wieder unattraktiver macht.

Leider fehlt nun jedoch vielen Maschinenbauingenieuren das Wissen, wie Koppelgetriebe synthetisiert werden können. Dieses wird versucht durch Softwarelösungen auszugleichen. Eine solche Software, welche einen Ingenieur bei der Synthese und Analyse von unter anderem Gelenkgetrieben unterstützt, ist beispielsweise der Mechanism Developer MechDev [1].

Als Ergebnisse für eine derartige softwaregestützte Auslegung bekommt der Nutzer meist die kinematischen Abmessungen des Getriebes. Dieses Getriebe muss dann jedoch weiterhin in einer CAD Software konstruiert werden. Das beinhaltet Schritte wie beispielsweise die Schichtung der Bauteile festzulegen, um Kollisionen der Gelenke zu vermeiden oder Endanschläge zu konstruieren. Dieser manuelle Arbeitsaufwand verhindert in viele Entwicklungsiterationen kurzer Zeit zu durchlaufen. Eine Software, welche dem Nutzer nicht nur bei der Synthese der kinematischen Abmessungen unterstützt, sondern auch die Konstruktion und Schichtung der Teile übernimmt, ist die Solid Geometry Toolbox (SG-Lib)[2–4] für Matlab.

2 Aufgabenstellung für Konstruktion

Im folgenden Beitrag soll gezeigt werden, wie die automatischen Konstruktionsfunktionen der SG-Library genutzt werden können, um ein Viergelenkgetriebe zu entwerfen und zu konstruieren.

Als Anwendungsbeispiel wurde hierfür ein Taschenmesser nach dem Beispiel des CRKT Provoke[®] gewählt, welches man in Abbildung 1 sehen kann.

An diesem Messer sieht man sogleich auch die Herausforderung, welche beim Konstruieren des Viergelenks entsteht. Die im Griff abgelenkten



Abb. 1: CRKT Provoke® Viergelenksmesser

Glieder müssen stark gekrümmt werden, wenn diese in der selben Ebene ohne Kollision bewegt werden sollen.

3 Umsetzung

Im folgenden Abschnitt wird aufgezeigt, wie mittels der SG-Library ein solches Viergelenksmesser entworfen werden kann. Anfangs wird auf die Angaben eingegangen, welche notwendig sind, um die Aufgaben für die SG-Library zu beschreiben. Anschließend wird gezeigt, wie die Maßsynthese der Kinematik durchgeführt werden kann. Danach wird die Konstruktion durchgeführt und abschließend werden die Anpassungen beschrieben, die noch manuell durchgeführt wurden.

3.1 Vorgaben

Damit die SG-Library Lösungen für eine Aufgabe erstellen kann, muss eine Aufgabe auf eine Art und Weise beschrieben werden, welche die Bibliothek versteht. Hierfür werden die Außenkonturen der für die Aufgabe relevanten Körper, wie beispielsweise die Klinge oder der Griff, als sogenannte CPL (Closed Point List) übergeben. Hierbei handelt es sich um eine Liste von Punkten, welche insgesamt eine geschlossene Kontur ergibt. Da die Konstruktion auf Basis der Klinge eines bestehenden Messers erstellt werden soll, wird ein Foto (siehe Abbildung 2) von diesem als Basis verwendet.

Mit dem folgenden Befehl kann ein Bild geladen, zugeschnitten und auf echte Größe skaliert werden.



Abb. 2: Spendermesser für die Klinge des Viergelenkmessers

```
img = imreadui(); % Laden der Messerklinge
```

Auf Basis dieses Bildes kann, mit den nachfolgenden Befehl, die Kontur der Klinge bzw. des Endeffektors durch Klicken und Ziehen von Punkten bestimmt werden. Diese Kontur wird dann unter dem Namen CPLM in ein Strukturarray (struct), welches alle der Informationen für die Problemstellung enthalten wird, abgespeichert.

```
PS.CPLE = CPLui(); % Kontur von Klinge
```

Dieser Prozess wird außerdem für Kontur des Griffes bzw. des Gestelles wiederholt. Zusätzlich wird auf die selbe Weise die Kontur einer Sperrfläche festgelegt, welche der Endeffektor nicht betreten darf. Diese Verbotzone wird genutzt, um sicherzustellen, dass die Klinge des Messers nie in den Bereich der Finger auf der Unterseite des Messers gelangen kann.

```
PS.CPLM = CPLui(); % Kontur des Griffes
PS.CPLW = CPLui(); % Verbotzone
```

Abschließend müssen für die Aufgabedefinition noch die zu erreichenden Posen des Endeffektors beschrieben werden. Am Beispiel des Viergelenkmessers geben wir zwei Posen vor. Die eine Pose ist die Klinge im ausgefahrenen Zustand und die andere Pose im einklappten Zustand. Mit der nachstehenden Funktion können die Posen ebenfalls durch einfachen Klicken von zwei Punkten festgelegt werden. Definiert werden die Posen dann jeweils durch einen Aufpunkt „A“ und die Richtung, welche durch den Punkt „B“ festgelegt ist.

```
Poses = Poseui(2);
PS.A = Poses.A; PS.B = Poses.B;
```

Da damit alle Vorgaben definiert sind, können die vorgegebenen Bereiche sowie die Posen angezeigt werden. Dies erzeugt den Plot, welcher in Abbildung 3 zu sehen ist.

```
SGfigure;  
Poseplot space(PS);  
Poseplot(PS);
```

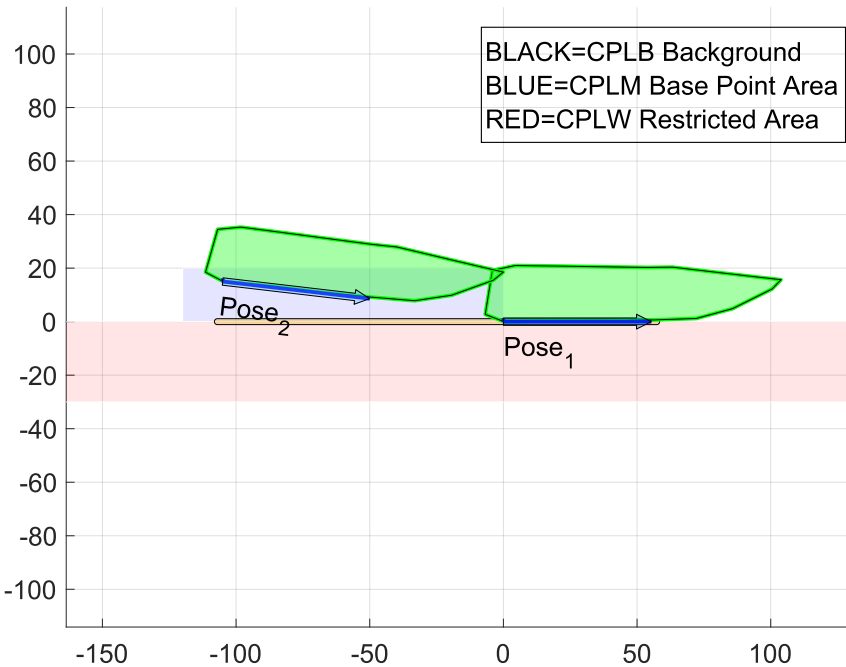


Abb. 3: Vorgaben für die automatische Konstruktion

Somit sind alle Vorgaben vorhanden, welche die SG-Library für die Maßsynthese des Viergelenks benötigt.

3.2 Kinematische Maßsynthese

Die Maßsynthese des Viergelenks kann am Beispiel des Viergelenkmessers mit dem nachfolgenden Befehl gestartet werden.

```
ps_sol = fourBarposesyntheses(PS,[1 2],...
    20,[1 2],[1 2],...
    'break coll CPL0 CPLW');
```

Als ersten Parameter wird an die Funktion das PS struct übergeben, in dem die Aufgabenstellung definiert ist.

Mit dem zweiten Parameter wird der Funktion gesagt, welche Posen sie verwenden soll. Da in diesem Beispiel nur zwei Posen definiert wurden, werden hier nur die Nummern der beiden Posen 1 und 2 übergeben.

Der dritte Parameter der Funktion gibt an, mit wie vielen Punkten der Endeffektor bzw. das Gestell gerastert werden soll, um nach Lösungen zu suchen. Am Beispiel des Viergelenkmessers sollen 20 Punkte verwendet werden.

Mit dem vierten Parameter wird festgelegt, in welcher Reihenfolge die Posen durchlaufen werden sollen, wenn sich die Kurbel gegen den Uhrzeigersinn dreht. Da im Beispiel des Viergelenkmessers nur zwei Posen durchlaufen werden und kein Motor die Drehrichtung vorgibt, ist die Vorgabe in diesen spezifischen Fall egal.

Nicht egal für das Beispiel ist jedoch der fünfte Parameter der Funktion. In diesem wird festgelegt, ob die Bewegung in bestimmten Posen durch Anschläge begrenzt werden soll. Hier soll die Bewegung nur zwischen der ersten und zweiten Pose möglich sein.

Der letzte Parameter der Funktion gibt dem Nutzer die Möglichkeit, Ergebnisse mit Filtern aus der Lösungsmenge auszuschließen. Im Beispiel werden Ergebnisse verworfen, wenn es zu einer Kollision zwischen der Klinge und der Sperrzonen kommen würden. In der Ausgabe der Funktion, welcher in Abbildung 4 zu sehen ist, ist das vorgehen der Funktion erkennbar. Mit den Vorgaben sind grundsätzlich 442000 Kombinationen möglich. Mit den Einschränkungen und Bewegungsvorgaben sind jedoch nur noch 86 Lösungen übrig.

Mit dem nachfolgenden Befehl können dann einzelne Lösungen oder Gruppen von Lösungen angezeigt werden. Hier im Beispiel wird die

```

fourBarposesyntheses: Number of Gridpoints should be: 20
PoseaddGPL: CPLE 0.1 mm Buffer for 19 attachment points
checkfourbar2Poseattachpermutation: Selected poses are [1,2]
checkfourbar2Poseattachpermutation: Number of grid points found in GPL is: 19
checkfourbar2Poseattachpermutation: Grid distance used for effektor is: 7
checkfourbar2Poseattachpermutation: Grid distance for base points is: 7
checkfourbar2Poseattachpermutation: There are 442K combinations to check!
checkfourbar2Poseattachpermutation: There are angle limitations: [1,2]
fourBarposesyntheses: 125 solution were found using 2-Pose-Synthesis.
checkfourbar123solutions: There are 125 solutions to check!
checkfourbar123solutions: Shape buffer for the effector is 0.0 and buffer for links is 0.0
checkfourbar123solutions: Constraints to check: "break coll CPL0 CPLW"
checkfourbar123solutions: =====
checkfourbar123solutions: There are 86 solutions left!
Poseplotsolution: Plot a limited selection of 10 of 86 solutions

```

Abb. 4: Ausgabe der Synthesefunktion

Lösung mit der Nummer 59 angezeigt und animiert. Der erzeugte Plot ist in Abbildung 5 zu sehen.

```
fourBarposeplotsolution(ps_sol ,59 , 'animate')
```

3.3 Automatische Konstruktion

Wenn nun eine Lösung gefunden wurde, welche dem Nutzer gefällt, kann diese automatisch konstruiert werden. Hierfür sind zwei Funktionsaufrufe notwendig. Im ersten Schritt wird die Schichtung und die Kontur alle Glieder des Viergelenks erstellt. Zu diesen Gliedern gehören nicht nur die vier Hauptglieder des Viergelenks, sondern auch bis zu vier Abstandhalter um die Glieder in den unterschiedlichen Ebenen zu verbinden.

Für das Beispiel des Viergelenks wird das struct mit den Lösungen der Maßsynthese als erster Parameter übergeben.

Als zweiten Parameter wird die Nummer der gewünschten Lösung übergeben.

Das dritte Parameterset das übergeben wird, sind die Abmessungen der im Gestell angelenkten Glieder. In diesem Beispiel sollen die Glieder einen Breite von 6 mm haben und durch 1,5 mm Achsen verbunden werden. Die Dicke der Glieder sollen 2 mm sein.

Als viertes Parameterset wird die gewünschte Schichtung übergeben. Hierbei geben die Zahlen an, in welcher Ebene sich der Griff, die Klinge,

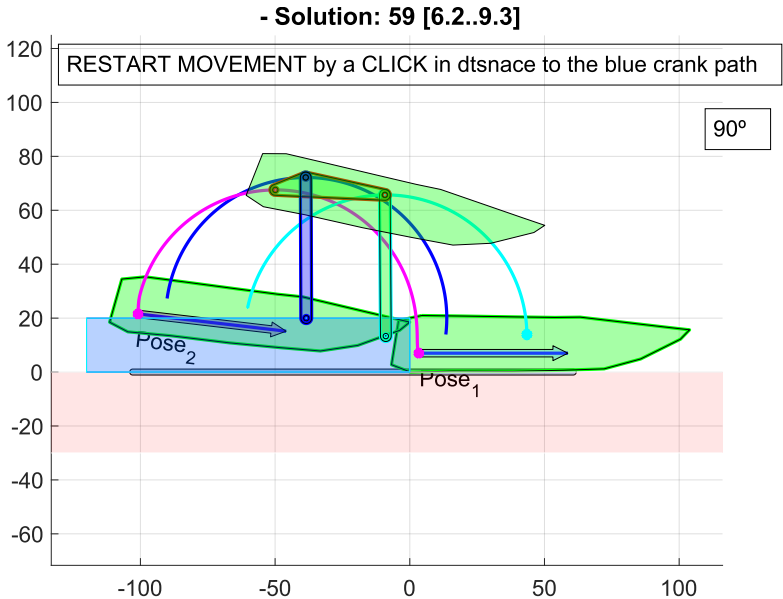


Abb. 5: Animation der Lösung Nummer 59

die beiden im Gestell angelenkten Glieder und das restliche Weltssystem sein sollen.

Der letzte Parameter gibt an, welcher Algorithmus für die Generierung der im Gestell angelenkten Glieder verwendet werden soll. Zur Auswahl stehen hier der A* Algorithmus oder Bezier Kurven.

```
CLLL = fourBarPoselayering(ps_sol, 59, ...
    [6 1.5 2], [0 2 1 3 0], "Bezier");
```

Abschließend können mit dem nachstehenden Befehl die so generierten Konturen in Körper umgewandelt werden. Die standard Verbindungsart ist hierbei mittels DIN912 Schrauben und DIN 985 Muttern. Die so generierten Körper werden als STL Dateien abgespeichert. Zusätzlich werden eine Animation der bewegten Körper und mehrere Ansichten der Baugruppe abgespeichert um die Montage einfacher zu gestalten. Eine solche Ansicht der Baugruppe ist in Abbildung 6 zu sehen.

```
fourBarCLLL2SGdesign(CLLL, [6 1.5 2]);
```

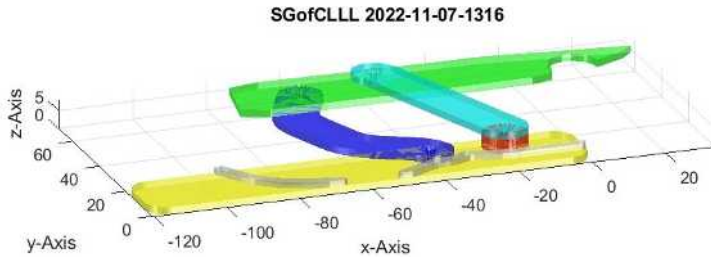


Abb. 6: 3D-Ansicht der Baugruppe für eine einfachere Montage

3.4 Anpassungen

Abschließend können bei Bedarf die STL-Dateien entweder in Matlab oder in einer CAD Software noch weiter angepasst werden. Am Beispiel des Messers wurden noch zwei Griffschalen hinzugefügt und an das Schwingenglied eine Rastfläche für den Daumen des Nutzers hinzugefügt. Eine Rendering einer Messerhälfte kann in Abbildung 7 betrachtet werden. Eine Verriegelung der Klinge konnte nicht hinzugefügt werden, da das Messer sonst nach §42a WaffG unter das Führungsverbot fallen würden [5].

4 Zusammenfassung

Im dem Beitrag wurde am Beispiel eines Taschenmessers gezeigt, wie ein Viergelenk mittels der SG-Library konstruiert werden konnte, ohne dass sich der Nutzer mit der Geometrie beschäftigen musste. Hierbei konnte sich der Nutzer komplett auf die Funktion konzentrieren. Das fertige Messer kann in Abbildung 8 betrachtet werden.



Abb. 7: Nachbearbeitung des Model im CAD System



Abb. 8: Fotos des fertigen Messers

Literatur

- [1] Müller, M. et al. „MechDev—A New Software for Developing Planar Mechanisms“. In: *Interdisciplinary Applications of Kinematics*. Hrsg. von Andrés Kecskeméthy et al. Cham: Springer International Publishing, 2019, S. 167–175. isbn: 978-3-030-16423-2. doi: 10.1007/978-3-030-16423-2_15.
- [2] Lüth, T. C. „SG-Library: Entwicklung Einer Konstruktiven MATLAB-Toolbox Zur Räumlichen Modellierung von Körpern, Gelenken Und Getrieben“. In: *11. Kolloquium Getriebetechnik* (2015). doi: 10.14459/2015md1276136.
- [3] Lüth, T. C. und Irlinger, F. „A Method for the Automatic Design of Mechanisms on the Example of a Four-Bar Linkage“. In: *ROMANSY 24 - Robot Design, Dynamics and Control*. Hrsg. von Andrés Kecskeméthy und Vincenzo Parenti-Castelli. Cham: Springer International Publishing, 2022, S. 12–36. isbn: 978-3-031-06409-8. doi: 10.1007/978-3-031-06409-8_2.
- [4] Lüth, M. „Gestaltung von Passungen Zwischen FDM-Druck- Und Normteilen Für Die Automatische Konstruktion von Mechanismen“. In: *15. Kolloquium Getriebetechnik*. Mainz, Sep. 2023, S. 161. isbn: 978-3-95886-502-0.
- [5] Bundesamt für Justiz. *Waffengesetz (WaffG) §42a Abschnitt 1 Satz 3*.

Automatisches Folgen von Personen und Gestensteuerung für einen mobilen Roboter mithilfe von bildbasierter menschlicher Posenerkennung

Dingzhi Zhang*; Franz Irlinger*, Tim C. Lüth*

* TU München, Lehrstuhl für Mikrotechnik und Medizingerätetechnik
[dingzhi.zhang; irlinger; tim.lueth]@tum.de

Kurzfassung

Die Komplexität der Robotersteuerung und des Bedienkonzeptes steht oft einer schnellen Verbreitung des Robotereinsatzes im Alltag entgegen. In diesem Beitrag werden intuitive Bedienmethoden für einen mobilen Roboter vorgestellt, wie das automatische Folgen des Benutzers und die Gestensteuerung unter Verwendung von bildbasierter menschlicher Posenerkennung. Die Positionsbestimmung des Benutzers relativ zum Roboter mittels Posenerkennung wird aus geometrischen Beziehungen abgeleitet. Außerdem wird in diesem Beitrag das Trainieren eines individuellen Posenklassifikators für die Gestensteuerung vorgestellt.

Title / Abstract

Automatic Human Following and Gesture Control for a Mobile Robot using Vision-Based Human Pose Estimation

The complexity of robot control and operation is often an obstacle to the widespread use of robots in everyday life. This paper presents intuitive control methods for a mobile robot, such as automatic human following and gesture control using vision-based human pose estimation. The determination of the user's position relative to the robot using pose

estimation is derived from geometric relationships. This work also describes the training of a custom pose classifier for gesture control.

1 Einführung

Trotz der rasanten Entwicklung in der Robotik ist der Einsatz mobiler Roboter im Alltag noch nicht weit verbreitet. Die Komplexität der Benutzer- und Programmierschnittstellen stellt für technisch nicht versierte Personen oft eine Hürde dar. In diesem Beitrag wird eine intuitive Bedienungsmethode mithilfe von bildbasierter menschlicher Posenerkennung vorgestellt, die zum einen das automatische Folgen einer Person und zum anderen die Robotersteuerung über Gesten ermöglicht. Die Kombination beider Funktionen ermöglicht eine Steigerung der Benutzerfreundlichkeit der eigenentwickelten mobilen Fahrplattform „Carrier Bot“ mit einer „Teach-and-Repeat“-Programmierung [1].

Die „Teach-and-Repeat“-Programmierung des „CarrierBot“ basiert auf dem Ansatz von Programming by Demonstration [2]. Dieser Ansatz erlaubt es dem Benutzer, einen neuen Roboterpfad ohne Programmierung zu definieren. Die Demonstration neuer Pfade für den Roboter kann durch den Benutzer mittels manueller Führung oder durch automatisches Folgen des Benutzers erfolgen.

Derzeit wird die automatische Verfolgung auf Basis der relativen Positionsbestimmung durch die Laufzeitmessung von drei Ultra-Wideband (UWB)-Sensoren realisiert, von denen zwei als Empfänger am Roboter montiert sind und einer als Sender vom Benutzer getragen wird. Das Sendermodul kann gleichzeitig als Fernbedienung verwendet werden. Allerdings weisen die UWB-Signale bei Umwelteinflüssen wie Hindernissen im Signalweg eine reduzierte Positionierungsgenauigkeit auf [1]. Gängige bildbasierte Verfahren wie Farberkennung, Merkmalserkennung sowie Personenerkennung [3–5] bieten zwar eine bessere Positionierungsgenauigkeit und -präzision, jedoch nicht ausreichend Informationen zur Steuerung des Roboters. Das in diesem Beitrag vorgestellte Verfahren unterscheidet sich von diesen herkömmlichen Methoden dadurch, dass zur Positionsbestimmung von Personen nicht die Bounding Box der im Bild erkannten Person, sondern die Gelenkposen verwendet wird. Die erkannte Posenposition lässt sich zusätzlich

durch ein neuronales Netz in benutzerdefinierte Gestenbefehle klassifizieren, sodass der Roboter durch Gesten gesteuert werden kann.

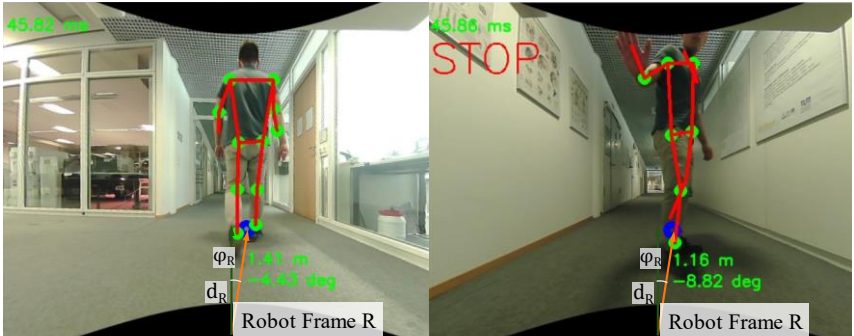


Abb. 1: Relative Positionsbestimmung unter Verwendung der menschlichen Posenerkennung. Beispielbilder aus der Perspektive der Roboterkamera, links: Folgezustand, rechts: Stop-Zustand durch Gestensteuerung.

2 Positionsbestimmung mittels menschlicher Posenerkennung

Im Folgenden wird der Einsatz der bildbasierten menschlichen Posenerkennung zur relativen Positionsbestimmung für das automatische Folgen von Personen sowie die Gestensteuerung für mobile Roboter vorgestellt.

Tab. 1: Schlüsselpunkte der Posenerkennung aus MoveNet [7]

Nr.	Schlüsselpunkt	Relative Koordinate	Konfidenz-Score
1	Nase	y_0, x_0	s_0
2	Linkes Auge	y_1, x_1	s_1
3	rechtes Auge	y_2, x_2	s_2
4	Linkes Ohr	y_3, x_3	s_3
5	Rechtes Ohr	y_4, x_4	s_4
6	Linke Schulter	y_5, x_5	s_5
7	Rechte Schulter	y_6, x_6	s_6
8	Linker Ellbogen	y_7, x_7	s_7

Nr.	Schl�sselpunkt	Relative Koordinate	Konfidenz- Score
9	Rechter Ellbogen	y_8, x_8	s_8
10	Linkes Handgelenk	y_9, x_9	s_9
11	Rechtes Handgelenk	y_10, x_10	s_10
12	Linke H�fte	y_11, x_11	s_11
13	Rechte H�fte	y_12, x_12	s_12
14	Linkes Knie	y_13, x_13	s_13
15	Rechtes Knie	y_14, x_14	s_14
16	Linkes Fu�gelenk	y_15, x_15	s_15
17	Rechtes Fu�gelenk	y_16, x_16	s_16

Mittels der menschlichen Posenbestimmung, wie beispielsweise MoveNet [6], lassen sich Schl sselpunkten eines Menschen (Gelenkpositionen) im Bild erkennen (vgl. Abb. 1) und deren relative Pixelkoordinaten ausgeben. Diese Schl sselpunkte werden durch das Posenerkennungsmodell vorgegeben. Tab. 1 zeigt beispielhaft die Ausgabe der Posenerkennung mittels Single-Pose MoveNet, wobei y_i , x_i , s_i die yx-Koordinaten (normiert auf den Bildbreite und -h he, im Bereich [0.0, 1.0]) und die Konfidenzwerte des i-ten Gelenks sind.

Das Eingangsbild ist vor der Positionsbestimmung mit Ber cksichtigung der Verzerrungsparameter zu entzerren. Zur Positionsbestimmung vom Benutzer kann der Mittelpunkt des linken und rechten Fu gelenks (Nr. 16 und 17) verwendet werden. In Abb.1 wird der Mittelpunkt in blau dargestellt. Die Bestimmung der relativen Position zwischen Benutzer und Roboter basiert auf einer geometrischen Abbildungsgleichung zwischen dem Kamerakoordinatensystem C und dem Bildkoordinatensystem Img (siehe Abb. 2). cF ist die Brennweite der Kamera in L ngeneinheiten von C . Die Kamera ist fest am Roboter montiert und hat daher eine feste Kamerah he h_c . Es wird angenommen, dass der Benutzer w hrend des Trackingprozesses auf dem Boden geht. Somit ist der vertikale Abstand zwischen dem Mittelpunkt der Fu gelenke und dem Ursprung des Kamera-Koordinatensystem cY_h gleich der Kamerah he h_c .

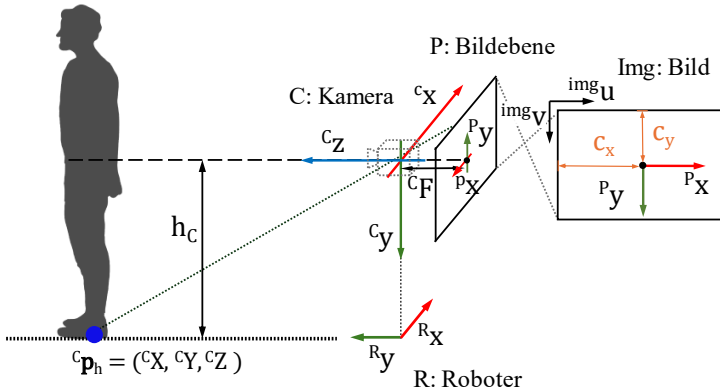


Abb. 2: Geometrische Beziehungen für die Positionsbestimmung.

Die Position des Benutzers im Roboter-Koordinatensystem $({}^R X_h, {}^R Y_h)$ entspricht den Koordinaten $({}^c Z_h, {}^c X_h)$ im Kamera-Koordinatensystem und kann wie folgt ausgedrückt werden:

$${}^R Y_h = {}^c Z_h = f_y \frac{{}^c Y_h}{(img_v_h - c_y)} \quad (1)$$

$${}^R X_h = {}^c X_h = \frac{{}^c Z_h}{f_x} (img_u_h - c_x) \quad (2)$$

wobei f_x und f_y die Brennweite in Pixelgröße in x- bzw. y-Richtung sind. c_x und c_y sind die Pixelkoordinaten des Hauptpunktes. Die Parameter (f_x, f_y, c_x, c_y) werden zusammen mit den Verzerrungsparametern durch die Kamerakalibrierung mit einem geeigneten Kameramodell bestimmt. Die in Echtzeit ermittelten Positionsergebnisse werden in Abb. 1 in Polarkoordinaten des Roboters dargestellt.

3 Gestensteuerung durch Posenerkennung und Posenklassifikation

Die Posenbestimmung ermöglicht neben der Positionsbestimmung auch die Implementierung einer Gestensteuerung. Dazu werden zusätzliche Gelenkpositionen verwendet.

Abb.1 zeigt eine Not-Stop-Gestensteuerung, die im Rahmen dieser Arbeit entwickelt wird. Durch das Heben einer oder beider H ande wird der Not-Stop aktiviert. Die Erkennung dieser Geste erfolgt durch die Auswertung der Pixelposition zwischen Handgelenk und Ellenbogen.

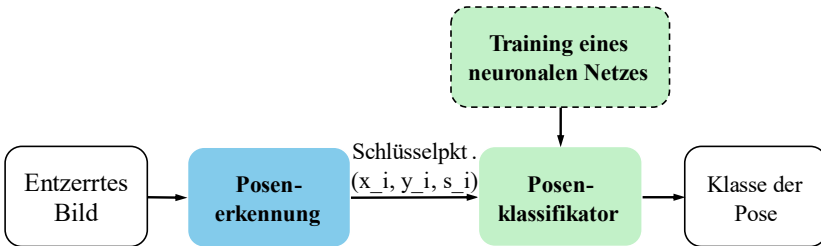


Abb. 3: Schematischer Ablauf der Gestensteuerung mit Posenerkennung und -klassifikation.

Mithilfe eines weiteren Klassifikators (z.B. eines neuronalen Netzes) lassen sich mehrere komplexere Posen erkennen und anschlieend in Steuerbefehle f ur den Roboter umzusetzen. Abb. 3 zeigt den schematischen Ablauf der Gestensteuerung mit Posenerkennung und Posenklassifikation. Ausgehend von einem entzerrten Kamerabild wird zun achst die Posenerkennung mithilfe eines vortrainierten Pose Estimation Models (z.B. MoveNet) durchgef uhrt. Die Ausgabe der Posenerkennung sind die in Tab.1 dargestellten normalisierten Pixelkoordinaten der erkannten Gelenkpunkte (x_i , y_i) und die zugeh origen Konfidenzwerte (s_i) des i -ten Gelenks. Diese Daten werden dann einem zweiten neuronalen Netz zur Klassifikation der erkannten Pose zugef uhrt. Ausgabe des Prozesses ist die Klasse der Pose, die dann in Steuerbefehle f ur den Roboter konvertiert werden kann.

Durch das Trainieren eines benutzerdefinierten Posenklassifikators ist es m oglich, eine nutzerspezifische Gestensteuerung zu realisieren. Das Trainieren eines individuellen Posenklassifikators kann die folgenden Schritte beinhalten: Datensammlung, Datenaufbereitung, Modellauswahl, Trainieren des Modells, Test und Validierung des Modells. Im Folgenden wird der Ablauf des Trainierens eines Posenklassifikators im Rahmen vorliegender Arbeit dargestellt.

Im ersten Schritt werden Bilder der gewünschten Gesten für die Robotersteuerung aufgenommen. Anschließend werden diese 80% der Bilder in einen Trainingssatz und 20% in in einen Testsatz aufgeteilt.

Tab. 2: Überblick über die Datensammlung für das Trainieren eines individuellen Posenklassifikators

Nr.	Klasse	Anzahl der Bilder im Trainingssatz	Anzahl der Bilder im Testsatz
1	Emergency stop	22	5
2	Left	31	7
3	Right	35	8
4	Stop	35	8
5	Walk	32	7

Tab.2 gibt einen Überblick über die definierten Posenklassen und die Anzahl der Bilder im Trainings- und im Testsatz.

Im zweiten Schritt werden die Bilder, sowohl im Trainings- als auch im Testsatz mit einem Posenerkennungsmodell (z.B. MoveNet) auf die Schlüsselpunkte der Gelenke analysiert, um die eigentlichen Eingabedaten für den Posenklassifikator zu erhalten. Dabei werden nur solche Bilder verwendet, bei denen alle Konfidenzwerte der Gelenkpunkte größer als 0,1 sind. Um die Gültigkeit des Posenklassifikators unabhängig von der Dimension und Auflösung des Eingangsbildes zu gewährleisten, werden in diesem Schritt normalisierte Koordinaten der Gelenkpunkte verwendet.

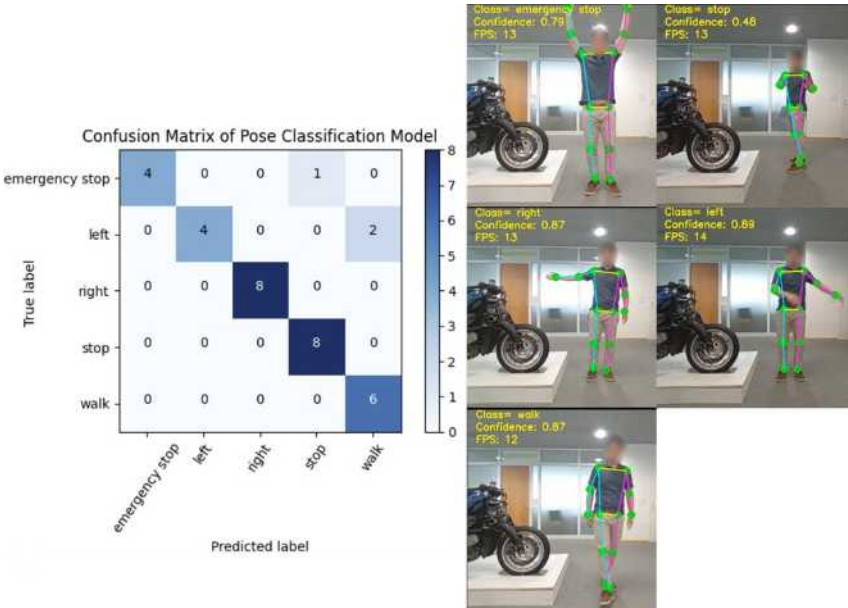


Abb. 4: Wahrheitsmatrix des trainierten Posenklassifikators (links) und Ausschnitte der selbsttrainierten Posenerkennung und -klassifikation aus dem Live-Kamerabild (rechts).

Für den Posenklassifikator wird ein neuronales Netz mit 7 Sichten und insgesamt 13061 Parameter definiert. Nach dem Training mit den gültigen Bildern aus dem Trainingsatz werden diese mit den Daten im Testsatz verifiziert. Insgesamt erreicht der trainierte Posenklassifikator mit den gültiger Testdaten eine Genauigkeit von 91%. Die Wahrheitsmatrix ist in Abb.4 dargestellt.

Die Positionsbestimmung und Gestenerkennung wird unter Verwendung von MoveNet [6] in Tensorflow Lite durchgeführt. Zur Beschleunigung der Inferenz von MoveNet wird ein Google Coral USB Accelerator auf einem Low-Cost Einplatinenrechner verwendet. Bei einer Auflösung von 640 x 480 Pixel wird eine durchschnittliche Framerate von über 21 FPS erzielt. Die Posenerkennung mit anschließender selbsttrainierter Posenklassifikation ohne Hardwarebeschleunigung erreicht eine durchschnittliche Framerate von 12-14 FPS. Ausschnitte der selbsttrainierten Posen-

erkennung und -klassifikation sind auf der rechten Seite von Abb. 4 abgebildet.

4 Zusammenfassung und Ausblick

Die menschliche Posenbestimmung liefert sowohl Informationen zur Positionsbestimmung als auch zur Gestenbestimmung. Dies ermöglicht die Implementierung von Personen-Tracking und Gestensteuerung für mobile Roboter. Zukünftig besteht die Möglichkeit, eine größere Anzahl von Schlüsselpunkten des Menschen zur Positionsbestimmung zu verwenden, um zusätzlich eine Schätzung der relativen Rotation des Benutzers zum Roboter durchzuführen. Eine potentielle Optimierung der Gestensteuerung könnte durch die Verwendung eines geeigneten Tracking-Algorithmus realisiert werden, um eine realistischere und aussagekräftigere Vorhersage der Gelenkposition bei Erkennung von Gelenkpositionen mit schlechten Konfidenzwerten oder beim Verlust der Posenerkennung zu erreichen.

Literatur

- [1] D. Zhang, L. Birner, V. Zinkernagel, C. Rehekampff, D. Burschka, and T. C. Lueth, "Carrier Bot: A UWB-based Human Following Mobile Platform for Intra-Office Transport with an Intuitive Teach-and-Repeat Programming," in *2023 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Koh Samui, Thailand, 2023, pp. 1–6.
- [2] B. Siciliano and O. Khatib, Eds., *Springer Handbook of Robotics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.
- [3] K. Koide and J. Miura, "Convolutional Channel Features-Based Person Identification for Person Following Robots," in *Advances in Intelligent Systems and Computing, Intelligent Autonomous Systems 15*, M. Strand, R. Dillmann, E. Menegatti, and S. Ghidoni, Eds., Cham: Springer International Publishing, 2019, pp. 186–198.
- [4] T. Yoshimi *et al.*, "Development of a Person Following Robot with Vision Based Target Detection," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, 2006, pp. 5286–5291.

- [5] C. Schlegel, J. Illmann, H. Jaberg, M. Schuster, and R. Wörz, "Vision Based Person Tracking with a Mobile Robot," in *Proceedings of the British Machine Vision Conference 1998*, Southampton, 1998, 42.1-42.10.
- [6] *MoveNet: Ultra fast and accurate pose detection model*. [Online]. url: <https://www.tensorflow.org/hub/tutorials/movenet> (besucht: May 1 2024).
- [7] google, "movenet: A convolutional neural network model that runs on RGB images and predicts humanjoint locations of a single person." url: <https://www.kaggle.com/models/google/movenet/> (besucht am 02.08.2024).

Vier Wege zum Ball'schen Punkt

Stefan Gössner*

* Fachhochschule Dortmund, Fachbereich Maschinenbau
stefan.goessner@fh-dortmund.de

Kurzfassung

Ausgehend von der Betrachtung der Bahnkrümmung eines Punktes der bewegten Ebene wird über den Wendekreis und die Gleichung von Euler-Savary die Kreispunktkurve hergeleitet. Im Schnittpunkt dieser Kurve mit dem Wendekreis liegt der Ballsche Punkt, der von hoher praktischer Bedeutung für die Maßsynthese von Koppelmechanismen ist. In einem anschaulichen Beispiel wird das Vorgehen demonstriert und drei weiteren Verfahren gegenübergestellt.

Die Betrachtung bedient sich der symplektische Geometrie im Vektorraum \mathbb{R}^2 als innovativer Ansatz.

Titel / Abstract

Four Ways to Ball's Point.

Discussing the curvature of a point on the moving plane, the circle-point-curve is derived via inflection circle and the equation of Euler-Savary. Ball's point, which is of high practical importance in mechanism design, lies at the point of intersection of circle-point-curve and inflection circle. There are three more ways to determine Ball's point, which are presented briefly in an illustrative mechanism example.

Symplectic geometry in vector space \mathbb{R}^2 is used for the derivation as a new innovative approach.

1 Einführung

Während der Betrachtung der Kinematik einer bewegten Ebene ist häufig der Ball'sche Punkt von besonderem Interesse. Dieser - nach *Robert Stawell Ball* [1] benannte - Punkt durchläuft zum betrachteten Zeitpunkt gerade einen Flachpunkt (Undulationspunkt) seiner Bahn und liegt mit vier unendlich benachbarten Punkten auf einer Geraden [7]. So resultiert im Bereich dieses Bahnpunktes ein signifikant langes Geradenstück seiner Bahn. Die Kenntnis dieses Punktes zur Wahl als Koppelpunkt ist daher für die Synthese von Geradführungsgetrieben von hoher praktischer Bedeutung.

Der Ball'sche Punkt kann auf vier verschiedene Arten gefunden werden

1. Schnittpunkt von Wendekreis und Kreis-Punkt-Kurve
2. Schnittpunkt der Wendekreise erster und zweiter Ordnung
3. Bereis'sche Konstruktion
4. Vektorgleichung zur Bereis'schen Konstruktion

Der erste Ansatz, der den Ball'schen Punkt als spezielles Element der Kreispunktkurve (Kurve mit stationärer Krümmung) bestimmt, wird in dieser Arbeit genauer analysiert. Eine Diskussion dieser Kurve ist in mehreren Publikationen zu finden [2,3,4,5]. In diesem Beitrag wird eine kinematische Herleitung der Kreispunktkurve zunächst vektoriell begonnen und im Anschluss skalar beendet. Letzteres lehnt sich eng an die Diskussion der Kreispunktkurve von Dijkstra an [2].

Die sich daraus ergebenden Gleichungen werden auf einen anschaulichen Beispielmechanismus angewandt und mit den Ergebnissen der obigen Methoden 2, 3 und 4 verglichen.

Der symplektische Vektorraum \mathbb{R}^2 ist ausgestattet mit einer kompatiblen, komplexen Struktur, die die Wirkung eines Orthogonaloperators (analog zur imaginären Einheit i) hat. Die Anwendung dieses Operators auf einen Vektor wird als Tildesymbol „ \sim “ über der betroffenen Vektorvariablen notiert.

2 Krümmung

Wir betrachten eine bewegte Ebene mit bekanntem Geschwindigkeitspol P , die eine Winkelgeschwindigkeit ω besitzt.

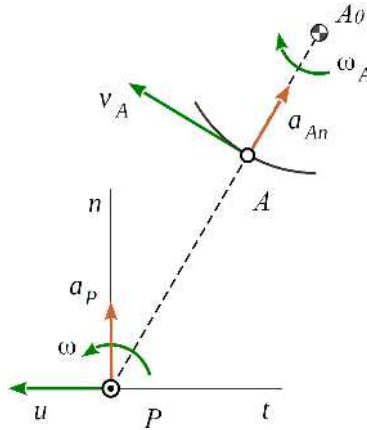


Bild. 1: Bahn des Punktes einer bewegten Ebene

Die kinematischen Gleichungen nach Euler lauten für den Punkt A [5]

$$\mathbf{r}_A = \mathbf{r}_P + \mathbf{r}_{PA} \quad (1)$$

$$\mathbf{v}_A = \underbrace{\mathbf{v}_P}_0 + \omega \tilde{\mathbf{r}}_{PA} \quad (2)$$

$$\mathbf{a}_A = \dot{\omega} \tilde{\mathbf{r}}_{PA} - \omega^2 \mathbf{r}_{PA} \quad (3)$$

Ein Punkt A dieser Ebene folgt einer Bahn, für die wir den Schmiegekreis und damit den Krümmungsmittelpunkt A_0 identifizieren. Die Punkte A und A_0 werden als *konjugierte Krümmungsmittelpunkte* bezeichnet. Sie liegen zusammen mit dem Geschwindigkeitspol P auf einer gemeinsamen Geraden - dem *Polstrahl*. Geschwindigkeit \mathbf{v}_A und Normalbeschleunigung \mathbf{a}_{An} des Punktes A lassen sich als momentane Rotation um den Krümmungsmittelpunkt A_0 mit der Winkelgeschwindigkeit ω_A auffassen, also $\mathbf{v}_A = \omega_A \tilde{\mathbf{r}}_{A_0A}$ und $\mathbf{a}_{An} = \omega_A^2 \mathbf{r}_{A_0A}$. Diese beiden Beziehungen lassen sich zu $\mathbf{a}_{An} = -\omega_A \tilde{\mathbf{v}}_A$ kombinieren. Nach Multiplikation mit

$\tilde{\mathbf{v}}_A$ erhalten wir den Ausdruck $\omega_A = \frac{\tilde{\mathbf{v}}_A \mathbf{a}_{An}}{v_A^2}$. Wegen des projektiven Charakters des Skalarprodukts können wir $\tilde{\mathbf{v}}_A \mathbf{a}_{An}$ durch $\tilde{\mathbf{v}}_A \mathbf{a}_A$ ersetzen und gelangen über $\mathbf{v}_A = \omega_A \tilde{\mathbf{r}}_{A_0A}$ zur Lage des Krümmungsmittelpunkts

$$\mathbf{r}_{A_0A} = \frac{v_A^2}{\tilde{\mathbf{v}}_A \mathbf{a}_A} \tilde{\mathbf{v}}_A. \quad (4)$$

Gleichung (4) ist das vektorielle Äquivalent zur bekannten skalaren Beziehung für den Bahnkrümmungsradius $\rho = \frac{1}{\kappa} = \frac{(x^2 + y^2)^{\frac{3}{2}}}{\dot{x}\dot{y} - \dot{y}\dot{x}}$.

3 Wendekreis

Definition:

Alle Punkte einer bewegten Ebene, die gerade einen Wendepunkt ihrer Bahn durchlaufen, liegen auf einem Kreis, dem *Wendekreis*.

Solch ein Punkt – sagen wir A – hat keine Normalbeschleunigung. Vielmehr ist dessen Beschleunigung \mathbf{a}_A zur Geschwindigkeit \mathbf{v}_A gleichgerichtet. Die Bedingung der Kollinearität lautet $\mathbf{a}_A \tilde{\mathbf{v}}_A = 0$. Setzen wir die Ausdrücke (2) und (3) ein, resultiert $(\mathbf{a}_P + \dot{\omega} \tilde{\mathbf{r}}_{PA} - \omega^2 \mathbf{r}_{PA})(-\omega \mathbf{r}_{PA}) = 0$.

Ausmultiplizieren liefert $\mathbf{r}_{PA}^2 - \frac{\mathbf{a}_P}{\omega^2} \mathbf{r}_{PA} = 0$ und die quadratische Ergänzung führt auf $(\mathbf{r}_{PA} - \frac{\mathbf{a}_P}{2\omega^2})^2 = (\frac{\mathbf{a}_P}{2\omega^2})^2$. Dies ist die vektorielle Form der Kreisgleichung $(\mathbf{p} - \mathbf{p}_0)^2 = R^2$, was zu zeigen war.

Der Pol ist stets ein Punkt des Wendekreises, da $\mathbf{a}_P \tilde{\mathbf{v}}_P = 0$ mit $\mathbf{v}_P = \mathbf{0}$ gilt. Der dem Pol gegenüberliegende Kreispunkt ist der Wendepol W (Bild 2). Dessen Lage kann der Kreisgleichung entnommen werden.

$$\mathbf{r}_{PW} = \frac{\mathbf{a}_P}{\omega^2}. \quad (5)$$

Die Polbeschleunigung \mathbf{a}_P hat die Richtung der Polbahnnormalen [5] (Bild 1).

4 Gleichung von Euler-Savary

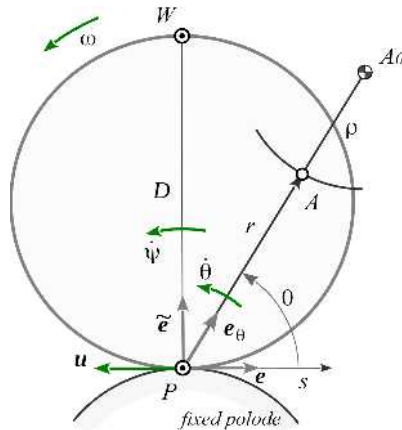


Bild. 2: Punktbahn Kinematik

Definition:

Die Gleichung von *Euler-Savary* beschreibt die Krümmungsverhältnisse der Bahn eines Punkts der bewegten Ebene.

4.1 Vektorgleichung

Einsetzen der kinematischen Beziehungen (2) und (3) in (4) ergibt

$$\mathbf{r}_{AA_0} = \frac{\omega^2 \mathbf{r}_{PA}^2}{-\omega \mathbf{r}_{PA} (\mathbf{a}_P + \dot{\omega} \tilde{\mathbf{r}}_{PA} - \omega^2 \mathbf{r}_{PA})} (-\omega \mathbf{r}_{PA}) .$$

Ausmultiplizieren, Ersetzen der Polbeschleunigung \mathbf{a}_P mittels Gleichung (5) und Kürzen des Bruchs führt zur rein geometrischen, vektoriellen Form der Euler-Savary Gleichung

$$\mathbf{r}_{AA_0} = \frac{\mathbf{r}_{PA}^2}{\mathbf{r}_{PW} \mathbf{r}_{PA} - \mathbf{r}_{PA}^2} \mathbf{r}_{PA} . \quad (6)$$

Sind zwei Paare konjugierter Krümmungsmittelpunkte bekannt – etwa $\{A, A_0\}$ und $\{B, B_0\}$, lässt sich Gleichung (6) zweimal hinschreiben und so eine vektorielle Gleichung für die Wendepollage synthetisieren [5].

$$\mathbf{r}_{PW} = \frac{1}{\tilde{\mathbf{r}}_{PA} \mathbf{r}_{PB}} \left(\mathbf{r}_{PB}^2 \left(\frac{\mathbf{r}_{PB}^2}{\mathbf{r}_{BB_0} \mathbf{r}_{PB}} + 1 \right) \tilde{\mathbf{r}}_{PA} - \mathbf{r}_{PA}^2 \left(\frac{\mathbf{r}_{PA}^2}{\mathbf{r}_{AA_0} \mathbf{r}_{PA}} + 1 \right) \tilde{\mathbf{r}}_{PB} \right) \quad (7)$$

4.2 Skalare Gleichung

Wir formulieren die Vektoren in Gleichung (6) bezüglich der symplektischen Basis $\{\mathbf{e}, \tilde{\mathbf{e}}\}$, ausgerichtet an Polbahntangente und -normale, um (Bild 2). Die Ersetzung $\mathbf{r}_{PA} = r \mathbf{e}_\theta$, $\mathbf{r}_{AA_0} = \rho \mathbf{e}_\theta$ und $\mathbf{r}_{PW} = D \tilde{\mathbf{e}}$ darin liefert

$$\rho \mathbf{e}_\theta = \frac{r^2}{D r \tilde{\mathbf{e}} \mathbf{e}_\theta - r^2} r \mathbf{e}_\theta .$$

Die Multiplikation mit dem Einheitsvektor \mathbf{e}_θ und die Identität $\tilde{\mathbf{e}} \mathbf{e}_\theta = \sin \theta$ resultiert in der bekannten skalaren, quadratischen Form der Euler-Savary Gleichung [2,3,4].

$$\rho = \frac{r^2}{D \sin \theta - r} \quad (8)$$

5 Die Kreispunktkurve

Definition:

Der geometrische Ort aller Punkte der bewegten Ebene mit gegenwärtiger stationärer Krümmung ist die Kreispunktkurve.

Wir beginnen mit dieser Zeitableitung von Gleichung (1): $\dot{\mathbf{r}}_A = \dot{\mathbf{r}}_P + \dot{\mathbf{r}}_{PA}$. Darin ist $\dot{\mathbf{r}}_P$ die Polwechselgeschwindigkeit \mathbf{u} und $\dot{\mathbf{r}}_{PA}$ die zeitliche Änderung des Vektors zwischen den Punkten A und P. $\dot{\mathbf{r}}_A$ setzt sich zusammen aus der momentanen Rotation von A um P mit ω und dem Abrollen $\dot{\psi}$ der Gangpolbahn (nicht gezeigt in Bild 2) auf der Rastpol-

bahn mit dem momentanen Krümmungsradius R_0 . Aus dem Zusammenhang von Polbeschleunigung a_p und Polwechselgeschwindigkeit $u = \frac{\tilde{a}_p}{\omega}$ [5] resultiert mit Gleichung (5) der Ausdruck $u = \omega r_{PW}$. Damit haben wir die folgenden Zusammenhänge:

$$\begin{aligned}\dot{r}_A &= (\omega + \dot{\psi}) r \tilde{e}_\theta \\ \dot{r}_P = \mathbf{u} &= -\omega D \mathbf{e} = -\dot{\psi} R_0 \mathbf{e} \\ \dot{r}_{PA} &= \frac{d}{dt}(r \mathbf{e}_\theta) = \dot{r} \mathbf{e}_\theta + \dot{\theta} r \tilde{e}_\theta\end{aligned}$$

Deren Einsatz in $\dot{r}_A = \dot{r}_P + \dot{r}_{PA}$ führt auf $(\omega - \dot{\theta} + \dot{\psi}) r \tilde{e}_\theta = -\omega D \mathbf{e} + \dot{r} \mathbf{e}_\theta$. Multiplikation hiervon (i) mit \mathbf{e}_θ und (ii) mit \tilde{e}_θ liefert uns zwei nützliche Ausdrücke.

$$\begin{aligned}\dot{r} &= \omega D \mathbf{e} \mathbf{e}_\theta = \omega D \cos \theta \\ \dot{\theta} &= \frac{\omega}{r} D \mathbf{e} \tilde{e}_\theta + \omega + \dot{\psi} = -\frac{\omega}{r} (D \sin \theta - r) + \omega \frac{D}{R_0}\end{aligned}$$

5.1 Herleitung der Kreispunktkurve

Die Bedingung einer stationären Krümmung für Punkte der Kreispunktkurve ist gleichbedeutend mit $\dot{\rho} = 0$. Die Zeitableitung von (8) lautet

$$\dot{\rho} = \frac{2 r \dot{r} (D \sin \theta - r) - r^2 (\dot{D} \sin \theta + \dot{\theta} D \cos \theta - \dot{r})}{(D \sin \theta - r)^2}$$

Mit den obigen nützlichen Ausdrücken resultiert (siehe [2,7])

$$\dot{\rho} = \frac{3 \omega \rho^2 D^2}{r^2} \left(\frac{\sin \theta \cos \theta}{r} - \frac{\sin \theta}{m} - \frac{\cos \theta}{n} \right) \quad \text{mit} \quad \frac{1}{m} = \frac{\dot{D}}{3 \omega d^2}, \quad \frac{1}{n} = \frac{D + 2 R_0}{3 D R_0}$$

Wegen $\dot{\rho} = 0$ muss die runde Klammer verschwinden, also:

$$\frac{\sin \theta \cos \theta}{r} = \frac{\sin \theta}{m} + \frac{\cos \theta}{n} \quad (9)$$

5.2 Zwei bekannte Punkte der Kreispunktkurve

Die Bestimmung der Koeffizienten m und n konstant für jede Lage der bewegten Ebene ist aufwendig [2]. Wenn wir jedoch zwei Punkte mit konstanter Bahnkrümmung identifizieren können - sie sich also auf Gerade oder Kreis bewegen, müssen diese auf der Kreispunktkurve liegen. Es lässt sich Gleichung (9) zweimal auf diese Punkte – sagen wir A und B – anwenden und daraus das gesuchte m und n gewinnen.

$$\begin{aligned}
 m &= \frac{r_A r_B (\cos \theta_A \sin \theta_B - \sin \theta_A \cos \theta_B)}{\cos \theta_A \cos \theta_B (r_A \sin \theta_B - r_B \sin \theta_A)} = \frac{K}{M} \\
 n &= \frac{r_A r_B (\cos \theta_A \sin \theta_B - \sin \theta_A \cos \theta_B)}{\sin \theta_A \sin \theta_B (r_B \cos \theta_A - r_A \cos \theta_B)} = \frac{K}{N}
 \end{aligned}
 \tag{10}$$

5.3 Die Kreispunktkurve

Invertierung von Gleichung (9) und Einsetzen der Koeffizienten (10) liefert die Kreispunktkurve in polarer Form (Bild 3).

$$r(\theta) = \frac{K \sin \theta \cos \theta}{M \sin \theta + N \cos \theta}
 \tag{11}$$

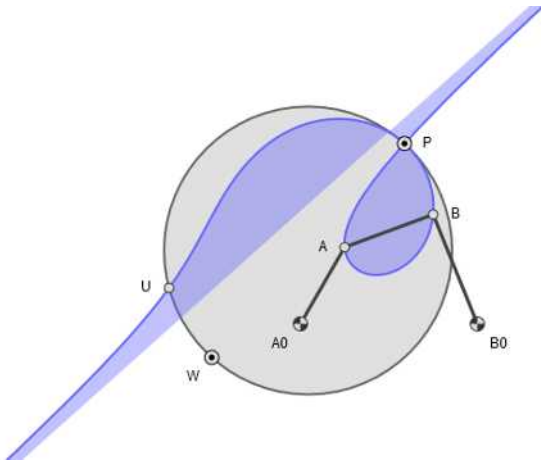


Bild. 3: Kreispunktkurve der Koppel eines Viergelenks

6 Ball'scher Punkt

Definition:

Ein Punkt U der bewegten Ebene, der gerade einen Flachpunkt¹ seiner Bahn durchläuft, die dort also stationär die Krümmung Null besitzt, wird Ball'scher Punkt² genannt.

Demnach muss die Geschwindigkeit $\dot{\mathbf{p}}$ eines solchen Punkts kollinear mit sowohl dessen Beschleunigung $\ddot{\mathbf{p}}$ als auch dessen Ruck $\dddot{\mathbf{p}}$ sein, also muss $\ddot{\mathbf{p}} \dot{\mathbf{p}} = \ddot{\mathbf{p}} \ddot{\mathbf{p}} = 0$ gelten. Diese Bedingung erfüllt neben dem Schnittpunkt der Wendekreise erster und zweiter Ordnung auch der Schnittpunkt von Wendekreis und Kreispunktkurve (Bild 3).

Die Gleichung des Wendekreises bezüglich der Basis $\{\mathbf{e}, \tilde{\mathbf{e}}\}$ (Bild 2) lautet $r(\theta) = D \sin \theta$. Gleichsetzen mit Beziehung (11) liefert den Winkel θ_U des Undulationspunkts.

$$\tan \theta_U = \frac{K - DN}{DM} \quad (12)$$

Den zugehörigen Vektor \mathbf{r}_{PU} erhalten wir durch Drehung und Skalierung des Einheitsvektors \mathbf{e} mit θ_U bzw. $r(\theta_U)$, also

$$\mathbf{r}_{PU} = D \sin \theta_U (\cos \theta_U \mathbf{e} + \sin \theta_U \tilde{\mathbf{e}}) .$$

Ausklammern von $\cos \theta_U$ und die Verwendung des Zusammenhangs

$\sin \phi \cos \phi = \frac{\tan \phi}{1 + \tan^2 \phi}$ führt auf die Vektorgleichung des Ball'schen Punkts.

$$\mathbf{r}_{PU} = \frac{D \tan \theta_U}{1 + \tan^2 \theta_U} (\mathbf{e} + \tan \theta_U \tilde{\mathbf{e}}) \quad (13)$$

In Verbindung mit Ausdruck (12) ist diese Beziehung erfreulicherweise trigonometriefrei.

¹ Auch Undulationspunkt

² Nach Sir Robert Stawell Ball, Irland (1840-1913)

7 Beispiel: Ball'scher Punkt der Koppel eine Viergelenks ...

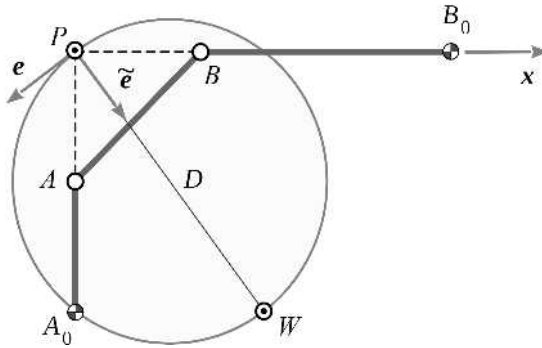


Bild. 4: Beispiel Viergelenk

Der Ball'sche Punkt der Koppel eine Viergelenks in gezeigter Stellung (Bild 4) sei mittels vier verschiedener Methoden zu bestimmen. Gegeben sind die Vektoren – normalisiert bezüglich der Kurbellänge.

$$r_{AA_0} = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, r_{BB_0} = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, r_{PA} = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, r_{PB} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, r_{PW} = \frac{1}{2} \begin{pmatrix} 3 \\ -4 \end{pmatrix}, D = r_{PW} = \frac{5}{2}$$

Der Vektor des Wendekreises r_{PW} folgt aus Gleichung (7).

7.1 ... mittels Kreispunktkurve

Vektor r_{PW} liefert die symplektische Basis $e = -\frac{\tilde{r}_{PW}}{r_{PW}} = \frac{1}{5} \begin{pmatrix} -4 \\ -3 \end{pmatrix}, \tilde{e} = \frac{1}{5} \begin{pmatrix} 3 \\ -4 \end{pmatrix}$.

Daraus lassen sich die trigonometrischen Ausdrücke für die Winkel θ_A und θ_B bestimmen.

$$\cos \theta_A = -e \tilde{x} = \frac{3}{5}, \sin \theta_A = -\tilde{e} \tilde{x} = \frac{4}{5}, \cos \theta_B = -e x = -\frac{4}{5}, \sin \theta_B = \tilde{e} x = -\frac{3}{5}$$

Hiermit erhalten wir aus (10) die Koeffizienten $K=1, M=\frac{12}{125}, N=\frac{84}{125}$ und nun führt die Beziehung (11) auf $\tan \theta_U = -\frac{17}{6}$.

Die Lage des Ball'schen Punkts (Bild 5) ergibt sich letztlich aus (13) zu

$$\mathbf{r}_{PU} = \frac{17}{26} \begin{pmatrix} 3 \\ -2 \end{pmatrix}$$

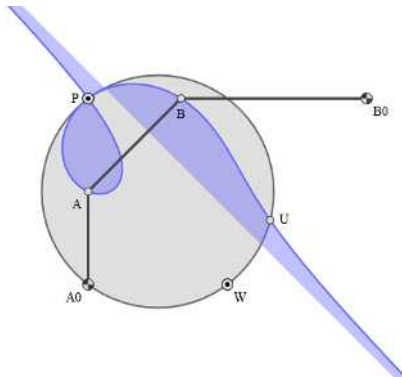


Bild. 5: Kreispunktkurve und Ball'scher Punkt des Beispiels

7.2 ... mittels Wendekreise erster und zweiter Ordnung

Die Gleichung des Wendekreises zweiter Ordnung sei ohne Herleitung angegeben. Jene kann hier nachgelesen werden [7].

$$\mathbf{r}_{PW_2} = \frac{r_{PA}^2}{(\tilde{\mathbf{r}}_{PW} \mathbf{r}_{PA})(\tilde{\mathbf{r}}_{PA} \mathbf{r}_{PB})} \left(\frac{(\tilde{\mathbf{r}}_{PW} \mathbf{r}_{PB})(\mathbf{r}_{PW} - \mathbf{r}_{PB}) \mathbf{r}_{PB}}{r_{PB}^2} - \frac{(\tilde{\mathbf{r}}_{PW} \mathbf{r}_{PA})(\mathbf{r}_{PW} - \mathbf{r}_{PA}) \mathbf{r}_{PB}}{r_{PA}^2} \right) \tilde{\mathbf{r}}_{PA}$$

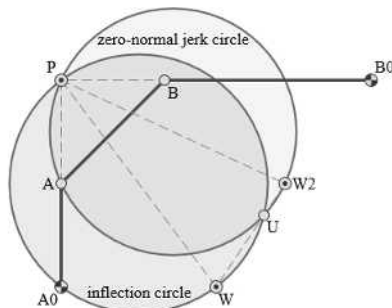


Bild. 6: Wendekreise erster und zweiter Ordnung

Das Einsetzen der Beispielwerte ergibt $r_{PW_2} = \frac{1}{6} \begin{pmatrix} 13 \\ -6 \end{pmatrix}$. Der Vektor von W nach W_2 ist $r_{WW_2} = r_{PW_2} - r_{PW}$ (Bild 6) und die Undulationspunktlage ist dann

$$r_{PU} = \frac{\tilde{r}_{WW_2} r_{PW}}{r_{WW_2}^2} \tilde{r}_{WW_2} = \frac{17}{26} \begin{pmatrix} 3 \\ -2 \end{pmatrix}.$$

7.3 ... mittels Konstruktion nach Berezis

Von Rudolf Berezis [8] stammt ein ansprechendes Konstruktionsverfahren des Ball'schen Punkts [5,7].

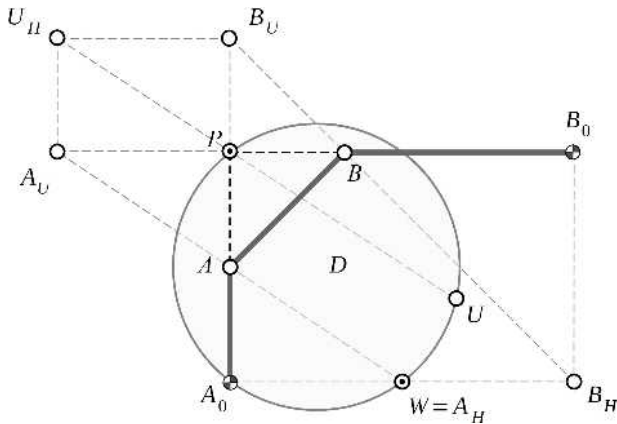


Bild. 7: Konstruktionsverfahren von Berezis

Das Verfahren geht wie folgt:

1. Ermittlung des Wendepols und -kreises.
2. Im Schnittpunkt der Orthogonalen zu PA_0 in A_0 und der Parallelen zu PA_0 durch W liegt der Hilfspunkt A_H .
3. Im Schnittpunkt der Orthogonalen zu PB_0 in B_0 und der Parallelen zu PB_0 durch W liegt der Hilfspunkt B_H .

4. Im Schnittpunkt der Geraden AA_H und der Orthogonalen zu PA durch P liegt der Hilfspunkt A_U .
5. Im Schnittpunkt der Geraden BB_H und der Orthogonalen zu PB durch P liegt der Hilfspunkt B_U .
6. Im Schnittpunkt der Parallelen zu PA durch A_U und der Parallelen zu PB durch B_U liegt der Hilfspunkt U_H .
7. Der Schnittpunkt von PU_H mit dem Wendekreis ist der Ball'sche Punkt U.

7.4 ... mittels Vektorgleichung des Ball'schen Punkts

Wenn jeder Schritt der Bereis'schen Konstruktion durch einen beschreibenden, geschlossenen Vektorzug formuliert wird, führt die Lösung dieses vektoriellen Gleichungssystems auf eine schöne, symmetrische Vektorgleichung des Ball'schen Punkts [7]. Sie lautet

$$\mathbf{r}_{PU} = \frac{\mathbf{r}_{PW} \mathbf{r}_{PU_H}}{r_{PU_H}^2} \mathbf{r}_{PU_H} \quad \text{mit} \quad \mathbf{r}_{PU_H} = \frac{(\tilde{\mathbf{r}}_{PW} \mathbf{r}_{PA}) \mathbf{r}_{BB_0} - (\tilde{\mathbf{r}}_{PW} \mathbf{r}_{PB}) \mathbf{r}_{AA_0}}{\tilde{\mathbf{r}}_{AA_0} \mathbf{r}_{BB_0}}$$

Hierin ist \mathbf{r}_{PU_H} der Vektor vom Pol P zum Hilfspunkt U_H in Bild 7. Einsetzen der gegebenen Werte liefert das bekannte Ergebnis.

$$\mathbf{r}_{PU} = \frac{17}{26} \begin{pmatrix} 3 \\ -2 \end{pmatrix}$$

8 Zusammenfassung

Die Kenntnis des Ball'schen Punkts ist von hohem praktischen Nutzwert während des Mechanismendesigns. Es gibt unterschiedliche Vorgehensweisen, sich der Lage dieses Undulationspunkts zu nähern. Der Fokus in diesem Beitrag liegt auf der Betrachtung der Kreispunktkurve.

Die Herleitung der Kreispunktkurve basiert einerseits auf der vektoriellen Anwendung der symplektischen Geometrie im \mathbb{R}^2 . Andererseits

lehnt sich der skalare Teil an der vorbildlichen Vorgehensweise von Dijkman an [2]. Schließlich wird der Ball'sche Punkt im Schnittpunkt der Kreispunktkurve und des Wendekreises gefunden.

Die Anwendung des gefunden Ergebnisses wird an einem anschaulichen Beispiel demonstriert und drei weiteren Vorgehensweisen gegenübergestellt

9 Referenzen

- [1] Robert Stawell Ball,
url:https://de.wikipedia.org/wiki/Robert_Stawell_Ball (bes. 07.08.2024).
- [2] Dijkman, E.A.. Motion Geometry of Mechanisms. 1976 Cambridge University Press, isbn: 0-521-20841-6.
- [3] Hartenberg, R.S., Denavit J.E.A. Kinematic Synthesis of Linkages. McGraw-Hill, 1964.
- [4] Bottema O., Roth B. Theoretical Kinematics. Dover Publications, 1990.
- [5] Gössner S., Mechanismentechnik. Logos Berlin, 2017, isbn: 978-3-8325-4362-4.
- [6] Gössner S., Ball's Point Construction Revisited. Mechanism and Machine Science, 2020
- [7] Modler K.-H., Getriebetechnik. Springer, 1995, isbn: 3-540-57001-2.
- [8] Rudolf Bereis, url:https://de.wikipedia.org/wiki/Rudolf_Bereis (bes. 07.08.2024).

Untersuchung von Methoden zur Synthese von Pick-and-Place Bewegungen

Georg Steinert *; Jens-Peter Majschak**

* TU Dresden, Professur für Verarbeitungsmaschinen/Verarbeitungstechnik
georg.steinert.@tu-dresden.de

** TU Dresden, Professur für Verarbeitungsmaschinen/Verarbeitungstechnik
jens-peter.majschak.@tu-dresden.de

Kurzfassung

Dieser Betrag beleuchtet Methoden zur Synthese von Bewegungen für Pick-and-Place Aufgaben. Solche Aufgaben treten in Verarbeitungsmaschinen zunehmend auf. Dabei werden je nach Anwendungsfall verschiedene Anforderungen an das Bewegungsdesign gestellt. Es existieren viele verschiedene Ansätze zur Synthese einer Pick-and-Place Bewegung. Anhand eines konkreten Anwendungsfalls werden, basierend auf einem Bewegungsplan, Bewegungsdesigns mit Hilfe der verschiedenen Methoden berechnet. Diese werden anschließend anhand verschiedener Kriterien beurteilt und eingeordnet.

Titel / Abstract

Investigation of methods for the synthesis of pick-and-place motions. This article examines methods for the synthesis of motions for pick-and-place tasks. Those tasks are increasingly encountered in processing machines. Depending on the application, different aspects on the motion design are required. Consequently, there are various approaches to the synthesis of a pick-and-place motion. Based on a specific application and a motion plan, motion designs are calculated using the various methods. These are assessed and categorised using various criteria.

1 Motivation und Problemstellung

Pick-and-Place Prozesse haben sich in Verarbeitungsmaschinen in den vergangenen Jahrzehnten immer weiter etabliert [1]. Frei programmierbare Industrieroboter in Kombination mit flexiblen Greifersystemen erlauben die Handhabung vielfältiger Objekte. Die genannten Prozesse sind in der Regel zyklisch und folgen den in Abb. 1 dargestellten Abschnitten.

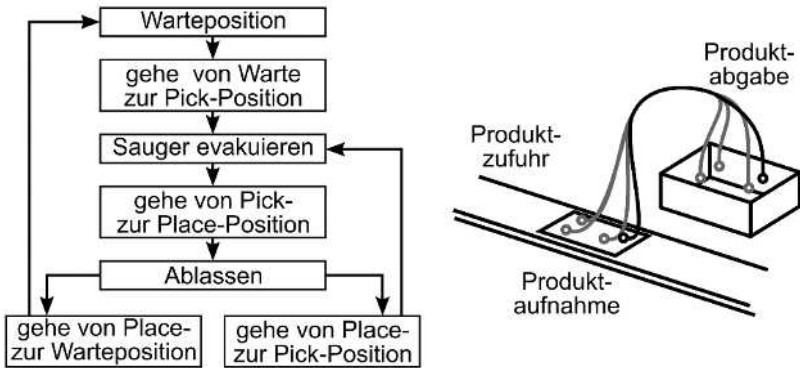


Abb. 1: Abschnitte und schematische Darstellung eines Pick-and-Place Prozesses.

Üblicherweise wird bei den verwendeten Industrierobotern auf Systemangebote zurückgegriffen. Das bedeutet, dass Mechanik, Motoren und Motion-Control System vom gleichen Anbieter bezogen werden. Diese bieten wiederum vorgefertigte Synthesemethoden an, um die benötigten Bewegungsdesigns zu berechnen und umzusetzen. Üblicherweise ist es aber auch möglich, eigene Methoden zu implementieren.

Die folgenden Ausführungen sollen anhand eines exemplarischen Anwendungsfalls durchgeführt werden. Dieses ist aus der Praxis entlehnt und zeichnet sich durch die in Tab. 1 aufgeführten Parameter aus. Das Produkt soll mit Hilfe eines Vakuum-Sauggreifers gegriffen werden. Es wird sich bei der Betrachtung auf die Transportphase, also den Abschnitt zwischen Produktaufnahme und Produktabgabe beschränkt, da in dieser erfahrungsgemäß die meisten Fehlerfälle durch Kontaktverluste auftreten [2]. Aus der Bewegungsaufgabe kann ein Bewegungsplan (s. Abb. 2) abgeleitet werden, welcher nach [3] der Ausgangspunkt für die Bewegungssynthese ist.

Tab. 1: Parameter des betrachteten Anwendungsfalls

Parameter	Symbol	Wert
Hub in x-/y-Richtung	x_{Hub}/y_{Hub}	300 mm / 400 mm
Endposition in z-Richtung	z_{End}	-70 mm
Höhe der Kartonkante	z_{Kart}	20 mm
Transportzeit	t_{trans}	300 ms
Zuführbandgeschwindigkeit	v_B	0,33 m/s
Objektmasse	m	250 g
Haftreibungskoeffizient	μ_H	0,33

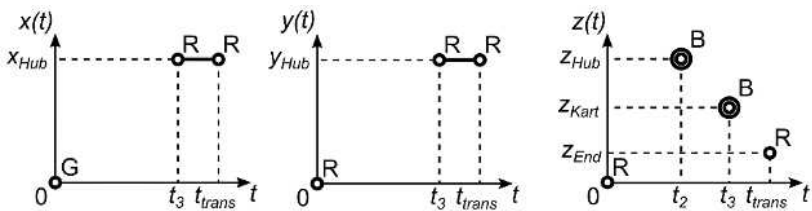


Abb. 2: Bewegungsplan für die betrachtete Bewegungsaufgabe.

Die Bewegungssynthese bezeichnet den Vorgang zur Erstellung eines Bewegungsdesigns [3]. An dieser Stelle soll in Anlehnung an die Definition von BRAUNE zur Getriebesynthese [4] ergänzt werden, dass die zugrundeliegende Bewegungsaufgabe möglichst gut oder zumindest akzeptabel erfüllt wird. Die zu verfolgende Zielstellung kann unterschiedlich ausfallen. Vorstellbar sind beispielsweise die Minimierung der Taktzeit, des Effektivmoments oder des maximalen Antriebsmoments unter Beibehaltung einer möglichst hohen Funktionssicherheit.

In diesem Fall soll das Kontaktverhalten der Wirkpaarung herangezogen werden. Im betrachteten Fall handelt es sich, wie in Abb. 3 schematisch dargestellt, um einen Vakuum Sauggreifer, welcher ein Objekt mit definierter Masse und Abmessung hält.

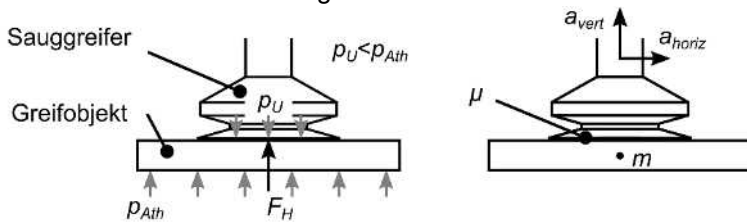


Abb. 3: Wirkpaarung

Um einen Kontaktverlust zu vermeiden muss die Haltebedingung (Gleichung (1)) erfüllt sein:

$$F_{Halt} \geq F_{Res} \quad (1)$$

Dabei lassen sich Haltekraft F_{Halt} und resultierende Kraft F_{Res} wie in Gleichung (2) berechnen [5]. A_{eff} bezeichnet hierbei die effektive Saugfläche des Greifers, $\Delta p = p_{Ath} - p_U$ die Druckdifferenz zwischen Atmosphäre und Saugerinnenraum, m die Masse des Objektes, g die Erdbeschleunigung, μ_H den Haftreibungskoeffizienten zwischen Sauger und Objekt und a_i die Beschleunigung in horizontaler und vertikaler Richtung.

$$A_{eff} \cdot \Delta p \geq m \left(g + a_{vert} + \frac{1}{\mu_H} |a_{horiz}| \right) \quad (2)$$

Ein Prozess ist umso robuster [6], je höher die Differenz zwischen Haltekraft und resultierender Kraft ausfällt. Die Haltekraft des Sauggreifers soll an dieser Stelle als konstant angenommen werden. Da auch Masse des Objekts, Reibwerte und Erdbeschleunigung als konstant betrachtet werden ergibt sich die Zielstellung, die auftretenden Massenkräfte zu minimieren. Dafür werden verschiedene Synthesemethoden kurz vorgestellt und mit einem Standardberechnungsverfahren verglichen. Die Möglichkeiten, ein Bewegungsdesign auf Grundlage des vorgestellten Bewegungsplans zu erstellen sind faktisch grenzenlos. Es wird sich daher auf drei ausgewählte Verfahren beschränkt, die, neben dem Standardverfahren, im Folgenden beschrieben werden.

2 Syntheseverfahren

2.1 Standardverfahren (Stand)

Verbreitet im Bereich der Roboterautomatisierung sind entkoppelte Bewegungsplanungen [7]. Die Methode gliedert sich in zwei Teilprobleme: die Bahnplanung, welche als Ergebnis eine geometrische Kurve zwischen Start- und Endpunkt liefert und die Trajektorienplanung, bei der ein geeignetes Geschwindigkeitsprofil berechnet wird, mit welchem die Bahn abgefahren wird [7].

Im vorliegenden Fall besteht die Bahn aus drei Abschnitten (s. Abb. 4): zwei Ellipsenbögen zwischen Startpunkt, Scheitelpunkt und Kartonkante und einem Geradenabschnitt zwischen Kartonkante und Endpunkt. Als Geschwindigkeitsprofil wird ein 7-Segmente Profil [8] eingesetzt.

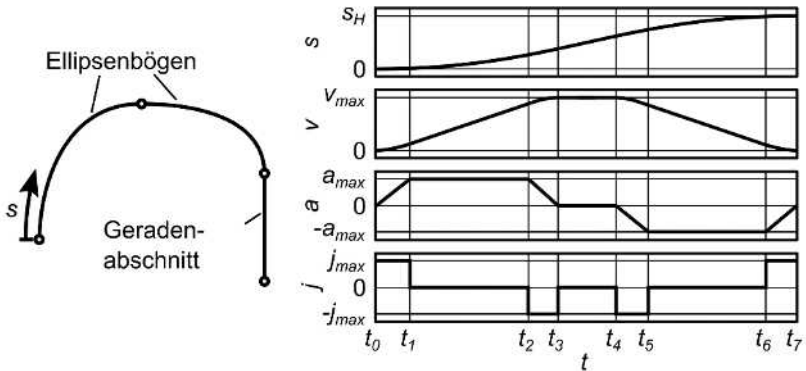


Abb. 4: Standardverfahren mit Bahnform, bestehend aus Ellipsenbögen und einem Geradenabschnitt (links) und das Geschwindigkeitsprofil (rechts), mit welchem die Bahn abgefahren wird.

Um die Geschwindigkeitsrandbedingung am Startpunkt zu gewährleisten, wird die Bewegung $\mathbf{f}(t) = [x(t), y(t), z(t)]^T$ mit der des Bandes $\mathbf{g}(t) = [v_B, 0, 0]^T$ überlagert.

$$\mathbf{f}_1(t) = h(t) \cdot \mathbf{g}(t) + (1 - h(t)) \cdot \mathbf{f}(t) \quad (3)$$

Die Funktion $h(t)$ ist hierbei eine stetige monoton steigende Funktion mit den Randpunkten $h(0) = 0$ und $h(t_B) = 1$. In diesem Fall ein Polynom 5. Grades.

2.2 Polynom 5. Grades mit Wendepunktverlagerung (optiPoly5)

Dieses Verfahren beruht auf dem interaktiven Bewegungsdesign [9] und wurde bereits in [10] vorgestellt. Hierbei wird der z-Verlauf durch zwei Polynomabschnitte 5. Grades erzeugt, welche jeweils einen verlagerbaren Wendepunkt am Zeitpunkt t_{WP1} bzw. t_{WP2} besitzen. Der Scheitelpunkt des Verlaufs ist ebenfalls verschiebbar gestaltet. Die Verläufe in x- und y-Richtung werden durch ein Polynom 5. Grades mit anschließender Rastphase umgesetzt. Das Vorgehen wurde aus pragmatischen Gründen gewählt, da es relativ einfach in eine bestehende Anlage implementiert werden konnte. Die Zeitpunkte der beiden Wendepunkte sowie Höhe und Zeitpunkt des Scheitelpunktes ergeben 4 Variablen $\mathbf{p} = [t_{WP1}, t_2, t_{WP2}, z_{Hub}]^T$, unter Vorgabe derer ein Bewegungsdesign $[x, y, z]^T = \mathbf{f}(\mathbf{p}, t)$ berechnet werden kann.

Die Ermittlung der Variablen erfolgt durch ein Optimierungsverfahren, welches sich folgendermaßen formulieren lässt:

$$\mathbf{p}^* = \arg \min_{\mathbf{p}} J(\mathbf{p}) \quad (4)$$

Die Zielfunktion $J(\mathbf{p})$ berechnet sich dabei aus einer modifizierten Form des rechten Terms von Gleichung (2), wobei die Summanden vernachlässigt werden, welche nicht vom verwendeten Bewegungsdesign abhängig sind.

$$J(\mathbf{p}) = \max_t \left(\frac{1}{\mu_H} \cdot (\ddot{x}(\mathbf{p}, t)^2 + \ddot{y}(\mathbf{p}, t)^2) + \ddot{z}(\mathbf{p}, t)^2 \right) \quad (5)$$

Anstelle der Beträge von Vertikal- und Horizontalbeschleunigung werden die quadrierten Werte verwendet. Das liegt darin begründet, dass mit dem vorgestellten Lösungsverfahren andernfalls keine sinnvolle Lösung gefunden wird.

Als Nebenbedingung wird festgelegt, dass die Zeitpunkte der Wendepunkte bzw. des Scheitelpunktes zeitlich aufeinander folgen, und dass sich der Scheitelpunkt oberhalb der Kartonkante befinden soll.

Das Problem wurde in der Programmierumgebung MATLAB implementiert und mit dem numerischen Solver „fmincon“ [11] gelöst.

2.3 Polynom 7. Grades (optiPoly7)

Anstelle mehrerer Polynomabschnitte verwendet dieses Verfahren für den Verlauf in Z-Richtung einen einzigen Polynomabschnitt 7. Grades.

$$z(t) = A_0 + A_1 t + A_2 t^2 + A_3 t^3 + A_4 t^4 + A_5 t^5 + A_6 t^6 + A_7 t^7 \quad (6)$$

Durch die Vorgabe von Lage, Geschwindigkeit und Beschleunigung in den Randpunkten entsteht ein zweifach unterbestimmtes Gleichungssystem zur Berechnung der Koeffizienten A_i .

$$\begin{aligned} z(0) &= 0 & z(t_{trans}) &= z_{End} \\ \dot{z}(0) &= 0 & \dot{z}(t_{trans}) &= 0 \\ \ddot{z}(0) &= 0 & \ddot{z}(t_{trans}) &= 0 \end{aligned} \quad (7)$$

Diese beiden Freiheitsgrade werden durch Zeitpunkt und Höhe des Scheitelpunktes definiert,

$$z(t_2) = z_{Hub} \quad \dot{z}(t_2) = 0 \quad (8)$$

welche als Variablen $\mathbf{p} = [t_2, z_{Hub}]^T$ in ein Optimierungsproblem einbezogen werden. Dieses wird identisch zu dem in Kapitel 2.2 vorgestellten Verfahren formuliert und unter Verwendung der selben Zielfunktion (Gleichung (5)) gelöst.

2.4 Optimalsteuerung (optiSteu)

Im Gegensatz zu den in Kapitel 2.2 und 2.3 vorgestellten Ansätze erlaubt es die Methode der optimalen Steuerung [12] Bewegungsdesigns zu ermitteln, ohne dass zuvor eine Berechnungsvorschrift festgelegt werden muss, die auf wenigen Parametern basiert. Auf eine genauere Beschreibung des Verfahrens wird an dieser Stelle verzichtet. Es sei diesbezüglich auf [3] und [13] verwiesen.

Ein numerisches Lösungsverfahren wurde ebenfalls in MATLAB implementiert. Als Zielfunktion wird wie zuvor Gleichung (5) verwendet. Die Rand- und Nebenbedingungen ergeben sich aus dem Bewegungsplan. Aufgrund der sich ergebenden großen Anzahl an Variablen wird die Software Bibliothek IPOPT [14] genutzt, welche durch Verwendung des MUMPS Solvers in diesem Fall deutlich schneller als der ebenfalls geeignete „fmincon“-Solver ist. Einige Anpassungen, beispielsweise die von BETTS [13] beschriebene Handhabung von „minmax“-Problemen durch sogenannte Slack-Variablen wurden vorgenommen, um einen funktionierenden Lösungsalgorithmus zu erhalten.

3 Vergleich

3.1 Grundsätzliches

Zuerst sollen die Bewegungsdesigns für sich betrachtet werden (s. Abb. 5). Obwohl die Lösungen aller vier betrachteten Verfahren den Bewegungsplan erfüllen, sind deutliche Unterschiede sogar in der Lagefunktion erkennbar. Ebenfalls sichtbar sind Knicke und Sprünge in der 1. und 2. Ableitung der Lagefunktionen beim Standardverfahren sowie beim Verfahren mit optimaler Steuerung. Diese Bewegungen sind somit ruckbehaftet.

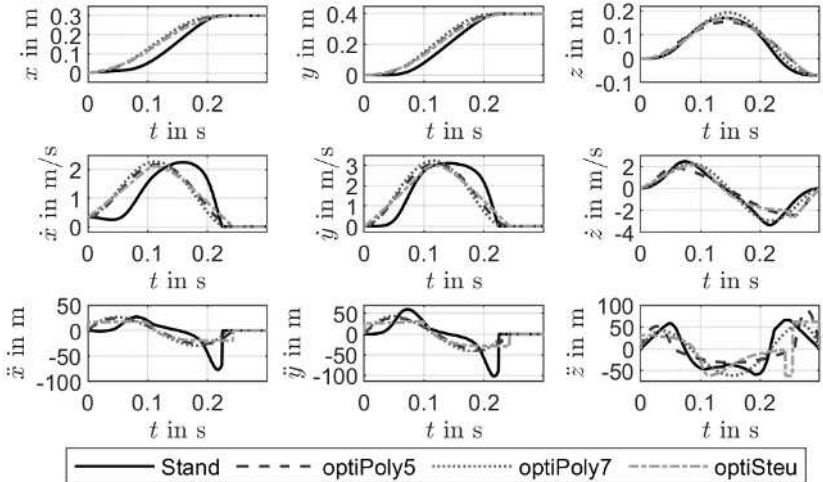


Abb. 5: Bewegungsdesigns. Lage-, Geschwindigkeits- und Beschleunigungsverläufe in x-, y- und z-Richtung.

Eine Analyse der Kennwerte nach VDI 2143 [15] in Form eines Kennwertkreises [16] ist in Abb. 6 dargestellt. Verwendet wird jeweils der größte Maximalwert der Verläufe in x-, y- und z-Richtung. Anstelle der Absolutwerte wird der Quotient vom jeweils besten (also niedrigsten) Kennwert der vier untersuchten Ansätze und dem vorliegenden Kennwert dargestellt. Ein höherer Wert ist demnach besser. Diese Darstellung legt nahe, dass der Ansatz mit Polynom 7. Grades besonders gut abschneidet, da er bei vielen Kennwerten einen vergleichsweise guten Wert aufweist.

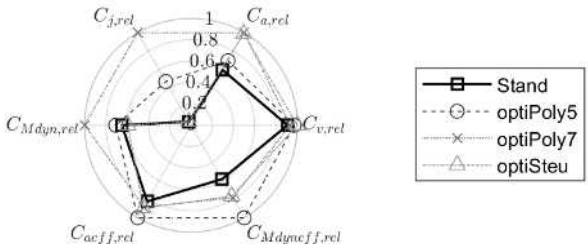


Abb. 6: Kennwertkreis mit den Quotienten aus Minimalwert und vorliegendem Kennwert. Betrachtet wird der jeweilige Maximalwert von x-, y- oder z-Verlauf.

3.2 Betrachtung der Wirkpaarung

Ziel der Optimierung ist die Minimierung der resultierenden Kraft an der Kontaktstelle. Diese kann mit Hilfe von Gleichung (2) berechnet werden. Die tatsächliche Zielfunktion wurde aber, aufgrund oben genannter Umstände angepasst. Es soll daher sowohl die der Optimierung zugrundeliegende Zielfunktion (genauer gesagt die Funktion innerhalb des Maximum Operators) und die resultierende Kraft betrachtet werden.

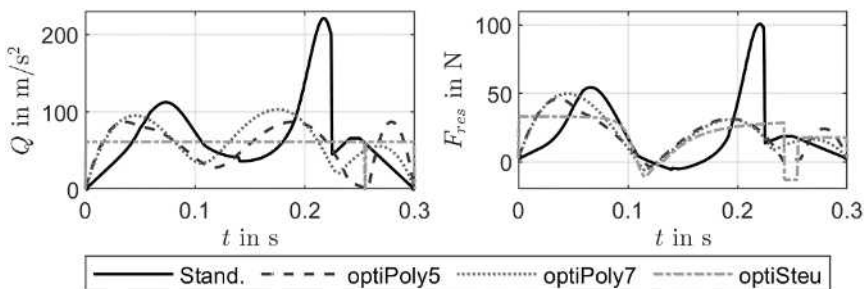


Abb. 7: Links: Funktion innerhalb des Maximum Operators der Zielfunktion, Rechts: resultierende Kraft

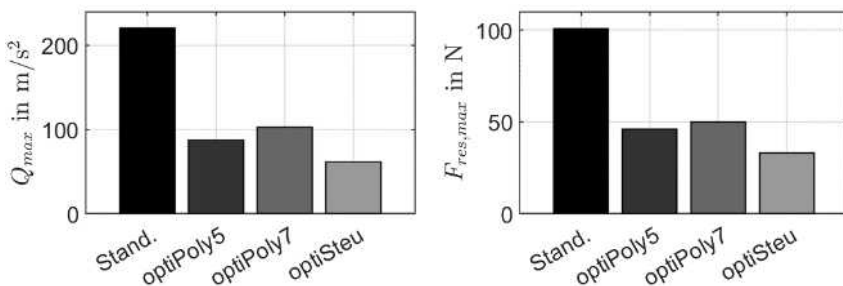


Abb. 8: Links: Wert der Zielfunktion des Optimierungsproblems, Rechts: Maximalwert der resultierenden Kraft

Deutlich zu erkennen ist, dass alle Verfahren eine deutliche Verbesserung gegenüber dem Standardverfahren bewirken. Die größte Verbesserung, sowohl bei der Zielfunktion, als auch beim Maximalwert der resultierenden Kraft erreicht der Ansatz mit optimaler Steuerung. In welchem Maße eine Optimierung nach dem Maximum der resultierenden Kraft eine weitere Verbesserung bewirken kann, bleibt an dieser Stelle offen. Dafür sind andere als die hier verwendeten mathematischen Verfahren nötig.

3.3 Betrachtung des Handhabungssystems

Für den betrachteten Anwendungsfall soll eine Delta-Kinematik eingesetzt werden. Da diese eine nichtlineare Übertragungsfunktion zwischen Antrieb und Abtrieb besitzt, kann man nicht zwangsläufig von den Vergleichsgrößen am Abtrieb auf solche am Antrieb schließen. Deshalb wurde ein kinetostatisches Modell unter Verwendung der MATLAB Toolbox „Simscape“ gebildet (s. Abb. 9) und genutzt, um die zu erwartenden Antriebsbelastungen zu bestimmen. Die Maße und Trägheitseigenschaften der Kinematik wurden in Orientierung an einen realen Anwendungsfall gewählt.

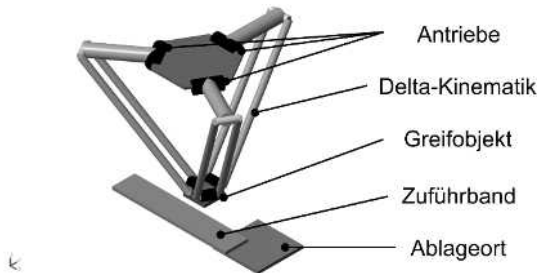


Abb. 9: Darstellung des „Simscape“-Modells zur Prozesssimulation

Von besonderem Interesse sind dabei die an den Antrieben erforderlichen Momente, welche sich aus den Massen- und Gewichtskräften sowie Prozesskräften ergeben. Auch hier erkennt man eine deutliche Reduktion des Maximalwerts der drei Optimierungsansätze gegenüber dem Standardansatz. Besonders gering fällt das Antriebsmoment bei der Optimalen Steuerung aus. Der Ansatz mit Polynom 7. Grades weist den Verlauf mit den geringsten Steigungen bzw. Krümmungen auf.

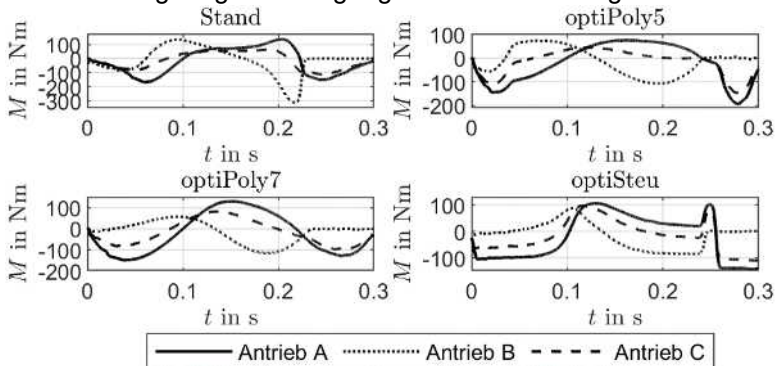


Abb. 10: Antriebsmomente der drei Antriebe für den betrachteten Prozess.

3.4 Möglichkeiten zur Implementierung

Obwohl das Hauptaugenmerk in diesem Beitrag auf der Berechnung und theoretischen Analyse der Bewegungsdesigns liegt, sollen an dieser Stelle noch Überlegungen zur Implementierung angebracht werden. Bis auf den Standardansatz enthalten alle vorgestellten Verfahren einen numerischen Optimierungsalgorithmus, welcher nicht ohne Weiteres auf der Steuerung ausgeführt werden kann. Da Start- und Endpunkt variieren können, ist eine Anpassung der Bewegungsdesigns im Betrieb aber notwendig. Es bieten sich verschiedene potentielle Vorgehensweisen an, um dem zu begegnen. Eine Auswahl wird kurz andiskutiert:

1. Worst-Case-Ansatz: Die Optimierung läuft im zeitunkritischen Bereich, also vor dem Maschinenbetrieb ab. Die Randbedingungen bilden dabei den zu erwartenden Worst-Case ab, also die Bewegungsaufgabe mit dem größtmöglichen Hub. Im Betrieb wird das so zuvor ermittelte Bewegungsdesign auf die tatsächlich auftretenden Randbedingungen skaliert. Die somit gefundene Lösung ist demnach nicht optimal. Da mit sinkendem Hub auch die zu erwartende Beschleunigung und somit die resultierende Kraft an der Kontaktstelle sinken sollte, ist dies aber vermutlich unproblematisch.
2. Kennfeldansatz: Wie in [17] beschrieben werden im Vorfeld eine definierte Anzahl an optimalen Lösungen berechnet, welche systematisch über den Parameterraum (mögliche Start- und Endpunkte) verteilt sind. Durch Interpolation zwischen den optimalen Lösungen kann der gesamte Parameterraum abgebildet werden. Das so entstandene Kennfeld wird auf der Steuerung gespeichert. Im Betrieb kann das benötigte Bewegungsdesign aus dem Kennfeld ausgelesen werden.
3. Lernende Methoden: [18] zeigt ein Verfahren, wie man auch zeitkritische Berechnungen durch Verwendung von Methoden des maschinellen Lernens im Maschinenbetrieb vornehmen kann.

4 Zusammenfassung und Ausblick

Dieser Beitrag befasste sich mit der Bewegungssynthese für Pick-and-Place Prozesse. Neben einem etablierten Standardverfahren wurden drei weitere Ansätze vorgestellt, welche jeweils einen Optimierungsvorgang beinhalteten. Es handelte sich dabei um bekannte Methoden, die auf einen exemplarischen Anwendungsfall angepasst wurden. Ziel war es, die zu erwartete Belastung des Kontakts zwischen Greifer und

Greifobjekt zu reduziert. Ein Simulationsmodell einer Handhabungseinrichtung erlaubte abschließend die Abschätzung zu erwartender Prozessgrößen im praktischen Einsatz. Hinsichtlich der Kontaktbelastung ließ sich mit der Methode der optimalen Steuerung das beste Bewegungsdesign im Vergleich berechnen. Zieht man weitere Vergleichskriterien, beispielsweise die Kenngrößen nach VDI 2143 heran, so stellen sich die beiden Ansätze mit Polynomabschnitten als gute „Allrounder“ heraus. Hinsichtlich der Implementierung konnten an dieser Stelle nur erste Ansätze diskutiert werden. Die im Kapitel 3.4 beschriebenen Herausforderungen könnten Gegenstand zukünftiger Arbeiten sein.

5 Literatur

- [1] DLG e.V. Fachzentrum Lebensmittel, „DLG Trendmonitor 2023 - Roboter in der Lebensmittel und Getränkeindustrie,“ Frankfurt am Main, 2023.
- [2] F. Gabriel, P. Bobka und K. Dröder, *Model-Based Design of Energy-Efficient Vacuum-Based Handling Processes*, 5. C. C. o. M. Systems, Hrsg., Procedia CIRP, 2020.
- [3] C. Troll, *Synthese und Umsetzung optimaler Bewegungen unter Berücksichtigung der einstellbaren Maschinendrehzahl*, T. U. Dresden, Hrsg., 2021.
- [4] R. Braune, *Genaulagen-Synthese von ebenen Koppelgetrieben mit aufgabenspezifisch konzipierten Bearbeitungsstrategien*, Aachen: Springer Vieweg Verlag, 2021.
- [5] S. Hesse, *Grundlagen der Handhabungstechnik*, München: Carl Hanser Verlag GmbH & Co. KG, 2016.
- [6] P. A. Gellerich, „Resilienz in der Praxis - Gestaltung robuster Verarbeitungsvorgänge,“ in *VVD 2024 Verarbeitungsmaschinen und Verpackungstechnik - Tagungsband*, Dresden/Radebeul, Selbstverlag der TU Dresden, 2024, pp. 355-374.
- [7] A. Schwarz, *Automatisierte Bewegungsplanung flexibler Handhabungsgeräte in der verarbeitenden Industrie [Dissertation]*, Dresden: TU Dresden, 2010.
- [8] B. Alpers, „On fast jerk-, acceleration- and velocity-restricted motion functions for online trajectory generation,“ *Robotics*, 05 10 2021.

- [9] R. Braune, „Bewegungs-Design mit interaktiver Variation von Stützwerten - Intuitive Bewegungsoptimierung durch Ausnutzung zugelassener Gestaltungsspielräume,“ in *VDI-Berichte 1966 zur VDI-Getriebetagung.*, Düsseldorf, VDI-Verlag, 2006, pp. 47-76.
- [10] G. Steinert und U. Jensen, „Bewegungsoptimierung bei Pick-and-Place Prozessen,“ in *VVD 2024 Verarbeitungsmaschinen und Verpackungstechnik - Tagungsband*, Dresden/Radebeul, Selbstverlag der TU Dresden, 2024, pp. 29-44.
- [11] MathWorks, „fmincon,“ [Online]. Available: <https://de.mathworks.com/help/optim/ug/fmincon.html>. [Zugriff am 14. 6. 2022].
- [12] O. Föllinger, *Optimale Regelung und Steuerung*, 3. verbesserte Auflage Hrsg., Wien: R. Oldenbourg Verlag München, 1994.
- [13] J. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, Philadelphia: SIAM, 2010.
- [14] A. Wächter, *An Interior Point Algorithm for Large-Scale Nonlinear Optimization with Applications in Process Engineering*, Pittsburgh: Carnegie Mellon University, 2002.
- [15] VDI-Gesellschaft Produkt- und Prozessgestaltung, *VDI 2143 Blatt 1: Bewegungsgesetze für Kurvengetriebe; Theoretische Grundlagen*, Berlin: Beuth Verlag GmbH, 1980.
- [16] M. Berger, J. Matthes und P. Kitzig, „Kennwertkreis und normierte Bewegungsgestaltung - Möglichkeiten zur optimalen Auslegung von Servoantriebssystemen,“ in *VDI Bewegungstechnik 2008. VDI Bericht 2050*, Düsseldorf, VDI-Verlag GmbH, 2008, pp. 17-34.
- [17] O. Holowenko, C. Troll, S. Ihlenfeldt und J.-P. Majschak, „A Novel Adaptive Interpolation Approach for the Implementation of Operating-Speed-Dependent Motion Profiles in Processing Machines,“ *Engineering Science and Technology, an International Journal*, 2021.
- [18] D. Gerth, F. Thiessenhusen und M. Döring, „Möglichkeiten zum Einsatz von Reinforcement Learning für das Motion Planning schneller Roboterbewegungen,“ in *VVD 2024 Verarbeitungsmaschinen und Verpackungstechnik - Tagungsband*, Dresden/Radebeul, Selbstverlag der TU Dresden, 2024, pp. 51-76.

Normierte Doppel-S-Profile für die Bewegungsaufgaben der VDI-Richtlinie 2143

Deborah Dietterle*; Burkhard Alpers*

* Hochschule Aalen, Fakultät Maschinenbau

deborah.dietterle@studmail.htw-aalen.de bzw. burkhard.alpers@hs-aalen.de

Kurzfassung

So genannte Doppel-S-Profile als Bewegungsfunktionen ermöglichen es, Maximalwerte von Geschwindigkeit, Beschleunigung und Ruck über einem ganzen Intervall konstant zu halten und damit höhere Spitzenwerte zu vermeiden. In normierter Form sind sie für die Bewegungsaufgaben Rast-in-Rast und Geschwindigkeit-in-Geschwindigkeit in die Neufassung der VDI-Richtlinie 2143 aufgenommen worden. Im vorliegenden Beitrag wird gezeigt, wie man Berechnungsgleichungen für die verbliebenen Bewegungsaufgaben aufstellt und durch geeignete Parameterwahl stochastisch oder interaktiv einen guten Kennwertemix erzielen kann.

Double-S-Profiles for motion tasks in VDI 2143 / Abstract

So-called Double-S-Profiles allow to realize maximum values for velocity, acceleration and jerk over a time interval thus avoiding higher maximum values for these quantities. They have been included in the new edition of the VDI Guideline 2143 for specific motion tasks (rest-in-rest, constant velocity in constant velocity). In this contribution, it is shown how equations for the remaining motion tasks can be developed and how one can stochastically or interactively generate motion functions with a good mixture of characteristic values.

1. Einleitung

Die VDI-Richtlinie 2143 [1] bietet für die abschnittsweise Erstellung von Bewegungs- und Übertragungsfunktionen normierte Bewegungsgesetze an. Als Gütekriterien für die anwendungsgerechte Auswahl eines Gesetzes sind u.a. das Betragsmaximum der 1., 2. und 3. Ableitung sowie des Produkts aus 1. und 2. Ableitung genannt (C_v, C_a, C_j, C_{Mdyn}). Mit niedrigen Kennwerten kann man Bewegungen realisieren, die geringe Werte bei Maximalgeschwindigkeit bzw. -beschleunigung bzw. -ruck aufweisen und bezüglich Schwingungsanregung, Energieeffizienz oder erforderlichem Antriebsmoment günstige Eigenschaften besitzen [1, 2].

Für die Bewegungsaufgaben Rast-in-Rast (RR) und Geschwindigkeit-in-Geschwindigkeit (GG) sind in die Neufassung der VDI-Richtlinie 2143 sogenannte Doppel-S-Profile [3] aufgenommen worden, die es ermöglichen, Maximalwerte von Geschwindigkeit, Beschleunigung und Ruck über einem ganzen Intervall konstant zu halten und damit höhere Spitzenwerte zu vermeiden. In [4] wird gezeigt, dass sich mit den Doppel-S-Profilen in der GG-Bewegungsaufgabe je nach Randwertkonfiguration günstigere Kennwerte als bei den Vergleichsgesetzen realisieren lassen. Im vorliegenden Beitrag wird dargelegt, wie man Gleichungen für die Berechnung von Doppel-S-Profilen für die verbliebenen Bewegungsaufgaben der Richtlinie erstellt, so dass nun auch für diese Aufgaben neben dem Polynom 5. Grades, der harmonischen Kombination und dem modifizierten Beschleunigungstrapez derartige Bewegungsgesetze zur Verfügung stehen. Für die Rast-in-Umkehr Bewegungsaufgabe (RU) werden konkret die Berechnungsgleichungen angegeben und gelöst. Dabei sind bis zu fünf Parameter (in Grenzen) frei wählbar vorzugeben, wobei nicht alle Vorgaben zu zulässigen Lösungen führen. Daher werden dem Leser auf Basis einer Analyse der Lösungsformeln und der Entstehung unzulässiger Lösungen Handlungsanweisungen für das interaktive Vorgehen bei der Lösungsfindung gegeben. Ferner wird gezeigt, wie man durch stochastische Lösungssuche einen Lösungsraum erzeugen kann, aus dem sich der Leser eine Anfangsnäherung mit akzeptablen Kennwerten zur weiteren interaktiven Verbesserung aussuchen kann. Anhand von Beispielen wird nachgewiesen, dass man mittels der Doppel-S-Profile gegenüber den bisher zu Verfügung stehenden Bewegungsgesetzen einzelne, manchmal auch alle Kennwerte verbessern und einen größeren Einfluss auf den Kennwertemix nehmen kann.

2. Doppel-S-Profile in der Richtlinie 2143

In Formel (1) und Abbildung 1 ist die stückweise konstante 3. Ableitung des Doppel-S-Profiles für die Bewegungsaufgabe GG angegeben [1, 3]. Wählt man dort die Werte für j_1 und j_2 sowie die Abschnittslängen so, dass die Funktion symmetrisch zu $z = 0.5$ ist, so erhält man das Doppel-S-Profil für die RR Aufgabe mit $Z_{j1} = Z_{j2} = Z_j$.

$$f'''(z) = \begin{cases} j_1 & \text{für } 0 \leq z < z_1 \\ 0 & \text{für } z_1 \leq z < z_2 \\ -j_1 & \text{für } z_2 \leq z < z_3 \\ 0 & \text{für } z_3 \leq z < z_4 \\ -j_2 & \text{für } z_4 \leq z < z_5 \\ 0 & \text{für } z_5 \leq z < z_6 \\ j_2 & \text{für } z_6 \leq z < 1 \end{cases} \quad (1)$$

Bei der Aufgabe RR können die Werte für $Z_j = Z_{j1} = z_1$ und $Z_a = z_3$ vorgegeben werden, wobei $Z_j \leq \frac{Z_a}{2}$ und $Z_a \leq 1$ gelten muss. Alternativ kann man auch die Werte für C_v und C_j vorgeben, wobei diese so zu wählen sind, dass Z_j reell und die Bedingungen für Z_j und Z_a erfüllt sind. Aus [1, Gl. (194)] folgt $Z_j = \frac{Z_a}{2} - \sqrt{\frac{Z_a^2}{4} - \frac{C_v}{C_j}}$, d.h. wählt man C_j groß genug, so kann man die Bedingungen sicher erfüllen.

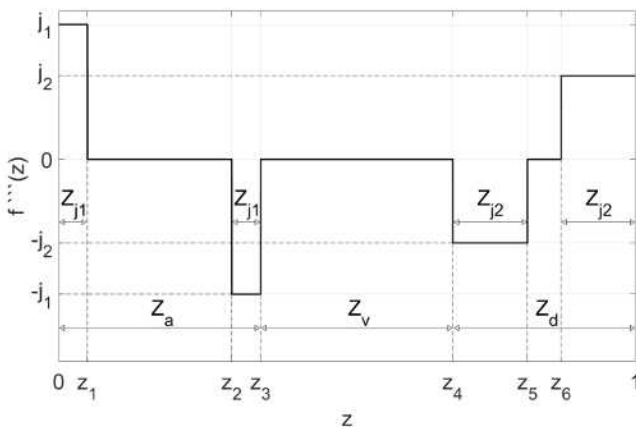


Abb. 1: Normiertes Doppel-S-Profil für GG: 3. Ableitung

Bei der Aufgabe GG kann man zunächst die Werte für $q_{ad} = \frac{Z_a}{Z_d}$ und für das Maximum der 1. Ableitung f'_{max} vorgeben. Damit lassen sich die Abschnittslängen Z_a, Z_v, Z_d bestimmen [3]. Ergibt sich dabei ein negatives Z_v , so ist f'_{max} zu verringern. Des Weiteren sind die Werte für j_1 und j_2 vorzugeben, wobei in [1] Untergrenzen angegeben sind, die $Z_{j1} \leq \frac{Z_a}{2}$ und $Z_{j2} \leq \frac{Z_d}{2}$ sicherstellen. Man kann hier also durch Beachtung der Vorgaben und Hinweise zügig eine gültige Parameterkonfiguration finden.

3. Doppel-S-Profil in der RU-Aufgabe

Das zuvor beschriebene Vorgehen der Entwicklung von Bewegungsgesetzen aus einer stückweise konstanten 3. Ableitung lässt sich auf weitere Bewegungsaufgaben übertragen, wie nachfolgend am Beispiel der RU Aufgabe gezeigt wird. Formel (1) gilt weiterhin (siehe Abb. 2), wobei die Abschnittslängen Z_{j21} und Z_{j22} aber in der Regel nicht gleich sind. Durch abschnittsweise Integration erhält man daraus unter Berücksichtigung der RU spezifischen Randbedingungen ($f(0) = 0, f(1) = 1, f'(0) = f'(1) = 0, f''(0) = 0, f''(1) = -a_U$) die Funktionsverläufe der Bewegungsgesetze 0. bis 2. Ordnung.

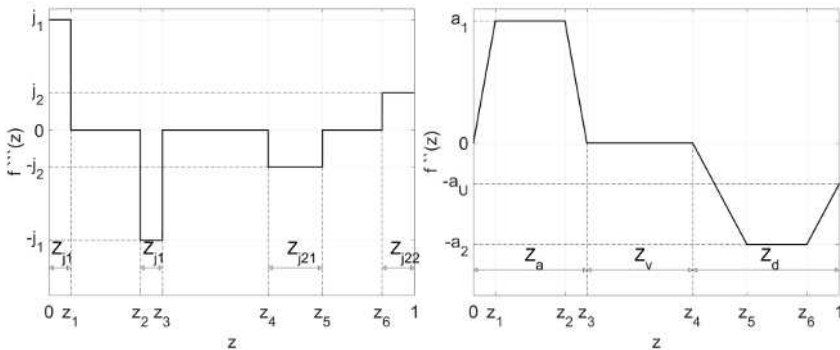


Abb. 2: Normiertes Doppel-S-Profil für RU: Bewegungsgesetze 3. (links) und 2. Ordnung (rechts)

Durch Betrachtung der Verläufe können insgesamt sieben Gleichungen formuliert werden, die die Zusammenhänge zwischen Abschnittslängen und Extrema beschreiben: Formel (2) ergibt sich aus Abb. 2 (rechts), die Formeln (3) und (4) aus den mit den Beträgen der normierten

Beschleunigung korrelierenden Flächeninhalten unter dem Graphen von $f'''(z)$ (vgl. Abb. 2 links). Formel (5) ergibt sich durch die Summation aller orientierten Flächeninhalte unter dem Graphen von $f'''(z)$. Aus $f'(1) = 0$ folgt in Formel (6), dass unter dem Graphen von $f'''(z)$ der orientierte negative Flächeninhalt des Verzögerungsbereichs Z_d denselben Betrag wie der orientierte positive Flächeninhalt des Beschleunigungsbereichs Z_a haben muss. Dieser positive Flächeninhalt entspricht wiederum f'_{max} (vgl. Formel (7)). Formel (8) ergibt sich aus der dreifachen Integration von Formel (1) zum Bewegungsgesetz 0. Ordnung und Einsetzen der Randbedingung $f(1) = 1$.

$$Z_a + Z_v + Z_d = 1 \quad (2)$$

$$a_1 = j_1 Z_{j1} \quad (3)$$

$$a_2 = j_2 Z_{j21} \quad (4)$$

$$a_U = -j_2 (Z_{j22} - Z_{j21}) \quad (5)$$

$$f'_{max} = \frac{1}{2} a_2 Z_{j21} + a_2 (Z_d - Z_{j21} - Z_{j22}) + \frac{1}{2} (a_2 + a_U) Z_{j22} \quad (6)$$

$$f'_{max} = a_1 \cdot (Z_a - Z_{j1}) \quad (7)$$

$$\begin{aligned} & -\frac{j_1 Z_a^2 Z_{j1}}{2} + \frac{j_1 Z_a Z_{j1}^2}{2} + j_1 Z_a Z_{j1} - \frac{j_2 Z_d^2 Z_{j21}}{2} \\ & + \frac{j_2 Z_d Z_{j21}^2}{2} - j_1 Z_{j1}^2 - \frac{j_2 Z_{j21}^3}{6} \\ & + \frac{j_2 Z_{j22}^3}{6} = 1 \end{aligned} \quad (8)$$

Die enthaltenen Größen müssen alle reell sein und darüber hinaus folgenden Bedingungen genügen:

$$0 \leq Z_a, Z_v, Z_d, Z_{j1}, Z_{j21}, Z_{j22} \leq 1 \quad (9)$$

$$a_1, a_2, a_U, f'_{max}, j_1, j_2 \geq 0 \quad (10)$$

$$Z_{j1} \leq \frac{Z_a}{2} \quad (11)$$

$$Z_{j21} \geq Z_{j22} \quad (12)$$

$$Z_d \geq Z_{j21} + Z_{j22} \quad (13)$$

$$a_1 \leq j_1 \quad (14)$$

$$a_2 \leq j_2 \quad (15)$$

$$a_U \leq a_2 \quad (16)$$

Da in den Formeln (2) bis (8) insgesamt zwölf unterschiedliche Größen vorkommen und sieben Gleichungen vorhanden sind, werden fünf

Größen $(a_1, a_2, a_U, f'_{max}, j_1)$ ausgewählt, die vom Anwender vorzugeben sind, und die sieben verbleibenden Größen $(j_2, Z_a, Z_v, Z_d, Z_{j_1}, Z_{j_{21}}, Z_{j_{22}})$ als unbekannt angenommen. Mit der Auswahl der angegebenen Größen kann insbesondere direkt auf die geschwindigkeits- und beschleunigungsrelevanten Kennwerte Einfluss genommen werden. Es liegt dann ein nichtlineares 7x7-Gleichungssystem vor, das mit Hilfe eines Computeralgebrasystems (z.B. Symbolic Math Toolbox in Matlab® oder Maple®) nach den unbekanntenen Größen gelöst werden kann, um diese in Abhängigkeit von den vorgegebenden Größen zu beschreiben. Anschließend können gegebene und gesuchte Größen in die Bewegungsgesetze 0. bis 3. Ordnung eingesetzt werden, um den Vorgaben des Anwenders entsprechende Funktionsverläufe zu erzeugen.

Statt der angegebenen Auswahl der vorgegebenen Größen können auch in Analogie zum Vorgehen bei der GG Bewegungsaufgabe als weitere Größe das Verhältnis $q_{ad} = \frac{z_a}{z_d}$ eingeführt und vorgegeben und dann die ruckbezogenen Größen j_1 und j_2 anstelle von a_1 und a_2 in der Vorgabe genutzt werden. In diesem Beitrag wird nur die Vorgabe von $(a_1, a_2, a_U, f'_{max}, j_1)$ betrachtet, weitere Varianten werden in [5] untersucht.

Da für RU einige von RR und GG bekannte, aus der Symmetrie der Graphen der Bewegungsgesetze resultierende Vereinfachungen entfallen, ist es schwieriger, gültige und insbesondere bezüglich der Kennwerte gute Parameterbesetzungen (Werte für die vorgegebenden Größen) zu finden. Um zu gültigen Parameterbesetzungen zu gelangen und diese nach dem Ermessen des Anwenders optimieren zu können, wurden zwei Methoden entwickelt:

- **Stochastischer Ansatz:** Für die vorgegebenden Größen werden Zufallszahlen innerhalb eines als technisch relevant erachteten Bereichs generiert. Mit diesen Zufallstupeln werden mithilfe der sich aus den Gleichungen (2) bis (8) ergebenden Berechnungsformeln die gesuchten Größen ermittelt. Anschließend erfolgt für jedes Tupel anhand der Formeln (9) bis (16) eine Prüfung, ob alle Größen innerhalb des jeweils für sie zulässigen Wertebereichs liegen. Tupel, die dies nicht erfüllen, werden aussortiert, und an den verbleibenden Tupeln wird eine Kennwertbestimmung für C_a , C_v , C_j und $C_{M_{dyn}}$ vorgenommen. Die ausgewählten Kennwerte der gültigen Tupel werden anschließend in einem Diagramm (siehe Abb. 3) dargestellt. Der Anwender kann sich damit einen visuellen Eindruck des

- Lösungsraums verschaffen und bei Betrachtung mithilfe eines Mathematik-Programms bereits gezielt ein bestimmtes Tupel (als Anfangsnäherung) auswählen.
- **Interaktiver Ansatz:** Anhand einer Analyse der mathematischen Struktur der Lösungsformeln kann antizipiert werden, welche Fehler für die Werte der gesuchten Größen (d.h. nicht-reelle Werte oder Verletzungen von (9) bis (16)) in Abhängigkeit von den Besetzungen der vorgegebenden Größen zu erwarten sind. Ebenso lässt sich beweisen, dass unter Berücksichtigung bestimmter Grenzen für die vorgegebenden Größen einzelne Fehlerarten im Vorfeld ausgeschlossen sind oder gleichzeitig eliminiert werden. Ergänzend lassen sich aus den Lösungsformeln teils logisch, teils heuristisch begründete Hinweise zur Anpassung der gesuchten mithilfe der gegebenen Größen formulieren, um zu einer gültigen Lösung zu kommen (Beispiele siehe unten). Alternativ kann anhand dieser Hinweise auch eine mit dem stochastischen Ansatz ermittelte Anfangsnäherung verbessert werden.

Der **stochastische Ansatz** wurde in Matlab® für die oben angegebene Variante mit Vorgabe von $(a_1, a_2, a_U, f'_{max}, j_1)$ folgendermaßen umgesetzt: Zunächst wird eine vom Anwender festgelegte Anzahl n an Zufallstupeln innerhalb fester Grenzen generiert. In Anlehnung an die Bilder 24 und 26 in [1] werden als Grenzen exemplarisch $a_1 \in [0; 15]$, $a_2 \in [0; 15]$, $a_U \in [0; 15]$, $f'_{max} \in [0; 15]$ und $j_1 \in [0; 300]$ gewählt. Anschließend werden die gesuchten Größen wie angegeben berechnet und auf Gültigkeit untersucht. Für die gültigen Tupel ergeben sich die Kennwerte wie folgt:

$$C_a = \max \{a_1, a_2\} \quad (17)$$

$$C_v = f'_{max} \quad (18)$$

$$C_j = \max \{j_1, j_2\} \quad (19)$$

$$C_{M \text{ dyn}} = \max |f'(z) \cdot f''(z)| \quad (20)$$

Die Bestimmung von $C_{M \text{ dyn}}$ erfolgt durch Abtastung der Betragsfunktion aus Formel (20) mit der Schrittweite $\Delta z = 0,01$, da für RU keine gesonderte analytische Beschreibung der Berechnungsformel für $C_{M \text{ dyn}}$ existiert. Im vorliegenden Beispiel werden für 10000 zufallsgenerierte,

gültige Tupel Kennwerte bestimmt. Erstellt man von diesen Kennwerten einen Scatter-Plot in Matlab® (vgl. Abb. 3 links), wobei drei Größen auf den Achsen abgetragen werden und die vierte (in Abb. 3: $C_{M\ dyn}$) farbkodiert wird, ist eine schalenartig aufgebaute Punkt看ke zu erkennen. Insbesondere in Bereichen, in denen C_v , C_a und C_j gleichzeitig besonders niedrig wären, liegen keine gültigen Tupel vor. Ebenso werden nicht alle zuvor festgelegten Intervalle der vorzugebenden Größen komplett ausgereizt, da sich diese Größen gegenseitig so begrenzen, dass für sämtliche Tupel $C_v \in (1; 2)$ gilt, obwohl zuvor Zufallszahlen im Bereich $f'_{max} \in [0; 15]$ generiert wurden. Die Werte für $C_{M\ dyn}$ sind schalenartig in die Bildebene hinein gestaffelt. Die äußerste „Schale“ in dunklem Grau befindet sich dem Betrachter zugewandt vorne in der Bildebene und ist sichtbar gegen den lösungsleeren Raum abgegrenzt. In die Bildebene hinein steigt $C_{M\ dyn}$ immer weiter an. Ist ein ausgeglichener Kennwertemix erwünscht, bietet es sich daher an, Tupel aus diesem Bereich zu entnehmen. Um die umfangreiche und etwas unübersichtliche Punkt看ke auszudünnen, kann man unter den Kennwerttupeln diejenigen entfernen, bei denen alle Kennwerte größer gleich denjenigen eines anderen Tupels sind. Dann bleiben – bezogen auf die Menge der stochastisch gefundenen und gültigen Kennwerttupel – die so genannten Pareto-Optima übrig (vgl. [6]), die näherungsweise die „Pareto-Front“ darstellen (siehe Abb. 3, rechts). Dies erleichtert dem Benutzer die Auswahl.

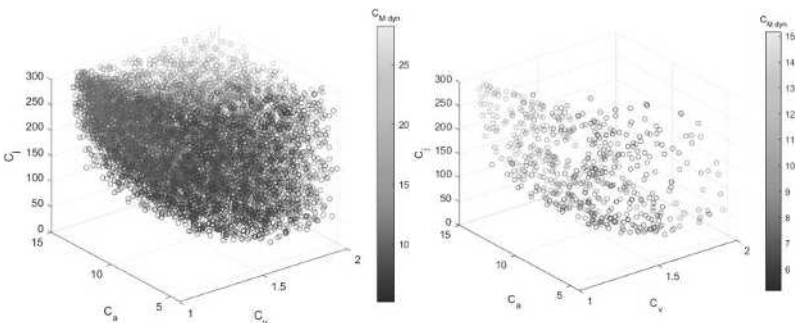


Abb. 3: Punkt看ke aus mit dem stochastischen Ansatz gefundenen Kennwerttupeln (links); bereinigte Punkt看ke aus Pareto-Optima (rechts)

Sollte kein Mathematikprogramm zur Realisierung des stochastischen Ansatzes zur Verfügung stehen, bietet sich die Verwendung des **interaktiven Ansatzes** an. Dieser basiert auf der Prüfung der

Bedingungen aus den Formeln (9) bis (16) in einer festen Abfolge durch den Anwender und der manuellen Anpassung der Parameterbesetzungen, falls diese nicht erfüllt sind. Durch logische Argumentation kann die Anzahl zu prüfender Bedingungen auf fünf reduziert werden. Zum Beispiel kann man schlussfolgern, dass, wenn a_1, f'_{max} und (gemäß vorherigem Beweis) Z_a reell sind, nach Formel (7) auch Z_{j1} reell sein muss, oder dass mit $j_1 \geq a_1 > 0$ nach Formel (3) auch $Z_{j1} \leq 1$ folgt (siehe dazu ausführlich [5]). In Kombination mit den Handlungshinweisen, wie die vorzugebenden Größen in jedem Schritt anzupassen sind, ergibt sich hieraus ein Algorithmus, der systematisch abgearbeitet werden kann (siehe Abb. 4).

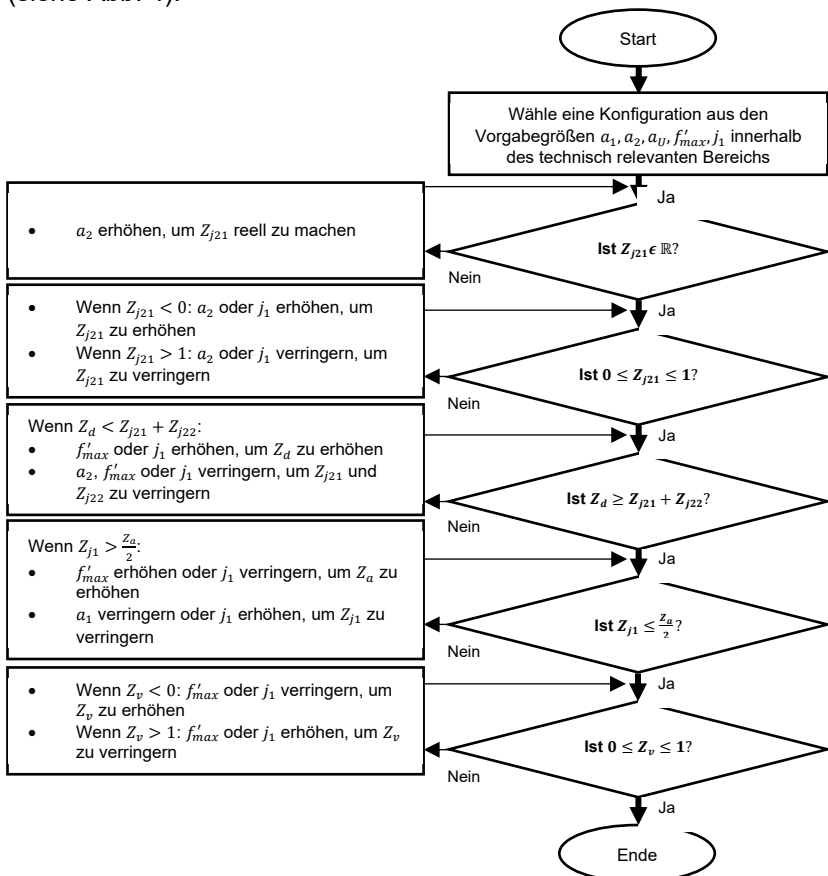


Abb. 4: Algorithmus für das interaktive Vorgehen beim Doppel-S-Profil für RU

Mit den beiden vorgestellten Ansätzen sind die Doppel-S-Profile für RU gut handhabbar. Darüber hinaus ist von Interesse, inwiefern mit diesen Bewegungsgesetzen eine weitere Verbesserung der Kennwerte gegenüber den in der Richtlinie vorgestellten Bewegungsgesetzen für diese Bewegungsaufgabe, der harmonischen Kombination und dem modifizierten Beschleunigungstrapez, möglich ist. Letztere enthalten noch jeweils einen Wendepunktparameter λ , der bei gegebener Umkehrbeschleunigung aber bereits festgelegt ist. Der Vergleich stochastisch ermittelter Kennwerttupel für die hier betrachtete Variante der Doppel-S-Profile mit den ausgeglichenen Kennwerten der beiden anderen Bewegungsgesetze bei $\lambda \approx 0,5$ zeigt, dass das Doppel-S-Profil mit einzelnen Tupeln bis zu drei Kennwerte entweder der harmonischen Kombination oder des modifizierten Beschleunigungstrapezes unterbieten kann, jedoch nicht beide Bewegungsgesetze gleichzeitig und auch nicht ein Bewegungsgesetz in allen vier Kennwerten gleichzeitig. Einen entscheidenden Vorteil bietet das Doppel-S-Profil bei Anwendungsszenarien, die es erfordern, dass sich der Wendepunktparameter von $\lambda \approx 0,5$ entfernt. In diesem Fall steigen der C_a - und $C_{M\ dyn}$ -Wert der harmonischen Kombination sowie beim modifizierten Beschleunigungstrapez zusätzlich der C_j -Wert stark an, sodass in solchen Fällen mit dem Doppel-S-Profil insgesamt niedrigere Kennwerte erzielt werden können. Ist beispielsweise am Umkehrtrand eine normierte Beschleunigung von $f_U'' = -3,4$ (also $a_U = 3,4$) gefordert, können stochastisch gleich mehrere Tupel des Doppel-S-Profils ermittelt werden, die die beiden anderen Bewegungsgesetze gleichzeitig in allen vier Kennwerten unterbieten (vgl. Tab. 2 und 3).

Tab. 2: Kennwerte der Vergleichsgesetze

Bewegungsgesetz	C_v	C_a	C_j	$C_{M\ dyn}$
Modifiziertes Beschleunigungstrapez ($\lambda = 0,3858$)	1,899	6,014	97,96	9,45
Harmonische Kombination ($\lambda = 0,254$)	1,615	9,986	247,0	9,047

Tab. 3: Kennwerte des Doppel-S-Profiles für RU

Bewegungsgesetz	C_v	C_a	C_j	$C_{M\ dyn}$
Doppel-S-Profil mit kleinstem C_v	1,42	5,93	84,54	6,63
Doppel-S-Profil mit kleinstem C_a	1,56	5,01	78,83	6,80
Doppel-S-Profil mit kleinstem C_j	1,60	5,57	39,28	6,62
Doppel-S-Profil mit kleinstem $C_{M\ dyn}$	1,47	5,97	79,87	6,38

Dies hat den weiteren Vorteil, dass dem Anwender beim Doppel-S-Profil in Grenzen noch Spielraum verbleibt, die verschiedenen Kennwerte gegeneinander abzuwägen und einen für das jeweilige Anwendungsszenario relevanten Kennwert noch etwas weiter abzusenken, ohne in einem anderen Kennwert deshalb schlechter abzuschneiden als die Vergleichsgesetze.

4. Übertragung auf weitere Bewegungsaufgaben

Bei der Übertragung auf weitere Bewegungsgesetze können Formel (1) sowie die bei der Definition des Doppel-S-Profiles für RU eingeführten Größen weitgehend beibehalten werden. Sollte $f''(0)$ vorgegeben sein, so sind innerhalb von Z_a statt der zwei Abschnitte mit Länge Z_{j1} zwei Abschnitte mit Längen Z_{j11} und Z_{j12} vorzusehen, die nicht gleich zu sein brauchen. Die Beziehungsgleichung lautet dann analog zu Formel (5):

$$f''(0) = j_1(Z_{j12} - Z_{j11}) \quad (21)$$

Bei der abschnittweisen Integration von (1) sind jeweils die Anfangsbedingungen (also $f'(0)$, $f''(0)$) zu berücksichtigen, so dass sich auch $f(z)$ ändert. In den Gleichungen (6) bzw. (7) ist f'_{max} durch den Ausdruck $f'_{max} - f'(1)$ bzw. $f'_{max} - f'(0)$ zu ersetzen. Die einzuhaltenden Bedingungen (9)-(16) sind entsprechend zu erweitern. Der stochastische Ansatz lässt sich problemlos auf andere Bewegungsaufgaben übertragen, während beim interaktiven Ansatz die Argumentation bezüglich des Einflusses gewisser Größen aufgrund komplexerer Ausdrücke schwieriger wird.

5. Zusammenfassung und Ausblick

Im vorliegenden Beitrag wird dargelegt, wie man Doppel-S-Profile für alle Bewegungsaufgaben der VDI-Richtlinie 2143, die solche Profile bislang nur für die Aufgaben RR und GG enthält, aufstellen und somit die

Richtlinie erweitern kann. Das Problem, für die Parameter Werte zu finden, die zu einer gültigen Konfiguration führen, wird anhand der Bewegungsaufgabe RU durch einen stochastischen und einen interaktiven Ansatz mit Hinweisen sinnvoller Parametervariationen gelöst. Darstellungen in Matlab®, die direkt die zugehörige Kennwertkonfiguration in einem Plot angeben, ermöglichen es, schnell zu einer Konfiguration mit anwendungsgerechtem Kennwertemix nahe der Pareto-Front zu gelangen. Der stochastische Ansatz lässt sich problemlos von der RU Aufgabe auf die weiteren Bewegungsaufgaben übertragen. Es bleibt zu untersuchen, ob sich auch entsprechende Hinweise für die interaktive Vorgehensweise formulieren lassen, wenn noch weitere Parameter wie Randableitungen hinzukommen, die die Formelanalyse erschweren. Hier könnte ein KI-basierter Ansatz hilfreich sein, der Strategien anhand von vielen Versuchen erlernt.

Literatur

- [1] VDI 2143 Bewegungsgesetze für Kurvengetriebe und Motion-Control-Systeme: Blatt 1. Gründruck, Berlin: Beuth-Verlag 2023.
- [2] Speicher, Th. Kennwertorientierte Methoden zur Auswahl und Synthese von Bewegungsgesetzen für Motion-Control-Systeme. Chemnitz: Universitätsverlag Chemnitz 2021.
- [3] Alpers, B., Berger, M. und Speicher, Th. Zur Einbeziehung zeitoptimierter Bewegungsfunktionen in den abschnittsorientierten Ansatz der VDI Richtlinie 2143, in: Falkenstein, J., Woernle, Chr. (Hrsg.) 14. Kolloquium Getriebetechnik Berlin: Logos Verlag 2021, S. 23-32.
- [4] Alpers, B. Vergleich der GG-Übertragungsfunktionen aus der Neufassung der VDI Richtlinie 2143, in: Corves, B., Hüsing, M. (Hrsg.) 15. Kolloquium Getriebetechnik. Aachen: Verlag Mainz 2023, S. 23-43.
- [5] Dietterle, D. Normierte Doppel-S-Profile für Bewegungsaufgaben, Bachelorarbeit, Hochschule Aalen 2024.
- [6] Schumacher, A. Optimierung mechanischer Strukturen. Grundlagen und industrielle Anwendungen. Berlin – Heidelberg: Springer 2005.

Parameterstudie zur gezielten Rekonfiguration nachgiebiger Mechanismen

Vivien, Platl*; Lena, Zentner*

Technische Universität Ilmenau, Fachgebiet Mechanik Nachgiebiger Systeme

*vivien.platl@tu-ilmenau.de

Kurzfassung

Nachgiebige Mechanismen besitzen im gefertigten Zustand für ihre Anwendung optimale, aber unveränderliche Eigenschaften. Eine gezielte Rekonfiguration während ihres Einsatzes ermöglicht die Erschließung neuer Anwendungsgebiete. Deswegen wurden erste Parameterstudien durchgeführt, welche die Auswirkung einer gezielten kontinuierlichen Rekonfiguration auf die Eigenfrequenzen nachgiebiger Mechanismen aufzeigen sollen.

Title / Abstract

Parameter study for the targeted reconfiguration of compliant mechanisms

In their finished state, compliant mechanisms have optimal but unchangeable properties for their application. Targeted reconfiguration during their use would make it possible to open up new areas of application. For this reason, initial parameter studies were carried out to show the effect of targeted continuous reconfiguration on the natural frequencies of compliant mechanisms.

Methodik

Nachgiebige Mechanismen (NM) werden bei der Entwicklung gezielt hinsichtlich gewünschter funktionsgebender Eigenschaften optimiert, wodurch diese zwar optimal, aber unveränderlich sind. Unter anderem zählen dazu die Eigenfrequenzen, welche maßgeblich das dynamische Verhalten der NM beeinflussen [1]. Um künftig mehrere Anwendungsgebiete für einen einzelnen NM zu erschließen, wird eine Rekonfiguration angestrebt. Diese kann durch Vorspannung im NM oder mit dem Gestell, Nutzung des Durchschlageffekts, Verschieben/Verdrehen von Verbindungsstellen oder deren Freilassen/Verbinden erfolgen. Erste Parameterstudien in *CaTEf* [1] untersuchen die Auswirkungen einer Rekonfiguration, realisiert durch die Variation inhärenter Winkel, auf die Eigenfrequenzen. Sie dienen als Grundlage für zukünftige Untersuchungen zur kontinuierlichen Rekonfiguration und deren Auswirkungen auf die funktionsgebenden Eigenschaften von NM während des Einsatzes.

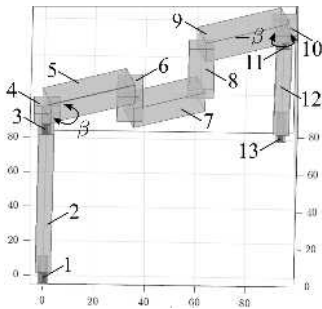


Abb. 1: Darstellung einer Rekonfiguration in *CaTEf*.

Tab. 1: Zusammenhang von Winkel und Eigenfrequenz.

β [°]	natural frequency [Hz]
± 0	6.07
± 10	6.14
± 20	6.37
± 30	6.75
± 40	7.26
± 45	7.55

Literatur

- [1] Platl, V. und Zentner, L. „An analytical method for calculating the natural frequencies of spatial compliant mechanisms“. In: *Mechanism and Machine Theory* 175 (2022), S. 17. doi: <https://doi.org/10.1016/j.mechmachtheory.2022.104939>.

Die Software Mechanism Developer - nun verfügbar für Lehre und Industrie

Vincent Brünjes*; Mathias Hüsing*; Burkhard Corves*

* RWTH Aachen, Institut für Getriebetechnik, Maschinendynamik und Robotik
bruenjes@igmr.rwth-aachen.de

Kurzfassung

Am Institut für Getriebetechnik, Maschinendynamik und Robotik der RWTH Aachen wird seit einiger Zeit die Software „Mechanism Developer“ (MechDev) zur interaktiven Auslegung und Analyse von ebenen Getrieben entwickelt [1-2] und dessen aktuelle Fortschritte regelmäßig publiziert [3-5]. Diese Software ist ab sofort allgemein verfügbar und lizenzierbar. MechDev wird sowohl für Lehrzwecke als auch für industrielle Anwendungen empfohlen. Kern der Software ist ein leistungsfähiges Analysetool zur kinematischen und kinetostatischen Analyse ebener ungleichmäßig übersetzender Getriebe, welches ständig weiterentwickelt wird.

Titel / Abstract

The Software Mechanism Developer – now available for Education and Industry.

The Institute of Mechanism Theory, Machine Dynamics and Robotics of RWTH Aachen University has been developing the "Mechanism Developer" (MechDev) software for the interactive design and analysis of planar transmissions for some time [1-2] and regularly publishes its current progress [3-5]. This software is now generally available to the public. MechDev is recommended for educational purposes as well as for industrial applications. The core features of the software constitute a

powerful toolkit for the kinematic and kinetostatic analysis of planar linkages, which is constantly being improved on.

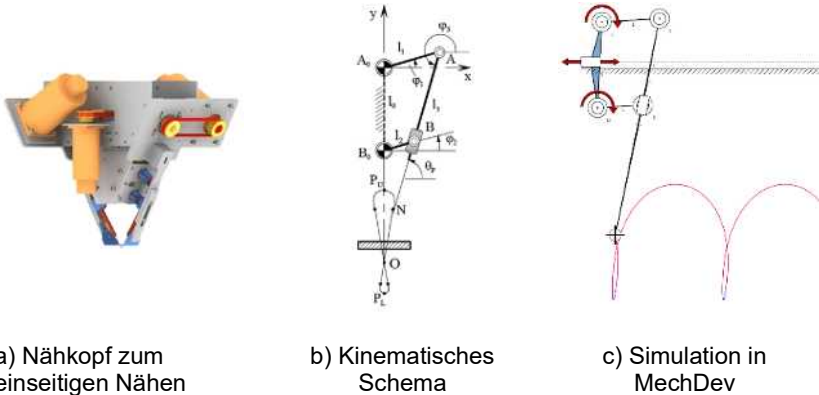
1 Die Software MechDev

Der Wunsch jedes Anwenders ist, die komplexen Problemstellungen bei Analyse und der Auslegung von ungleichmäßig übersetzenden Getrieben möglichst intuitiv und leicht handhabbar zu lösen. Der Entwickler möchte dann die Ergebnisse geeignet nutzen. Genau dieses soll MechDev in idealer Weise bieten.

Durch den Programmaufbau ist die Einarbeitungszeit in MechDev äußerst gering. Es bedarf keiner umfangreicher Softwareerfahrung um an den Kern, nämlich das Verständnis und die Auslegung des Getriebes vorzudringen. Das macht den Einstieg und auch den Wiedereinstieg sowohl in einer Getriebeauslegung als auch in der Software verblüffend einfach und effizient. Damit eignet sich die Software hervorragend für kinematisch unerfahrene Personen und Studenten. Die leicht zugänglichen Features von MechDev bieten dem erfahrenen Kinematiker einen idealen Werkzeugkoffer zur Lösung komplexer Aufgabenstellungen. Ein Produktentwickler kann auf seine gewohnten Tools bauen und gleichzeitig mit MechDev den Auslegungsspielraum erweitern.

Mit Mechdev können nicht nur ebene ungleichmäßig übersetzende Koppelgetriebe, sondern auch Kurvengetriebe und Räder bzw. Koppelrädergetriebe behandelt werden. Zur Synthese von Kurvengetrieben stehen gängige Bewegungsgesetze zu Verfügung. Diese bilden auch eine ideale Auslegungsbasis für Servoantriebe.

In diesem Beitrag wird die Software MechDev interaktiv vorgestellt. Dabei wird gezeigt, welche Möglichkeiten zur Analyse sowie der interaktiven Auslegung bestehen. Durch wenige Eingaben kann damit ein Urteil über die mögliche Verwendung eines Getriebes getroffen werden. Dies wird im Vortrag anhand eines Anwendungsbeispiels aus dem Bereich der einseitigen Nähetechnik vorgestellt, siehe Abb. 1.



a) Nähkopf zum
einseitigen Nähen

b) Kinematisches
Schema

c) Simulation in
MechDev

Abb. 1: Nadelführungsgetriebe

Literatur

- [1] Müller, M., Mannheim, T., Hüsing, M., Corves, B. „MechDev—A New Software for Developing Planar Mechanisms”. In: *Mechanisms and Machine Science*. Interdisciplinary Applications of Kinematics. Hrsg. von A. Kecskeméthy, F. Geu Flores, E. Carrera und D. Elias. Vol. 71. Springer, Cham, 2019, S. 167-175. isbn: 978-3-030-16423-2. doi: 10.1007. url: <https://rdcu.be/dGvJ4> (besucht am 29.04.2024)
- [2] Corves, B., Huesing, M., Müller, M. „A Model-View-Controller-Based Software Approach for the Interactive Design of Planar Mechanisms”. In: *Mechanism and Machine Science. Lecture Notes in Mechanical Engineering*. Hrsg. von Dibakar Sen, Santhakumar Mohan und Gondi Kondaiah Ananthasuresh. Springer, Singapur, 2020, S. 255–266. isbn: 978-981-15-4476-7. doi: 10.1007. url: <https://rdcu.be/dGeZ1> (besucht am 29.04.2024).
- [3] Beckermann, A., Müller, M., Hüsing, M., Corves, B. „Analytical Kinematic Analysis of Cam Mechanisms in MechDev”. In: *Advances in Mechanism and Machine Science*. IFToMM WC 2019. Hrsg. von T. Uhl. Vol 73. Springer, Cham, 2019, S. 429-438. isbn: 978-3-030-20131-9. doi: 10.1007. url: <https://rdcu.be/dGvxs> (besucht am 29.04.2024).
- [4] Knobloch, T., Brünjes, V., Hüsing, M. und Corves, B. „Analyse von Koppel-, Kurven-, Räder- und Räderkoppelgetrieben mit Mechanism

Developer (MechDev)“. In: *Tagungsband Getriebetagung 2022*. Getriebetagung 2022. Hrsg. von Maik Berger, Burkhard Corves und Tim Lüth. Logos Verlag GmbH, Berlin, 2022, S. 23-31. isbn: 978-3-8325-5552-8. doi: 10.30819. url: <https://www.logos-verlag.de/cgi-bin/engbuchmid?isbn=5552> (besucht am 29.04.2024).

- [5] Knobloch, T., Brünjes, V., Hüsing, M. und Corves, B. „Auslegung ebener Getriebe mit der Software Mechanism Developer (MechDev)“. In: *12. SAXSIM - SAXON SIMULATION MEETING*. 12. Anwendertreffen SAXSIM. Hrsg. von Maik Berger. Universitätsverlag Chemnitz, 2023. isbn: 978-3-96100-112-5. doi: 10.51382. url: <https://nbn-resolving.org/urn:nbn:de:bsz:ch1-qucosa2-852861> (besucht am 29.04.2024).

Transversalsymmetrische Gelenke im Berechnungstool "CoMSys"

Hannes, Jahn*; Lena, Zentner*

* Technische Universität Ilmenau, Fachgebiet Mechanik Nachgiebiger Systeme
hannes.jahn@tu-ilmenau.de

Kurzfassung

Vorgestellt wird eine Weiterentwicklung des Berechnungstools *CoMSys* [1] zur Analyse und Optimierung von nachgiebigen Mechanismen. Das auf der Theorie großer Verformungen stabförmiger Strukturen basierende Programm ist ein effizientes Werkzeug um nachgiebige Mechanismen hinsichtlich ihrer Verformung zu berechnen. Neben den derzeit implementierten vollsymmetrischen Gelenken sollen auch transversalsymmetrische Gelenke implementiert werden. Diese werden mit Hilfe einer bereits vorgestellten Methode [2] in die Benutzeroberfläche integriert.

Title / Abstract

Transversely symmetrical hinges in the "CoMSys" calculation tool
A further development of the calculation tool *CoMSys* [1] for the analysis and optimization of compliant mechanisms is presented. Based on the theory of large deformations of rod-like structures, *CoMSys* is an efficient tool for calculating the deformation of compliant mechanisms. In addition to the currently implemented fully symmetric hinges, transversely symmetric hinges are also to be implemented. These are integrated into the user interface using previously described methods [2].

Methodik

Für die Berechnung transversalsymmetrischer Gelenke wird die Balkenachse (siehe Abb. 1 (a), orange) entlang der halben Höhe des Gelenks approximiert [2]. Basierend darauf werden die Krümmung und die Höhe des Gelenks in Abhängigkeit davon berechnet. Um den Übergang in der Balkenachse zwischen den benachbarten Abschnitten und dem Gelenk korrekt zu beschreiben, wird ein Hilfselement (Abb. 1 (a), Balkenachse in rot) mit unendlicher Steifigkeit eingeführt. Dadurch können verschiedene Gelenkhöhen und -radien des Gelenks festgelegt werden. Durch die Implementierung dieser automatisierten Gelenkberechnung in *CoMSys* können auf einfachste Weise unterschiedlichste Mechanismen realisiert und durch das Lösen eines nichtlinearen Gleichungssystems ausgewertet werden. Eine mögliche Berechnung ist in Abb. 1 (b) angedeutet.

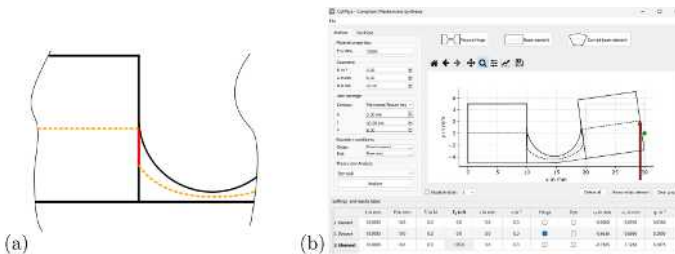


Abb. 1: Transversalsymmetrisches Gelenk: (a) darstellung der neutralen Faser; (b) die GUI *CoMSys* mit verformten Gelenk.

Gefördert durch die Deutsche Forschungsgemeinschaft (DFG) - (ZE 714/22-1)

Literatur

- [1] Jahn, H., Henning, S. und Zentner, L. *CoMSys - ein GUI-basiertes Berechnungstool zur Analyse und Optimierung nachgiebiger Mechanismen*. 2022. doi: 10.17185/DUEPUBLICO/75432.
- [2] Jahn, H., Fröhlich, T. und Zentner, L. *Analytical Description of Transversally Symmetric Hinges with Semicircular Contours*. 2024. doi: 10.1007/978-3-031-45709-8_49.

Einordnung der IKBT und Anwendung mit Blender – eine Methode zur Generierung der analytischen IK serieller Roboter

Jascha, Paris

Demcon, Jascha.Paris@Demcon.com

Kurzfassung

Die Berechnung der Inversen Kinematik (IK) bildet zusammen mit einer grafischen Repräsentation des kinematischen Systems eine Basis für die Auslegung, Simulation und Steuerung von Robotern. Die vorteilhafte, analytische Lösung der IK industrieller Roboter ist in der Literatur bekannt, dennoch zeigen Zhang und Hannaford mit der IKBT einen weiteren Ansatz für automatisch generierte, kompakte, analytische Lösungen. Dieser Beitrag bespricht die IKBT und ordnet sie im Vergleich zu anderen Ansätzen ein. Anschließend wird gezeigt, wie die IKBT zusammen mit Blender für die kinematische Analyse im System Engineering genutzt werden kann.

Title / Abstract

Review of IKBT and application with Blender – a method for generating the analytical IK of serial robots

The calculation of the inverse kinematics (IK) together with a graphical representation of the kinematic system forms a basis for the design, simulation and control of robots. The advantageous, analytical solution of the IK of industrial robots is known in literature, but Zhang and Hannaford show another approach for automatically generated, compact,

analytical solutions with the IKBT. This article discusses the IKBT and compares it to other approaches. Furthermore, it is shown how the IKBT can be used together with Blender for kinematic analysis within system engineering.

1 Einführung

Die Berechnung der Inversen Kinematik (IK) bildet die Basis der Auslegung, Simulation und Steuerung von Robotern. Bei vielen Anwendungen ist es vorteilhaft, wenn eine analytische Lösung der IK genutzt werden kann. Insbesondere trifft dies auf numerische Optimierungen zu, bei denen die gleiche Aufgabe vielfach berechnet werden muss.

Zur Analyse von Automatisierungssystemen mit industrieller Robotik kann der Entwicklungs- oder Systemingenieur auf viele verschiedene Softwarelösungen zurückgreifen, von denen einige in Abschnitt 2.1 vorgestellt werden. Soll explizit eine analytische Lösung oder eine Implementierung außerhalb verfügbarer Software – bspw. zur Anwendung eigener Algorithmen – verwendet werden, kann auf vorhandene Lösungen der IK in der Literatur zurückgegriffen werden, siehe [1].

Dennoch wird im populären open-source Robot Operating System (ROS) zumeist auf numerische Lösungen mittels IKFast (Zitationen: 721, GitHub-Sterne von MoveIt und Rave: 2288) [2] zurückgegriffen.

Zhang und Hannaford hingegen versprechen mit der IKBT und Verhaltensbäumen direkt analytische Lösungen und den passenden Quellcode für serielle Roboter, die mittels DH Parameter beschrieben werden, zu erstellen [3]. Dies ist als Software-Werkzeug vielversprechend, aber bisher nicht weit verbreitet (Zitationen: 21, GitHub-Sterne: 175). Daher wird der Ansatz in diesem Beitrag besprochen und in Abschnitt 2.2 vorgestellt. In Abschnitt 4.1 wird die IKBT mit anderen Lösungen verglichen und hinsichtlich Richtigkeit sowie Leistungsfähigkeit evaluiert. Abschließend wird der Nutzen der IKBT im System Engineering in Abschnitt 4.2 diskutiert.

Um einen vollständigen Ansatz zu präsentieren, wird zusätzlich die Visualisierung und kinematische Analyse mit dem open-source Programm Blender vorgestellt, siehe Abschnitt 3.1. Es wird in Abschnitt 3.2 ein

Algorithmus gezeigt, mit dem die DH-Parameter für die IKBT aus 3D-Modellen berechnet werden können. Zusammen kann Blender mit der IKBT für die effiziente Analyse von Robotikanwendungen verwendet werden.

2 Stand der Technik

Innerhalb dieses Abschnitts wird zunächst ein Überblick über Anwendungen zur Bewegungsanalyse von Robotern im industriellen Umfeld gegeben, siehe Abschnitt 2.1. Anschließend wird das Softwarewerkzeug IKBT von Zhang und Hannaford [3] vorgestellt sowie besprochen, siehe Abschnitt 2.2.

2.1 Software zur Bewegungsanalyse von Robotern

Alle großen Roboterhersteller bieten eigene Software zur Programmierung und Simulation ihrer Roboter an, siehe Abb. 1 für einige Beispiele. Vorteilhaft ist, dass die exakten Roboterdaten und Randbedingungen hinterlegt sind. Nachteilig ist, dass es sich meist um geschlossene Systeme handelt. Nicht immer werden alle Daten zur Analyse ausgegeben und die Modellierung der Umgebung muss importiert und dann in der jeweiligen Software angepasst werden. Zudem werden meist nur die firmeneigenen Roboter unterstützt, sodass mehrere Programme erlernt und ggf. mehrere Lizenzen erworben werden müssen.



(a) Robot Studio [4]

(b) Kuka Sim[5]

(c) MotoSim [6]

Abb. 1: Software von Roboterherstellern zur Programmierung und Simulation

Darüber hinaus gibt es Software von Drittanbietern, die viele verschiedene Roboter vordefiniert enthalten und auch zur Simulation sowie

teilweise zur Programmierung verwendet werden können, siehe Abb. 2 für einige Beispiele. Vorteilhaft ist dabei, dass nur ein Programm für verschiedene Robotersysteme verwendet wird und häufig mehr Daten ausgegeben werden können. Weiterhin ist es aber nur begrenzt möglich die Software z.B. in eigenen Optimierungs-, KI-Trainings- oder Steuerungsumgebungen zu nutzen.

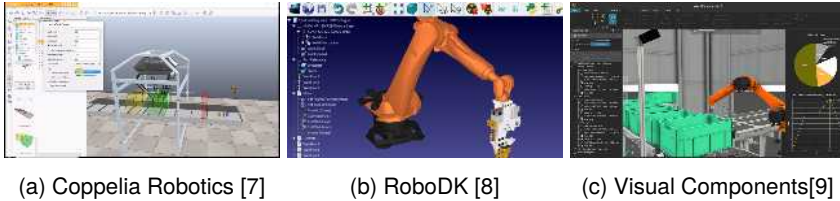


Abb. 2: Allgemeine Software zur Simulation und Programmierung von Robotern

Neben vollständiger Anwendungssoftware gibt es verschiedene open-source Werkzeuge zur Analyse von Robotern. Einige Beispiele sind OMPL [10], OpenRave [11] und das in ROS mit IKFast von OpenRave genutzte MoveIt [12]. Diese Werkzeuge sind flexibel, erfordern aber mehr Einarbeitung. MoveIt und IKFast werden innerhalb der ROS Umgebung genutzt, sodass viele Abhängigkeiten installiert werden müssen.

2.2 IKBT – Verhaltensbäume zur Lösung der inversen Kinematik

Die IKBT ist ein Expertensystem zur Lösung der IK mittels Verhaltensbäumen, das von Zhang und Hannaford entwickelt wird [3]. Die Abb. 3 gibt einen Überblick über den Verhaltensbaum. Die IKBT enthält bekannte Lösungsstrategien für einzelne Aufgabenstellungen der IK. In der Anwendung erstellt die IKBT zunächst einen Abhängigkeitsgraph der Gelenke. Daraus werden Gruppen gebildet und dann einzeln gelöst. Die gewonnene analytische Lösung wird abschließend mit einer numerischen Lösung verglichen und so verifiziert.

Die Ausgabe des Expertensystems ist ein Bericht als PDF und die IK als Quellcode für Python und C++. Es handelt sich also nicht um eine eigenständige Software, sondern vielmehr um die Generierung

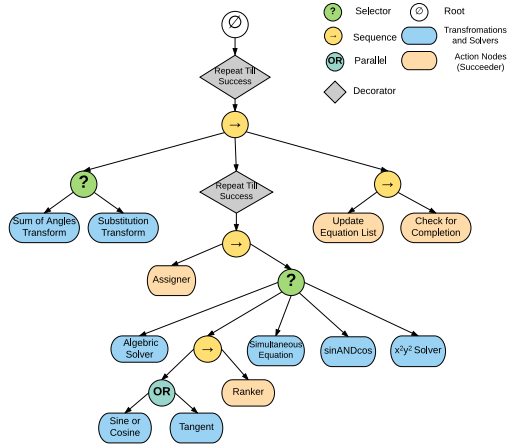
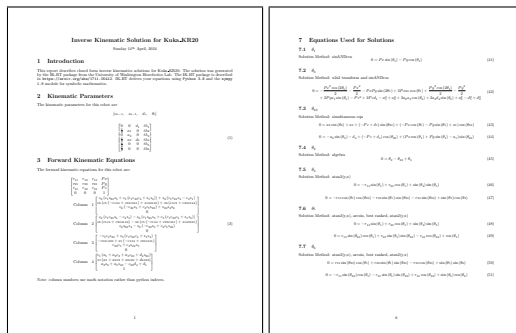


Abb. 3: Übersicht zur Struktur der IKBT [3]

von Code, der in andere Softwareumgebungen eingebunden werden kann. Der Bericht enthält die Aufgabenstellung, die Herleitung und das Ergebnis, siehe Abb. 4 für einen Auszug aus dem Bericht. Dadurch ist gut nachvollziehbar wie die IK gelöst wird.



(a) [7]

(b) [9]

Abb. 4: Auszug aus der dokumentierten Lösung der IK des KR20

Seit Juni 2024 steht eine aktualisierte Version der IKBT bereit [13], wobei die Neuerungen zum Teil noch nicht vollständig implementiert sind. Die

Autoren stellen selbst fest, dass die Lösung der IK tatsächlich mehr einem allgemeinen Graph als einem Baum entspricht. Dies führt bei der ursprünglichen IKBT zu unnötiger Komplexität. Ein erster Vergleich der beiden Lösungen kann das nicht bestätigen und zeigt eher, dass aktuell noch die ursprüngliche Version der IKBT kompaktere Lösungen liefert.

3 Integration der IKBT

Das Ergebnis der IKBT-Software ist ein Satz von Gleichungen zur Lösung der IK, der in quasi allen Softwareumgebungen genutzt werden kann. Zur Analyse von Robotern und deren Bewegung wird innerhalb einer Softwareumgebung zumeist eine Visualisierung und die einfache Verwendung einer Vielzahl von Robotern erwartet. In diesem Abschnitt soll eine weitere Möglichkeit dazu gezeigt werden, die beispielsweise im Systems Engineering oder in der Lehre genutzt werden kann.

Zur Visualisierung wird hier die open-source Animationssoftware Blender [14] verwendet. Einige Möglichkeiten zur Nutzung von Blender im Zusammenhang mit Kinematik und Robotik werden in Abschnitt 3.1 besprochen. Zur Analyse von weiteren oder neuen Robotern werden zum einen 3D-Modelle benötigt, die sich meist bei den Roboterherstellern herunterladen lassen, und zum anderen die DH-Parameter. Zur Vollständigkeit wird daher ein Algorithmus zur Ermittlung der DH-Parameter aus den geladenen 3D-Modellen in Abschnitt 3.2 vorgestellt.

3.1 Kinematik mit Blender

Blender ist eine open-source 3D-Entwicklungsumgebung und enthält Funktionen zur Modellierung, Animation, Simulation, Videobearbeitung und zum Rendering. Für die Anwendung im Bereich der Kinematik hervorzuheben ist die umfangreiche Python API von Blender. Darüber können alle Funktionen von Blender mit Python verwendet werden und so Objekte erzeugt, manipuliert, bewegt und gerendert werden. Die GNU GPL Lizenz von Blender erlaubt die freie kommerzielle Nutzung und hebt sich so von Anwendungen wie Unity oder der UnrealEngine ab.

Obwohl Blender auch das Rigging, also die Charakteranimation, beherrscht, sollen hier alle kinematischen Berechnung außerhalb Blenders durchgeführt werden. So wird vorgeschlagen, folgende Funktionen zur Analyse in Blender zu nutzen:

- Positionieren von Objekten über absolute oder relative Positionsvektoren und Quaternionen für die Rotationen, die über die Python API den Objekten zugeordnet werden
- Erstellen von zeitlichen Abfolgen über das Setzen von sog. Keyframes, also von Eigenschaften (z.B. einer Pose) mit Zeitstempel
- Erzeugen oder Kopieren von Objekten zur Darstellung von Posen, Hindernissen, etc.
- Laden von 3D Modellen von Robotern, Umgebungen, etc.
- Verwendung von Kurven zur Darstellung Trajektorien

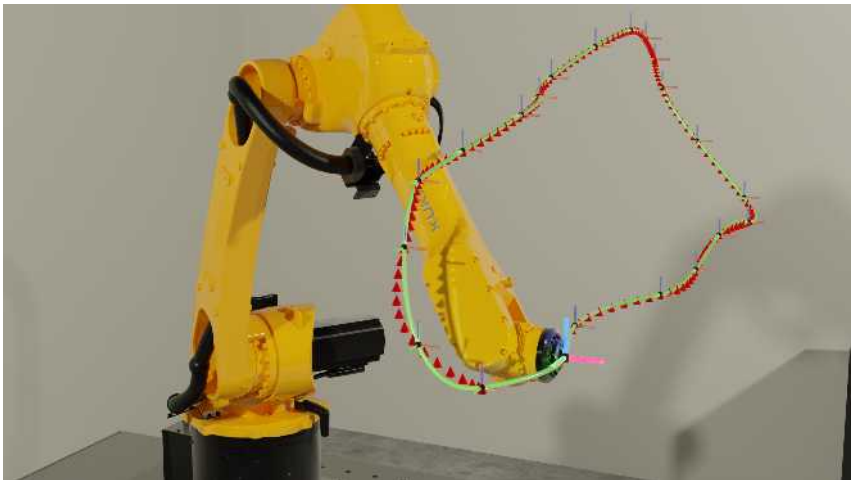


Abb. 5: Verwendung von Blender zur Darstellung kinematischer Systeme und Elemente wie Trajektorien oder Koordinatensysteme

Die Anwendung dieser Funktionen zeigt Abb. 5: Das Modell eines Kuka KR20 wird geladen und positioniert; die grüne Bahn wird in Blender

vorgegeben, in Python mit Polynomen angenähert und in rot dargestellt; einige Posen entlang der Bahn werden über Koordinatensysteme gezeigt.

Darüber hinaus kann das intuitive Benutzerinterface von Blender zum Betrachten der Umgebung und der zeitlichen Abfolge sowie zur Definition von Randbedingungen genutzt werden. Die Abb. 5 zeigt das Benutzerinterface mit der Animation einer Roboterbewegung.

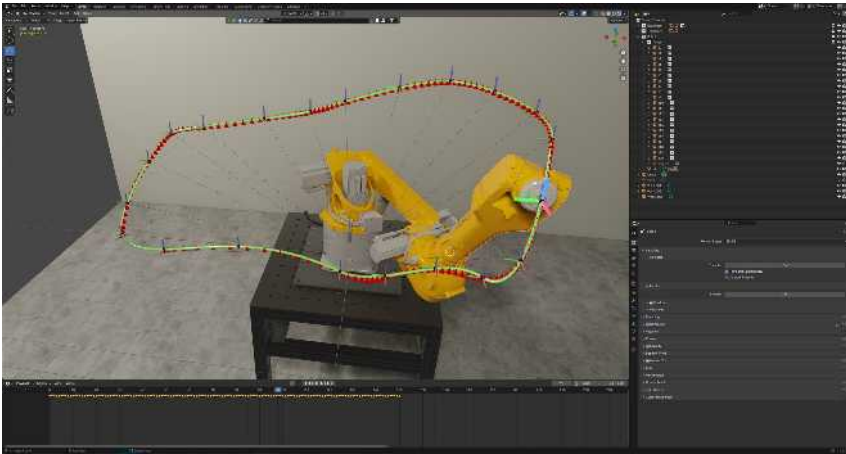


Abb. 6: Betrachtung kinematischer Abläufe mit der Blender Benutzeroberfläche

3.2 Ermittlung der DH-Parameter

Die Denavit-Hartenberg-Parameter bilden eine formalisierte Konvention zur Beschreibung der kinematischen Eigenschaften von Robotern, vergleiche [1]. Aufgrund des formalisierten Charakters liegt es nah die Parameter aus einem 3D-Modell abzuleiten. Allerdings sind teilweise Entscheidungen zu treffen und es muss den numerischen Ungenauigkeiten in 3D Modellen Rechnung getragen werden, sodass hier ein entsprechender Algorithmus gezeigt werden soll.

Zunächst wird ein 3D-Modell in Blender geladen. Die einzelnen Roboter-elemente werden hierarchisch angeordnet und die Koordinatensysteme in die Drehachsen gelegt, indem geeignete Flächen markiert und z.B.

deren Zentren als Koordinatensysteme übernommen werden. Die Posen der Robotererelemente können über die Blender API ausgelesen und in lokale Ortsvektoren umgerechnet werden. Zusätzlich müssen lediglich die Richtungsvektoren der Drehachsen angegeben werden. Mit dem Algorithmus 1 lassen sich dann die DH-Parameter direkt bestimmen.

Algorithmus 1 : Bestimme die DH Parameter aus 3D-Modelldaten

Data :

$\vec{p}_{\text{lokal},i}$: Relativer Ortsvektor der Drehgelenke

$\vec{r}_{\text{Achse},i}$: Drehachse eines jeden Robotererelements

Result :

a_i : DH-Parameter a eines jeden Robotererelements

b_i : DH-Parameter b eines jeden Robotererelements

α_i : DH-Parameter α eines jeden Robotererelements

$q_{\text{Versatz},i}$: Gelenkversatz zwischen 3D-Modell und DH

```

1  $\vec{O}_{\text{Korrektur}} = \vec{0}, \vec{Z}_1 = \text{Normalisiere}(\vec{r}_{\text{Achse},1}), \vec{X}_1 = \vec{e}_x$ 
2 for  $i \in \text{Robotererelemente}$  do
3    $\vec{Z}_{i+1} = \text{Normalisiere}(\vec{r}_{\text{Achse},i+1})$ 
4    $\vec{X}_{i+1} = \vec{Z}_i \times \vec{Z}_{i+1}$ 
5   if  $\|\vec{X}_{i+1}\| = 0$  then
6      $\vec{X}_{i+1} = \text{Normalisiere}(\vec{p}_{\text{lokal},i+1} - (\vec{X}_{i+1} \cdot \vec{Z}_{i+1}) \cdot \vec{Z}_{i+1})$ 
7    $|\alpha_i| = |\arccos(\vec{Z}_i \cdot \vec{Z}_{i+1})|$ 
8   if  $\|\text{RotiereUmVektor}(\vec{X}_{i+1} \cdot |\alpha_i|, \vec{Z}_i - \vec{Z}_{i+1})\| < \text{eps}$  then
9      $\alpha_i = |\alpha_i|$ 
10  else
11     $\alpha_i = -|\alpha_i|$ 
12   $\vec{O}_{i+1} = \vec{p}_{\text{lokal},i+1} + \vec{O}_{\text{Korrektur}}$ 
13   $\vec{O}_{i+1} = \vec{O}_{i+1} - (\vec{O}_{i+1} \cdot \vec{Z}_{i+1}) \cdot \vec{Z}_{i+1}$ 
14   $\vec{O}_{\text{Korrektur}} = \vec{p}_{\text{lokal},i+1} + \vec{O}_{\text{Korrektur}} - \vec{O}_{i+1}$ 
15   $q_{\text{Versatz},i} = \arctan2((\vec{X}_i \times \vec{X}_{i+1}) \cdot \vec{Z}_i, \vec{X}_i \cdot \vec{X}_{i+1})$ 
16   $a_i, b_i \leftarrow \text{AbstandEntlangLinie}(\vec{Z}_i, \vec{O}_{i+1}, \vec{X}_{i+1})$ 

```

Es empfiehlt sich die Berechnungen mit erhöhter Genauigkeit, in Python z.B. mit MPMath und 100 Dezimalstellen, durchzuführen. Abschließend sollten die numerischen Ungenauigkeiten von a_i und b_i nahe Null und die von α_i nahe $\frac{\pi}{2}$ gerundet werden.

Beim Bestimmen des Abstands zweier nachfolgender Koordinatensysteme entlang der Achsen zur Bestimmung von a_i und b_i , muss das Teilen durch Null vermieden werden, sodass das Vorgehen in Algorithmus 2

Algorithmus 2 : Finde den Abstand entlang einer Linie

```

Data :  $\vec{v}_1, \vec{p}, \vec{v}_2$ 
Result :  $a, b$ 
1  $k_a = 0, k_b = 0$ 
2 for  $i \leftarrow 1$  to 3 do
3   for  $i \leftarrow 1$  to 3 do
4     if  $i \neq j$  then
5       if  $|\vec{v}_{1,i}| > \sqrt{\text{eps}}$  then
6         if  $|\frac{\vec{v}_{1,j} \cdot \vec{v}_{2,i}}{\vec{v}_{1,i} \cdot \vec{v}_{2,j}}| > \sqrt{\text{eps}}$  then
7            $k_a = \frac{\vec{p}_j - \vec{v}_{1,j} \cdot \vec{p}_i / \vec{v}_{1,i}}{\vec{v}_{1,j} \cdot \vec{v}_{2,i} / \vec{v}_{1,i} - \vec{v}_{2,j}}$ 
8            $k_b = k_a \cdot \frac{p_i}{\vec{v}_{1,i}} + \frac{\vec{v}_{2,i}}{\vec{v}_{1,i}}$ 
9         else if  $\vec{v}_{1,j} \cdot \vec{v}_{2,i} = 0 \wedge \vec{v}_{1,j} \cdot p_i = 0 \wedge |\vec{v}_{2,j}| > \sqrt{\text{eps}}$  then
10           $k_a = -\frac{p_j}{\vec{v}_{2,j}}$ 
11  $a = \|\vec{v}_2 \cdot k_a\|, b = \|\vec{v}_1 \cdot k_b\|$ 

```

verwendbar ist.

Es ist zu beachten, dass die Genauigkeit der so bestimmten DH-Parameter von den verfügbaren 3D-Modellen und den ausgewählten Flächen zur Bestimmung eines Punktes der Drehachsen abhängig ist. Bei einigen zur Verfügung gestellten 3D-Modellen liegen die Ursprungskordinatensysteme einzelner Körper bereits auf den Drehachsen und somit ist die Genauigkeit dabei höher.

4 Anwendung im Systems Engineering

In Abschnitt 4.1 wird die IKBT mit der Berechnung nach Angeles [1] hinsichtlich Anwendung und Performance verglichen. Dabei wird die Berechnung der inversen Kinematik für verschiedene Roboter durchgeführt. Demcon ist eine niederländische Unternehmensgruppe, die Entwicklungsdienstleistungen und Maschinenbau von der ersten Idee bis zum fertigen Produkt in den Bereichen high-tech systems, life sciences & health, smart industry sowie weiteren anbietet. Das System Engineering ist bei Demcon von zentraler Bedeutung und führt die Entwicklung durch den gesamten Entwicklungsprozess. Inwieweit dabei die vorgestellte Kombination von IKBT und Blender genutzt werden kann diskutiert abschließend Abschnitt 4.2.

4.1 Anwendung mit verschiedenen Robotersystemen

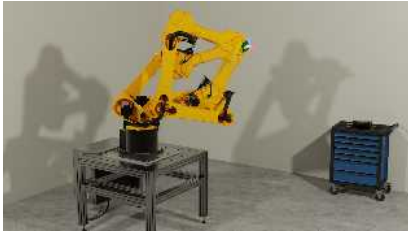
Für die Anwendung der IKBT zusammen mit Blender und für den Vergleich der IKBT mit der IK für entkoppelte ($a_4 = 0$, $a_5 = 0$ und $b_5 = 0$), serielle Roboter aus [1], werden vier verbreitete Industrieroboter mit jeweils unterschiedlichen DH-Parametern verwendet. Zusätzlich wird ein nicht entkoppelter Roboter, der UR10, verwendet, wobei die IK nach [15] als Referenz dient.

Tab. 1: Übersicht der zum Vergleich verwendeten Roboter

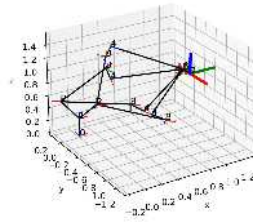
Roboter	a [mm]	b [mm]	α	T_{IK} [ms]	T_{IKBT} [ms]
ABB IRB1200	$a_2 = 350$ $a_3 = 42$	$b_1 = 180$ $b_4 = 348$	$\alpha_1 = \pi/2$	0.13	0.11
			$\alpha_3 = \pi/2$		
			$\alpha_4 = \pi/2$		
			$\alpha_5 = \pi/2$		
			$\alpha_1 = \pi/2$		
Kuka KR20	$a_1 = 200$ $a_2 = 780$ $a_3 = 170$	$b_1 = 250$ $b_4 = 823$	$\alpha_3 = \pi/2$	2.20	0.11
			$\alpha_4 = \pi/2$		
			$\alpha_5 = \pi/2$		
			$\alpha_1 = \pi/2$		
			$\alpha_3 = \pi/2$		
Yaskawa HC20DTP	$a_2 = 820$	$b_1 = 220$ $b_4 = 880$	$\alpha_3 = \pi/2$	0.13	0.10
			$\alpha_4 = \pi/2$		
			$\alpha_5 = \pi/2$		
			$\alpha_1 = \pi/2$		
			$\alpha_3 = \pi/2$		
Mitsubishi rv35	$a_1 = 150$ $a_2 = 870$	$b_1 = 431$ $b_4 = 1080$	$\alpha_3 = \pi/2$	2.34	0.11
			$\alpha_4 = \pi/2$		
			$\alpha_5 = \pi/2$		
			$\alpha_1 = \pi/2$		
			$\alpha_3 = \pi/2$		
UR10e	$a_2 = 613$ $a_3 = 571$	$b_1 = 81$ $b_4 = 174$ $b_5 = 120$	$\alpha_1 = \pi/2$	0.09	0.12
			$\alpha_4 = \pi/2$		
			$\alpha_5 = \pi/2$		
			$\alpha_1 = \pi/2$		
			$\alpha_4 = \pi/2$		

Die IK wird unter Verwendung von Python und den Paketen Numpy sowie Scipy implementiert. Dabei wird die IK nach [1] S.139 ff. mit Fallunterscheidung nach $a_1 = 0$, $\alpha_1 = \frac{\pi}{2}$ und allen übrigen Robotern implementiert. Bei letztem Fall ist ein Polynom vierter Ordnung zu lösen, was mit der passenden Funktion aus Scipy realisiert wird. Die ausgegebenen Gleichungen der IKBT werden größtenteils übernommen, jedoch werden die Übergabeparameter angepasst und rein informative Relativ-

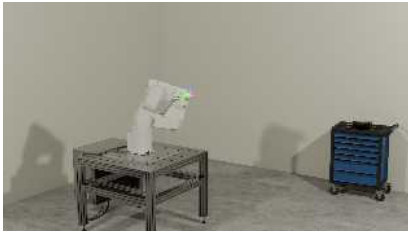
winkel aus den Berechnungen entfernt. Tabelle 1 zeigt die Roboter mit den jeweils korrekt ermittelten DH-Parametern.



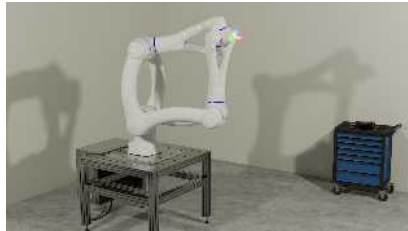
(a) KR20 in Blender



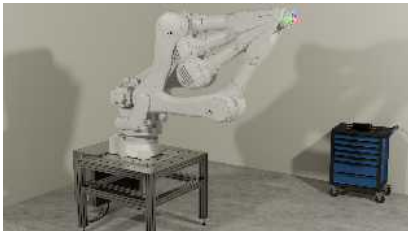
(b) KR20 in Python



(c) ABB IRB1200



(d) Yaskawa HC20DTP



(e) Mitsubishi rv35



(f) UR10e

Abb. 7: Die Lösung der IK für alle Konfigurationen verschiedener Roboter

Für jeden Roboter werden alle acht Konfiguration mit der IKBT berechnet und jeweils mit der numerischen Vorwärtskinematik als auch der Referenz-IK verglichen, siehe Abb 7. Alle Lösungen sind korrekt, lediglich für den UR10 werden von der IKBT für den Antriebswinkel θ_2 falsche Werte berechnet, wodurch die nachfolgend berechneten Winkel θ_3 und

θ_4 natürlich ebenfalls inkorrekt sind. Hinsichtlich der Berechnungsgeschwindigkeit sind die IKBT und die Referenz-IK ähnlich, solange nicht das Polynom vierter Ordnung gelöst werden muss. In diesen Fällen ist die IKBT deutlich schneller und dementsprechend zu empfehlen.

Für einen detaillierteren Leistungsvergleich soll erneut die Berechnung der IK des KR20 herangezogen werden. Wie oben beschrieben, muss für die Referenz-IK das Polynom vierter Ordnung gelöst werden, sodass die benötigt Zeit dafür mit über 2 ms pro Berechnung hoch ist. Die Lösung mittels IKBT benötigt weiterhin nur 0.1 ms pro Berechnung, wobei hier alle acht reellen Lösungen der IK zusammen berechnet werden. Wird der generierte Code dahingehend angepasst, dass nur eine Konfiguration berechnet wird, halbiert sich die Zeit auf 50 μ s. Python ist dafür bekannt, dass Schleifen nur relativ langsam ausgeführt werden. Dem kann durch die Verwendung von Vektoren mit allen zu berechnenden Posen und Numpy, kurz vektorisiertes rechnen, entgegengewirkt werden. So kann in diesem Beispiel die Rechenzeit auf 1.1 μ s pro IK reduziert werden. Der Anteil der reinen IK liegt bei 0.5 μ s, wobei die restliche Zeit für weitere Transformationen benötigt wird. Eine Übersicht über die verschiedenen Zeiten zur Berechnung der IK zeigt Tab. 2 Das IKFast-Modul aus OpenRave [16] soll Berechnungen hingegen in 4 μ s abschließen.

Tab. 2: Leistungsvergleich verschiedener Implementierungen der IK

Beschreibung	Wert
Roboter	Kuka KR20
T für IK	2.1351 ms
T für IKBT	0.1097 ms
T für einfache IKBT	0.0527 ms
T für vektorisierte IKBT	0.0011 ms
T nur für IK bei vektorisierter IKBT	0.0005 ms

Zwar kann IKBT noch nicht für alle Roboter die IK fehlerfrei generieren, dies lässt sich jedoch sofort feststellen und es muss dann auf andere

Lösungen zurückgegriffen werden. Ansonsten lassen sich einige Vorteile der IKBT gegenüber Alternativen wie der IK nach [1] oder IKFast identifizieren. Die Gleichungen sind leicht lesbar und in quasi allen Programmierumgebungen verwendbar. Die Lösungen sind kompakt und können schnell, in dem hier vorliegenden Vergleich am schnellsten, berechnet werden.

4.2 Nutzen der IKBT mit Blender im Systems Engineering

Durch die vorgestellte Integration von IKBT und Blender entsteht eine open-source Entwicklungsumgebung, um einfach verschiedene Robotersysteme oder andere Bewegungssysteme zu analysieren. In frühen Entwicklungsphasen stehen im System Engineering zumeist keine Roboterprogrammierer zur Verfügung und zudem ist der Aufbau einer Simulation aufwendig, was insbesondere beim Vergleich vieler verschiedener Konzepte ins Gewicht fällt. Der vorgestellte Ansatz bietet sich daher für die Evaluierung im Konzeptstadium an. Zudem können so bereits in der Angebotsphase erste visuell ansprechende Renderings oder Animationen erstellt werden. Ein weiterer Ansatz im System Engineering ist es, Modelle über den gesamten Entwicklungs- oder sogar Lebenszyklus zu verwenden. Die Größte Herausforderungen stellen dabei die Schnittstellen oder eben nicht vorhandene Schnittstellen spezialisierter Software dar. Die von der IKBT ermittelten Gleichungen sind strukturell einfach aufgebaut und greifen lediglich auf trigonometrische Funktionen zurück. Diese stehen in quasi allen Systemen zur Verfügung, sodass die von der IKBT erstellte IK durchgängig genutzt werden kann: Für die Visualisierung in Blender, die Simulation in einem Mehrkörpersimulationsmodell, auf der SPS und für die Visualisierung in der Webansicht des digitalen Zwillings. Es stellt sich die Frage, warum die Berechnung der IK im Bereich von μs optimiert werden soll, obwohl die meisten Steuerungen im Bereich von 1 ms verwendet werden. Dies hat vor allem mit dem Nutzen von Optimierungsverfahren und Ansätzen von maschinellem Lernen zu tun, denn hier müssen viele Berechnungen durchgeführt werden. Schnellere Algorithmen können bei diesen Methoden viel Rechenzeit sparen.

Literatur

- [1] Angeles, J. *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms*. Springer International Publishing, 2014. isbn: 9783319018515. doi: 10.1007/978-3-319-01851-5.
- [2] Diankov, R. „Automated construction of robotic manipulation programs“. In: (2010).
- [3] Zhang, D. und Hannaford, B. „IKBT: Solving symbolic inverse kinematics with behavior tree“. In: *Journal of Artificial Intelligence Research* 65 (2019), S. 457–486.
- [4] ABB AG. *Author Components in RSXML*. 15. Juli 2024. url: <https://developercenter.robotstudio.com/api/robotstudio/articles/Walkthroughs/AuthorComponentsInRxml.html>.
- [5] Kuka. *Roboterzellen leichter planen mit der Simulationssoftware KUKA.Sim 4.0*. 15. Juli 2024. url: <https://www.kuka.com/de-de/unternehmen/presse/news/2021/05/kuka-sim-simulation-software>.
- [6] Yaskawa. *MotoSim EG-VRC*. 15. Juli 2024. url: https://www.yaskawa.de/produkte/software/productdetail/product/motosim-eg-vc_1686.
- [7] Coppelia Robotics. *Create. Compose. Simulate. Any robot*. 15. Juli 2024. url: <https://coppeliarobotics.com/>.
- [8] RoboDK. *Simulieren Sie Roboter-Anwendungen*. 15. Juli 2024. url: <https://robodk.com/de/>.
- [9] Virtual Components. *Visual Components connector for NVIDIA Omniverse: a perfect recipe for manufacturing digitalization*. 15. Juli 2024. url: <https://www.visualcomponents.com/blog/visual-components-connector-for-nvidia-omniverse/>.
- [10] Open Motion Planning Library. *The Open Motion Planning Library*. 21. Juli 2024. url: <https://ompl.kavrakilab.org/>.
- [11] Open Rave. *Welcome to OpenRAVE*. 21. Juli 2024. url: <http://openrave.org/>.

- [12] MoveIt. *Moving robots into the future*. 21. Juli 2024. url: <https://moveit.ros.org/>.
- [13] Hannaford, B. und Zhang, D. *IKBT*. 29. Juli 2024. url: <https://github.com/uw-biorobotics/IKBT>.
- [14] Blender. *Blender 4.2 LTS*. 16. Juli 2024. url: <https://www.blender.org/>.
- [15] Hawkins, K. P. *Analytic Inverse Kinematics for the Universal RobotsUR-5/UR-10 Arms*. 7. Dez. 2013.
- [16] OpenRave. *IKFast: The Robot Kinematics Compiler*. 29. Juli 2024. url: http://openrave.org/docs/latest_stable/openravepy/ikfast/#using-generated-ik-files.

Entwurf eines Koppelgetriebes für die stufenlose Höhenverstellung einer landwirtschaftlichen Stelzradzugmaschine

Robert Konradt*, Stephan Schmidt**,
Sebastian Frasch*, Andreas Scholz*

* Otto-von-Guericke-Universität Magdeburg, Institut für Mobile Systeme,
Lehrstuhl für Mechatronik, Vorname.Nachname@ovgu.de

** Hochschule Merseburg, Fachbereich Ingenieur- und Naturwissenschaften,
Professur für Mechatronische Systeme, Stephan.Schmidt@hs-merseburg.de

Kurzfassung

Biologische Pflanzenschutzmittel sind in der Landwirtschaft häufig nur wirkungsvoll einsetzbar, wenn die zu schützenden Kulturen bereits hochstehend sind. Vor diesem Hintergrund ist ein Traktorkonzept mit hoher Bodenfreiheit entwickelt und prototypisch aufgebaut worden. Dessen Alleinstellungsmerkmal sind vier ebene Koppelgetriebe, die eine stufenlose Höhenverstellung der einzelnen Stelzen ermöglichen. Der Antrieb der Koppelgetriebe erfolgt durch jeweils einen Hydraulikzylinder.

Titel / Abstract

Design of a mechanical linkage for the continuous height adjustment of an agricultural high clearance tractor

Biological plant protection can often only be used effectively in agriculture if the crops to be protected have already grown high. In this setting, we have developed and built a prototype of a high clearance tractor. Its unique feature are four planar linkages, which allow for a continuous height adjustment of the individual stilts using hydraulic cylinders.

1 Motivation

Der Einsatz von chemischen Pflanzenschutzmitteln in der Landwirtschaft stellt eine große Belastung für die Umwelt dar und führt zur Beeinträchtigung der Biodiversität. Biologische Pflanzenschutzmittel, wie die Schlupfwespe Trichogramma brassicae, bieten eine nachhaltige und umweltfreundliche Alternative [1], [2]. Allerdings existiert derzeit keine wirtschaftliche Lösung zur effektiven Ausbringung. So sind verfügbare konventionelle landwirtschaftliche Fahrzeuge häufig nur für chemische Pflanzenschutzmittel und auf eine deutlich höhere Nutzlast ausgelegt, was sich negativ im Fahrzeuggewicht und im Anschaffungspreis widerspiegelt. Für die vom Gesetzgeber vorgegebene langfristige Reduzierung chemischer Insektizide ist daher die Konzipierung und Entwicklung eines Spezialfahrzeuges erforderlich.

Der Anwendungszeitraum der Schlupfwespe ist von dem Entwicklungsstadium des Parasiten, des Maiszünslers, und somit von der lokalen Tagestemperatursumme abhängig. Aus diesem Grund verschiebt sich der optimale Austragungszeitpunkt zwischen Süd- und Mitteleuropa von Ende Mai bis Anfang Juli. Dadurch ergibt sich örtlich ein kurzes Anwendungszeitfenster von nur wenigen Tagen. Dies erfordert eine hohe Mobilität und Verfügbarkeit der austragenden landwirtschaftlichen Maschine. Diese wurde daher so konzipiert, dass sie im eingefahrenen Zustand auf einem konventionellen PKW-Anhänger schnell zwischen den verschiedenen Einsatzorten transportiert werden kann, während sie im ausgefahrenen Zustand über den hochstehenden Maisbestand fahren kann (Abb. 1).

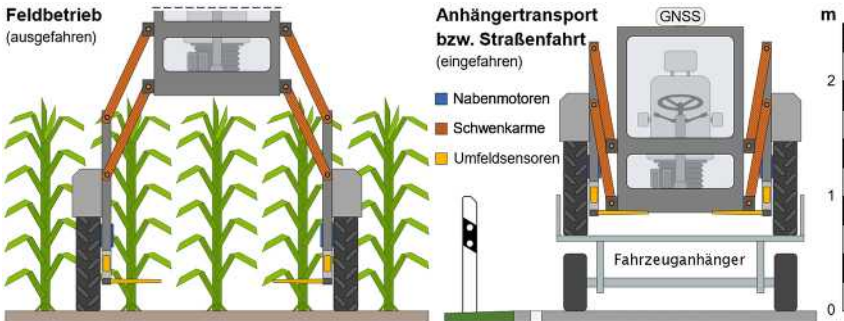


Abb. 1: Konzept der stufenlosen Höhen- und Spurweitenverstellung mittels ebener Koppelgetriebe.

2 Stand der Technik

Verfügbare Pflanzenschutzgeräte, wie selbstfahrende Feldspritzen, verwenden zum Verstellen oftmals Linearführungen. Ein bewährtes Fahrwerkskonzept stellt das sogenannte Längstandem dar (Abb. 2). Unebenheiten an einem Rad werden bei diesem Konzept durch den mittig positionierten Drehpunkt so ausgeglichen, dass im Vergleich zu herkömmlichen Pendelachsen der Fahrzeugaufbau nur geringfügig angehoben wird [3]. Die hydropneumatische Federung kann dabei ebenfalls zur Niveauregulierung genutzt werden. Neben dem Hoch- und Herunterfahren dienen die Verstellgetriebe auch dem Anpassen der Spurweite der Maschine an die unterschiedlichen Maisreihenabstände oder an die Breite der Fahrgasse der Pflugespur. Letztere wird bei der Aussaat vorgegeben und basiert auf dem Fuhrpark des jeweiligen Betriebes und kann daher von Feld zu Feld variieren [4], [5].

Der Einstellbereich für die Bodenfreiheit und Spurweite des Tandemfahrwerks ist ausreichend um die Anforderung eines flexiblen Einsatzes sowie der Einhaltung der gesetzlichen Vorgaben zu erfüllen. So wird zum Beispiel die *Amazone Pantera 4504-H* Feldspritze zur Einhaltung der Vorschriften zur Kippsicherheit und zur genehmigungs-freien Fahrt auf öffentlichen Straßen auf eine Transportbreite von 2,55 m in Verbindung mit einer Bodenfreiheit von 1,25 m verfahren [3]. Auf dem Feld kann diese auf 1,70 m und die Spurweite auf bis zu 2,60 m erhöht werden. Ein entscheidender Nachteil dieses Fahrwerkskonzepts ist jedoch, dass es aufgrund des limitierten Verstellbereichs nicht auf einem PKW-Anhänger transportiert werden kann.

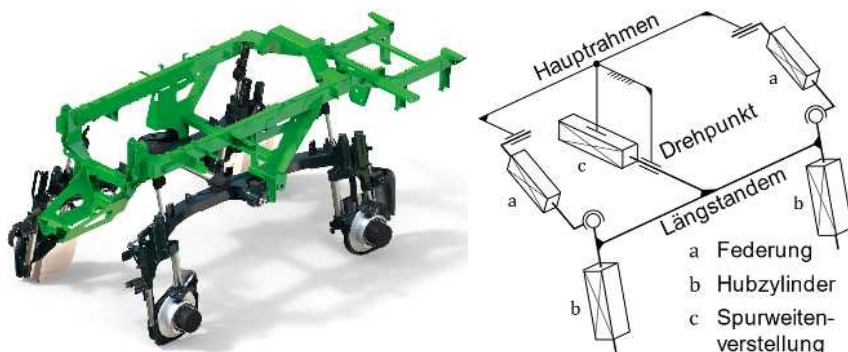


Abb. 2: Fahrwerksgetriebe einer *Amazone Pantera 4504-H* Feldspritze [3].

3 Getriebeauslegung

Der Fokus des Beitrags liegt auf der Auslegung und Gestaltung des Koppelgetriebes zur stufenlosen Höhen- und Breitenverstellung eines landwirtschaftlichen Spezialfahrzeuges. Im Rahmen der Entwicklung wurden zunächst die Anforderungen an das Getriebe identifiziert und definiert. Im weiteren Schritt wurde das Gesamtsystem in Form eines Mehrkörpermodells in der Simulationsumgebung *Matlab / Simulink* abgebildet. Auf Basis der Ergebnisse wurden schlussendlich die Komponenten ausgelegt und ausgewählt. Die Anforderungen begründen sich dabei auf der Produktidee eines kompakten, leichten und mobilen Geräteträgers zur wirtschaftlichen Ausbringung von biologischen Pflanzenschutzmitteln in hochstehenden Kulturen.

3.1 Anforderungsanalyse

Bei der Konzeption wurden zunächst alle Anforderungen an das Koppelgetriebe zur stufenlosen Höhenverstellung analysiert und ein aufgabenangepasster Entwurf durchgeführt. Das Hauptkriterium für die Auslegung war die Anpassungsfähigkeit der Stelzradzugmaschine, so dass die Größenvorgaben sowohl für den Einsatz auf dem Feld als auch für den Straßen- sowie Anhängertransport erfüllt sind (Abb. 1). Darüber hinaus soll das Konzept mindestens den gleichen Funktionsumfang in Hinblick auf die Spurweiten- und Höhenverstellung aufweisen wie die konkurrierenden, selbstfahrenden Feldspritzen (Abb. 2). Die Bodenfreiheit der Stelzradzugmaschine soll daher mindestens 1,80 m bei einer gewählten Fahrgasse von 2,25 m betragen. Und für den Transport soll eine Breite von 2,10 m, sowie eine Höhe von 2,50 m eingehalten werden.

In Hinblick auf den Einsatz in der Landwirtschaft gilt den Anforderungen bezüglich eines einfachen und robusten Getriebes mit wenigen Gelenken und Gliedern eine sehr hohe Priorität. Weitere Faktoren waren eine hohe Wartungsfreundlichkeit sowie eine möglichst kostengünstige Realisierbarkeit. Darüber hinaus muss das Verstellgetriebe zentrale Fahrwerksfunktionen (Tab. 1) wie das Federn und Lenken ermöglichen. Nichtsdestotrotz ist aufgrund der Vorgabe der Verladefähigkeit der maximal zu Verfügung stehende Bauraum stark begrenzt. Neben der bereits in Kapitel 2 aufgeführten Kippstabilität müssen bei der Gestaltung auch weitere Aspekte der funktionalen Sicherheit berücksichtigt werden.

3.2 Konzeption

Zum Erfassen und Vergleichen der verschiedenen Lösungsprinzipien zur Umsetzung des Verstellgetriebes wurde ein morphologischer Kasten erstellt. Durch die Kombination der Teilfunktionen wurde die Gesamtlösung generiert [6]. In Tab. 1 sind die verschiedenen Ansätze für die jeweiligen Getriebe für die Höhen- und Spurweitenverstellung sowie für die Lenkung und Federung aufgeführt. Die Abb. 3 und Abb. 4 stellen dabei noch einmal einige Varianten unterstützend dar.

Tab. 1: Morphologie der Fahrwerksgetriebe der Stelzradzugmaschine.

	Teilfunktion	Variante 1	Variante 2	Variante 3
1	Höhenverstellung	Schubgelenk / Linearführung	Drehgelenk, Längsachse	Drehgelenk, Querachse
2	Spurweitenverstellung	Schubgelenk / Linearführung	Drehgelenk, Längsachse	
3	Federung	Pendelachse (Abb. 3)	Tandemfahrwerk (Abb. 2)	Einzelrad (Abb. 4)
4	Lenkung	Einzelradlenkung (Abb. 2, Abb. 4)	Knicklenkung (Abb. 3)	Radseitenlenkung ¹
5	Antriebsfunktion	Hydraulisch	Elektrisch	Mechanisch

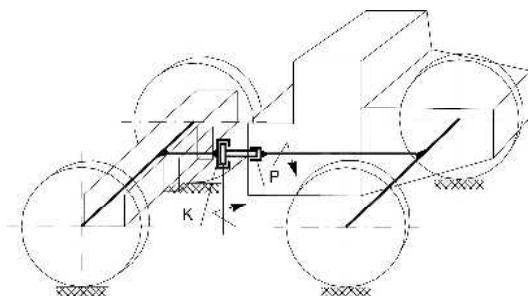


Abb. 3: Funktionsprinzip einer Knicklenkung (K) mit Pendelachse (P) [7].



Abb. 4: Federung und Lenkung des Züm 540 Geräteträgers [8].

¹ In der Literatur auch Antriebs-, Brems-, Skid- oder Panzerlenkung genannt [7].

Nach einer vollständigen Bewertung der einzelnen Varianten (Tab. 1) anhand eines Zielsystems wurde aufgrund von einschlägigen Vorteilen die Gesamtlösung 1-2, 2-2, 3-1, 4-2 und 5-1 sowie als zusätzliche Option für kleine Lenkeingriffe 4-3 gewählt. So erlaubt das zentrale Knickgelenk beispielsweise das unabhängige Lenken der Maschine in jeder Position des Koppelgetriebes, da die Räder, im Gegenteil zu einer Einzelradlenkung, stets parallel zum Vorder- oder Hinterwagen der Maschine sind. Der Nachteil, dass aufgrund der Entkopplung des Aufbaus für jedes Rad eine Stelze mit je einem Koppelgetriebe erforderlich ist, wird durch die Skalierbarkeit in der Fertigung aufgrund der punktsymmetrischen Gestaltung des Gesamtsystems wieder ausgeglichen. Darüber hinaus kann aufgrund der Kombination von Höhen- und Spurweitenverstellung auf einen aufwendigeren Aufbau mit mehreren Getrieben verzichtet werden. Aufgrund der Kopplung können jedoch Höhe und Breite nicht unabhängig voneinander verstellt werden. Da aber die Anpassung der Spurweite nur im ausgefahrenen Zustand bei großen Schwenkwinkeln θ notwendig ist, ist die Änderung der Fahrzeugbreite Δb aufgrund der Sinus-Komponente der Länge des Schwenkarmes $\overline{A_0 A}$ größer als der Kosinusanteil der Höhenänderung Δh (Gl. 1) (Abb. 7). Aus diesem Grund ist die Reduktion der Bodenfreiheit Δh akzeptabel und kann konstruktiv berücksichtigt werden. Da die Stelzen stufenlos und unabhängig voneinander Verfahren werden, kann das Getriebekonzept zukünftig auch zur Niveauregelung herangezogen werden [9], [10]. Der Radstand wurde so gewählt, dass die Räder sich in der breitesten Konfiguration mit dem Schwenkwinkel $\theta = 90^\circ$ sowie einen maximalen Lenkeinschlag der Knicklenkung von $\delta = 45^\circ$ nicht berühren.



Abb. 5: Funktionsmuster im Maisbestand und auf einem Anhänger.

$$\Delta b = \sin(\underbrace{180^\circ - \Delta\theta}_{\Delta\psi}) \cdot \overline{A_0 A} > \Delta h = \cos(\Delta\psi) \cdot \overline{A_0 A} \quad \text{für } \theta > 135^\circ \quad (1)$$

3.3 Gestaltung

Eine besondere Herausforderung bei der Getriebeauslegung bestand in der Realisierung des großen Schwingwinkelbereichs von $\Psi = 150^\circ$ bei gleichzeitig begrenztem Hub s des Hydraulikzylinders sowie dem zur Verfügung stehenden Bauraum. Hieraus ergeben sich insbesondere in den Endlagen hohe erforderliche Zylinderkräfte (Abb. 8). Dies ist auf ein deutliches Verlassen des in der Literatur als erforderlich beschriebenen Bereiches des Übertragungswinkels μ zwischen dem Schleifengelenk (5) und der Schwinde (2) zurückzuführen (Abb. 6 und Gl. 3) [11]. Mit den $n = 6$ Gliedern und den $g = 7$ Gelenken ergibt sich für das ebene Koppelgetriebe ein Freiheitsgrad von $F = 1$ (Gl. 2).

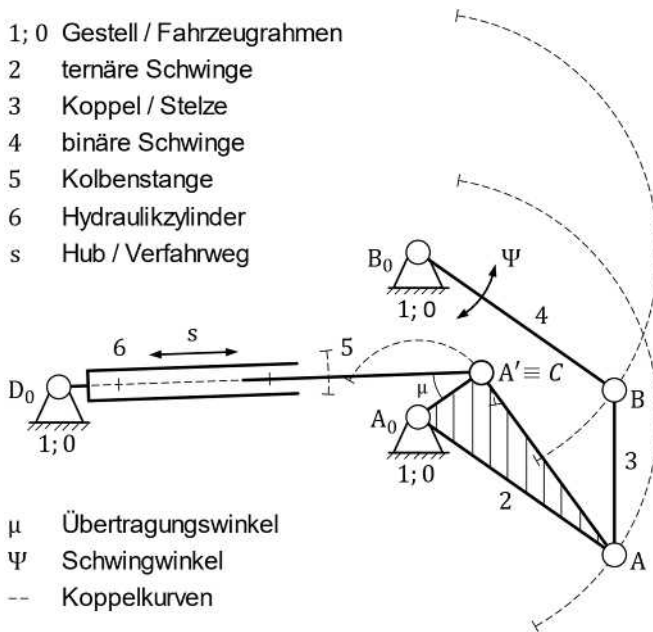


Abb. 6: Kinematik des hydraulisch angetriebenen Koppelgetriebes.

$$F = b \cdot (n - 1) - \sum_{i=1}^g (u_i) = 3 \cdot (6 - 1) - 7 \cdot \underbrace{(3 - 1)}_{u_i = b - f_i} = 1 \quad (2)$$

$$\mu_{\min} \geq \mu_{\text{erf}} \quad \text{mit} \quad 40^\circ \leq \mu_{\text{erf}} \leq 50^\circ \quad \text{hier:} \quad \mu_{\min} = 9^\circ \ll \mu_{\text{erf}} \quad (3)$$

Der vom Verfahrwinkel α abhängige Übertragungswinkel μ (Gl. 4) lässt sich dabei über die geometrischen Dreiecksbeziehungen des Kosinus- und Sinussatzes (Gln. 5 und 6) herleiten. Die Beziehung zwischen den beiden Winkeln wird durch die Wahl des durch die Konstruktion bedingten Lagerwinkels Λ sowie der Länge $\overline{A_0 A'}$ und Positionierung Φ des Hebelarms vorgegeben (Abb. 7).

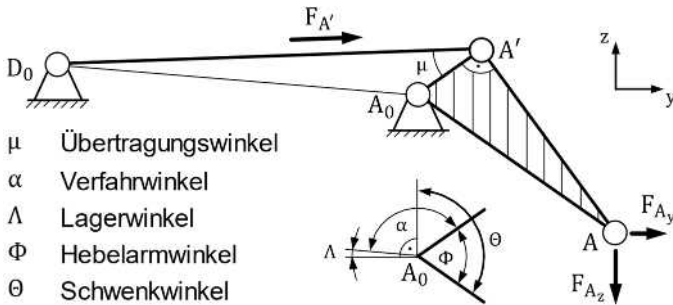


Abb. 7: Allgemeine Konfiguration des Koppelgetriebes mit Definition der relevanten Winkel.

$$90^\circ + \theta = \Lambda + \alpha + \Phi \rightarrow \alpha = 90^\circ + \theta - \Phi - \Lambda \quad (4)$$

$$\overline{D_0 A'} = \sqrt{(\overline{D_0 A_0})^2 + (\overline{A_0 A'})^2 - 2 \cdot \overline{D_0 A_0} \cdot \overline{A_0 A'} \cdot \cos \alpha} \quad (5)$$

$$\frac{\overline{D_0 A_0}}{\sin \mu} = \frac{\overline{D_0 A'}}{\sin \alpha} \rightarrow \mu = \sin^{-1} \left(\frac{\overline{D_0 A_0}}{\overline{D_0 A'}} \cdot \frac{\sin \alpha}{1} \right) \quad (6)$$

Im Rahmen der Getriebeauslegung wurden zunächst die erforderlichen Zylinderkräfte analytisch abgeschätzt. Dazu wurde vereinfachend angenommen, dass die Reibkraft am Rad vernachlässigbar klein ist und dadurch im Punkt A lediglich eine vertikale Aufstandskraft F_A wirkt, die einem Viertel des Fahrzeuggewichts entspricht. Entsprechend dieser Vereinfachungen kann die parallele Schwinde (4) des viergliedrigen Drehgelenkgetriebes unberücksichtigt bleiben. Da im eingefahrenen Zustand die Aufstandskraft F_A das Gelenk A' zu überstrecken droht, wurde $\mu_{\max} = 155^\circ$ gewählt, so dass eine ausreichende Winkelreserve zum gestreckten Zustand besteht. Im ausgefahrenen Zustand ist die Winkelreserve mit $\mu_{\min} = 9^\circ$ kleiner, da hier die Aufstandskraft F_A einer Überstreckung entgegenwirkt (Abb. 8). Abb. 9 verdeutlicht diesen Zusammenhang für den gesamten Bereich des Übertragungswinkels μ .

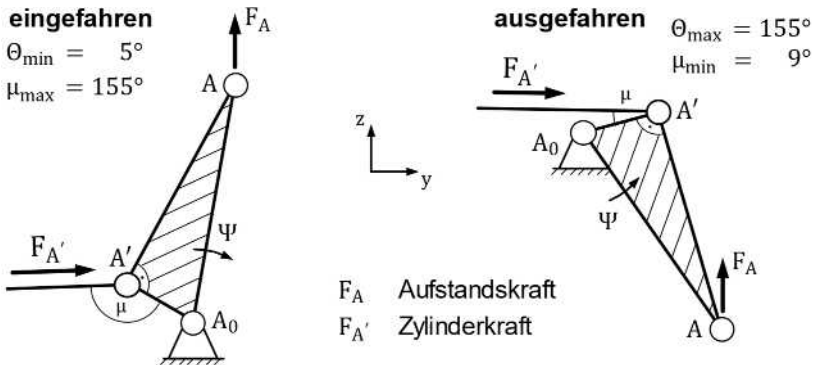


Abb. 8: Übertragungswinkel und Kräfte in den beiden Endlagen.

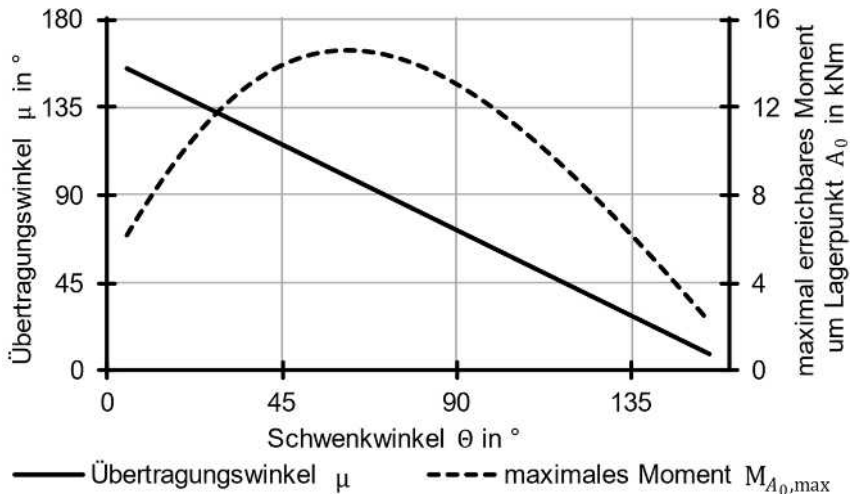


Abb. 9: Einfluss des Übertragungswinkels μ auf das vom Hydraulikzylinder maximal erreichbare Moment um den Lagerpunkt A_0 .

3.4 Mehrkörpersimulation

Zur weiteren Optimierung der Konfiguration des Verstellgetriebes in Bezug auf die auftretenden Lagerkräfte sowie zur Ermittlung der erforderlichen Antriebskräfte der Hydraulikaktoren wurde eine Mehrkörpersimulation erstellt. Die Modellierung des Gesamtsystems, bestehend aus den vier einzelnen Koppelgetrieben inklusive ihrer oberen Glieder, erfolgte mit der *Simscape Multibody Toolbox* (Abb. 10).

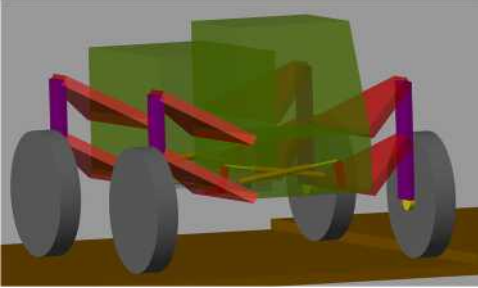


Abb. 10: Mehrkörpermodell während der Simulation des Bordsteinabdrückens.



Abb. 11: Schleppversuch zur Bestimmung des Reibwertes.

In der Simulation wurde der Lastfall des Bordsteinabdrückens nachgebildet. Hierbei wurde der Freiheitsgrad der Knicklenkung blockiert und die Translation der beiden linken Räder wurde quer zur Fahrtrichtung eingeschränkt. In der Simulation wurde auch die bisher vernachlässigte Reibkraft der Räder auf dem Boden berücksichtigt. Zur Abbildung der Radreibungen wurde die Formulierung nach Dahl [12] mit einem Exponentialfaktor von $d = 1$ verwendet, um auch Haftreibung darstellen zu können. Der Reibwert zwischen Reifen und Boden wurde mittels eines Schleppversuchs ermittelt. Dabei wurde die Stelzradzugmaschine bei blockierten Rädern gegen die Reibungskraft gezogen und die erforderliche Zugkraft gemessen (Abb. 11).

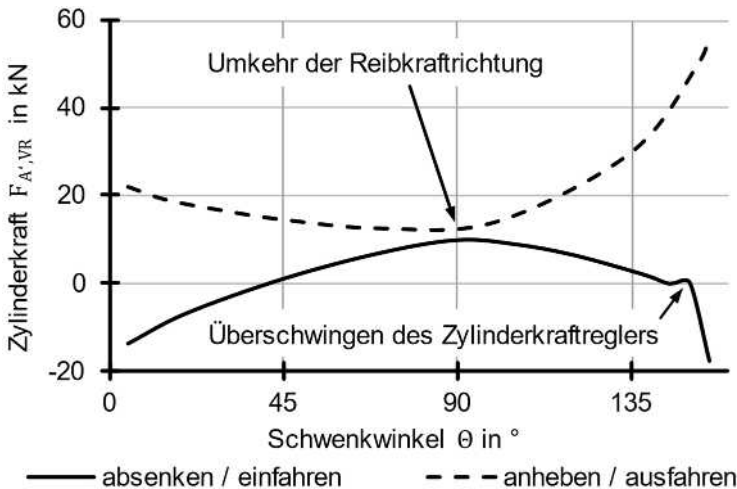


Abb. 12: Ergebnis der Mehrkörpersimulation der vorderen rechten Stelze.

Die Bewegung des Getriebes wurde aus numerischen Gründen mittels einer Regelung mit Sollwegvorgabe realisiert. Der Regler stellt die Zylinderkraft $F_{A'}$ in Abhängigkeit des Verfahrweges s des Hydraulikzylinders. Durch die Regelung des Hubs s ergeben sich jedoch Überschwinger bei den Zylinderkräften an den Umkehrpunkten. In Abb. 12 ist exemplarisch das Simulationsergebnis der Kraft $F_{A'}$ des Hydraulikzylinders der vorderen rechten Stelze über den Schwenkwinkel θ der Schwinge aufgetragen. Bei einem Winkel $\theta = 90^\circ$ hat die Maschine ihre maximale Breite erreicht. Die Verschieberichtung der Räder auf dem Boden und damit auch die Reibkraft kehren sich dann um.

4 Zusammenfassung und Ausblick

Im vorliegenden Beitrag wurde eine neuartige Stelzradzugmaschine zur praxistauglichen Ausbringung von biologischen Pflanzenschutzmitteln vorgestellt. Das kompakte und robuste Getriebekonzept erlaubt es dabei den Zielkonflikt zwischen einer hohen Bodenfreiheit und einer schnellen Verladefähigkeit aufzulösen. Bei der Gestaltung der Antriebsfunktion wurde im Sinne eines möglichst einfachen Aufbaus die Einsparung von Gliedern, Gelenken und Antrieben höher gewichtet als die Reduktion der auftretenden Stellkräfte. Das Konzept wurde mit Hilfe analytischer Berechnungen und einer Mehrkörpersimulation in Hinblick auf die maximal auftretenden Lagerkräfte optimiert.

Zukünftige Arbeiten umfassen die weitere Automatisierung der Maschine sowohl in Hinblick auf die Fahrzeugsteuerung der Niveauregulierung und Radseitenlenkung, als auch die Untersuchung und Implementierung einer prädiktiven Spurführung. Zudem ist eine Anpassung des Aufbaus in Hinblick auf eine wirtschaftliche Herstellung in Serie ausstehend.

Danksagung

Die Entwicklung und der prototypische Aufbau der Stelzradzugmaschine wurde als Verbundprojekt in Kooperation mit FAG - Fahrzeugwerk Aschersleben GmbH, Ematik GmbH sowie BIOCARE GmbH realisiert. Die Förderung erfolgte durch das Bundesministerium für Wirtschaft und Klimaschutz im Rahmen des Zentralen Innovationsprogramms Mittelstand (Förderkennzeichen 16KN074739).

Literatur

- [1] N.N.: TRICHOSAFE - Erfahrung und Qualität seit über 25 Jahren, Firmenschrift, BIOCARE Gesellschaft für biologische Pflanzenschutzmittel mbH, Einbeck, 2021.
- [2] N.N.: Vermeidung und Verminderung von Pestiziden in der Umwelt, Firmenschrift, Deutsche Bundestiftung Umwelt, Osnabrück, 2020.
- [3] N.N.: Amazone Pantera, Firmenschrift, AMAZONEN-WERKE H. DREYER SE & Co. KG, Hasbergen, 2023.
- [4] Y. Reckleben: Anbausysteme für Mais - Der richtige Reihenabstand, Zeitschriftenaufsatz, in Lohnunternehmen, 65. Jahrgang, Ausgabe 3, 2010, S. 28-30.
- [5] B. Blumenthal, H. Kirchmeier, M. Marx, R. Brandhuber, M. Demmel: Regelspurverfahren im Ackerbau, Zeitschriftenaufsatz, in Bayrische LfL, Schriftenreihe 2/2016, 2018. ISSN 1611-4159.
- [6] P. Naefe: Methodisches Konstruieren - Auf den Punkt gebracht. Wiesbaden: Springer Vieweg, 2018, S. 117-123. DOI: 10.1007 ISBN 978-3-658-22635-0.
- [7] P. Dudziński: Lenksysteme für Nutzfahrzeuge, Monographie. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2005, S.17, 20. DOI: 10.1007 ISBN 978-3-540-22788-5.
- [8] N.N.: Zürn 540 Hochrad-Geräteträger, Zürn Harvesting GmbH & Co KG Firmenschrift, Waldenburg, 2019.
- [9] R. Konradt, S. Schmidt: Design of an adjustable lightweight stilt tractor, Konferenzbeitrag, in LAND.TECHNIK AgEng 2023, VDI-Berichte, Band 2427, VDI Wissensforum GmbH, Hrsg., 2023, S. 289-294.
- [10] R. Konradt, H. Heidfeld, S. Schmidt, W. Beitzen-Heineke: Selbstfahrendes zweispuriges Stelzenfahrzeug, Offenlegungsschrift, Deutschland, DE 10 2022 114 860 A1, 2023.12.14.
- [11] H. Kerle, B. Corves, M. Hüsing: Getriebetechnik - Grundlagen, Entwicklung und Anwendung ungleichmäßig übersetzender Getriebe, 5 Auflage. Wiesbaden: Springer Vieweg, 2015, S.132. DOI: 10.1007 ISBN 978-3-658-10057-5.
- [12] P. R. Dahl: A solid friction model. TOR 158(3107-18), The Aerospace Corporation, El Segundo, CA, 1968, S.19.

Entwicklung eines Drahtfördersystems für Drähte mit sehr kleinem Durchmesser

Pascal Schnabel*; Holger Letsch**, Maik Berger***

* TU Chemnitz, Professur Montage- und Handhabungstechnik
pascal.schnabel@mb.tu-chemnitz.de

** TU Chemnitz, Professur Schweißtechnik
holger.letsch@mb.tu-chemnitz.de

*** TU Chemnitz, Professur Montage- und Handhabungstechnik
maik.berger@mb.tu-chemnitz.de

Kurzfassung

Beim gepulsten Laserstrahlschweißen kleiner Baugruppen erfolgt in vielen Fällen die Zugabe eines drahtförmigen Zusatzwerkstoffes zum Vermehren der Schmelze. Die derzeit existierenden automatisierten Drahtzuführsysteme versagen bei Drahtdurchmessern unter 0,3 mm und die Drahtförderung ist nur noch händisch möglich. Dieser Artikel beschreibt den Entwicklungsprozess eines auf einem Riementrieb basierenden Drahtvorschubsystems, das die Grenzen des Stands der Technik überwindet. Hierfür wird die Entwicklung des Lösungskonzeptes bis zum ersten funktionsfähigen Prototyp vorgestellt.

Titel / Abstract

Development of a precision wire feeder for small diameter wire

Pulsed laser beam welding of small assemblies often involves the addition of a wire-shaped filler material to increase the melt. The existing automated wire feeding systems fail with wire diameters below 0,3 mm and wire feeding is only possible manually. This article describes the development process of a wire feed system based on a belt drive that overcomes the limitations of the state of the art. The development of the solution concept up to the first functional prototype is presented, highlighting key challenges and solutions.

1 Einführung

Beim Laserstrahlschweißen kleiner Baugruppen erfolgt der Wärmeeintrag sehr konzentriert und lokal, wobei in ca. 25% der Fälle die Zugabe von Zusatzwerkstoff zum Vermehren der Schmelze vorteilhaft oder gar nötig ist. Die Zusatzwerkstoffe werden in Form von Drähten oder Stäben zum Laserstrahl geführt und dabei aufgeschmolzen.

Die manuelle Zufuhr des Zusatzwerkstoffes ist stark vom menschlichen Können abhängig, nicht reproduzierbar und unpräzise, was zu instabileren Schweißverbindungen führt. Aus diesem Grund existieren automatische Drahtvorschubsysteme, die den Zusatzwerkstoff mittels zweier gegenläufiger Rollen kontinuierlich in den Prozess einbringen [1-4]. Durch die bessere Handhabung und die erhöhte Präzision können die Qualität und die Reproduzierbarkeit der Schweißverbindung verbessert, ein größerer Durchsatz erzielt und Personalkosten eingespart werden.

Der Nachteil der Antriebsrollen ist die punktuelle Kontaktfläche, wodurch die übertragbare Schubkraft beschränkt ist und Schlupf entstehen kann. Bei Durchmessern von weniger als 0,3 mm ist ein schlupffreies Fördern der Drähte auch durch Optimierung der Rollen, z.B. durch Nuten und Riffelungen, nicht mehr möglich. Versuche mit dünnerem Material führten zu einem Verwirren und Verstopfen der Drahtförderer, was die technologischen Grenzen der aktuellen Systeme aufzeigt.

Für spezifische Anwendungsfälle, wie das Verschweißen dünnwandiger Kanülen oder Folien, sind Zusatzdrähte mit Durchmessern $< 0,3$ mm erforderlich, um eine unerwünschte Veränderung der Bauteilgeometrie durch übermäßigen Materialeintrag zu vermeiden.

Gegenwärtig ist die Zufuhr solch dünner Drähte nur manuell realisierbar, wodurch die Vorteile automatisierter Drahtfördersysteme nicht genutzt werden können und keine konstante Schweißnahtqualität gewährleistet ist. Aus den beschriebenen Ungenauigkeiten und Nachteilen der manuellen Materialeinbringung ergibt sich die dringende Notwendigkeit eines innovativen Drahtvorschubsystems.

Ziel des ZIM-geförderten Projekts „WireGuide“ war die Entwicklung eines automatischen Drahtvorschubsystems, das die Grenzen existierender Drahtvorschubsysteme überwinden kann. Eine Frage, die es dabei zu beantworten galt, war: Inwieweit lässt sich das Drahtvorschubsystem miniaturisieren und in ein handgeführtes Werkzeug integrieren?

Zusätzlich musste ein Vorgehen zur Auslegung des Vorschubsystems gefunden werden. Tabelle 1 fasst die wesentlichen Entwicklungsziele zusammen.

Tab. 1: Entwicklungsziele

Parameter	Entwicklungsziel
Drahtdurchmesser	0,1 – 0,5 mm
Drahtfördergeschwindigkeit	1 – 100 mm/s
Abmessungen Fördersystem	max. 150 * 40 * 40 mm
Kosten	≤1000€

2 Entwicklung des Lösungskonzeptes

2.1 Antriebskonzept

Zunächst wurde ausgehend von der vorangegangenen Markt- und Patentrecherche nach einem geeigneten Vorschubkonzept für Zusatzwerkstoffe mit rundem Querschnitt gesucht. Patentrechtlich geschützte Prinzipien sowie bereits marktverfügbaren und ungeeignete Konzepte konnten als erstes ausgeschlossen werden. Als Nächstes wurde versucht, durch Konzipierung geeigneter Wirkprinzipien, das Problem der aktuellen Drahtförderer, die punktuelle Übertragung der Vorschubkraft, zu überwinden. Hierfür konnten mechanische, pneumatische und magnetische Wirkprinzipien gefunden und anhand derer mehrerer Maschinenkonzepte entwickelt werden, die die Übertragung einer größeren Vorschubkraft ohne zu große Drahtverformung erlauben. Eine starke Drahtverformung führt zur plastischen Verformung des Drahtes, was ein verstärktes Ausknicken des Drahtes nach der Drahtauslassdüse zur Folge hat und die Präzision der Drahtzuführung beeinträchtigt. Bei der kraftschlüssigen Übertragung einer Vorschubkraft kann dies durch eine Vergrößerung der Kontaktfläche zwischen Antriebselement und/oder des Reibwertes erreicht werden.

Dieser Ansatz basierte auf der Überlegung, dass die maximal übertragbare Vorschubkraft F_{Vor} der Reibkraft zwischen Draht und Antriebselement entspricht. Die Reibkraft F_R wiederum ist abhängig vom Reibwert μ und der Normalkraft F_N . Begrenzt wird die maximale Anpresskraft F_F nur durch die Druckfestigkeit des Drahtwerkstoffes σ_{zul} sowie

der Kontaktfläche A . Dieser Zusammenhang lässt sich in den Gleichungen 1a bis 1c zusammenfassen. Diese Überlegung zeigte, dass bei einem Punkt- und Linienkontakt mit endlich kleiner Kontaktfläche eine sehr große Anpresskraft nötig ist und eine unzulässige Drahtverformung und Überschreitung der maximal zulässigen Druckfestigkeit zu erwarten ist.

$$F_{Vor} \leq F_R \quad (a); \quad F_R = F_N \cdot \mu \quad (b); \quad \sigma_{zul} \geq \frac{F_F}{A} = \frac{F_N}{A} \quad (c) \quad (1)$$

Die schrittweise Entwicklung mechanischer Wirkprinzipien ist in Abb. 1 dargestellt.

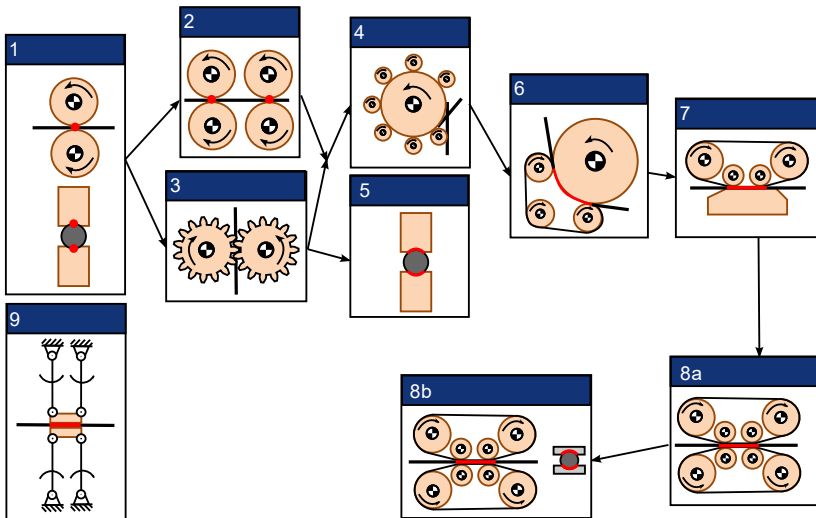


Abb. 1: schematische Darstellung der schrittweisen Entwicklung von mechanischen Vorschubprinzipien

Die aus den Wirkprinzipien entwickelten Maschinenkonzepte wurden im Anschluss anhand der Kriterien Maschinenkosten, Miniaturisierbarkeit, Komplexität, Drahtdurchmesserabhängigkeit, Flexibilität und Prozesssicherheit bewertet und miteinander verglichen. Magnetische Prinzipien erwiesen sich aufgrund der Notwendigkeit magnetischer Zusatzwerkstoffe oder höherer Kosten als nachteilig.

Pneumatische Wirkprinzipien erfordern eine zusätzliche Medienversorgung, wodurch die Portabilität und Flexibilität negativ beeinträchtigt wird. Konzepte auf Basis von Koppelgetrieben (9) führten zu komplexeren Konstruktionen, die die Miniaturisierbarkeit erschwerten und waren unflexibel. Konzepte mit starker Drahtverformung (4,6) erschweren die präzise Drahtzuführung aufgrund verstärkten Ausknickens nach der Drahtauslassdüse. Das Konzept eines doppelseitigen Zahnriemens (8) erzielte die beste Bewertung aufgrund seiner geringen Komplexität, flexiblen Einsetzbarkeit für verschiedene Drahtdurchmesser, Kosteneffizienz, guten Miniaturisierbarkeit und geringen Drahtverformung. Idealisiert betrachtet führt der Zahnriemen zu einem Linienkontakt (8a). Unter Berücksichtigung der Riemenverformung ergibt sich ein Flächenkontakt (8b). Somit konnte gegenüber dem Stand der Technik mit mehreren Punktkontakten, durch einen zweiseitigen Flächenkontakt, die Kontaktfläche vergrößert und die maximal mögliche Anpresskraft ohne unzulässige Drahtverformung gesteigert werden. Eine weitere Verbesserung der übertragbaren Kraft ist nur noch durch Vergrößerung der Kontaktfläche (längerer Riemen, Längsnut) oder durch eine Vergrößerung des Reibwertes möglich.

2.2 Gesamtkonzept

Das Problem der geforderten Miniaturisierung bei gleichzeitig begrenzten Fertigungskosten wurde durch die in Abb. 2 skizzierte Entkopplung des Antriebssystems vom Handwerkzeug mittels einer auswechselbaren, schlauchförmigen Verbindung gelöst.

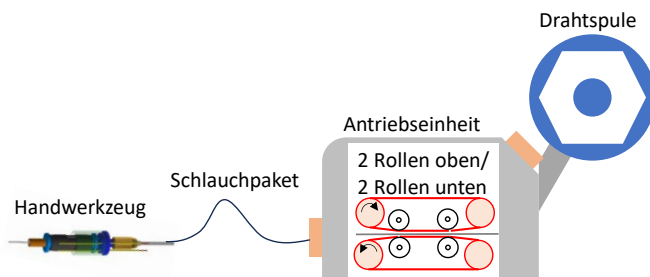


Abb. 2: schematische Darstellung des Gesamtkonzeptes

Der Schweißprozess gliedert sich in drei Phasen. Prozessbedingt ist zu Beginn des Schweißprozesses eine größere Menge an Zusatzwerkstoff zuzuführen, weshalb der Draht mit erhöhter Geschwindigkeit zu fördern ist (Schnellvorschub SVS). Im Anschluss wird mit normaler Fördergeschwindigkeit geschweißt (Normalvorschub NV). Am Ende des Prozesses wird, um das Festschweißen des Drahtes zu vermeiden, der Draht mit maximaler Geschwindigkeit zurückgezogen (Schnellrückzug SRZ). In Abb. 3 ist der idealisierte Geschwindigkeitsverlauf eines Drahtvorschubs dargestellt.

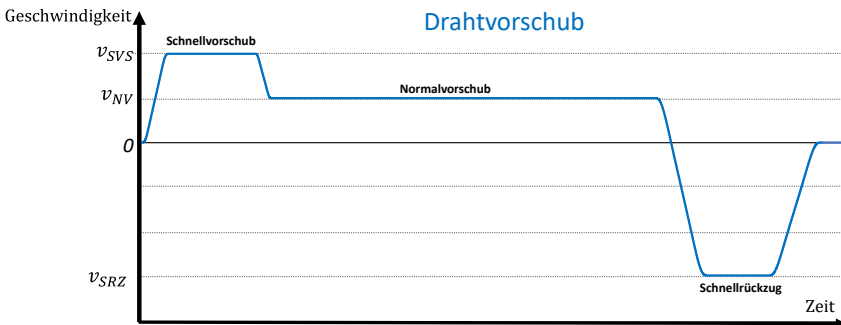


Abb. 3: idealisierter Geschwindigkeitsverlauf des Drahtvorschubs beim Laserschweißen

3 Auslegung und Dimensionierung

Zur Auslegung der Antriebskomponenten des Fördersystems war zunächst die notwendige Drahtvorschubkraft F_{Vor} zu bestimmen. Diese muss größer als die in der schlauchförmigen Verbindung auftretende Reibkraft $F_{Schlauch}$ und die zum Drahtabzug von der Rolle notwendige Kraft F_{Spule} sein. Die Reibkraft im Schlauch konnte experimentell mit einer Zugkraftwaage und einem Umlenkradius von 60 mm gemessen und mit den in [5] angegebenen Reibkräften verglichen werden. Der Maximalwert der beiden Werte 5 N wurde für $F_{Schlauch}$ angenommen. Die Kraft zum Abzug von der Spule ließ sich aus dem Massenträgheitsmoment der Spule und einer zuvor definierten Beschleunigung berechnen.

Die notwendige Beschleunigung beim Schnellvorschub wurde anhand der maximalen Pulsfrequenz des Laserschweißgerätes (30 Hz) sowie

dem empfohlenen Abstand der Drahtauslassdüse zum Laserspot (3 mm bei 0,3 mm Drahtdurchmesser [1]) bestimmt. Unter der Annahme einer konstanten Beschleunigung, einer maximalen Pulsfrequenz von 30 Hz sowie das der Draht nach dem 3. Puls den Laserspot erreicht, wurde eine Beschleunigung von $0,3 \text{ m/s}^2$ definiert. Die maximal mögliche Beschleunigung tritt beim Schnellrückzug auf und sollte das komplette Restmoment des Motors ausnutzen.

Als Nächstes erfolgte die Betrachtung der Wirkstelle zwischen Draht und Riemen. Zur Reduktion des Antriebsmoments musste die Materialpaarung Zahnriemen/ Zusatzwerkstoff einen möglichst großen Reibwert aufweisen und gleichzeitig verschleißarm sein. Eine Marktrecherche ergab, dass für die minimale und gewünschte Zahnriemengröße nur ein Polyurethanriemen standardmäßig verfügbar war. Aufgrund der Begrenzung durch die angestrebten Fertigungskosten (siehe Tab. 1) wurde dieser ausgewählt und Optimierungspotenzial, durch bspw. eine Riemenbeschichtung, zunächst offen gelassen. Bei einer idealisierten Betrachtung der Wirkstelle ergibt sich unter Vernachlässigung der Riemen- und Drahtverformung das in Abb. 4 skizzierte Reibmodell.

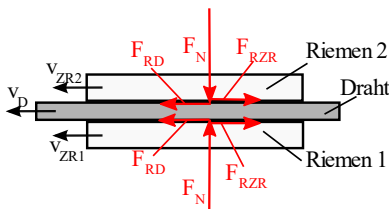


Abb. 4: Skizze des Reibmodells

Die von den Zahnriementrieben erzeugte Vorschubkraft ergibt sich aus der Summe der Tangentialkräfte beider Riemen. Um sicherzustellen, dass kein Schlupf auftritt, muss die Reibkraft zwischen Riemen und Draht am Draht F_{RD} größer als die Vorschubkraft sein. Über den Reibwert Stahl-Polyurethan ($\mu=0,2$) und einer Sicherheit gegen Rutschen S_R konnte im Anschluss die Normalkraft F_N bzw. die notwendige Anpresskraft F_F berechnet werden (Gl. 2).

$$F_F = S_R \cdot \frac{F_{Vor}}{\mu} \quad (2)$$

Zur Gewährleistung gleicher Umfangsgeschwindigkeiten von unterem und oberem Zahnriemen war eine Synchronisation der beiden Riementriebe notwendig. Insbesondere beim Richtungswechsel (Übergang von NV zu SRZ) können nicht synchron laufende Riemen zu Verspannungen führen, die das notwendige Antriebsmoment erhöhen [6]. Hierbei war zu beachten, dass aufgrund unterschiedlicher Drahtdurchmesser der Achsabstand beider Riementriebe einstellbar sein muss und die beiden Riementriebe gegenläufig angetrieben werden müssen. Nach einem Vergleich verschiedener in Frage kommender Getriebe wurde in Hinblick auf die angestrebte kompakte Baugröße und einfache Fertigung ein Zahnradgetriebe zur Synchronisation der beiden Riementriebe gewählt. Durch Verwendung einer achsabstandsunempfindlichen Evolventenverzahnung sollten die kleinen Abstandsunterschiede ($\pm 0,2$ mm) zwischen den beiden Riementrieben kompensiert werden. Möglichkeiten zur Einstellung bzw. Reduktion des Zahnradspiels [7] wurden zunächst nicht betrachtet.

Danach folgte die Gestaltung des Getriebes und kompakte Anordnung der einzelnen Zahnscheiben, Spannrollen und Zahnräder. Hierfür wurde zuerst die Kontaktstelle Draht/ Riemen betrachtet und die in [8] vorgestellten Konzepte verglichen. Der Vergleich zeigte, dass eine Vergrößerung des Umschlingungswinkel des Drahtes um den Riemen, der bei versetzten Führungsrollen auftritt, zur Vergrößerung der übertragbaren Vorschubkraft führt, aber gleichzeitig die Drahtverformung zunimmt. Zusätzlich entstehen durch versetzte Rollen erhebliche Relativbewegungen aufgrund voreilender und zurückbleibender Abschnitte, die zu vermeiden sind [7]. Eine größere Zahl an Führungsrollen erhöht die Zahl der Reibstellen sowie die Komplexität des Getriebes und steht einer miniaturisierten Baugröße entgegen. Um Komplexität und Baugröße zu minimieren, wurde entschieden, zwei Führungsrollen gegenüber anzuordnen.

Im Anschluss erfolgte die Anordnung der restlichen Zahnriemenscheiben, Spannrollen und Zahnräder. Die Anordnung ist in Abb. 5 dargestellt.

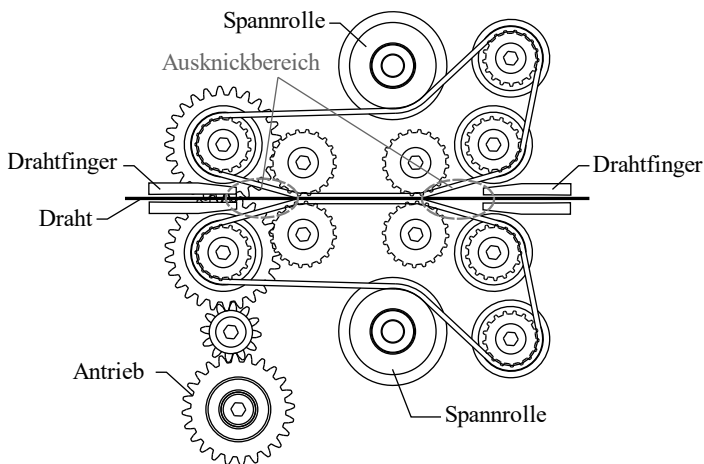


Abb. 5: Anordnung der Zahnräder, Zahnscheiben und Spannrollen

Zur Dimensionierung des Antriebs wurde ein vereinfachtes Berechnungsmodell entwickelt, das nachfolgend beschrieben wird. Das Antriebsmoment setzt sich aus dem notwendigen Beschleunigungsmoment M_a , dem notwendigen Moment zur Überwindung der Reibung M_R , dem Moment zur Erzeugung der benötigten Vorschubkraft M_{Vor} sowie dem Wirkungsgrad des Zahnradgetriebes η zusammen:

$$M_{an} = (M_a + M_R + M_{Vor}) \cdot \eta \quad (3)$$

Reibungsverluste entstehen u.a. an den Wellenlagerungen sowie den Verzahnungen der Zahnräder. Die Verluste an den Wellenlagerungen resultieren aus der Anpresskraft des oberen Zahnriemens, der durch die Vorspannkraft des Zahnriemens wirkende Kraft auf die Lager sowie dem Lagerdurchmesser. Weitere Verluste, die bspw. aus der Erwärmung der Getriebeglieder oder Verunreinigungen resultieren, wurden vernachlässigt.

Im Anschluss war das Beschleunigungsmoment anhand der zuvor definierten Beschleunigung, den Massenträgheitsmomenten der Getriebeglieder J_{red} , dem Durchmesser der Zahnscheibe d_{zS} sowie der Übersetzung des Zahnradgetriebes i ermittelbar (Gl. 4).

$$M_a = \sum_i J_{red_i} \cdot \frac{a}{d_{zS}} \cdot 2 \cdot i \quad (4)$$

Das Vorschubmoment M_{Vor} ergab sich aus der Vorschubkraft F_{Vor} sowie dem Zahnscheibendurchmesser d_{zS} und der Zahnradübersetzung i (Gl. 5). Das durch die Anpresskraft des oberen Riemens erzeugte Reibmoment M_R war über den Lagerdurchmesser d_{Lager} und den Lagerreibwert μ_{Lager} berechenbar (Gl. 6). Analog dazu konnte das durch die Vorspannkraft F_V erzeugte Reibmoment an jeder Lagerstelle M_{R_i} aus dem Umschlingungswinkel des Riemens α_i berechnet werden (Gl. 7). Die Vorspannkraft wurde nach den in [7] beschriebenen Hinweisen gewählt.

$$M_{Vor} = F_{Vor} \cdot \frac{d_{zS}}{2} \cdot i^{-1} \quad (5)$$

$$M_R = F_F \cdot \mu_{Lager} \cdot \frac{d_{Lager}}{2} \cdot i^{-1} \quad (6)$$

$$M_{R_i} = \frac{F_V}{\sin(\alpha)} \cdot \mu_{Lager} \cdot \frac{d_{Lager}}{2} \cdot i^{-1} \quad (7)$$

Basierend auf diesem Modell und unter Berücksichtigung der Massenträgheitsmomente der Getriebeelemente, Zahnriemenparameter, üblicher Wirkungsgrade von Zahnradgetrieben sowie standardmäßiger Lagerreibwerte wurde ein Motormoment von $M_{an} = 23 \text{ Ncm}$ berechnet. Mithilfe des berechneten Motormoments, dem Wunsch nach einer hohen Genauigkeit, einfachen Steuerung sowie kleiner Baugröße wurde ein Schrittmotor (Baugröße: Nema 17) mit einer Sicherheit von ca. 1,5 zum Antrieb des Vorschubsystems gewählt.

4 Konstruktion und Gestaltung

Die größte Herausforderung bei der Überführung des Lösungskonzeptes in eine konstruktive Gestaltung lag in der angestrebten Miniaturisierung des Antriebskonzeptes unter Berücksichtigung einer einfachen, kostengünstigen Fertigung und Montage. Dies wurde durch die Entkopplung

der Antriebseinheit vom Handwerkzeug mittels einer auswechselbaren, flexiblen schlauchförmigen Verbindung erreicht. Hierfür wurde durch eine Kombination aus Fräsen und additiver Fertigung ein ergonomisches Handgriffstück gefertigt und ein Knopf zur Steuerung des Drahtvorschubs vorgesehen. Der Knopf ist über eine im Schlauch integrierte Signalleitung mit der Antriebseinheit verbunden. Das innovative Handwerkzeug ist in Abb. 6 dargestellt.

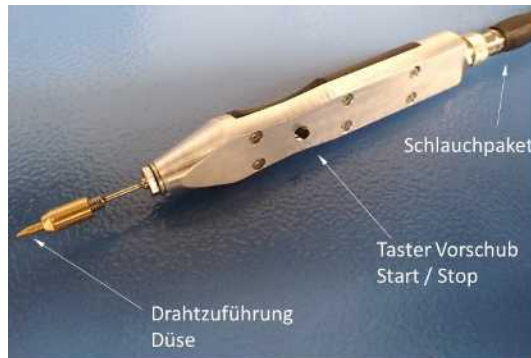


Abb. 6: Handwerkzeug

Weitere Herausforderungen bestanden in der Minimierung des möglichen Ausknickbereiches, der Fertigung der winzigen Einzelkomponenten und der Gewährleistung eines einfachen schnellen Zusatzwerkstoffwechsels. Hierfür wurde der Riementrieb möglichst nah an die Drahtein- und -auslaufstelle gelegt und ein Drahtführungsrohr („Drahtfinger“) an die Ein- und Auslaufseite des Riemens montiert (siehe Abb. 5). Das größte Problem war die Herstellung der extrem kleinen Bauteile. Speziell die winzigen Drahtführungsdüsen mussten mehrfach mittels Mikrobohren gefertigt werden, da mehrere Bohrungen ungenau bzw. fehlerhaft waren und anfangs unklar war, welcher Bohrungsdurchmesser der Drahtdüse eine präzise und gleichzeitig reibungsarme Drahtförderung erlaubt. Zur Gewährleistung einer einfachen Montage und zugleich kompakten Bauweise der Antriebseinheit ist der Riementrieb nur einseitig gelagert und der Abstand zwischen den beiden parallelen Riemen sowie deren Anpresskraft auf den Draht durch eine Schraube einstellbar. Durch den Einsatz einer Spannrolle ist die Spannung jedes Zahnriemens anpassbar.



Abb. 7: Prototyp des Drahtförderers

5 Zusammenfassung und Ausblick

Im Rahmen des Forschungsprojekts konnte erfolgreich das in Abb. 7 dargestellte funktionsfähige Drahtvorschubsystem entwickelt werden, das die in Tab. 1 definierten Ziele erfüllt. Das Vorschubsystem ist mit einer Größe von 115x140x100 mm sehr kompakt und in Kombination mit dem entwickelten Handwerkzeug zur Zufuhr von Drähten mit Durchmessern 0,1 bis 0,5 mm einsetzbar.

Zukünftig wird angestrebt das Antriebssystem weiter zu miniaturisieren und direkt in das Handwerkzeug zu integrieren, sodass auch stabförmige Zusatzwerkstoffe zuführbar sind. Des Weiteren soll der Wechsel des Zusatzwerkstoffes vereinfacht werden. Bei jedem Wechsel ist das Schlauchpaket auszutauschen, der Abstand zwischen beiden Riemen neu einzustellen und ein neuer Draht einzufädeln. Die ersten Versuche offenbarten, dass insbesondere das Neueinfädeln viel Fingerspitzengefühl erfordert und sehr zeitaufwendig ist.

Im Verlauf des Forschungsprojekts konnte ein erster vereinfachter Berechnungsansatz zur Auslegung riemenbasierter Vorschubsysteme, unter Vernachlässigung von Verformungen und Adhäsionswirkungen, aufgestellt und erfolgreich angewandt werden.

Unbeantwortet bleibt die Frage, welche Zahnriemen/ Zusatzwerkstoff Kombination die optimalen Fördereigenschaften ermöglicht und welchen Einfluss Verunreinigungen und Verschleiß auf den Förderprozess haben. Zukünftig wird angestrebt in weiteren experimentellen Unter-

suchungen die noch offenen Fragen zu beantworten und den Berechnungsansatz zu validieren.

Danksagung: Dieser Beitrag wurde gefördert durch das Bundesministerium für Wirtschaft und Energie aufgrund eines Beschlusses des Deutschen Bundestages.

Literatur

- [1] Ingenieurbüro Hannweber GmbH. Betriebsanleitung. Programmierbares Drahtvorschubsystem Lafet. Hrsg. von L&A Lasertechnik und Applikationen GmbH. 2015
- [2] Brandon, E., Hooper, F. und Reichenbach, M. „Precision wire feeder for small diameter wire“. US5137223A. National Technology and Engineering Solutions of Sandia LLC. 1982. url: <https://patentimages.storage.googleapis.com/46/3b/6b/b20cb12157ab64/US5137223.pdf> (besucht am 06.06.2024).
- [3] Oeste, J. „Schweißvorrichtung mit neuartiger Drahtvorschubvorrichtung“. DE102015003083A1. Alexander Binzel Schweisstechnik GmbH and Co KG. 2024. url: <https://patentimages.storage.googleapis.com/08/12/7a/13ac1c864e2e6c/DE102015003083A1.pdf> (besucht am 06.06.2024).
- [4] Alexander Binzel GmbH & Co KG. „Drahtvorschubsystem“. DE29915301U1. Alexander Binzel GmbH and Co KG. 2000. url: <https://register.dpma.de/DPMAregister/pat/PatSchrifteneinsicht?docId=DE29915301U1> (besucht am 06.06.2024).
- [5] Fichter, R.: „Optimale „Förderbarkeit“ von Schweißdraht. Prozess-Sicherheit in der Verarbeitung durch sauberen Draht mit idealen Fördereigenschaften“. Internetdokument. url: [www. https://www.georeinigungstechnik.de/fileadmin/user_upload/pdf/fo_rderbarkeit_von_schweissdraehten_version_12.04.2021.pdf](https://www.georeinigungstechnik.de/fileadmin/user_upload/pdf/fo_rderbarkeit_von_schweissdraehten_version_12.04.2021.pdf) (besucht am 16.04.2024).
- [6] Schumann, P.: „Treibkraft und Systemverhalten von Hüllandruckwinden“ Dissertation, Fakultät für Maschinenwesen der TU Dresden (2022): Dissertation
- [7] Roth, K.: Zahnradtechnik Evolventen-Sonderverzahnungen zur Getriebeverbesserung. Springer-Verlag, 2013, S. 127. isbn: 978-3-642-63778-0. doi: 10.1007/978-3-642-58890-7.

- [8] Perneder, R.: *Handbuch Zahnriementechnik. Grundlagen, Berechnung, Anwendungen*. Springer-Verlag, 2009, S. 216–219. isbn: 978-3-540-89321-9. doi: 10.1007/978-3-540-89322-6.

Erzeugung von Polygonprofilen für Welle-Nabe-Verbindungen

Uwe Bäsel

HTWK Leipzig, Professur Maschinenelemente und Getriebetechnik
uwe.baesel@htwk-leipzig.de

Kurzfassung

Polygonprofile werden als formschlüssige Welle-Nabe-Verbindungen eingesetzt. Die Geometrie der speziellen Polygonprofile P3G und P4C ist in den Normen DIN 32711-1 und DIN 32712-2 festgelegt. Historisch erfolgte die Fertigung von Polygonprofilen auf speziellen Werkzeugmaschinen. Ausgehend von der Definition der Profile werden in diesem Artikel mehrere kinematische Fertigungsvarianten hergeleitet und diskutiert, wobei sich insbesondere auch bekannte Varianten ergeben. Abschließend wird zum Vergleich kurz auf die CNC-Fertigung von Polygonprofilen eingegangen.

Title / Abstract

Generation of polygon profiles for shaft-hub connections

Polygon profiles are used as form-fit shaft-hub connections. The geometry of the special polygon profiles P3G and P4C is defined in the German industrial standards DIN 32711-1 and DIN 32712-2. Historically, polygon profiles were manufactured on special machine tools. Based on the definition of the profiles, several kinematic generation variants are derived and discussed in this article, whereby known variants arise in particular. Finally, the CNC manufacturing of polygon profiles is briefly discussed for comparison.

1 Einführung

Abb. 1 zeigt beispielhaft ein Zahnrad mit einem P3G-Polygonprofil nach DIN32711-1 [1].

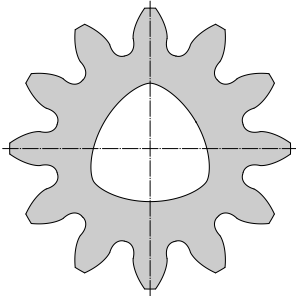


Abb. 1: Zahnrad mit Polygonprofil (P3G-Profil)

Im Sinne begrifflicher Klarheit geben wir ausgehend von den Normen DIN32711-1 [1] und DIN32711-2 [2] die folgende Definition (siehe auch [3], [4]):

Definition 1 Eine durch die reelle Parameterdarstellung

$$\left. \begin{aligned} x(\varphi) &= [R - e \cos(n\varphi)] \cos \varphi - ne \sin(n\varphi) \sin \varphi, \\ y(\varphi) &= [R - e \cos(n\varphi)] \sin \varphi + ne \sin(n\varphi) \cos \varphi, \end{aligned} \right\} 0 \leq \varphi \leq 2\pi, \quad (1)$$

gegebene parametrisierte Kurve mit natürlicher Zahl n und reellen Parametern $R > 0$ und $e > 0$ bezeichnen wir als P_n -Kurve.

Eine ebene konvexe Fläche nennen wir P_n G-Profil, wenn ihre Begrenzung durch die komplette Kurve (1) gegeben ist; wir nennen sie P_n C-Profil, wenn ihre Begrenzung abwechselnd stückweise durch (1) und Bögen eines Kreises mit Mittelpunkt im Koordinatenursprung definiert ist.¹

Abb. 2 zeigt ein Beispiel für eine P3-Kurve mit zugehörigem P3G-Profil. Abb. 3 zeigt ein Beispiel für eine P4-Kurve und ein P4G-Profil, das stückweise durch Teile dieser P4-Kurve und Bögen des Kreises mit Radius r_1 begrenzt wird.

¹Für die Welle betrachten wir das Material, für die Nabe, wie in Abb. 1, das Loch. Nicht alle Profile nach dieser Definition sind praxistauglich.

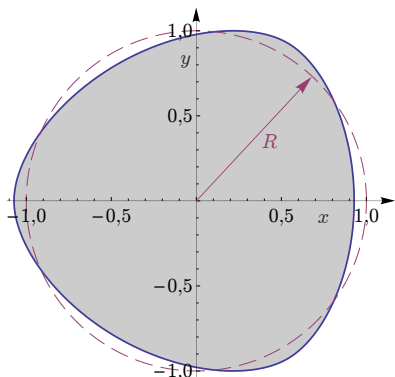


Abb. 2: P3-Kurve (blau) und P3G-Profil (schattiert) mit $R = 1$ und $e = 0,072$

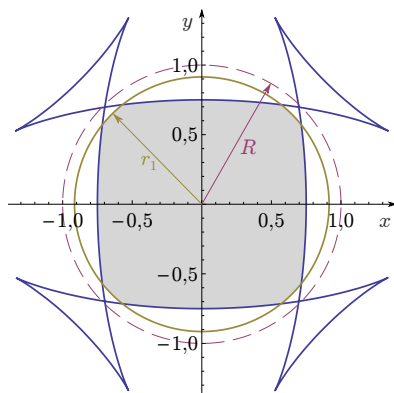


Abb. 3: P4-Kurve (blau) und P4C-Profil (schattiert) mit $R = 1$, $e = 0,25$ und $r_1 = 0,914$

Die Parameterdarstellung (1) kann man in der komplexen Form

$$z(\varphi) = e^{i\varphi} (R - e \cos(n\varphi) + ine \sin(n\varphi)) \quad (2)$$

schreiben. Für die Ableitung erhält man leicht

$$z'(\varphi) = ie^{i\varphi} \left(R + (n^2 - 1) e \cos(n\varphi) \right). \quad (3)$$

Eine P_n -Kurve kann nur dann die Begrenzung eines P_n G-Profiles sein, wenn sie keine Selbstüberschneidungen hat; die Kurve darf deswegen, abgesehen von Spitzen, keine Singularitäten haben. $z'(\varphi)$ kann nur dann gleich null sein, wenn der Ausdruck in den äußeren Klammern verschwindet. Ein P_n G-Profil erhält man deswegen nur, wenn dieser Ausdruck für $0 \leq \varphi < 2\pi$ größer oder gleich null ist. Wegen $-1 \leq \cos(n\varphi) \leq 1$ gilt dann

$$R - (n^2 - 1) e \geq 0, \quad n = 1, 2, 3, \dots, \quad (4)$$

also

$$e \leq \frac{R}{n^2 - 1}, \quad n = 2, 3, 4, \dots \quad (5)$$

Für $R = (n^2 - 1) e$ hat die Begrenzungskurve des P_n G-Profiles Spitzen.

2 Kinematische Erzeugung der Profilkurven

Aus (2) folgt²

$$z(\varphi) = \ell_2 e^{i\varphi} + \ell_3 e^{i[\pi-(n-1)\varphi]} + \ell_4 e^{i(n+1)\varphi} \quad (6)$$

mit

$$\ell_2 = R = \overline{A_0A}, \quad \ell_3 = \frac{1}{2}(n+1)e = \overline{AB}, \quad \ell_4 = \frac{1}{2}(n-1)e = \overline{BC} .^3$$

Die Strecke $\overline{A_0A}$ kann man als um den Koordinatenursprung A_0 mit Winkel φ rotierende Kurbel (Teil 2) betrachten (siehe Abb. 4).

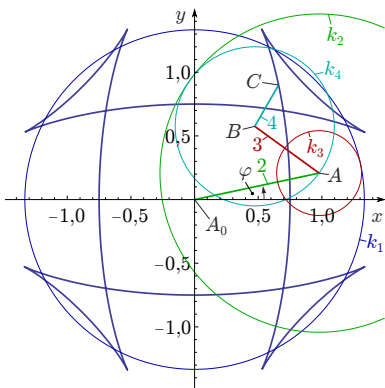


Abb. 4: Kinematische Erzeugung der P4-Kurve in Abb. 3

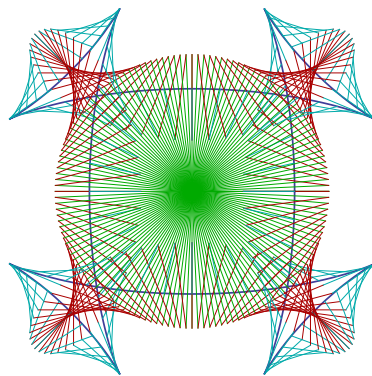


Abb. 5: Bewegungssequenz des Mechanismus in Abb. 4

Im Punkt A ist Teil 2 mit der Strecke AB (Teil 3) drehgelenkig verbunden. Punkt B ist der Drehpunkt für die Verbindung von AB und BC (Teil 4). Der Punkt C beschreibt die Profilkurve. Teil 1 ist das Gestell. Die Bewegung der Teile 2, 3 und 4 ist in Abb. 5 gezeigt. Störend an dieser Art der kinematischen Erzeugung, die mittels eines zweistufigen Planetengetriebes realisiert werden kann (entsprechende Wälzkreise sind in Abb. 4 eingezeichnet), ist, dass man ein Werkzeug mit Durchmesser gleich null benötigt.

²Bezüglich der einzelnen Schritte siehe [5].

³Siehe auch [6, Gleichung (5)].

3 Systematische Lösungsfindung und traditionelle Fertigung

Als Spezialfall von Gleichung (6) mit $R = 9$, $n = 3$ und $e = 1$ erhält man

$$z(\varphi) = 9e^{i\varphi} + 2e^{i(\pi-2\varphi)} + e^{4i\varphi} \quad (7)$$

und somit den Mechanismus und die Kurve mit (auf Länge 5 normierten) Tangentenvektoren in Abb. 6. (Der Mechanismus ist verdoppelt, wobei die Antriebswinkel φ um 180° versetzt sind.) Wir benutzen diese spezielle Variante nachfolgend beispielhaft als Referenz und entwickeln aus ihr systematisch weitere Lösungen zur Erzeugung derselben Konturkurve (P_n -Kurve). Dabei ergeben sich auch bekannte traditionelle Herstellungsverfahren.

Die drei Summanden auf der rechten Seite von (6) können in sechs verschiedenen Reihenfolgen angeordnet werden, was entsprechend auch für die Getriebeglieder (Strecken) des erzeugenden Mechanismus gilt. Die Verwendung der Reihenfolge

$$z(\varphi) = \ell_3 e^{i[\pi-(n-1)\varphi]} + \ell_4 e^{i(n+1)\varphi} + \ell_2 e^{i\varphi} \quad (8)$$

liefert den Mechanismus in Abb. 7. Man erkennt, dass hier im Unter-

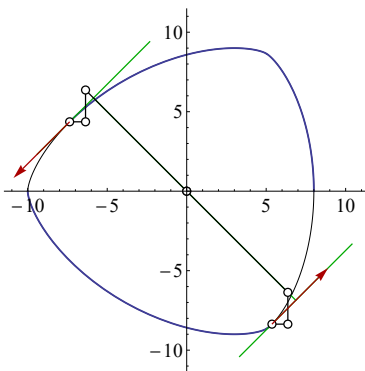


Abb. 6: Spezielle P3-Kurve mit Tangentenvektoren und erzeugendem Mechanismus

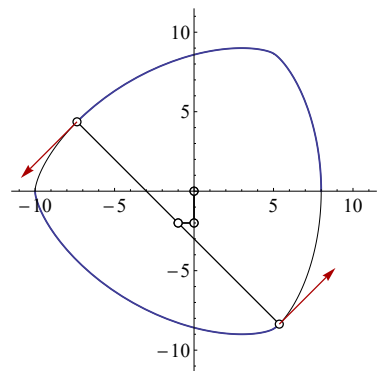


Abb. 7: Mechanismus nach Gleichung (8)

schied zu Abb. 6 - und das gilt allgemein, wie man leicht zeigt - die Tangentenvektoren orthogonal zum letzten Getriebeglied der Reihenfolge sind. (Auch in Abb. (4) war das nicht der Fall.) Diese Tatsache ermöglicht die Fertigung mit einem kreiszylindrischen Werkzeug (mit Radius r), wie z. B. einem Fräser oder einer Schleifscheibe. Das Werkstück ist gestellfest und die Kurve des Werkzeugmittelpunktes ist die Parallelkurve im Abstand r zur zu erzeugenden Konturkurve.

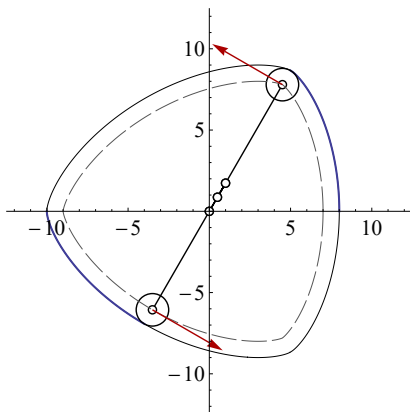
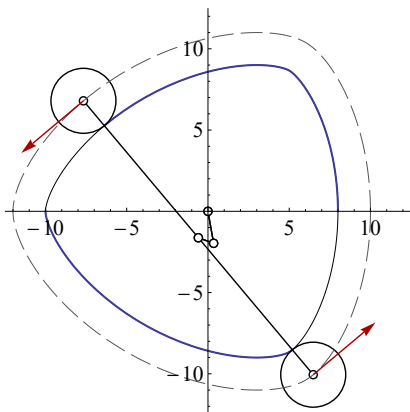


Abb. 8: Herstellung der Außenkontur Abb. 9: Herstellung der Innenkontur

Abb. 8 zeigt das Prinzip der Herstellung der äußeren Kontur (Welle) unter Verwendung eines Werkzeugs mit Radius $r = 2$, und Abb. 9 die Verwendung dieses Prinzips für die Herstellung der inneren Kontur der Nabe unter Verwendung eines Werkzeugs mit Radius $r = 1$. Für diese innere Kontur ist $r = 1$ der maximal mögliche Werkzeugradius, weil - wie man leicht nachrechnet - der kleinste Krümmungsradius der Kurve gleich 1 ist. Im Gegensatz dazu kann der Werkzeugradius r für die Wellenkontur beliebig groß gewählt werden.

Multiplikation von (8) mit $e^{-i\varphi}$ (kinematische Umkehr) ergibt

$$z(\varphi) e^{-i\varphi} = \ell_3 e^{i(\pi-n\varphi)} + \ell_4 e^{in\varphi} + \ell_2 . \tag{9}$$

Diese Gleichung bedeutet, dass nun die Kurve (das Werkstück) rotiert, während das Getriebeglied mit Länge ℓ_2 nur eine Translationsbewegung

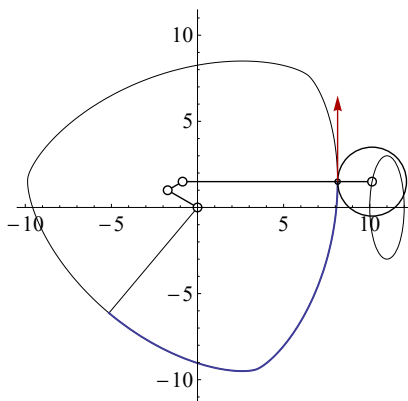


Abb. 10: Herstellung der Außenkontur mit rotierendem Werkstück (Welle)

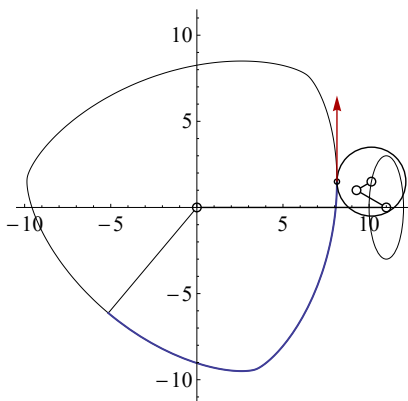


Abb. 11: Herstellung der Außenkontur mit rotierendem Werkstück (Welle)

ausführt (siehe Abb. 10). Wir betrachten die Bahnkurve der Werkzeugmittelpunktes. Die Gleichung dieser Kurve ist durch die rechte Seite von (9) gegeben. Es folgt, wobei wir die Reihenfolge der Summanden ändern, l_2 , l_3 und l_4 von (6) verwenden und mit r wiederum den Werkzeugradius bezeichnen,

$$\begin{aligned}
 & l_2 + r + l_3 e^{i(\pi-n\varphi)} + l_4 e^{in\varphi} \\
 &= l_2 + r - l_3 e^{-in\varphi} + l_4 e^{in\varphi} \\
 &= R + r - \frac{1}{2} (n+1) e e^{-in\varphi} + \frac{1}{2} (n-1) e e^{in\varphi} \\
 &= R + r - e \frac{e^{in\varphi} + e^{-in\varphi}}{2} + ne \frac{e^{in\varphi} - e^{-in\varphi}}{2}
 \end{aligned}$$

und somit

$$l_2 + r + l_3 e^{i(\pi-n\varphi)} + l_4 e^{in\varphi} = R + r - e \cos(n\varphi) + ine \sin(n\varphi). \quad (10)$$

Das ist die Parameterdarstellung einer Ellipse (vergl. (2)), die n -mal im mathematisch negativen Sinn durchlaufen wird, ihren Schwerpunkt im Punkt $R + r$ hat und ihren Startpunkt im Punkt $R + r - e$. Die Länge der kleinen Halbachse ist gleich e , die der großen gleich ne .

Schreibt man (9) in der Form

$$z(\varphi) e^{-i\varphi} = \ell_2 + \ell_3 e^{i(\pi-n\varphi)} + \ell_4 e^{in\varphi},$$

so erhält man den Mechanismus in Abb. 11. Die Parameterdarstellung der Bahnkurve des Werkzeugmittelpunktes ist wiederum durch (10) gegeben. Verglichen mit der Variante in Abb. 10 hat die Variante in Abb. 11 den Vorteil, dass ein Teil weniger bewegt werden muss. Die Variante in Abb. 11 ist im Wesentlichen die kinematische Umkehrung der Variante in Abb. 6. Das erkennt man mittels (3): Wenn wir ℓ_2 durch $\ell_2 + r$ ersetzen, und folglich R durch $R + r$, dann ändert der Tangentenvektor nur seine Länge, nicht aber seine Richtung. Der nicht gezeigte Tangentenvektor des Werkzeugmittelpunktes bezüglich der rotierenden Kurve in Abb. 11 hat dieselbe Richtung wie der gezeigte Tangentenvektor an die Kurve in ihrem momentan erzeugten Punkt. Das muss selbstverständlich der Fall sein, da die Werkzeugmittelpunktkurve bezüglich des rotierenden Werkstücks die Parallelkurve der zu erzeugenden Konturkurve ist. Die Variante in Abb. 11 ist im Wesentlichen die Lösung in [7, Abb. 9], [8, Abb. 2]. Auch dort beschreibt der Werkzeugmittelpunkt die oben erwähnte Ellipse (bzgl. des festen Teiles/Koordinatensystems).

Die elliptische Bahnkurve des Werkzeugmittelpunktes kann durch Abrollen eines Kreises (grün) mit Radius $\varrho_4 = (n + 1) e/2$ auf der Innenseite eines gestellfesten Kreises mit doppeltem Radius ϱ_2 (siehe Abb. 12) erzeugt werden; die Ellipse ist eine spezielle Hypotrochoide. Diese Kreise können die Teil- bzw. Wälzkreise eines außen- und eines innenverzahnten Zahnrades sein, die mechanische Lösung ist also ein Umlaufrädergetriebe. Es ist Zufall, dass das Werkzeug und der abrollende Kreis in Abb. 12 gleichen Radius $r = 2$ haben. Abb. 13 zeigt eine Variante mit Werkzeugradius $r = 10$.

Wir können (9) als

$$z(\varphi) e^{-i\varphi} - \ell_3 e^{i(\pi-n\varphi)} - \ell_4 e^{in\varphi} = \ell_2$$

und folglich auch als

$$\ell_3 e^{-in\varphi} + \ell_4 e^{i(\pi+n\varphi)} + z(\varphi) e^{-i\varphi} = \ell_2.$$

schreiben.

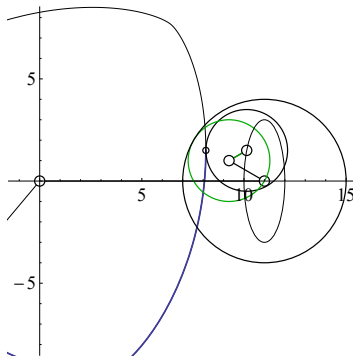


Abb. 12: Umlaufrädergetriebe zur Erzeugung der Bewegung des Werkzeugmittelpunktes in Abb. 11

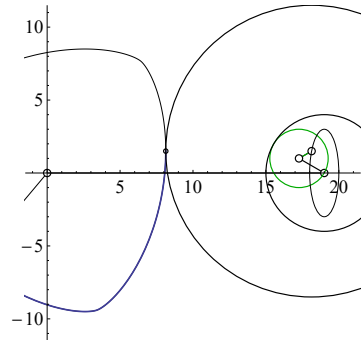


Abb. 13: Umlaufrädergetriebe wie in Abb. 12, aber größerer Werkzeugradius

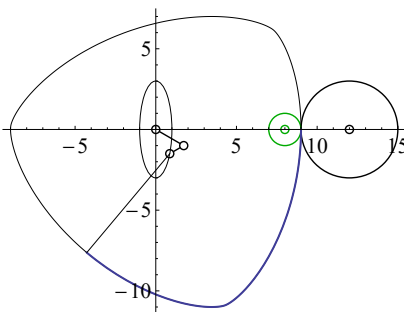
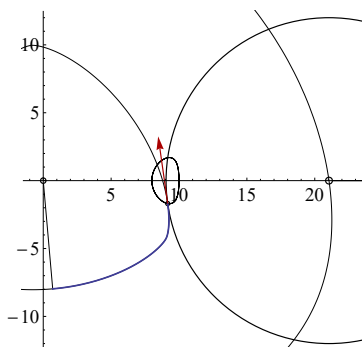
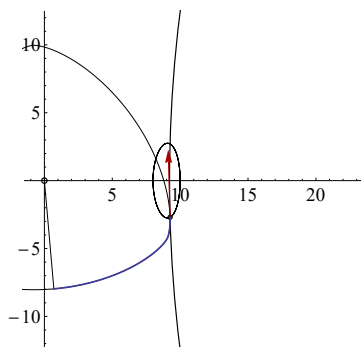


Abb. 14: Variante mit festen Werkzeugmittelpunkten

Das liefert die Lösung in Abb. 14, in der mögliche Werkzeuge mit festen Mittelpunkten für die Fertigung von Außen- bzw. Innenkontur gezeigt sind (siehe auch [8, Abb. 26]). Diese Variante ist im Wesentlichen eine kinematische Umkehrung der Lösung in Abb. 10 und hat den Vorteil, dass eine existierende ortsfeste Schleifscheibe verwendet und die Bewegung des Werkstücks durch eine Zusatzeinrichtung erzeugt werden kann; die Bewegung des Werkstücks ist jedoch schwierig zu realisieren, z. B. unter Verwendung des Umlaufrädergetriebes in den Abbildungen 12 und 13.

4 NC-Fertigung

Die bisher diskutierten Fertigungsvarianten erfordern spezielle Werkzeugmaschinen. Heutzutage werden für die Fertigung von P_n -Profilen flexible CNC-Maschinen bevorzugt. Für unsere Zwecke ist eine CNC-Maschine ausreichend, bei der der Werkzeugmittelpunkt eine gesteuerte Linearbewegung ausführen kann. Wir nehmen an, dass sich der Werkzeugmittelpunkt entlang der (horizontalen) x -Achse bewegt. Dann ist die aktuelle x -Koordinate des Werkzeugmittelpunktes der rechte Schnittpunkt zwischen der Parallelkurve im Abstand r zur um den Winkel $-\varphi$ gedrehten P_n -Kurve und der x -Achse (siehe Abbildungen 15 und 16).

Abb. 15: Radius $r = 12$ Abb. 16: Radius $r = 100$

Zur Berechnung dieses Schnittpunktes müssen wir zuerst den Wert ψ des Parameters der Kurve z im Schnittpunkt bestimmen, wofür wir in *Mathematica* unter Verwendung von `FindRoot` die Funktion $f_r(\varphi)$ mittels

```
f[\[CurlyPhi]_, r_] := \[Psi] /.
  FindRoot[Im[(z[\[Psi]] - I r z1[\[Psi]]/Abs[z1[\[Psi]])*
    Exp[-I \[CurlyPhi]]] == 0, {\[Psi], \[CurlyPhi]}
```

definieren, wobei z_1 die erste Ableitung von z (siehe (3)) ist. Die x -Koordinatenfunktion $x_r(\varphi)$ des Werkzeugmittelpunktes berechnet man dann mit

```
Re[(z[f[\[CurlyPhi], r]] - I r z1[f[\[CurlyPhi], r]])/
  Abs[z1[f[\[CurlyPhi], r]])*Exp[-I \[CurlyPhi]]]
```


Als Beispiele werden in Abb. 17 die Graphen der Funktionen f_1 und f_{12} gezeigt. Die zugehörigen Graphen der geschifteten Funktionen

$$\tilde{x}_r(\varphi) := x_r(\varphi) - R - r$$

für $r = 1$ und $r = 12$ zeigt Abb. 18.

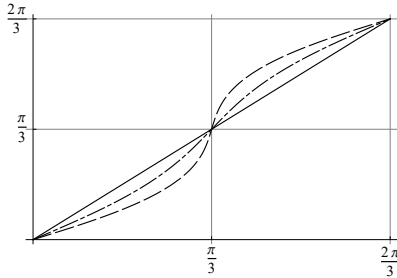


Abb. 17: f_1 (gestrichelt), f_{12} (strichpunktiiert) und f_∞

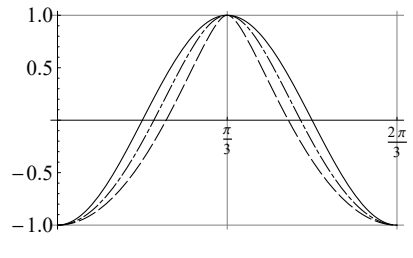


Abb. 18: \tilde{x}_1 (gestrichelt), \tilde{x}_{12} (strichpunktiiert) und \tilde{x}_∞

Für $r \rightarrow \infty$ konvergiert der Berührungspunkt zwischen Werkzeug und Werkstück gegen denjenigen in der traditionellen Fertigung mit rotierendem Werkstück (siehe Abbildungen 11, 12, 13), folglich $f_\infty(\varphi) := \lim_{r \rightarrow \infty} f_r(\varphi) = \varphi$ (siehe Abb. 17). Für $r \rightarrow \infty$ konvergiert die Bahnkurve der Berührungspunkts gegen die Ellipse mit der Parametergleichung

$$R - e \cos(n\varphi) + ine \sin(n\varphi)$$

(siehe Abbildungen 15, 16 und 10, 11; vergl. (2), (10)). Man beachte, dass $R - e \cos(n\varphi)$ die Stützfunktion der P_n -Kurve ist. Aus (10) folgt unmittelbar

$$\tilde{x}_\infty(\varphi) := \lim_{r \rightarrow \infty} \tilde{x}_r(\varphi) = -e \cos(n\varphi)$$

(siehe Abb. 18)

In DIN 3689-1 [9] ist die Verwendung von Hypotrochoiden als Profilkurven standardisiert, was uns in gewisser Weise zum Startpunkt der Entwicklung in Musyl [10] zurückbringt (siehe auch [11]), wo allerdings Epitrochoiden verwendet wurden. Selbstverständlich erlauben CNC-Maschinen die Fertigung fast beliebiger Profilkurven; jedoch macht die resultierende Flexibilität die praktische Anwendung komplexer und schwieriger zu handhaben.

Literatur

- [1] DIN32711-1. *Welle-Nabe-Verbindung – Polygonprofil P3G – Teil 1: Allgemeines und Geometrie*. 2023-09.
- [2] DIN32711-2. *Welle-Nabe-Verbindung – Polygonprofil P3G – Teil 2: Berechnung und Dimensionierung*. 2023-09.
- [3] HEXAGON software. *WN13 Software for calculation of PnG polygon profile shaft to collar connections*. 2020-2022. url: https://www.hexagon.de/wn13_e.htm.
- [4] HEXAGON software. *WN14 Software for calculation of PnC polygon profile shaft to collar joints*. 2020-2022. url: https://www.hexagon.de/wn14_e.htm.
- [5] Bäsel, U. „Application of complex-valued functions in plane differential geometry and kinematics“. In: arXiv:2310.10265v6 [math.DG] 5 Jul 2024 (2024). url: <https://arxiv.org/pdf/2310.10265>.
- [6] Schönwandt, U. „Neues Verfahren zum Drehen und Schleifen von Polygonprofilen“. In: *Zeitschrift für wirtschaftlichen Fabrikbetrieb* 84.8 (1989), S. 469–471.
- [7] Durrenbach, R. „Verbindungen von Welle und Nabe“. In: *Konstruktion* 6.10 (1954), S. 399–401.
- [8] Filemon, E. „Production and analysis of polygon profiles“. In: *Periodica Polytechnica (Mechanical Engineering)* 3.1 (1959), S. 81–112. url: <https://pp.bme.hu/me/article/view/6417/5522>.
- [9] DIN 3689-1. *Welle-Nabe-Verbindung – Hypotrochoidische H-Profile – Teil 1: Geometrie und Maße*. 2021-11.
- [10] Musyl, R. „Das K-Profil eine zyklische Kurve“. In: *Maschinenbau und Wärmewirtschaft* 1.3/4 (1946), S. 59–63.
- [11] Ziaei, M. „(2) Anpassungsfähige kontinuierliche Innen- und Außenkonturen für form- und reibschlüssige Verbindungen auf Basis der komplexen Zykloiden“. In: *VDI-Berichte* 2004 (2007), S. 277–294.

Auslegung eines monolithischen Positioniermechanismus als Fünfgelenk mit Abwurfmechanismus

Lucas Artmann*; Vitus Hensgens, Tim C. Lüth

* TU München, Lehrstuhl für Mikrotechnik und Medizingerätetechnik
lucas.artmann@tum.de

Kurzfassung

Vorge stellt wird ein monolithischer Mechanismus für die automatisierte Flüssigkeitshandhabung in der Zellzucht. Der Mechanismus nutzt Festkörpergelenke, um Abrieb und potenzielle Kontaminationsquellen zu eliminieren. Er ist als Fünfgelenk mit zwei Freiheitsgraden für die horizontale Positionierung der Pipettenspitze ausgelegt. Über einen weiteren Freiheitsgrad ist ein Wechsel zwischen Absaug- und Dosierpipette sowie das Abwerfen der Pipettenspitzen implementiert. Der gesamte Mechanismus ist so konstruiert, dass er keine über den Zellkulturen liegenden Motoren enthält, was das Risiko von Kontaminationen weiter reduziert. Der Mechanismus wird in einem Pipettierautomaten implementiert.

Design of a monolithic positioning mechanism as a 5-joint with ejection mechanism for pipette tips

A monolithic mechanism for automated liquid handling in cell culture is presented. The mechanism uses solid-state joints to eliminate abrasion and potential sources of contamination. It is designed as a five-bar linkage with two degrees of freedom for horizontal positioning of the pipette tip. A further degree of freedom is used to switch between the aspirating

and dispensing pipette and to eject the pipette tips. The entire mechanism is designed so that it does not contain any motors located above the cell cultures, which further reduces the risk of contamination. The mechanism is implemented in an automatic pipetting machine.

Eukaryotische Zellen benötigen für die erfolgreiche Kultivierung eine erhöhte Umgebungstemperatur, idealerweise 37°C [1, 2]. Hierfür werden Inkubatoren verwendet, Wärmeschränke die stabile Umgebungstemperatur schaffen. Oftmals mit zusätzlichen Funktionen wie Luftbefeuchtung und CO₂-Gehalt Steuerung [3, 4]. Die Zellen wachsen innerhalb von Gefäßen (Petrischalen, *wellplates* oder *flasks*), welche in der Regel verschlossen sind. In den Gefäßen befindet sich neben den Zellen noch Nährmedium welches die Nährstoffe für die Zellen zur Verfügung stellt. Die Nährstoffe werden von den Zellen verstoffwechselt und Abfallstoffe reichern sich im Medium an. Deshalb muss in bestimmten Intervallen das Medium gewechselt bzw. erneuert werden [4, 5].

Um das Medium zu wechseln, müssen die Gefäße geöffnet werden und sind somit anfällig für Verunreinigungen, deshalb werden die Gefäße für den Wechsel aus dem Inkubator herausgenommen und in eine Sterilwerkbank überführt [5]. Dort wird entweder manuell oder durch teilautomatisierte Pipettierautomaten das Medium gewechselt. Innerhalb der Sterilwerkbank herrscht eine laminare Luftströmung, um zu verhindern, dass Partikel aufgewirbelt werden und als Verunreinigungen in die Gefäße gelangen können [6]. Um jedoch die besten Umgebungsparameter für die Zellen zu gewährleisten ist ein Wechsel des Mediums innerhalb des Inkubators vorteilhaft, da es zu keinen Schwankungen der Umgebungsparameter kommt [1]. Neben dem Menschen sind aneinander bewegte Teile, bei denen Reibung zwischen unterschiedlichen Komponenten stattfindet, Erzeuger von Partikeln. Also klassische Lager und Gelenke [7].

3.1 Automatisierte Flüssigkeitshandhabung in Laboren

Es gibt hochentwickelte Roboter, DIY-Workstations mit Open-Source-Programmierung und Pipettierhilfsgeräte für die Flüssigkeitshandhabung. Erstere wie der „Microlab STARlet“ und der „Fluent 480“ sind modular erweiterbar, aber teuer (über 120.000 USD). DIY-Workstations, wie der EvoBot und der FINDUS Roboter, basieren auf dem Open-Source-Prinzip und erfordern mehr Zeit und Fachkenntnisse zur Anpassung. Pipettierhilfsgeräte wie der „Assist“ von Integra und der „epMotion® 96“

von Eppendorf automatisieren das Pipettieren, erfordert jedoch manuelle Spitzenaufnahme.



Abb. 1 (1) „Assist“ von INTEGRA Biosciences AG (INTEGRA Biosciences AG., 2021); (2) „MICROLAB STARlet“ mit „PCR Prep“ Konfiguration von Hamilton Company (Hamilton Company, 2022);

3.2 Positioniermechanismen

Gängige Anordnungen von Mechanismen sind kartesisch, zylindrisch, sphärisch, SCARA und anthropomorphisch. Letztere ähneln dem menschlichen Arm und werden auch Knickarmmanipulatoren genannt. Diese sind die häufigste Bauform und flexibel einsetzbar, haben jedoch Nachteile wie hohen Energieverbrauch und niedrige Steifigkeit. [9] Parallele Manipulatoren besitzen mehrere kinematische Ketten, die den Endeffektor mit der Basis verbinden, was zu höherer Steifigkeit und Genauigkeit führt. Sie sind komplexer in der Kinematik und haben einen kleineren Arbeitsraum. [10, 11]

3.3 Festkörpergelenke

Festkörpergelenke, auch nachgiebige Mechanismen genannt, sind Alternativen zu herkömmlichen Drehgelenken. Sie bestehen aus starren Segmenten, die durch biegeweiße Teile verbunden sind, und können entweder aus einem Bauteil (monolithisch) oder aus mehreren Teilen (hybrid) bestehen. Bei monolithischen Strukturen kann die Nachgiebigkeit entweder verteilt oder konzentriert auftreten, was ihre Struktur und Funktion beeinflusst. [7, 12] Festkörpergelenke werden durch ihre Freiheitsgrade und Form charakterisiert, wobei Bewegungsachsen mit geringerer Steifigkeit als Freiheitsgrade gelten. Das Pseudo Rigid Body Modell von

Howell vereinfacht diese Strukturen, indem es sie durch starre Segmente und Drehfedern darstellt, was die Berechnung großer Bewegungen und der benötigten Kräfte ermöglicht. Die Steifigkeit der Federn hängt von der Geometrie und dem E-Modul der Struktur ab. [13] Ein Vorteil nachgiebiger Mechanismen ist ihre Spiel- und Reibungsfreiheit, was sie für Reinraumumgebungen und präzise Positionieraufgaben geeignet macht. Zudem sind monolithische Mechanismen, die durch additive Fertigungsverfahren hergestellt werden, leichter zu miniaturisieren. [7]

3.4 Nachteile

Automatisierte Systeme müssen in Inkubatoren den Umgebungsbedingungen von 37 °C und 95% Luftfeuchtigkeit standhalten, was aktuelle Modelle wie der „Microlab® STARlet™“ und „Fluent 480“ nicht leisten können.

Roboter wie der Fluent 480 und Microlab® STARlet™, sind groß und reduzieren den nutzbaren Platz im Inkubator. Roboter wie der „Assist“ von Integra und „epMotion® 96“ von Eppendorf benötigen manuelles Eingreifen für Spitzenwechsel und Plattenhandhabung, was die Effizienz im Inkubator mindert.

Die Arbeitsräume monolithischer Positioniermechanismen sind begrenzt und oft aus Metall, fokussiert auf Nano- oder Mikropositionierung. Ihr Arbeitsbereich ist zu klein für standardisierte Multiwell-Platten, was ihre Anwendung in der Zellkultur einschränkt. Automatisierungslösungen müssen daher optimiert werden, um die Effizienz und Kapazität in der Zellkultur zu steigern. [12]

4 Eigener Ansatz

4.1 Erwarte Vorteile

Die Verwendung von Festkörpergelenken eliminiert den Abrieb, der in herkömmlichen Gelenken auftritt, und verringert somit das Risiko einer Kontamination der Zellen. Das automatisierte Abwerfen der Spitzen durch einen nachgiebigen Mechanismus bietet gegenüber dem manuellen Spitzenwechsel bisheriger Pipettierhilfen reduziert manuelle Eingriffe und minimiert das Risiko von Verunreinigungen. Die Abmessungen des Geräts sind speziell für diese Anwendung optimiert, was zu einer kompakteren Bauweise im Vergleich zu multifunktionalen Geräten führt. Durch die Verwendung von Kunststoffen wird das Gewicht im Vergleich

zu metallischen Konstruktionen erheblich reduziert, was die Portabilität und Handhabung des Geräts verbessert.

4.2 Lösungsstruktur

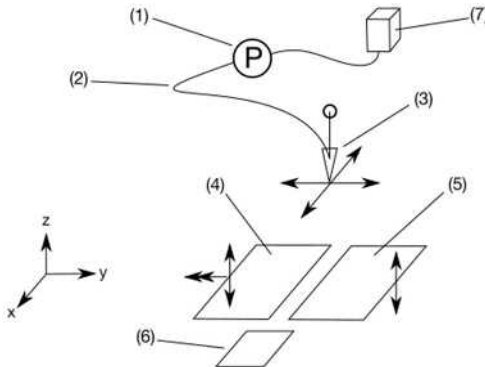


Abb. 2 Struktur des Apparats: (1) Pumpe; (2) Schlauch; (3) Spitze für die Aufnahme von Pipettenspitzen; (4) Zellzuchtgefäß; (5) Pipettenspitzenbox; (6) Abwuschale; (7) Flüssigkeitsbehälter

Die Struktur dieses Apparates ist in Abb. 2 dargestellt. Den zentralen Punkt des Apparats bildet eine Spitze (3), auf der Pipettenspitzen befestigt werden können. Diese Spitze wird für das Dosieren und Absaugen relativ zu dem Zellgefäß (4) und der Pipettenspitzenbox (5) positioniert. Für benutzte Pipettenspitzen, welche entsorgt werden sollen, ist eine Auffangschale (6) vorgesehen. Die Positionierung der Spitze in den horizontalen x- und y-Koordinaten wird durch einen monolithischen Mechanismus ermöglicht, welcher von Motoren angetrieben wird. Für die Veränderung des relativen vertikalen Abstands zwischen Spitze und Zellgefäß bzw. Pipettenspitzenbox sind Zellgefäß und Pipettenspitzenbox beweglich in z-Richtung. Für das Fördern der Flüssigkeiten sind Pumpen (1) verantwortlich. Die Pumpen werden durch Schläuche (2) mit der Spitze, welche die Pipettenspitzen aufnimmt, sowie den Flüssigkeitsbehältern (7) verbunden.

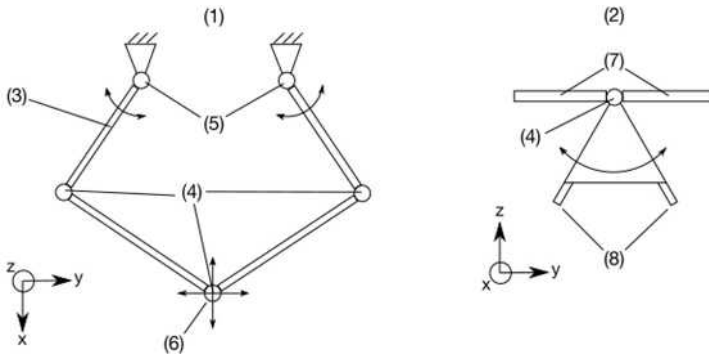


Abb. 3 Struktur des monolithischen Mechanismus: (1) Draufsicht des Fünfgelenks; (2) Frontalansicht der Spitzenaufnahme; (3) Starre Segmente; (4) Festkörpergelenke; (5) Servomotor; (6) Position der Spitzenaufnahme; (7) Starre Segmente des Fünfgelenks; (8) Befestigungspositionen für Pipettenspitzen

Der monolithische Mechanismus ist in Abb. 3 in zwei Ansichten dargestellt. Es wird die Draufsicht auf den Mechanismus gezeigt (1), welcher die Spitzenaufnahme (6) in x- und y-Koordinaten positioniert. Starre Segmente (3) sind durch Festkörpergelenke (4) miteinander verbunden. Für den Antrieb sorgen zwei Motoren, welche an Position (5) angebracht werden. Die Spitzenaufnahme an Position (6) ist in der Ansicht (2) in der Frontalansicht zu sehen. (7) zeigt starre Segmente des Fünfgelenks aus Ansicht (1). Pipettenspitzen können an zwei Positionen (8) befestigt werden. Eine Position dient der Dosierung, die andere dem Absaugen von Flüssigkeiten. Zur Aufnahme einer Spitze und zum Absaugen oder Dosieren von Flüssigkeiten muss die entsprechende Spitze lotrecht nach unten zeigen. Ein weiteres Festkörpergelenk (4) erlaubt diese Rotation.

5 Geräteaufbau

5.1 X,Y und Spitzenpositionierung

Die Länge der Kurbel beträgt 100 mm und die Länge der Koppel 150 mm. Der maximale Drehwinkel der Kurbeln beträgt aufgrund des Antriebs 180° . Die Form des daraus resultierenden Arbeitsraums (1) ist in Abb. 4 in blau dargestellt. Der Arbeitsraum des Fünfgelenks (5) deckt ein Zellgefäß (2), den Abwurfbehälter (4) sowie die Pipettenspitzenbox (3) ab.

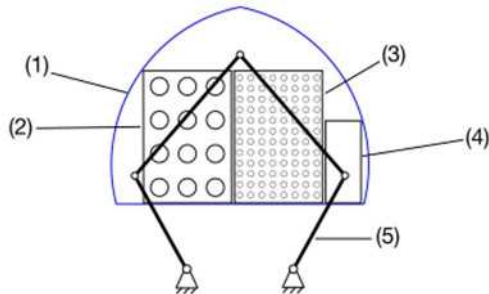


Abb. 4 Arbeitsraum des Fünfgelenks (1), Arbeitsraum (2), 12er Wellplatte als Beispiel für ein Zellgefäß (3), Pipettenspitzenbox (4), Abwurfbehälter (5), Fünfgelenk

Nachdem durch das Fünfgelenk der Punkt P (siehe Abb. 5) an gewünschten x- und y-Koordinaten positioniert wird, muss vertikal unterhalb dieses Punktes das Absaugen und Dosieren von Flüssigkeiten ermöglicht werden. Dafür muss der Anschluss der Pipettenspitze (4) vertikal unter dem Punkt P ausgerichtet werden. Es werden zwei Anschlüsse für Pipettenspitzen verwendet die durch Rotation variiert werden können. Der Drehpunkt dieser Rotation (3) liegt unterhalb des Punktes P, wodurch beide Anschlüsse eine vertikale Ausrichtung erreichen können. Die Umsetzung der Kinematik in einen Mechanismus mit Festkörpergelenken ist in Abb. 5 dargestellt. Die Spitzenaufnahme (5), welche die beiden Anschlüsse für Pipettenspitzen (4) besitzt, dreht sich in einem Festkörpergelenk (3) relativ zur Koppel b. Durch dieses Festkörpergelenk ist die Spitzenaufnahme mit der Koppel b verbunden, welche ebenfalls mit Punkt P verbunden ist. Die Koppel garantiert deshalb, dass sich der Drehpunkt (3) der Spitzenaufnahme stets vertikal unter Punkt P befindet.

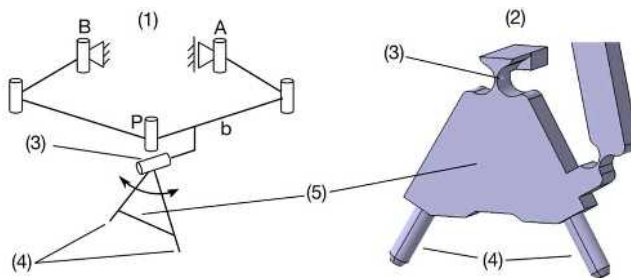


Abb. 5 Drehung der Anschlüsse für Pipettenspitzen; Die Kinematik ist auf der linken Seite (1) die konstruktive Umsetzung auf der rechten Seite (2) dargestellt;

Die Spitzenaufnahme (5) besitzt zwei Anschlüsse für Pipettenspitzen (4); Die Spitzenaufnahme kann sich relativ zur Koppel b im Drehpunkt (3) drehen

Für die gezielte Drehung der Spitzenaufnahme wird eine vertikale Bewegung des Punktes A durch die in Abb. 6 dargestellte Kinematik in eine Drehung der Spitzenaufnahme übersetzt. Die Koppel b aus Abb. 5 wird dafür in drei Segmente b₁, b₂ und b₃ unterteilt, welche durch Festkörpergelenke verbunden sind (siehe Abb. 6). Während das Segment b₃ in vertikaler Richtung nicht verschiebbar ist, kann das Segment b₁ simultan mit dem Lagerpunkt A vertikal verschoben werden. Sämtliche in Abb. 6 rot dargestellte Teile bewegen sich vertikal, sobald der Punkt A vertikal bewegt wird. Da die Koppel b₁ mit der Spitzenaufnahme verbunden ist, wird durch die vertikale Bewegung der Koppel b₁ eine Drehbewegung der Spitzenaufnahme erzeugt. Der benötigte Anschluss für die Pipettenspitzen kann folglich durch die Verschiebung des Punktes A in die Vertikale gebracht werden.

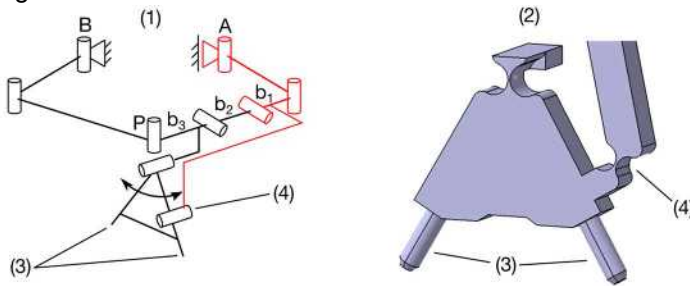


Abb. 6 Die Kinematik (1) zeigt die Unterteilung der Koppel in drei Segmente b₁, b₂ und b₃. Der benötigte Anschluss für Pipettenspitzen (3) kann in die Vertikale gebracht werden, wenn die rot dargestellten Teile über das Gelenk (4) den Spitzenhalter drehen. Die konstruktive Umsetzung ist in (2) dargestellt.

Die Spitzenaufnahme und ihre Verbindung zu Koppel b₃ ist in der Frontalansicht (2) gezeigt. Die Verbindung zwischen b₃ und Spitzenaufnahme besitzt weitere Festkörpergelenke.

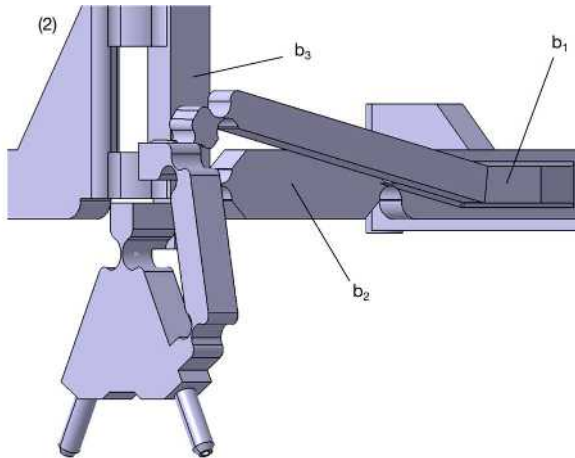


Abb. 7 Frontalansicht der Spitzenaufnahme inkl. Verbindung zur Koppel. Konstruktive Umsetzung der Verbindung von Koppel und Spitzenaufnahme für das Steuern der Drehung. Die Koppel ist in drei Segmente b_1 , b_2 und b_3 unterteilt, welche durch Festkörpergelenke verbunden sind. (1)

5.2 Abwurfmechanismus

Beim Abwurfmechanismus soll auf Motoren verzichtet werden, welche sich direkt oberhalb der Zellen befinden. Dafür wird auf bereits vorhandene Aktoren zurückgegriffen. Dabei wird die Drehbewegung der Spitzenaufnahme für den Abwurf der Spitze genutzt. Soll der Abwurf von Pipettenspitzen unabhängig von der aktuellen x- und y-Position möglich sein, so muss die Bewegung der Koppel b_1 in vertikaler Richtung in das Abwerfen der Pipettenspitze übersetzt werden.

Für das Abwerfen der Pipettenspitze wird eine Gabel verwendet, welche in Abb. 8 dargestellt ist. Die Gabel (2) ist mit der Spitzenaufnahme über ein Festkörpergelenk (1) verbunden. Um dieses Gelenk dreht sich die Gabel und bewegt sich dabei entlang des Anschlusses für die Pipettenspitzen. Befindet sich auf diesem Anschluss eine Pipettenspitze, so wird diese durch die Gabel herunter geschoben. Die Gabel berührt dabei jedoch lediglich die Pipettenspitze und nicht den Anschluss. So wird das Anhaften von Abrieb am Anschluss vermieden.

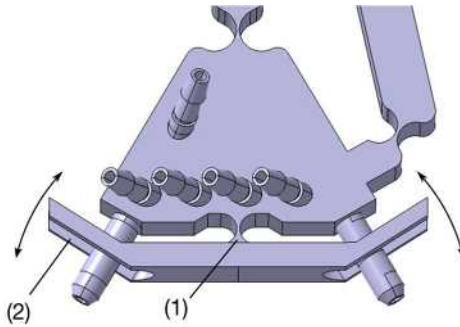


Abb. 8 Gabel für den Abwurf von Pipettenspitzen; (1) Drehpunkt der Gabel; (2) Gabel, welche die Pipettenspitze abwirft

Die Kinematik des Spitzenabwurfs ist so gestaltet, dass die Gabel genug Platz für eine Pipettenspitze lässt, wenn der entsprechende Anschluss vertikal nach unten ausgerichtet ist. Wird dieser Anschluss jedoch aus der Vertikalen herausgedreht, weil der andere Anschluss in die Vertikale gebracht werden soll, so bewegt sich die Gabel entlang des Anschlusses und wirft die Pipettenspitze ab.

Die Realisierung ist in Abb. 9 dargestellt. Durch das Hinzufügen eines weiteren Lagerpunkts A, welcher relativ zum Drehpunkt der Spitzenaufnahme B unbeweglich ist, wird ein Viereck gebildet. Die vier Gelenke sind mit A, B, C und D bezeichnet. Die Kurbel dieses Vierecks bildet die Spitzenaufnahme, welche die Punkte B und C verbindet. Die Gabel, welche die Pipettenspitzen abwirft, ist die Koppel. Komplettiert wird das Viereck durch die Schwinge (3). Die Kurbel dieses Getriebes wird durch die Rotation der Spitzenaufnahme bewegt. Es ist zwangsläufig und gibt für jede Auslenkung der Spitzenaufnahme eine definierte Position der Gabel vor. Je nach Rotationsrichtung der Spitzenaufnahme wird die Schwinge (3) auf Druck oder Zug belastet. Wird die Schwinge durch Druck belastet, kann es zum Knicken der Schwinge kommen. Eine geknickte Schwinge ist nicht in der Lage, den benötigten Druck für das Abwerfen der Pipettenspitze zu liefern. Als Lösung dieses Problems setzt dieser Mechanismus auf ein weiteres Viereck durch das Hinzufügen einer weiteren Schwinge (4).

Die konstruktive Umsetzung der Kinematik (1) ist auf der rechten Seite (2) der Abb. 9 dargestellt. Die Gelenkpunkte A und B werden in der Kon-

struktur nicht umgesetzt. Stattdessen ist die gesamte Schwinge nachgiebig ausgeführt und verformt sich sehr leicht unter Druckbelastung. Wird die Schwinge jedoch auf Zug belastet, gibt sie nur sehr wenig nach und bewegt dadurch die Gabel. Durch die langen, nachgiebigen Abschnitte bleibt das Moment gering, welches der Drehung der Spitzenaufnahme entgegenwirkt. Gleichzeitig kann die auf Zug belastete Seite die benötigten Kräfte liefern, welche für das Abstoßen der Pipettenspitzen benötigt werden. Das Gelenk C, welches Gabel und Spitzenaufnahme verbindet, ist analog zu den Gelenken des Fünfgelenks geteilt ausgeführt. Das geteilte Gelenk ist in der isometrischen Ansicht in Abb. 10 mit (1) gekennzeichnet.

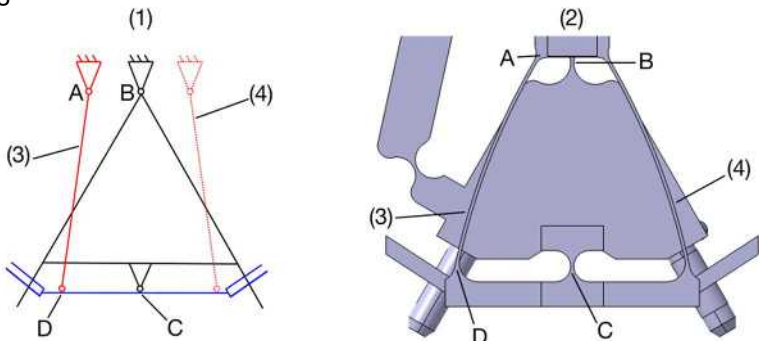


Abb. 9 Viergelenk zur Bewegung des Spitzenabwurfmechanismus (1), Kinematik des Viergelenks (2), Konstruktive Umsetzung (3), Schwinge des Viergelenks (4), zusätzliche Schwinge des parallelen Viergelenks A, B, C, D Gelenke des Viergelenks

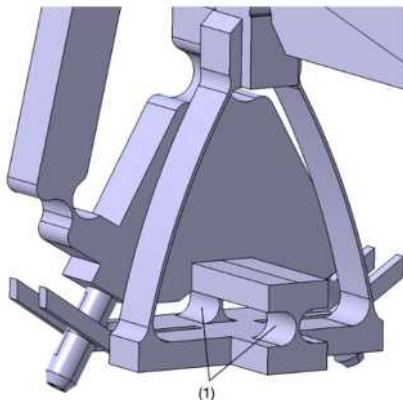


Abb. 10 Isometrische Ansicht des Spitzenabwurfmechanismus; (1) geteiltes Gelenk der Gabel

6 Experiment

Es wurde qualitativ untersucht, ob der entwickelte Mechanismus an vorgegebenen x und y Koordinaten Pipettenspitzen aufnehmen kann und diese im Anschluss wieder abwerfen kann. Beides wurde visuell überprüft.

6.1 Versuchsaufbau

Es wurden die x - und y -Koordinaten einer Pipettenspitze in der Pipettenspitzenbox als Startwert vorgegeben. Die Pipettenspitzen sind innerhalb der Box matrixförmig verteilt. Zeilen und Spalten haben jeweils 10 mm Abstand. Die Zielkoordinaten werden entsprechend dieser Abstände gewählt. 8 Pipettenspitzen wurden in einer Zeile hintereinander angefahren. Im Anschluss wurden, beginnend mit dem Startwert, 10 Pipettenspitzen in einer Spalte angefahren. Überprüft wurde, ob die Pipettenspitze aufgenommen wurde und im Anschluss abgeworfen werden konnte. Die Pipettenspitzenbox mit 5 ml Pipettenspitzen wurde auf den dafür vorgesehenen Hubtisch des Apparats positioniert, sodass sie in der Vertiefung der Ablagefläche steht. Der Deckel wurde aufgeklappt und von der Box entfernt.

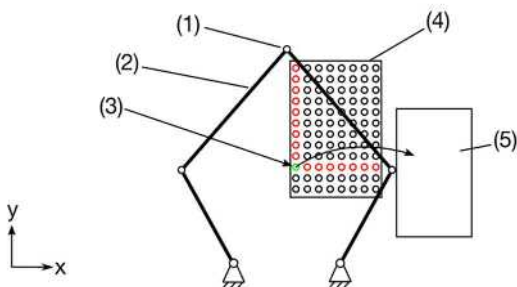


Abb. 11 Zielpositionen des Mechanismus. (1) Spitzenaufnahme; (2) Fünfgelenk des monolithischen Mechanismus; (3) Startposition zur Aufnahme der Spitzen (grün), danach der Zeile und Spalte folgend die nächsten Zielpositionen (rot); (4) Pipettenspitzenbox; (5) Abwurfbehälter

Der Mechanismus wurde durch die Software in den Zustand gefahren, in der die Spitzenaufnahme mit dem Anschluss für das Absaugen senkrecht nach unten zeigt. Das Übereinstimmen der Position der Spitzenaufnahme und der Startposition wurde überprüft. Insgesamt wurden die

Zeilen und Spalten achtmal abgefahren. Es wurde registriert, ob die Pipettenspitze im Pipettenspitzenblock von der Spitzenaufnahme getroffen wurde und ob sie im Anschluss im Abwurfbehälter abgeworfen wurde.

6.2 Ableitung der Ergebnisse aus den Messwerten

Für das Abfahren der Zeile wurden zwei Quotienten gebildet, welche das erfolgreiche Treffen der Spitze und das erfolgreiche Aufnehmen der Pipettenspitzen beschreiben. Dafür wurde ein Quotient aus den positiven Ergebnissen und der Gesamtanzahl der Messungen berechnet. Für das Abfahren der Spalte wurden diese zwei Quotienten ebenfalls berechnet.

6.3 Zusammenfassung der Ergebnisse

Der Mechanismus konnte 97 %, der in einer Zeile befindlichen Pipettenspitzen, mit der Spitzenaufnahme treffen. 84 % der Spitzen wurden erfolgreich aufgenommen und im Anschluss vom Mechanismus abgeworfen.

In einer Spalte wurden 98% der Pipettenspitzen von der Spitzenaufnahme getroffen. Für ein erfolgreiches Anfahren der y-Koordinate mussten Werte vorgegeben werden, welche von der tatsächlichen Position abweichen. Für Bewegungen der Spitzenaufnahme in y-Richtung muss der Mechanismus seine Festkörpergelenke stärker Verformen als für eine Bewegung in x-Richtung. Große Auslenkungen der Festkörpergelenke führen zu Abweichungen von der angenommenen Kinematik. Der Spitzenabwurfmechanismus wirft 100 % der Spitzen ab, welche von der Spitzenaufnahme aufgenommen werden.

Zusammenfassend lässt sich sagen, dass der entwickelte Mechanismus mithilfe der vorgenommenen Optimierung der Steuerung, vorgegebene Zielpositionen anfahren kann. Aufgenommene Pipettenspitzen werden zuverlässig abgeworfen.

Literatur

- [1] B. Alberts, *Lehrbuch der Molekularen Zellbiologie*. [Place of publication not identified]: [John Wiley and Sons], 2014.
- [2] A. Berk *et al.*, *Molekulare Zellbiologie*. De Gruyter, 1996.
- [3] S. Schmitz und C. Desel, *Der Experimentator Zellbiologie*, 1. Aufl. Berlin, Heidelberg: Springer Berlin Heidelberg, 2018. [Online]. Verfügbar unter: <http://nbn-resolving.org/urn:nbn:de:bsz:31-e-pflicht-1586572>

- [4] D. Boujard, B. Anselme, C. Cullin und C. Raguénès-Nicol, Hg., *Zell- und Molekularbiologie im Überblick*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014.
- [5] S. Schmitz, *Der Experimentator: Zellkultur*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2020.
- [6] G. Gstraunthaler und T. Lindl, *Zell- und Gewebekultur: Allgemeine Grundlagen und spezielle Anwendungen*, 7. Aufl. Berlin, Heidelberg, s.l.: Springer Berlin Heidelberg, 2013. [Online]. Verfügbar unter: <http://dx.doi.org/10.1007/978-3-642-35997-2>
- [7] A. Raatz, *Stoffschlüssige Gelenke aus pseudo-elastischen Formgedächtnislegierungen in Pararellrobotern*. Zugl.: Braunschweig, Techn. Univ., Diss., 2006. Essen: Vulkan-Verl., 2006.
- [8] H. Tegally, J. E. San, J. Giandhari und T. de Oliveira, "Unlocking the efficiency of genomics laboratories with robotic liquid-handling" (eng), *BMC genomics*, Jg. 21, Nr. 1, S. 729, 2020, doi: 10.1186/s12864-020-07137-1.
- [9] J. Mareczek, *Modellbildung von Kinematik und Dynamik*. Berlin, Heidelberg: Springer Vieweg, 2020.
- [10] B. Siciliano, *Robotics: Modelling, Planning and Control*. London: Springer London, 2009.
- [11] J. Baillieul und T. Samad, Hg., *Encyclopedia of systems and control*, 2. Aufl. Cham: Springer, 2021]-.
- [12] S. Linß, *Ein Beitrag zur geometrischen Gestaltung und Optimierung prismatischer Festkörpergelenke in nachgiebigen Koppelmechanismen*. Zugl.: Ilmenau, Techn. Univ., Diss., 2015. Ilmenau: Univ.-Verl. Ilmenau, 2015.
- [13] L. L. Howell, "Compliant Mechanisms" in *Encyclopedia of Nanotechnology*, B. Bhushan, Hg., Dordrecht: Springer Netherlands, 2016, S. 604–611, doi: 10.1007/978-94-017-9780-1_302.

Entwurf von topologieoptimierten nachgiebigen Beinen für vierbeinige Roboter

Yilun Sun*, Tim C. Lüth**

TU München, Lehrstuhl für Mikrotechnik und Medizingerätetechnik
*yilun.sun@tum.de, **tim.lueth@tum.de

Kurzfassung

Roboterbeine sind eine wichtige Komponente des vierbeinigen Roboters, um verschiedene Bewegungsabläufe zu ermöglichen. Obwohl die konventionellen, auf starren Gelenken basierenden Beine im Allgemeinen robuste Bewegungen ausführen können, haben sie immer noch das Problem der schlechten Abdichtung, wenn sie in komplexem und flüssigem Terrain eingesetzt werden. Um dieses Problem zu lösen, wurden in den letzten Jahren vollständig nachgiebige Beine mit monolithischer Struktur eingeführt, um die Kompaktheit des Systems und die strukturelle Nachgiebigkeit von vierbeinigen Robotern zu verbessern. In dieser Arbeit stellen wir eine auf Topologie-Optimierung basierende Methode vor, um ein effizientes Design von nachgiebigen Roboterbeinen zu erreichen. Um die strukturelle Steifigkeit und die Biegeflexibilität des realisierten Beins auszugleichen, wird ein multi-objektiver Optimierungsalgorithmus verwendet. Eine Reihe von Entwurfsfällen wird vorgestellt, um das Entwurfsprinzip und das analytische Verfahren der beschriebenen Methode zu veranschaulichen. Die Ergebnisse haben gezeigt, dass ein vierbeiniger Roboter mit den optimierten Beinen erfolgreich stabile und kontinuierliche geradlinige Gehbewegungen ausführen kann.

Title / Abstract

Design of topology-optimized compliant legs for quadruped robots

Robotic legs are an important component of the quadruped robot for achieving different motion gaits. Although the conventional rigid-link-based legs can generally perform robust motions, they still have the issues with poor sealing when operating in complex and liquid terrains. To cope with this problem, fully compliant legs with monolithic structure have been introduced in recent years to improve the system compactness and structural compliance of quadruped robots. In this work, we present a topology-optimization-based method to achieve efficient design of compliant robotic legs. In order to balance the structural stiffness and bending flexibility of the realized leg, a multi-objective optimization algorithm is utilized. A series of design cases are presented to illustrate the design principle and analytical procedure of the proposed method. In addition, experimental evaluation is also performed, and the results have demonstrated that, a quadruped robot with the optimized legs can successfully achieve stable and continuous straight-line walking motions.

Literatur

- [1] Sun, Y., Zong, C., Pancheri, F., Chen, T., & Lueth, T.C. "Design of topology optimized compliant legs for bio-inspired quadruped robots". *Scientific Reports* 13, 4875 (2023). <https://doi.org/10.1038/s41598-023-32106-5>

Validierung einer energieeffizienten Antriebsmethode von Kurvengetrieben an einem Prüfstand

Thomas Knobloch, Lukas Kaboth, Rufat Asgarzade,
Mathias Hüsing, Burkhard Corves

RWTH Aachen University, Institut für Getriebetechnik, Maschinendynamik und
Robotik
knobloch@igmr.rwth-aachen.de

Kurzfassung

Vor dem Hintergrund des Klimawandels, sich wandelnder Gesetzgebung und gestiegenen Ansprüchen an die Energieeffizienz ist die Industrie stetig auf der Suche nach Potenzialen zur Kostenminimierung. Nicht nur in der Verpackungsindustrie, sondern auch in den Bereichen der Handhabungs- und Umformtechnik werden zum Großteil ungleichmäßig übersetzende Getriebe zur Erzeugung bestimmter Bewegungsabläufe genutzt. Schon seit mehreren Jahrzehnten gibt es Ansätze, derartige Getriebe energieeffizienter zu betreiben, wobei die sogenannte Eigenbewegung vielversprechend ist. Die praktische Umsetzung in der Industrie ist mit der Entwicklung von Servomotoren möglich geworden, da sie die Eigenbewegung mit hoher Genauigkeit abbilden können. Die Kombination aus ungleichmäßig übersetzenden Getrieben und einer variablen Antriebsgeschwindigkeit ist seitdem Gegenstand der Forschung. Dieser Beitrag präsentiert den Vergleich und die Diskussion der Messergebnisse eines Kurvengetriebes, das einmal konstant und zum anderen servogesteuert angetrieben wird, so dass damit das Kurvengetriebe effizienter betrieben wird.

Title / Abstract

Validation of an Energy-efficient Drive Method for Cam Mechanisms on a Test Bench

Due to the climate change, changing legislation and increased demands on energy efficiency, the industry is constantly looking for ways to minimise costs. Not only in the packaging industry, but also in the fields of handling and forming technology, non-uniformly translating mechanisms are used to generate certain motions. For several decades now, there have been attempts to operate such mechanisms in a more energy-efficient manner, with so-called Eigenmotion being a promising approach. Practical implementation in industry has become possible with the development of servo motors, as they can reproduce the Eigenmotion with high precision. The combination of non-uniformly translating mechanisms and a variable drive speed has been the subject of research ever since. This article presents a comparison and discussion of the measurement results of a cam mechanism driven at a constant angular velocity and an energy-efficient cam mechanism.

1 Einführung

Um den Ausstoß von Treibhausgasen zu reduzieren, soll dieser durch verschiedene Maßnahmen begrenzt werden. Einerseits soll die Stromerzeugung aus erneuerbaren Energien gefördert werden. Andererseits soll der Verbrauch durch eine erhöhte Energieeffizienz vermindert werden. So soll nach §4 Abs. 2 des Energieeffizienzgesetzes der Endenergieverbrauch bis 2045 um 45 % gegenüber 2008 gesenkt werden [1]. Ein wesentlicher Anteil des Energieverbrauchs entfällt auf die Bereitstellung von mechanischer Energie. Hierfür wird ungefähr ein Viertel des deutschen industriellen Endenergieverbrauchs benötigt [2]. Hohen Bedarf an mechanischer Energie haben insbesondere die Fahrzeug-, Textil- und Handhabungstechnik sowie der Bereich der Verpackungsmaschinen. Argumente für den Einsatz von ungleichmäßig übersetzenden Getrieben in diesen Bereichen sind vor allem mechanische Robustheit, Zuverlässigkeit und Wirtschaftlichkeit [3]. In einigen industriellen, hochdynamischen Anwendungen sind während

des Arbeitsprozesses in einer Maschine kurze Stillstände des Abtriebsgliedes erforderlich, sogenannte Rasten. Dafür werden häufig Kurvengetriebe verwendet, da diese im Gegensatz zu Koppelgetrieben eine exakte Rast erzeugen können. Gegenstand dieses Forschungsprojekts ist die Anwendung von Maßnahmen zur Steigerung der Energieeffizienz auf Rastgetriebe mit Kurvenscheiben, insbesondere die Entwicklung von Synthesemethoden und die Untersuchung des Eigenbewegungsbetriebs. Dafür wurde am Institut für Getriebetechnik, Maschinendynamik und Robotik (IGMR) ein Prüfstand konstruiert. Das CAD-Modell des Versuchsgetriebes ist in Abb. 1 abgebildet.

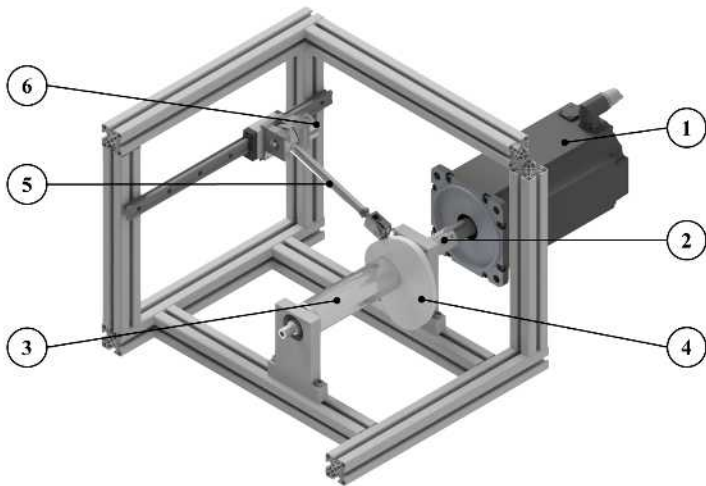


Abb. 1: Dreigliedriges Kurvengetriebe im CAD-Modell

Die Antriebswelle (3) ist über eine Schalenkupplung (2) mit einem Servomotor (1) verbunden. Auf der Welle wird die Kurvenscheibe (4) montiert. Der Rollenhebel (5) wird am Gestell befestigt. Zur Messung des Winkelverlaufs wird ein Drehgeber an der Drehachse des Rollenhebels befestigt. Als Zwanglaufsicherung des Rollenhebels kommt eine Feder zum Einsatz, die im CAD-Modell nicht dargestellt ist.

2 Eigenbewegung

Die theoretischen Grundlagen der Eigenbewegung sind seit vielen Jahren bekannt [4]. Allerdings ist es erst seit der Entwicklung von Servomotoren möglich, die nicht konstante Geschwindigkeit der Eigenbewegung zu applizieren. Die Eigenbewegung eines Mechanismus beschreibt eine Bewegung, die die Einhaltung eines konstanten Energieniveaus voraussetzt. Bei einem festgelegten Startwinkel φ_0 ergibt sich:

$$E_{ges} = E_{kin}(\varphi) + E_{pot}(\varphi) = E_{kin}(\varphi_0) + E_{pot}(\varphi_0) = konst. \quad (1)$$

Hierbei bezeichnen E_{kin} die kinetische und E_{pot} die potentielle Energie. Die kinetische Energie setzt sich zusammen als:

$$E_{kin}(\varphi) = \frac{1}{2} J_{red}(\varphi) \cdot \dot{\varphi}^2 \quad (2)$$

J_{red} bezeichnet das auf die Antriebswelle reduzierte Massenträgheitsmoment. Daraus resultiert:

$$\frac{1}{2} J_{red}(\varphi) \cdot \dot{\varphi}^2 + E_{pot}(\varphi) = \frac{1}{2} J_{red}(\varphi_0) \cdot \dot{\varphi}_0^2 + E_{pot}(\varphi_0) \quad (3)$$

Umformen ergibt:

$$\dot{\varphi}_e(\varphi) = \sqrt{\frac{\frac{1}{2} J_{red}(\varphi_0) \cdot \dot{\varphi}_0^2 + E_{pot}(\varphi_0) - E_{pot}(\varphi)}{\frac{1}{2} J_{red}(\varphi)}} \quad (4)$$

Die reale Eigenbewegung ist zusätzlich durch ihre Abhängigkeit von Reibungsverlusten und Nutzmomenten gekennzeichnet. Je näher ein Mechanismus im Bereich der Eigenbewegung betrieben wird, desto weniger Energie wird bei dessen Betrieb verbraucht [4].

Zwar gehören die Grundlagen der Eigenbewegung längst dem Stand der Technik an, dennoch kommt die Anwendung der Eigenbewegung bisher nur in wenigen Fällen zum Einsatz. Die Methoden zum Auslegen von energieeffizienten Getrieben sind noch relativ neu und noch immer Gegenstand der Forschung.

Am IGMR wurden bereits einige Forschungen zur Eigenbewegung von Koppelgetrieben und Kurvengetrieben durchgeführt und veröffentlicht [5–8].

3 Entwicklung des Getriebes

In Abb. 2 ist eine Skizze des Getriebes dargestellt.

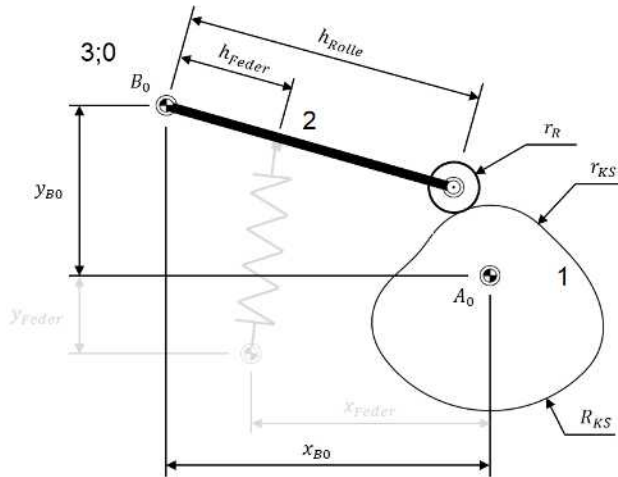


Abb. 2: Skizze des Prüfstands mit kinematisch relevanten Größen

Der Rollenhebel des Kurvengetriebes soll eine Rast-in-Rast-Bewegung ausführen, wie sie in Abb. 3 dargestellt ist.

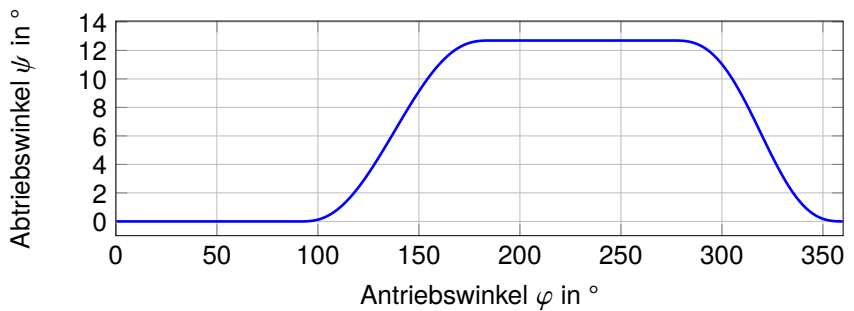


Abb. 3: Übertragungsfunktion

Zwischen den Rasten wird jeweils ein Polynom fünften Grades eingesetzt. Das Getriebe wird anschließend mit der institutseigenen Software *Me-*

chanism Developer [9] synthetisiert und dimensioniert. In Tab. 1 sind die geometrischen Größen des Getriebes angegeben.

Tab. 1: Geometrische Größen des Getriebes

Größe	Wert in mm
x_{B0}	209,0
y_{B0}	122,2
h_{Rolle}	199,5
h_{Feder}	72,5
r_r	13,0
r_{KS}	47,0
R_{KS}	86,9

Die am Institut gefertigte Kurvenscheibe ist in Abb. 4 dargestellt.

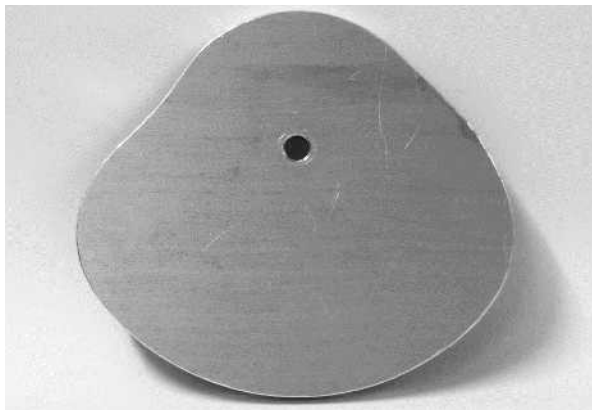


Abb. 4: Kurvenscheibe aus Aluminium

Der aufgebaute Prüfstand ist in Abb. 5 abgebildet. Als Zwanglaufsicherung kommt eine Feder mit einer Federsteifigkeit von $275,6 \text{ N/m}$ zum Einsatz.



Abb. 5: Dreigliedriges Kurvengetriebe am Prüfstand

4 Modell

Um die Eigenbewegung simulativ zu berechnen, wird ein Mehrkörpermodell des Getriebes in der Software *Matlab/Simulink* [10] erstellt. Die Eigenbewegung kann anschließend direkt aus dem Modell berechnet werden. In Abb. 6 sind der simulierte Verlauf des Antriebswinkels φ_e und die simulierte Antriebswinkelgeschwindigkeit $\dot{\varphi}_e$ des Getriebes in Abhängigkeit von der Zeit t in Eigenbewegung abgebildet.

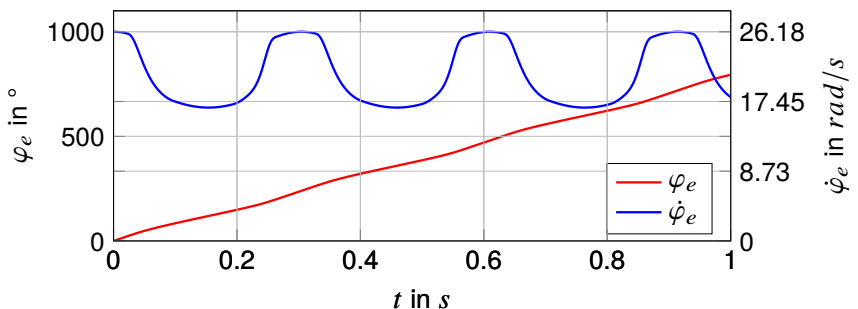


Abb. 6: Simulierte Eigenbewegung des Getriebes bei einer mittleren Drehzahl von $196,75 \text{ min}^{-1}$. Rot: Antriebswinkel φ_e . Blau: Antriebswinkelgeschwindigkeit $\dot{\varphi}_e$

5 Versuche

Zur Verifizierung der höheren Energieeffizienz des Kurvengetriebes in Eigenbewegung im Vergleich zu einer konstanten Antriebswinkelgeschwindigkeit sollen Antriebsleistung und Drehmoment des Servomotors aufgezeichnet werden. Zunächst wird das Kurvengetriebe mit einer konstanten Winkelgeschwindigkeit von $196,75 \text{ min}^{-1}$ angetrieben. Anschließend wird über das *Simulink*-Modell eine Eigenbewegung bei einer mittleren Drehzahl von $196,75 \text{ min}^{-1}$ berechnet, die dann am Servomotor appliziert wird. Durch die Identität der Werte von konstanter Winkelgeschwindigkeit und mittlerer Winkelgeschwindigkeit der Eigenbewegung wird sichergestellt, dass die Periodendauer für eine Umdrehung der Kurvenscheibe in beiden Fällen unverändert ist. In Abb. 7 sind die gemessenen Drehmomentverläufe bei Betrieb des Getriebes mit einer konstanten Antriebswinkelgeschwindigkeit von $\varphi = 196,75 \text{ min}^{-1}$ (rot) sowie bei Betrieb des Getriebes in Eigenbewegung bei einer mittleren Winkelgeschwindigkeit von $\varphi_{e,m} = 196,75 \text{ min}^{-1}$ (blau) dargestellt.

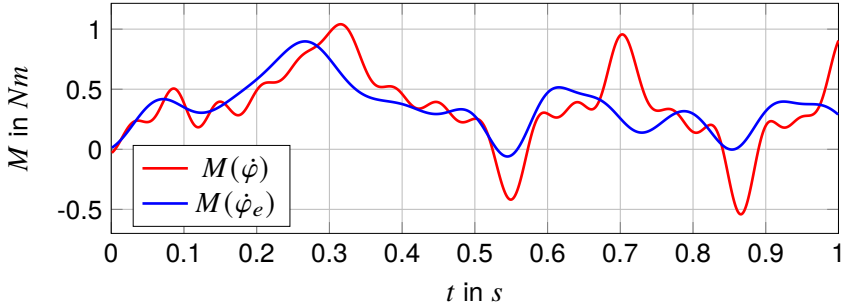


Abb. 7: Gemessenes Antriebsmoment M bei Betrieb des Getriebes mit konstanter Antriebswinkelgeschwindigkeit $\dot{\varphi}_{konst}$ (rot) und in Eigenbewegung $\dot{\varphi}_e$ (blau) bei mittleren Drehzahlen von jeweils $196,75 \text{ min}^{-1}$

Es zeigt sich, dass bereits im Anlaufvorgang eine Reduktion des maximalen Antriebsmoments erwirkt werden kann. Im stationären Betrieb in Eigenbewegung kann das maximale Antriebsmoment um 73 % gesenkt werden. In Abb. 8 sind die aufgenommenen Leistungen bei Betrieb des Getriebes mit konstanter Antriebswinkelgeschwindigkeit und in Eigenbewegung

abgebildet.

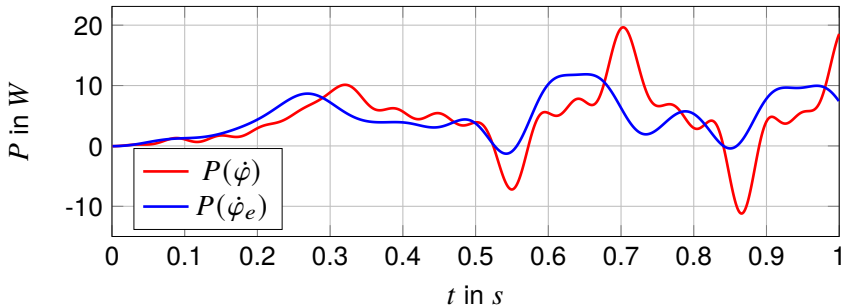


Abb. 8: Gemessene Leistung P bei Betrieb des Getriebes mit konstanter Antriebswinkelgeschwindigkeit $\dot{\varphi}_{konst}$ (rot) und in Eigenbewegung $\dot{\varphi}_e$ (blau) bei mittleren Drehzahlen von jeweils $196,75 \text{ min}^{-1}$

Hierbei kann im Eigenbewegungsbetrieb neben den um 49 % geringeren Leistungsspitzen eine Reduktion der Verlustleistung um bis zu 94,7 % gegenüber dem Betrieb mit konstanter Antriebswinkelgeschwindigkeit nachgewiesen werden. Die prozentual eingesparte Leistung im stationären Eigenbewegungsbetrieb im Vergleich zum Betrieb mit konstanter Antriebswinkelgeschwindigkeit liegt bei 14 %. In Abb. 9 sind die gemessenen Winkelverläufe des Rollenhebels bei Betrieb des Getriebes mit konstanter Antriebswinkelgeschwindigkeit (rot) sowie bei Betrieb des Getriebes in Eigenbewegung (blau) aufgezeichnet.

Durch die unterschiedlichen Antriebswinkelgeschwindigkeiten ergeben sich bei gleicher Kontur der Kurvenscheibe Unterschiede im Winkelverlauf des Abtriebsgliedes. Durch eine anschließende Optimierung des Getriebes, insbesondere der Kontur der Kurvenscheibe, lassen sich diese Unterschiede minimieren.

6 Zusammenfassung und Ausblick

In diesem Beitrag konnte gezeigt werden, dass die Anwendung der Eigenbewegung auf ein dreigliedriges Kurvengetriebe zu einem reduzierten Energieverbrauch, geringeren Leistungsspitzen und zu einem verringerten maximalen Antriebsmoment führt. In einem nächsten Schritt soll die Kontur

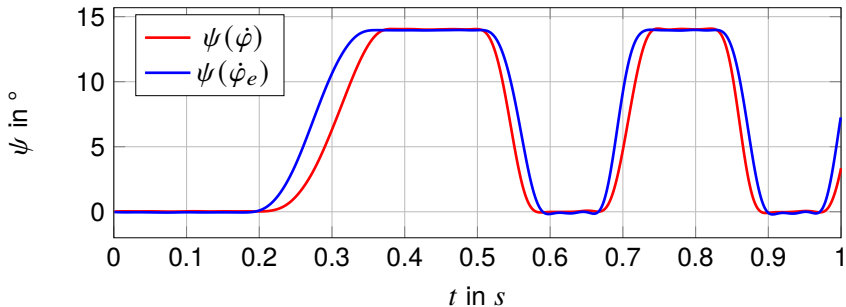


Abb. 9: Gemessener Verlauf des Abtriebswinkels ψ bei Betrieb des Getriebes mit konstanter Antriebswinkelgeschwindigkeit $\dot{\varphi}_{konst}$ (rot) und in Eigenbewegung $\dot{\varphi}_e$ (blau) bei mittleren Drehzahlen von jeweils $196,75 \text{ min}^{-1}$

der Kurvenscheibe derart verändert werden, um in Betrieb des Getriebes in Eigenbewegung weiterhin den geforderten Verlauf des Abtriebswinkels zu erzielen.

Literatur

- [1] *Energieeffizienzgesetz*. Bundesamt für Justiz. url: <https://www.gesetze-im-internet.de/eneffg/BJNR1350B0023.html> (besucht am 25.04.2024).
- [2] Umweltbundesamt. *Endenergieverbrauch nach Energieträgern und Sektoren*. 2015. url: <https://www.umweltbundesamt.de/daten/energie/energieverbrauch-nach-energietraegern-sektoren> (besucht am 25.04.2024).
- [3] Kerle, H., Corves, B. und Hüsing, M. *Getriebetechnik: Grundlagen, Entwicklung und Anwendung ungleichmäßig übersetzender Getriebe*. Springer Fachmedien Wiesbaden, 2015. isbn: 9783658100575. doi: 10.1007/978-3-658-10057-5.
- [4] Dresig, H. und Vul'fson, I. I. *Dynamik der Mechanismen*. Springer Vienna, 1989. isbn: 9783709190357. doi: 10.1007/978-3-7091-9035-7.

- [5] Schwarzfischer, F., Hüsing, M. und Corves, B. „The Dynamic Synthesis of an Energy-Efficient Watt-II-Mechanism“. In: *Multibody Mechatronic Systems*. Springer International Publishing, Sep. 2017, S. 213–222. doi: 10.1007/978-3-319-67567-1_20.
- [6] Schwarzfischer, F. „Die Eigenbewegung der Schubkurbel“. In: *IFToMM D-A-CH Konferenz / Dritte IFToMM D-A-CH Konferenz 2017* (2017). doi: 10.17185/DUEPUBLICO/43389.
- [7] Schwarzfischer, F., Hüsing, M. und Corves, B. „Dynamic synthesis of energy-efficient mechanisms“. In: *International Journal of Mechanisms and Robotic Systems* 4.4 (2018), S. 401. doi: 10.1504/ijmrs.2018.096301.
- [8] Knobloch, T., Hüsing, M. und Corves, B. „Effiziente Erzeugung von Rasten hochdynamischer Getriebe in der Produktentwicklung“. In: *IFToMM D-A-CH Konferenz vol. 2023, Neunte IFToMM D-A-CH Konferenz 2023: 16./17. März 2023* (2023), Universität Basel. doi: 10.17185/DUEPUBLICO/77387.
- [9] Müller, M. et al. „MechDev—A New Software for Developing Planar Mechanisms“. In: *Mechanisms and Machine Science*. Springer International Publishing, 2019, S. 167–175. isbn: 9783030164232. doi: 10.1007/978-3-030-16423-2_15.
- [10] MathWorks. *MATLAB SIMULINK*. url: <https://de.mathworks.com/products/simulink.html> (besucht am 25. 04. 2024).

VR-gestützte interaktive Ansteuerung einer Montagezelle mit Delta-Endeffektor

Akhilraj Anilkumar*; Claas Duarte Nanninga*; Mike Männel*;
Maik Berger*

*TU Chemnitz, Professur Montage- und Handhabungstechnik
akhilraj.chamakalayil-anilkumar@mb.tu-chemnitz.de
claas-immanuel.duarte-nanninga@mb.tu-chemnitz.de
mike.maennel@mb.tu-chemnitz.de
maik.berger@mb.tu-chemnitz.de

Kurzfassung

Intuitive kopf- oder handbewegungsbasierte Steuerungen erhöhen die Flexibilität und Anpassungsfähigkeit teilautomatisierter Vorrichtungen, indem sie die Mensch-Maschine-Interaktion verbessern. Der vorliegende Beitrag behandelt die Ansteuerung eines servomotorisch angetriebenen kartesischen Portals mit drei Linearachsen und einer Drehachse mittels Bewegungserfassung. An der vierten Achse des Portals ist ein Delta-Roboter angeflanscht, der sich auch mit Handbewegungen steuern lässt. Über eine Tiefenkamera oder einen VR-Controller wird die Handposition erfasst, während das VR-Headset die Kopfposition verfolgt, welche das Motion-Control-System in eine Sollposition des Endeffektors umrechnet.

Title / Abstract

Use of Depth Cameras for Hand Intuitive Control of Linear Axes in Assembly Cells

Intuitive hand or head motion based controls enhance the flexibility and adaptability of partially automated assembly processes by improving human-machine interaction. This paper discusses the control of an assembly cell with three linear axes and one rotary axis, all driven by servo

motors, using motion tracking. A delta robot is attached to the fourth axis of the cell. A depth camera or VR controller tracks the hand position, and VR headset tracks the head position to operate the end effector via a programmable logic controller (PLC).

1 Einführung

Die Programmierung herkömmlicher Bewegungssteuerungen erfordert ein nicht unbeträchtliches Grundwissen und Erfahrung bei Inbetriebnahme und Modifikation. Möchte man Bewegungen vorgeben, sind beispielsweise Werkzeuge wie die Richtlinie VDI 2143 [1] oder darüber hinaus das grafisch-interaktive Bewegungsdesign [2] üblich. Um Nutzern mit weniger Erfahrung in diesem Themengebiet die Bewegungssteuerung zugänglich zu machen, wurden in der jüngeren Vergangenheit Möglichkeiten aufgezeigt, mit denen man Kinematiken auf interaktive Weise ansteuern kann [3]. Die Idee bestand darin, mit verschiedenen Eingabemethoden und -geräten, wie Joystick, Tiefenkamera zur Verfolgung der Handbewegung oder Verfolgung der Mausposition, die Sollwertvorgabe zu vereinfachen.

Darüber hinaus lässt sich die Maschinenbedienung sowohl um eine visuelle als auch um eine haptische Rückmeldung für den Nutzer erweitern. Für die haptische Interaktion in der Virtual-Reality (VR) stehen Force-Feedback-Handschuhe wie der SenseGlove Nova [4] zur Verfügung. Der in [5] vorgestellte und im Vergleich preiswerte Force-Feedback-Handschuh kann ebenfalls in Verbindung mit einem VR-Headset genutzt werden. Er bietet dem Bediener beim Greifen virtueller Objekte eine kinästhetische Kraftrückmeldung auf die Fingerspitzen. Da diese Konstellation auch eine Verfolgung der Handposition im Raum ermöglicht, ist eine interaktive Sollwertvorgabe für Bewegungssteuerungen denkbar.

Der vorliegende Beitrag soll aufzeigen, wie sich alternative Eingabegeräte mit klassischen Motion-Control-Systemen verbinden lassen und welche Gestalt die dafür notwendige Softwarearchitektur annehmen kann.

2 Versuchsstand

2.1 Allgemeiner Aufbau

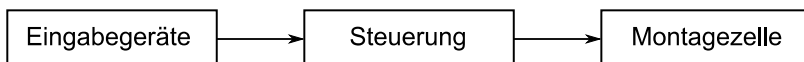


Bild 1: Übersicht

Der gesamte in dieser Vorstudie verwendete Versuchsstand lässt sich grob in die drei Ebenen Eingabegeräte, Steuerung und Montagezelle gliedern (Bild 1).

Durch das Human-Machine-Interface (HMI) gibt der Maschinenbediener interaktiv mit verschiedenen Eingabegeräten seine Wunschposition vor. Diese Vorgaben wandeln im Anschluss die Steuerungsprogramme für das Motion-Control-System der Montagezelle in Sollwerte für die darin verbauten Servomotoren um.

2.2 Eingabegeräte

Zum Einsatz kommen in diesem Vorhaben für Motion-Control-Systeme alternative Bedienerchnittstellen.



Bild 2: Tiefenkamera

Eine solche Tiefenkamera (Bild 2) arbeitet mit Infrarot-Kameras, deren erzeugte Bilder ein spezielles vom Hersteller angebotenes Programm in Handpositionen umwandelt.

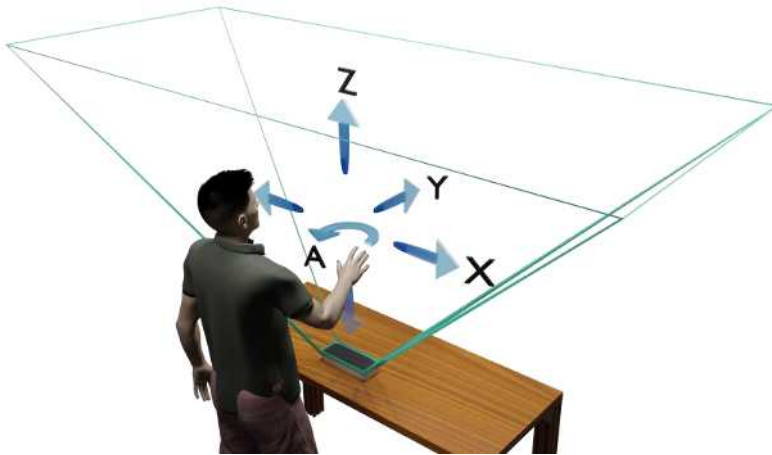


Bild 3: Anwendungsszenario der Tiefenkamera mit relevanten Koordinaten

Dabei lassen sich alle sechs Freiheitsgrade (drei translatorische und drei rotatorische) einzeln auswerten, wobei in Bild 3 nur die für diesen Beitrag relevanten Koordinaten eingezeichnet sind. Darüber hinaus ist auch die Erfassung der Gelenkpositionen aller Finger möglich.

Die Anwendungsmöglichkeiten sind mannigfaltig. So konnten in früheren Arbeiten ein Knickarmroboter oder auch ein Delta-Roboter mit einer solchen Kamera angesteuert werden [6, 7].

Eine weitere Eingabemöglichkeit stellt ein VR-System dar. Dazu gehören ein Headset und zwei Controller (Bild 4). Dieses ist in der Lage, die Einzelbewegungen der drei zugehörigen Geräte in allen sechs Freiheitsgraden zu verfolgen. Zusätzlich bietet es den Einblick in eine virtuelle Umgebung, in der sich der Benutzer umsehen und bewegen kann.

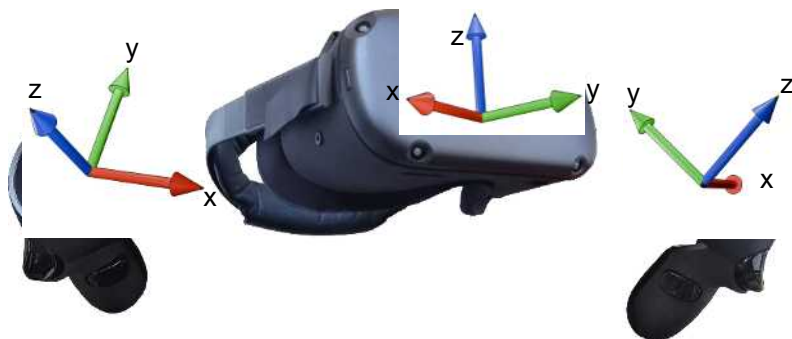


Bild 4: VR-System bestehend aus Headset und Controllern (Oculus Quest)

Es ist weiterhin möglich, den in [5] aufgezeigten haptischen Handschuh für die Steuerung der Montagezelle zu nutzen (Bild 5). Dieser Handschuh besteht aus einer Basis (1), die auf der Hand befestigt ist. Auf der Basis sind mehrere Kraftübertragungseinheiten aufgebracht, die mit der jeweiligen Fingerspitze verbunden sind. Bei einem Null-Kraft-Zustand können die Finger frei bewegt werden. Wenn in der VR die virtuelle Hand ein Objekt greift, wird dies als Force-Feedback-Zustand bezeichnet. In diesem Zustand wird die Drehung des Antriebsgliedes (2) durch den Servomotor (3) gesperrt, welche zur Sperrung der Fingerbeugung führt. Zur Verfolgung der Handposition wird der VR-Controller am Arm des Bedieners befestigt. Die Messung der Fingerbeugung erfolgt durch ein am Antriebsglied befestigten Potentiometer (4). Zur Erkennung des virtuellen Greifens, Darstellung der VR-Szene und zum Antreiben der Servomotoren wird die VR-fähige Spieleengine Unity verwendet.

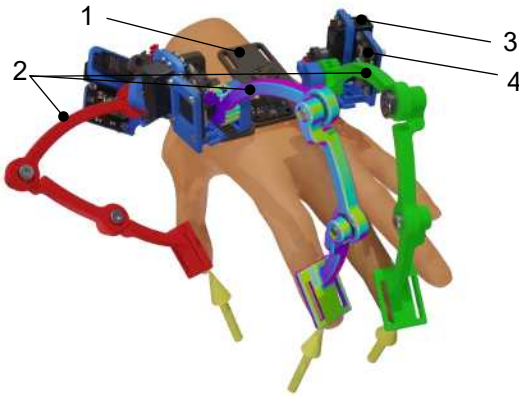


Bild 5: Haptischer Handschuh (Eigenentwicklung)

2.3 Steuerung und Montagezelle

Zur Bewegungssteuerung des Versuchsstandes gehört eine Soft-SPS der Firma Beckhoff, mit physisch getrennter Entwicklungs- und Laufzeitumgebung. Auf dem Entwicklungsrechner laufen die Programme zur Auswertung der Eingabegeräte und Visualisierung, während der Industrie-PC (IPC) mit Laufzeitumgebung die Sollwerte über EtherCAT an die Servoverstärker eines Motion-Control-Systems (MCS) von Bosch Rexroth überträgt.

Die Montagezelle (Bild 6) besteht aus einem kartesischen Portal (1) mit drei orthogonalen Achsen x , y und z sowie einer zusätzlichen a -Achse, mit der sich ein Endeffektor um die z -Achse drehen lässt. Die aufgezählten Achsen werden jeweils von einem Servomotor angetrieben, welche Teil des oben genannten MCS sind. Darüber hinaus ist ein für Montagelinien übliches Transfersystem (2) verbaut, welches jedoch erst in zukünftigen Arbeiten Verwendung finden soll.

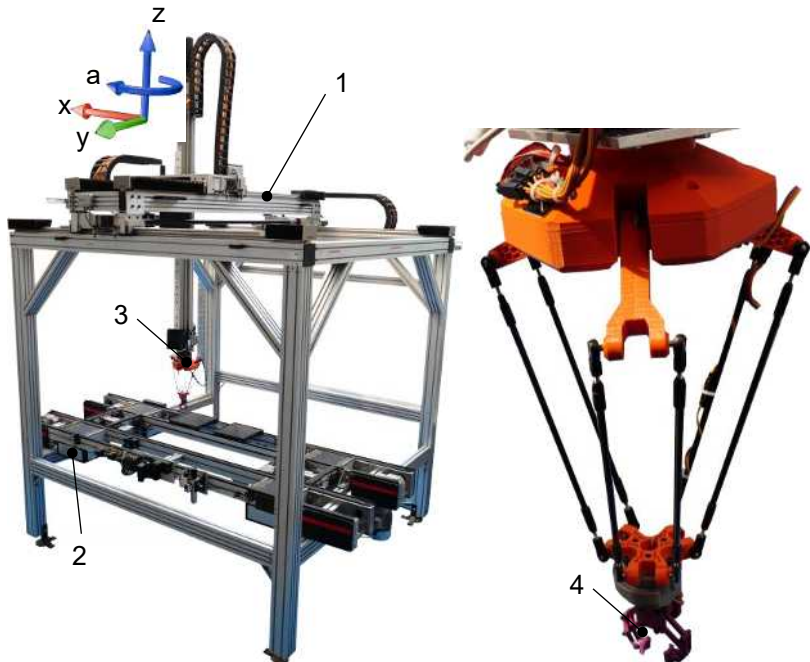


Bild 6: Montagezelle komplett (links), Detailaufnahme Delta-Roboter (rechts)

An der a-Achse des Portals ist ein Deltaroboter (3) befestigt, angetrieben durch drei Kleinstservomotoren. Den Endeffektor selbst bildet ein ebenfalls servomotorisch angetriebener Zwei-Backen-Greifer (4), der sich für einfache Handhabungsaufgaben nutzen lässt. Diese vier Aktoren erhalten ihre Sollwerte von einem Tinkerforge-Mikrocontroller, der mit dem besagten Entwicklungsrechner verbunden ist.

Durch diese Kombination lassen sich die Vorteile eines großen Arbeitsraumes durch das kartesische Portal einerseits und guter dynamischer Eigenschaften durch den Delta-Roboter andererseits nutzen.

Ferner sind Operationen denkbar, indem sich das Portal auf einen Werkstückträger des Transfersystems aufsynchrisiert und der Delta-Roboter dabei etwaige Manipulationen durchführt.

3 Softwarearchitektur

Gegenstand der folgenden Betrachtung ist der modulare Aufbau der Softwarearchitektur (Bild 7) zur Ansteuerung des Versuchsstandes. Diese weist im Wesentlichen drei Komponenten auf. Dazu gehören die Mensch-Maschine-Schnittstelle, die Auswerteprogramme und das SPS-Programm des Motion-Control-Systems.

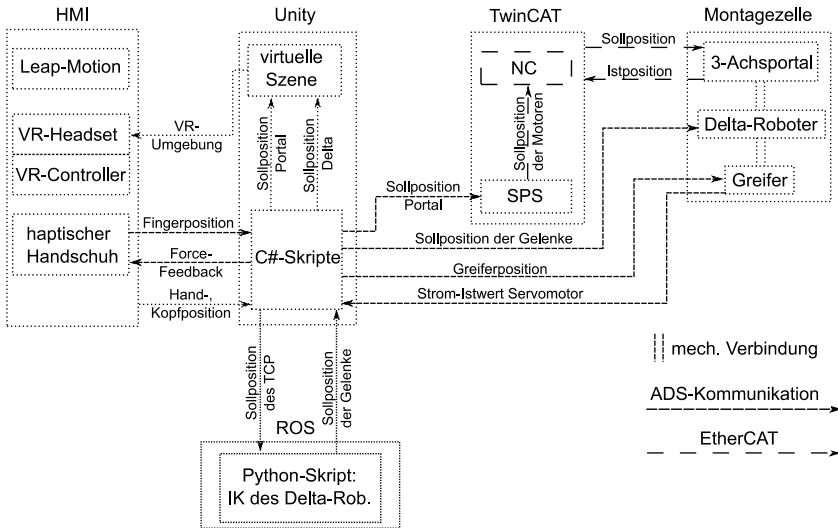


Bild 7: Datenfluss

Die Mensch-Maschine-Schnittstelle (HMI) umfasst die in Kapitel 2.2 beschriebenen Eingabegeräte. Davon erfassen die Leap-Motion-Kamera und der VR-Controller die Hand- und das VR-Headset die Kopfposition des Benutzers. Der haptische Handschuh überträgt die Fingerposition. Diese so gewonnenen Echtzeitdaten verarbeiten C#-Skripte innerhalb der Unity-Umgebung mithilfe inverser Kinematik sowohl in die Sollposition des kartesischen Portals als auch in die des TCP des Delta-Roboters. Dessen inverse Kinematikberechnung zur Ermittlung der drei Gelenkpositionen erfolgt jedoch in einem externen ROS-Programm unter Linux (siehe dazu auch [6] und [8]). Die Ausgabeergebnisse bilden einen

externen Sollwertgenerator sowohl für den virtuellen als auch den realen Versuchsstand.

Die Kommunikation zwischen Unity und dem SPS-Programm läuft über das Kommunikationsprotokoll ADS, sodass die Positionswerte aller vier Achsen des Portals in Form von Variablen verfügbar sind. Diese speist die Funktion MC_ExtSetPointGenFeed in die jeweilige NC-Achse, die wiederum mit einem Glättungsfilter einen Durchschnittswert einer Reihe von Positionssollwerten erzeugt. Diese dann geglätteten Sollwerte gelangen über den Ethernet-Feldbus EtherCAT zu den Servoverstärkern und damit den Servomotoren des Portals.

Die Datenkommunikation zwischen Unity und dem Micro-Controller des Delta-Roboters erfolgt über TCP/IP, dessen Servomotoren mit Hilfe von Pulsweitenmodulation angesteuert werden. Kommt es nun zu einem erfolgreichen Greifvorgang, misst der Micro-Controller einen erhöhten Strombedarf des Greiferservomotors, den wiederum ein C#-Skript in einer Variablen speichert. Bei Überschreitung eines Grenzwertes spricht der Force-Feedback-Handschuh an wodurch der Benutzer eine haptische Bestätigung erfährt.

4 Virtual-Reality-Visualisierung

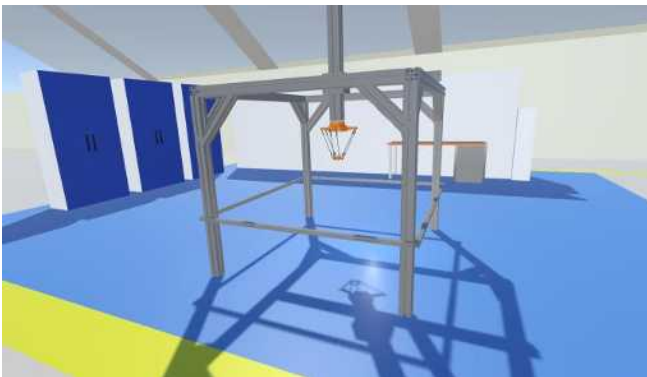


Bild 8: VR-Szene

Die VR-Szene enthält ein detailliertes Modell des Labors, in dem sich die Montagezelle befindet (Bild 8). Die Oculus Quest wird als VR-Headset eingesetzt, um eine immersive und interaktive Umgebung zu schaffen. Wenn das VR-Headset oder die Controller als Eingabegerät ausgewählt werden, bewegt der Benutzer im virtuellen Raum den Controller oder das

Headset zur gewünschten Zielposition des Endeffektors der Montagezelle. Der Endeffektor folgt in Echtzeit der Kopf- oder Handbewegung. Wird der Leap-Motion-Sensor als Eingabegerät gewählt, so wird der Arbeitsbereich der Montagezelle virtuell und skaliert über dem Leap-Motion-Sensor platziert. Die erfasste Handposition wird direkt in die Sollposition des Endeffektors umgewandelt, wodurch der Endeffektor den Handbewegungen des Benutzers folgt und in VR visualisiert wird. Die VR-Visualisierung bietet somit eine immersive Möglichkeit, die Montagezelle zu betrachten und zu steuern. Der Benutzer kann sich frei in der virtuellen Umgebung bewegen und die Steuerung der Montagezelle aus verschiedenen Perspektiven beobachten. Dies verbessert das Verständnis der Funktionsweise der Montagezelle und ermöglicht eine detaillierte Überprüfung und Optimierung der Steuerungsprozesse.

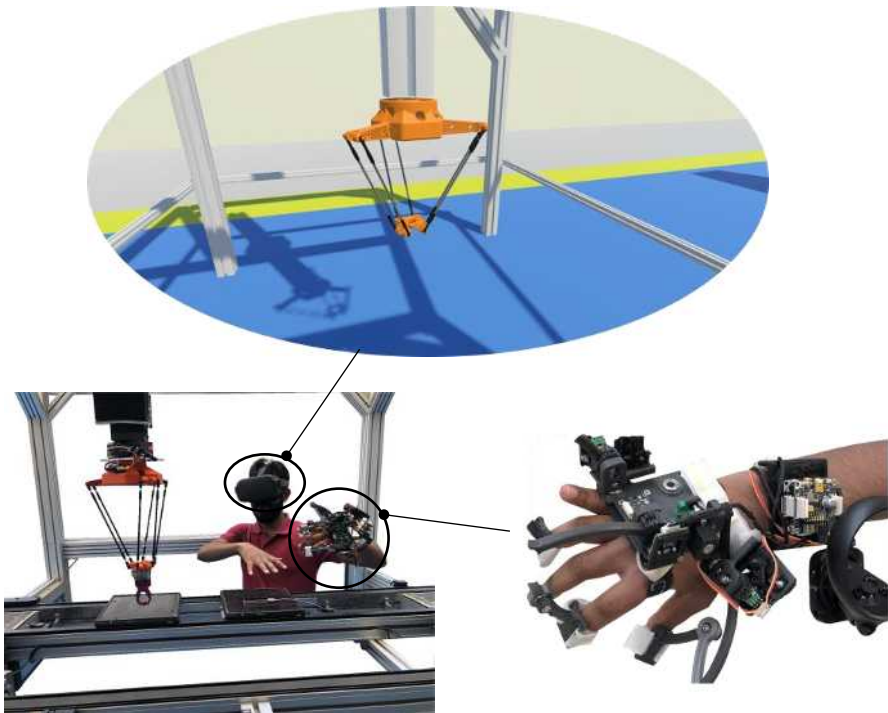


Bild 9: Bewegungssteuerung der Montagezelle

5 Zusammenfassung

Die skalierbare modular aufgebaute Architektur des Systems integriert mehrere Steuerungsebenen, die getrennt voneinander funktionieren. Neue Sensoren, Aktoren oder Softwaremodule können problemlos integriert oder bestehende Komponenten ersetzt werden, ohne das gesamte System neu zu gestalten.

Der wissenschaftliche Beitrag dieser Arbeit liegt in der innovativen Integration und Verteilung verschiedener Steuerungssysteme und Technologien. Diese Architektur bietet eine flexible, skalierbare und modulare Plattform, die als Grundlage für eine Vielzahl zukünftiger Entwicklungen und Forschungen in der Robotik und Automatisierung dienen kann.

6 Ausblick

Neben den in dieser Vorstudie vorgestellten Eingabegeräten sollen in folgenden Arbeiten noch weitere in Betrieb genommen werden. Dazu gehört das mittlerweile weit verbreitete Touchscreen, bei dem sich mit dem Finger gezeichnete Wegkurven in Sollwerte umrechnen lassen. Des Weiteren ist die Verwendung von IMU-Sensoren denkbar, die aus einer Verdrehung eines Smartphones eine Geschwindigkeitsvorgabe ermöglicht.

Durch Probandenstudien lässt sich dann die für unterschiedliche Antriebssysteme jeweils geeignete Beschaffenheit der Mensch-Maschine-Schnittstelle ermitteln und damit eine Auswahl aus der Vielzahl an Eingabegeräten treffen.

Ferner sind Berechnungsalgorithmen zu entwickeln, die aus eingegebenen Sollwertvorgaben Bewegungsprofile berechnen, zum Beispiel möglichst energieeffizient oder taktzeitminimal. Mit solchen Bewegungsprofilen ließen sich dann Visualisierungen im Rahmen einer virtuellen Inbetriebnahme realisieren.

Darüber hinaus sind telemanipulative Fernwartungen vorstellbar, die es dem Maschinenbediener erlauben, nicht mehr vor Ort und damit möglichst zeitnah auf etwaige spontane Formatwechsel oder auch Störungen im Maschinenablauf reagieren zu können.

7 Literaturverzeichnis

- [1] VEREIN DEUTSCHER INGENIEURE: *VDI 2143 Bewegungsgesetze für Kurvengetriebe und Motion-Control-Systeme : Blatt 1*. Berlin : Beuth Verlag, 2024
- [2] HEINE, Andreas ; MATTHES, Jörg ; SCHULZ, Ringo ; BERGER, Maik: *MOCAD - Ein grafisch interaktives Werkzeug zur Berechnung und Optimierung von Kurvengetrieben und Motion Control Systemen*. Düsseldorf : VDI Verlag, 2010 (VDI-Berichte 2116)
- [3] WABNITZ, Philipp ; HEINE, Andreas ; CHEN, Lian ; TEICHGRÄBER, Carsten ; BERGER, Maik: PAROHA: Konzeptstudie eines Parallelkinematik-Roboter-Handgelenks für spezifische Handhabungsaufgaben in einem MRK-Szenario. In: *Fachtagung MECHATRONIK*. Paderborn, 27.-28.03.2019, S. 192–196
- [4] SENSEGLOVE: *The New Sense in VR for enterprise*. URL <https://www.senseglove.com/product/nova/> – Überprüfungsdatum 2024-07-23
- [5] ANILKUMAR, Akhilraj ; WABNITZ, Philipp ; BERGER, Maik: Entwicklung eines haptischen Handschuhsystems für Virtual-Reality-Anwendungen. In: *Getriebetagung 2022*. 22.-23.09.2022 : Logos Verlag Berlin, S. 215–223
- [6] ANILKUMAR, Akhilraj ; BRINSTER, Luise: Simulation eines parallelen Roboters mit GeoGebra und ROS. In: *12. SAXSIM 2023*.
- [7] ANILKUMAR, Akhilraj ; BERGER, Maik: *Entwicklung eines ROS-Kommunikationssystems für die robotische Telemanipulation mit haptischem Feedback mittels Handbewegung in einem MRK-Szenario* : 9. IFToMM D-A-CH Konferenz, 2023
- [8] ANILKUMAR, Akhilraj ; BERGER, Maik: Digitaler Zwilling durch Verbindung von ROS-Kompatiblen Robotern mit Virtual-Reality zur Echtzeitvisualisierung von Roboterbewegung. In: *13. SAXSIM 2024*.

2D “TenSens”-Sensor zur simultanen Messung von Kräften und Momenten

Lukas Merker*, Thomas Kissinger*, Valter Böhm**,
David Herrmann**, Lena Zentner*

* TU Ilmenau, Fakultät für Maschinenbau
lukas.merker@tu-ilmenau.de

** OTH Regensburg, Fakultät Maschinenbau

Kurzfassung

Vorgestellt wird ein 2D-Kraft-Momentensensor, der mittels eines faseroptischen Sensors die Bestimmung äußerer Kräfte in einer Ebene sowie Momente in einer Achse senkrecht zu dieser Ebene ermöglicht. Das Sensorkonzept kommt durch den multifunktionalen Einsatz eines vorgespannten faseroptischen Sensors als Messmittel und als tragendes Strukturelement ohne einen Deformationskörper aus, und bietet damit Potentiale für den Leichtbau, die Reduzierung des Fertigungsaufwands sowie der mechanischen Modellkomplexität.

Title / Abstract

2D “TenSens” sensor for simultaneous force/torque measurement

This paper presents a 2D force/torque sensor, which uses an optical fiber sensor to determine external forces in a plane and moments in a normal direction to that plane. The sensor concept does not require a deformation body due to the multifunctional use of a prestressed optical fiber sensor acting as both a measuring device and a structural component, and thus offers potential for lightweighting as well as the reduction of manufacturing and model complexity.

Aufbau des Sensors und Untersuchungsmethodik

Der Sensor besteht gemäß der Patentanmeldung [1] aus zwei starren Körpern I und II, die an ihren Eckpunkten durch vier Messabschnitte (MA) der Länge l_i eines einzigen faseroptischen Sensors (FOS) gemäß Abb. 1 miteinander verbunden sind. Der FOS besitzt einen Ein- und Ausgang für ein eingekoppeltes Lichtsignal, das entlang des FOS insgesamt acht Reflexionselemente (zwei pro MA) durchläuft. Dies ermöglicht eine orts aufgelöste Detektion von dehnungsinduzierten Änderungen der optischen Wegdifferenz in den MA mittels Range-Resolved Interferometry [2]. Im Referenzzustand des Sensors sind alle MA vorgespannt. In einem simulationsbasierten Konzeptnachweis wird der Messbereich des Sensors anhand der nichtlinearen Modellgleichungen so bestimmt, dass für alle einwirkenden Kraft-Momenten-Paare \vec{F} , \vec{M} die Forderung $0 < \varepsilon_i < \varepsilon_{max} \forall i$ gilt, wobei ε_i die Dehnung des i -ten MAs und ε_{max} die maximal zulässige Dehnung des FOS ist. Es wird gezeigt, dass \vec{F} und \vec{M} analytisch eindeutig aus den Längen der MA bestimmbar sind.

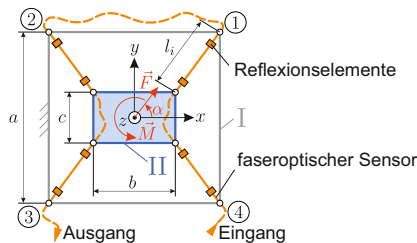


Abb. 1: Sensorvorrichtung

Literatur

- [1] Merker, L. et al. „Selbsttragende Vorrichtung (TenSens-Plattform) mit vorgespannten optischen Fasern zur Ermittlung einer äußeren Kraft [...]“ Patentanmeldung, DRN: 2023102717405500DE. 2023.
- [2] Kissinger, T., Charrett, T. O. und Tatam, R. P. „Range-resolved interferometric signal processing using sinusoidal optical frequency modulation“. In: *Optics express* 23.7 (2015), S. 9415–9431.

Dieser Tagungsband präsentiert die Fachbeiträge der Getriebetagung 2024 in München, die im Rahmen einer langjährigen Tradition stattfindet. Die alle zwei Jahre gemeinsam von den Technischen Universitäten Aachen, Chemnitz und München organisierte Konferenz ist ein bedeutendes Forum für den Austausch neuester Entwicklungen und Forschungsergebnisse im Bereich der Getriebetechnologie.

Im Zentrum dieser Tagung stehen die digitalen Werkzeuge im Entwicklungsprozess, die klassische Getriebesynthese sowie die Integration von Mechanismen und Getrieben in industriellen Anwendungen. Der globale Wettbewerb und die damit einhergehend kürzeren Entwicklungszyklen erfordern den Einsatz innovativer Lösungsstrategien für Antriebs- und Bewegungsaufgaben. Ziel ist es, eine schnelle, ressourcen- und energieeffiziente Herstellung von Antriebslösungen zu ermöglichen. Dies gelingt nur durch eine stetig wachsende, integrative und interdisziplinäre Verknüpfung aller Fachgebiete innerhalb eines mechatronischen Entwicklungsprozesses.

Im Zusammenspiel mit der Robotik und den Methoden des Rapid-Prototyping bietet sich die Möglichkeit, schnelle, individualisierte und kostengünstige Bewegungssysteme sowie Roboter der nächsten Generation zu realisieren.