

# Automatisierte Generierung von Ordnungsschikanen für Vibrationswendelförderer mithilfe von Reinforcement Learning

Julia Cosima Isabell Stocker

Vollständiger Abdruck der von der TUM School of Engineering and Design der  
Technischen Universität München zur Erlangung einer

Doktorin der Ingenieurwissenschaften (Dr.-Ing.)

genehmigten Dissertation.

Vorsitz: Prof. Dr.-Ing. Johannes Fottner

Prüfer der Dissertation:

1. Prof. Dr.-Ing. Gunther Reinhart
2. Prof. Dr. ir. Daniel J. Rixen

Die Dissertation wurde am 09.06.2022 bei der Technischen Universität München  
eingereicht und durch die School of Engineering and Design am 31.08.2022 ange-  
nommen.



## **Abstract**

The thesis shows a consistent method for the automated generation of orienting devices for vibratory bowl feeders using reinforcement learning, which makes it possible to shorten the development time for orienting devices and increase the quality of the results.

Existing partial solutions are adapted and merged, and missing components are newly developed. The method is validated by means of a prototypical implementation.



## **Zusammenfassung**

In der Arbeit wird eine durchgängige Methode zur automatisierten Generierung von Ordnungsschikanen für Vibrationswendelförderer mithilfe von Reinforcement Learning vorgestellt, welche die Verkürzung der Entwicklungszeit für Ordnungsschikanen sowie die Steigerung der Ergebnisqualität ermöglicht.

Bestehende Teillösungen werden angepasst und zusammengeführt sowie fehlende Bausteine neu entwickelt. Die Methode wird anhand einer prototypischen Umsetzung validiert.



## Geleitwort der Herausgeber

Die Produktionstechnik ist in Zeiten globaler Herausforderungen, wie der Klimakrise, dem Mobilitätswandel und der Überalterung der Gesellschaft in westlichen Ländern, für eine nachhaltige Weiterentwicklung unserer Industriegesellschaft von zentraler Bedeutung. Der Einfluss eines Industriebetriebs auf die Umwelt und die Gesellschaft hängt dabei entscheidend von den eingesetzten Produktionsmitteln, den angewandten Produktionsverfahren und der eingeführten Produktionsorganisation ab. Erst das optimale Zusammenspiel von Mensch, Organisation und Technik erlaubt es, alle Potenziale für den Unternehmenserfolg auszuschöpfen. Dabei muss größtes Augenmerk darauf gelegt werden, möglichst ressourcenschonend, effizient und resilient zu werden, um flexibel im volatilen Produktionsumfeld zu agieren.

Um in dem Spannungsfeld Nachhaltigkeit, Komplexität, Kosten, Zeit und Qualität bestehen zu können, müssen Produktionsstrukturen ständig neu überdacht und weiterentwickelt werden. Dabei ist es notwendig, die Komplexität von Produkten, Produktionsabläufen und -systemen einerseits zu verringern und andererseits besser zu beherrschen.

Ziel der Forschungsarbeiten des *iwb* ist die ständige Verbesserung von Produktentwicklungs- und Planungssystemen, von Herstellverfahren sowie von Produktionsanlagen, Betriebsorganisation, Produktions- und Arbeitsstrukturen sowie Systeme zur Auftragsabwicklung werden unter besonderer Berücksichtigung der Anforderungen des Personals sowie von Nachhaltigkeitsaspekten entwickelt. Die dabei eingesetzten rechnergestützten und Künstliche-Intelligenz-basierten Methoden und die notwendige Steigerung des Automatisierungsgrades dürfen jedoch nicht zu einer Verfestigung arbeitsteiliger Strukturen führen. Fragen der optimalen Einbindung ökologischer und sozialer Aspekte in alle Planungs- und Entwicklungsprozesse spielen deshalb eine sehr wichtige Rolle.

## Zusammenfassung

---

Die im Rahmen dieser Buchreihe erscheinenden Bände stammen thematisch aus den Forschungsbereichen des *iwb*. Diese reichen von der Entwicklung von Produktionssystemen über deren Planung bis hin zu den eingesetzten Technologien in den Bereichen Fertigung und Montage. Die Steuerung und der Betrieb von Produktionssystemen, die Qualitätssicherung, die Verfügbarkeit und die Autonomie sind Querschnittsthemen hierfür. In den *iwb*-Forschungsberichten werden neue Ergebnisse und Erkenntnisse aus der praxisnahen Forschung des Institutes veröffentlicht. Diese Buchreihe soll dazu beitragen, den Wissenstransfer zwischen dem Hochschulbereich und den Anwendenden zu verbessern.

*Rüdiger Daub*

*Gunther Reinhart*

*Michael Zühl*



## Danksagung

Die vorliegende Dissertation entstand während meiner Arbeit als wissenschaftliche Mitarbeiterin am Institut für Werkzeugmaschinen und Betriebswissenschaften (*iwb*) der Technischen Universität München.

Besonders danken möchte ich meinem Doktorvater Herrn Prof. Dr.-Ing. Gunther Reinhart und Herrn Prof. Dr.-Ing. Michael F. Zäh für ihre stets großzügige Unterstützung und Förderung im Rahmen meiner Forschungstätigkeit sowie im Institutsleitungskreis. Bei Herrn Prof. Reinhart bedanke ich mich darüber hinaus für die konstruktive und wohlwollende Betreuung meiner Arbeit sowie bei Herrn Prof. Dr.-Ing. Johannes Fottner für die Übernahme des Vorsitzes der Prüfungskommission. Des Weiteren danke ich Herrn Prof. Dr. ir. Daniel Rixen für die aufmerksame Durchsicht meiner Arbeit sowie die Übernahme des Koreferats.

Ein wesentlicher Erfolgsfaktor bei der Erstellung meiner Dissertation waren die zahlreichen Diskussionen mit wohlwollenden und erfahrenen Kolleginnen und Kollegen zu allen fachlichen und persönlichen Problemstellungen. Daher möchte ich mich besonders bei meinen *iwb*-Kolleginnen und Kollegen Gregor Lux-Gruenberg, Marco Ulrich, Veit Hammerstingl, Corinna und Stefan Liebl, Till Günther, Andreas Bachmann, Susanne Vernim, Anna Kollenda, Daria Leiber und Ulrich Teschemacher bedanken, von denen ich fachlich wie auch persönlich sehr viel lernen konnte. Den beiden letztgenannten danke ich zudem für die aufmerksame und kritische Durchsicht meiner Arbeit und ihre wertvollen Hinweise. Auch allen Studierenden, die ich im Rahmen meiner Promotionszeit betreuen durfte, bin ich zu großem Dank für viele wertvolle fachliche Beiträge zu meiner Dissertation verpflichtet.

Besonderen Dank möchte ich meiner Familie und meinen Freunden aussprechen, ohne die mein Weg und die Vollendung dieser Arbeit nicht möglich

## Zusammenfassung

---

gewesen wären. Auch wenn meine Eltern Sonnhild und Hans diesen besonderen Meilenstein leider nicht mehr miterleben können, empfinde ich tiefste Dankbarkeit für die immerwährende Unterstützung all meiner Vorhaben. Dafür möchte ich meiner hinzugewonnenen Familie, bestehend aus Uli, Lilli, Peter, Sonja und Tobias, für die viele schöne gemeinsame Zeit und den Rückhalt danken. Bei meinen Freundinnen Anita Huhn und Luisa Zylka bedanke ich mich von ganzem Herzen für ihre moralische Unterstützung in all den Jahren und die gründliche Durchsicht meiner Arbeit.

Lieber Uli, du hast mir in zahlreichen fachlichen Diskussionen häufig entscheidende Impulse gegeben. Deine besonders kritische Durchsicht meiner Dissertation hat meine Arbeit noch einmal erheblich vorangebracht. Jedoch mindestens genauso bedeutend war für mich dein unerschütterliches Vertrauen in meinen Erfolg und meine Fähigkeiten sowie deine positive Grundhaltung. Eine Doktorarbeit ist auf moralischer Ebene stets „Teamwork“, insb. in der Partnerschaft, daher danke ich dir aus tiefstem Herzen für deine Unterstützung und Rücksichtnahme in den letzten Jahren. Ich bin sehr froh, dass wir unsere Dissertationen nun beide ins Ziel gebracht haben und freue mich auf unsere gemeinsame Zukunft als Familie mit vielen spannenden neuen Herausforderungen.

München, im September 2022

*Cosima Stocker*

# Inhalt

<b>Abstract</b>	<b>I</b>
<b>Zusammenfassung</b>	<b>III</b>
<b>Abkürzungsverzeichnis</b>	<b>XIII</b>
<b>Verzeichnis der Formelzeichen</b>	<b>XV</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Ausgangssituation und Problemstellung . . . . .	1
1.2 Zielsetzung . . . . .	4
1.3 Spezifizierung des Betrachtungsbereichs . . . . .	5
1.4 Wissenschaftstheoretische Einordnung . . . . .	6
1.5 Methodische Vorgehensweise und Aufbau der Arbeit . . . . .	10
<b>2 Stand der Erkenntnisse</b>	<b>13</b>
2.1 Begriffsdefinitionen . . . . .	13
2.2 Vibrationswendelförderer . . . . .	16
2.2.1 Produktionstechnische Bedeutung und Einsatzgebiete . . . . .	17
2.2.2 Aufbau und Funktionsweise . . . . .	20
2.2.3 Fördergut und Ordnungsschikanen . . . . .	22
2.3 Auslegung von Ordnungsschikanen . . . . .	29
2.3.1 Konventionelle Auslegung . . . . .	29
2.3.2 Methodische Unterstützung bei der Auslegung . . . . .	32
2.3.3 Unterstützung der Auslegung durch Simulationsverfahren . . . . .	37
2.3.4 Automatisierte Auswahl und Sequenzierung von Schikanenelementen . . . . .	40
2.4 Forschungsdefizit . . . . .	42
2.5 Reinforcement Learning als Verständnisgrundlage . . . . .	43
2.5.1 Künstliche Intelligenz und maschinelles Lernen . . . . .	44

2.5.2	Reinforcement Learning im Detail . . . . .	53
<b>3</b>	<b>Anforderungsanalyse und Lösungsansatz</b>	<b>59</b>
3.1	Anforderungen an die Methode . . . . .	59
3.1.1	Allgemeine Anforderungen . . . . .	59
3.1.2	Spezifische Anforderungen . . . . .	60
3.2	Lösungsansatz: Methode zur automatisierten Generierung von Ordnungsschikanen . . . . .	62
<b>4</b>	<b>Methodenschritt 1: Datenaufbereitung</b>	<b>67</b>
4.1	Überblick . . . . .	67
4.2	Automatisierte Fördergut-Analyse . . . . .	69
4.2.1	Extraktion von Fördergut-Charakteristika . . . . .	69
4.2.2	Ähnlichkeit-Analyse . . . . .	76
4.3	Entwurf einer erweiterbaren CAD-Bibliothek . . . . .	83
4.3.1	Auswahl und Parametrierung von Schikanenelementen	83
4.3.2	Struktur der CAD-Bibliothek . . . . .	87
<b>5</b>	<b>Methodenschritt 2: Bewertung der Schikanenelemente</b>	<b>91</b>
5.1	Überblick . . . . .	91
5.2	Simulationsbasierte Bewertung von Schikanenelementen . . . .	93
5.2.1	Bewertung von Schikanenelementen . . . . .	93
5.2.2	Umsetzbarkeit mit der Physiksimulation . . . . .	96
5.3	Berechnungsvorschriften für Simulationsläufe . . . . .	97
5.3.1	Automatisierte Versuchsplanung . . . . .	97
5.3.2	Konvergenz der Simulationsergebnisse . . . . .	99
<b>6</b>	<b>Methodenschritt 3: Generierung der Ordnungsschikane</b>	<b>101</b>
6.1	Überblick . . . . .	101
6.2	Aufbau des Reinforcement-Learning-Systems . . . . .	104
6.2.1	Modellierung des zugrundeliegenden Optimierungs- problems . . . . .	104
6.2.2	Struktur und Komponenten des Reinforcement-Learning- Systems . . . . .	105
6.3	Reinforcement Learning zur automatisierten Generierung von Ordnungsschikanen . . . . .	109
6.3.1	Lernprozess und Training . . . . .	109

6.3.2	Prädiktion von Ordnungsschikanen . . . . .	112
6.4	Simulative Evaluation und Optimierung . . . . .	114
6.4.1	Erzeugung der Ordnungsschikane aus dem Design- entwurf . . . . .	114
6.4.2	Simulation der Ordnungsschikane und Parameter- optimierung . . . . .	115
<b>7</b>	<b>Implementierung und Evaluierung der Auslegungsmethode</b>	<b>117</b>
7.1	Rahmenbedingungen . . . . .	119
7.2	Automatisierte Generierung einer Ordnungsschikane . . . . .	119
7.2.1	Datenaufbereitung . . . . .	119
7.2.2	Bewertung der Schikanenelemente . . . . .	123
7.2.3	Generierung der Ordnungsschikane . . . . .	124
7.2.4	Evaluation der Ordnungsschikane . . . . .	127
7.3	Kritische Bewertung der Forschungsergebnisse . . . . .	128
7.3.1	Richtigkeit der Ergebnisse . . . . .	128
7.3.2	Anwendbarkeit im industriellen Kontext . . . . .	130
7.3.3	Wirtschaftlichkeit . . . . .	132
7.3.4	Allgemeingültigkeit . . . . .	134
7.3.5	Zusammenfassung . . . . .	135
7.3.6	Bewertung der Zielerreichung . . . . .	135
<b>8</b>	<b>Zusammenfassung und Ausblick</b>	<b>139</b>
8.1	Zusammenfassung . . . . .	139
8.2	Ausblick . . . . .	140
	<b>Literatur</b>	<b>143</b>
	<b>Betreute Studienarbeiten</b>	<b>181</b>
	<b>Anhang</b>	<b>193</b>



## Abkürzungsverzeichnis

<b>ACM</b>	Association for Computing Machinery
<b>ACS</b>	American Chemical Society
<b>ASME</b>	American Society of Mechanical Engineers
<b>BMWi</b>	Bundesministerium für Wirtschaft und Energie
<b>CAD</b>	Computer Aided Design
<b>CIRP</b>	Collège International pour la Recherche en Productique
<b>CNN</b>	Convolutional Neural Network
<b>DNN</b>	Deep Neural Network
<b>DRM</b>	Design Research Methodology
<b>ECMA</b>	European Computer Manufacturers Association
<b>EU</b>	Europäische Union
<b>IEEE</b>	Institute of Electrical and Electronics Engineers
<b>IFRR</b>	International Foundation of Robotics Research
<b>IFTtoMM</b>	International Federation for the Promotion of Mechanism and Machine Science
<b>IfW</b>	Institut für Weltwirtschaft
<b>IPA</b>	Fraunhofer-Institut für Produktionstechnik und Automatisierung
<b>KI</b>	Künstliche Intelligenz
<b>KMU</b>	Kleine und mittelständische Unternehmen
<b>MEP</b>	Markov-Entscheidungsprozess
<b>ML</b>	Machine Learning
<b>NN</b>	Neuronales Netz
<b>OG</b>	Orientierungsgrad
<b>OS</b>	Ordnungsschikane/n
<b>OZ</b>	Ordnungszustand
<b>PG</b>	Positionierungsgrad
<b>ReLU</b>	Rectified Linear Unit
<b>RL</b>	Reinforcement Learning

## Abkürzungsverzeichnis

---

<b>RSJ</b>	Robotic Society of Japan
<b>SAC</b>	Soft Actor Critic
<b>SE</b>	Schikanenelement/e
<b>SL</b>	Supervised Learning
<b>StMWi</b>	Bayerisches Staatsministerium für Wirtschaft, Landesentwicklung und Energie
<b>UL</b>	Unsupervised Learning
<b>UML</b>	Unified Modeling Language
<b>VCI</b>	Verband der Chemischen Industrie
<b>VDI</b>	Verein Deutscher Ingenieure
<b>VDMA</b>	Verband Deutscher Maschinen- und Anlagenbau
<b>VWF</b>	Vibrationswendelförderer



## Verzeichnis der Formelzeichen

### Griechische Buchstaben

$\epsilon$	Effizienz eines SE oder einer OS
$\gamma$	Diskontierungsfaktor (RL)
$\pi$	Strategie eines RL-Agenten
$\tau$	Target-Smoothing-Koeffizient (RL)

### Lateinische Buchstaben

$a_t$	Aktion eines RL-Agenten zum Zeitpunkt $t$
$\mathcal{A}$	Menge der zulässigen Aktionen (RL)
$b$	neuronenspezifische Verzerrung
$d_k$	Schraubenkopfdurchmesser
$d_s$	Schaftdurchmesser einer Schraube
$h_k$	Schraubenkopfhöhe
$i$	Laufvariable, Index
$k$	Laufvariable, Index
$l$	Gesamtlänge einer Schraube
$m$	Masse
$M$	Tupel zur Beschreibung eines MEP (RL)
$n$	Laufvariable, Index
$o$	Ausgangssignal eines NN
$O_i$	Orientierung eines Fördergut-Teils
$\mathcal{O}$	Menge der möglichen Orientierungen eines Fördergut-Teils

## Verzeichnis der Formelzeichen

---

$p$	Übergangswahrscheinlichkeit von $s_t$ nach $s_{t+1}$
$\mathcal{P}$	Menge der Übergangswahrscheinlichkeiten (RL)
$q_\pi$	Aktions-Wertefunktion eines RL-Agenten
$r_t$	Belohnung eines RL-Agenten zum Zeitpunkt $t$
$r_{t+1}$	Belohnung eines RL-Agenten zum Zeitpunkt $t + 1$
$R_t$	Kumulierte Belohnung eines RL-Agenten zum Zeitpunkt $t$
$\mathcal{R}$	Menge der erzielten Belohnungen (RL)
$s_t$	Zustand einer RL-Umgebung zum Zeitpunkt $t$
$s_{t+1}$	Zustand einer RL-Umgebung zum Zeitpunkt $t + 1$
$S$	Schwerpunkt
$\mathcal{S}$	Menge der Zustände (RL)
$t$	Zeitpunkt, Laufvariable, Index
$T$	Finaler Zeitschritt
$t_{max}$	Maximale Anzahl an Zeitschritten (RL)
$v_\pi$	Zustands-Wertefunktion eines RL-Agenten
$\bar{v}$	Mittlere Fördergeschwindigkeit
$\vec{v}$	Vorzugslagen-Vektor
$w_i$	Gewicht eines NN
$x$	x-Koordinate
$x_i$	Eingangssignal eines NN
$y$	y-Koordinate
$z$	Laufvariable, z-Koordinate
$\vec{z}$	Zielorientierungs-Vektor

# 1 Einleitung

## 1.1 Ausgangssituation und Problemstellung

Deutschlands Leistungsfähigkeit im internationalen Wettbewerb sowie das hohe Maß an individuellem und gesellschaftlichem Wohlstand basieren zu einem großen Teil auf der fast schon traditionellen Stärke der deutschen produzierenden Industrie (BMW I 2019a, S. 4). Diese Stärke zeigt sich im hohen Anteil des verarbeitenden Gewerbes von über 20% an der gesamten deutschen Bruttowertschöpfung (VCI 2021, S. 7; STMW I 2020, S. 14).

Der durch die Globalisierung der Märkte zunehmende internationale Wettbewerb zwingt deutsche Unternehmen dazu, die Produktlebenszyklen zu verkürzen und bei starkem Preisdruck kundenindividuelle Produkte herzustellen (SCHENK 2015, S. 1, S. 75). Wie sich u. a. durch die Covid-19-Pandemie in den Jahren 2020 und 2021 intensiver denn je offenbart hat, bergen Unterbrechungen und Störungen internationaler Lieferketten ein hohes Risiko für die deutsche Industrie in Form von Kosten und Stillstandszeiten (STMW I 2020, S. 4; IFW 2020, S. 19 f.). Auch internationale Ereignisse wie Handelskonflikte in Verbindung mit der offensiven Umsetzung protektionistischer Strategien, wie z. B. in China und den USA, bis hin zu politischen Instabilitäten und Kriegen können in Zukunft die globalen Lieferketten beeinträchtigen (IFW 2020, S. 23; BMW I 2019a, S. 4).

Aus den eben genannten Gründen ist es erforderlich, die Resilienz der deutschen Unternehmen, also deren Widerstandsfähigkeit und Robustheit gegenüber außerordentlichen Ereignissen zu stärken (STMW I 2020, S. 4; IFW 2020, S. 32). Dazu kann die Erhaltung sowie die Rückverlagerung von Produktionsstätten nach Deutschland beitragen. Die Unternehmen sehen sich jedoch in Deutschland mit Herausforderungen wie einem durch den demografischen

## 1 Einleitung

---

Wandel zunehmenden Fachkräftemangel und im internationalen Vergleich relativ hohen Lohnstückkosten konfrontiert (IFW 2020, S. 17).

Wichtige Technologien zur Bewahrung der Wettbewerbsfähigkeit hiesiger Unternehmen unter den im vorigen Absatz genannten Bedingungen sind die Digitalisierung und Automatisierung von Prozessen (STMWI 2020, S. 4). Durch den Einsatz neuer Technologien der Industrie 4.0 (z. B. intelligente, vernetzte Produktionsressourcen oder digitale Methoden zur Entwicklung, Planung und Steuerung) können die Effizienz der Produktion gesteigert sowie deren Kosten verringert werden. Bei der Anwendung dieser digitalen Technologien bestehen allerdings bei vielen kleinen und mittelständischen Unternehmen (KMU)<sup>1</sup> zum Teil erhebliche Defizite (BMW 2019b, S. 3; IFW 2020, S. 23, S. 27). Zudem sind Automatisierungsansätze tendenziell unflexibel und erst bei hohen produzierten Stückzahlen wirtschaftlich, wogegen KMU meist Klein- und Mittelserien mit hohem Individualisierungsgrad fertigen (LOTTER & WIENDAHL 2012, S. 3). Die vor allem für KMU eingeschränkte Nutzbarkeit von Digitalisierungs- und Automatisierungslösungen ist insb. deshalb problematisch, weil fast 90% der Betriebe im verarbeitenden Gewerbe in Deutschland zu den KMU zählen (STMWI 2020, S. 25).

Automatisierte Montageanlagen benötigen Ordnungssysteme zur geordneten Bereitstellung von Bauteilen (HESSE & MALISA 2016, S. 441 ff.). Häufig eingesetzte Beispiele sind Industrieroboter mit Greifvorrichtungen, Speicher, Magazine oder Vibrationswendelförderer (VWF). Ordnungssysteme sind für einen wesentlichen Anteil der Montageanlagenkosten verantwortlich, sodass sie einen großen Hebel zur Rationalisierung darstellen (SCHMID 2006, S. 16). Zur Zuführung von ungeordneten Kleinteilen werden in automatisierten Montageanlagen in 80% der Fälle VWF eingesetzt (ROCKLAND 1995, S. 30), wie jüngere Arbeiten sowie eine aktuelle Umfrage qualitativ bestätigen (KRAUEL 2020<sup>2</sup>; LOTTER & WIENDAHL 2012, S. 246 f.; MATHIESEN 2017, S. 12 f.). VWF nehmen ungeordnetes Fördergut auf und vereinzeln und sortieren dieses mithilfe von Ordnungsschikanen (OS). Bei Änderungen am Produktionssystem, die in der

---

<sup>1</sup>Nach Definition der EU gelten Unternehmen mit weniger als 250 Mitarbeitenden sowie entweder einem Jahresumsatz von höchstens 50 Mio. € oder einer Jahresbilanzsumme von höchstens 43 Mio. € als KMU (EU 2003).

<sup>2</sup>Siehe hierzu Abschnitt „Betreute Studienarbeiten“.

## 1.1 Ausgangssituation und Problemstellung

---

Klein- und Mittelserienfertigung häufig erforderlich sind, weist dieser Prozess eine geringe Flexibilität auf: Prinzipbedingt müssen OS in VWF stets an ein spezielles Fördergut (selten: an eine sehr beschränkte Auswahl mehrerer Teile) angepasst werden (FELDMANN ET AL. 2014, S. 371; MATHIESEN 2017, S. 13).

Die Auslegung von OS ist geprägt von einer weitgehend manuellen Vorgehensweise und hat sich in den letzten Jahrzehnten kaum verändert (MATHIESEN 2017, S. 31; HESSE 2000b, S. 40). Der Auslegungsprozess gestaltet sich langwierig und erzielt oftmals nicht zufriedenstellende Ergebnisse, da das Vorgehen wenig systematisch vonstatten geht. Bislang werden OS auf Basis von Erfahrungswissen manuell in CAD<sup>3</sup> konstruiert, konventionell oder additiv gefertigt und anschließend experimentell angepasst. In komplexen Fällen erfolgt selbst die Konstruktion am Versuchsstand durch Fügen und Ausprobieren. Ein umfassender Vergleich von Lösungsalternativen ist i. d. R. aus Gründen der Wirtschaftlichkeit und Ressourcenintensität nicht möglich (HOFMANN 2014, S. 2, S. 20–23). Dadurch ist der Prozess zeit- und kostenintensiv und bewirkt im späteren Betrieb oftmals signifikante Prozessstörungen aufgrund der suboptimalen Auslegung (HOFMANN & STIRNER 2017). Gespräche mit Fachpersonal belegen Entwicklungszeiten von bis zu vier Monaten und meist tägliche Störungen im Betrieb (vgl. KRAUEL 2020). Bisherige Ansätze, diesen Prozess durch passende Softwarewerkzeuge zu unterstützen, lieferten keine in der Praxis anwendbaren Ergebnisse und sind deshalb gerade für KMU überhaupt nicht einsetzbar (z. B. BERKOWITZ & CANNY 1996; DALLINGER ET AL. 2012; M. JIANG ET AL. 2003).

Des Weiteren erfordert die Auslegung und Inbetriebnahme von OS viel Erfahrung und Expertenwissen (HESSE 2010, S. 195; MATHIESEN 2017, S. 31). Im Zuge des demografischen Wandels und dem damit einhergehenden Fachkräftemangel wird es immer schwieriger, junge Mitarbeitende für diese Art der Tätigkeit zu gewinnen, wie Aussagen aus der Praxis bestätigen. Durch die Abhängigkeit des Auslegungsergebnisses von personenspezifischem Fachwissen laufen VWF-Hersteller Gefahr, langfristig ihre Wettbewerbsfähigkeit einzubüßen, sobald das benötigte erfahrene Personal das Unternehmen verlässt (ABELE & REINHART 2011, S. 19–21; KRAUEL 2020).

---

<sup>3</sup>CAD: Computer Aided Design.

# 1 Einleitung

---

Zusammenfassend ist für produzierende Unternehmen in Deutschland (speziell KMU) langfristig ein wirtschaftlicher Betrieb nur mithilfe von Automatisierungslösungen möglich. Die Rationalisierung und Flexibilisierung der automatisierten Zuführtechnik stellt dabei einen großen Hebel zur Überwindung der bisherigen Hemmnisse dar. Für den wirtschaftlichen Einsatz zur automatisierten Zuführung in der Klein- und Mittelserienfertigung muss insb. die Auslegung von neuen OS schneller und effizienter werden (MATHIESEN 2017, S. 6 f.).

## 1.2 Zielsetzung

Auf Grundlage der in Abschnitt 1.1 beschriebenen Problemstellung ist das übergeordnete wissenschaftliche Ziel dieser Arbeit eine **durchgängige Methode zur automatisierten Generierung von OS für VWF**. Dies erfordert die Analyse bereits vorhandener Teilbausteine, deren Anpassung, Zusammenführung sowie Glättung und Schließung vorhandener Lücken der erarbeiteten durchgängigen Methode. Folgende Teilziele sollen damit erreicht werden:

**Verkürzte Entwicklungszeit für OS:** Die Methode soll aufgrund der durchgängigen Struktur, welche Brüche und somit Zeit- und Informationsverluste vermeidet, und der Nutzung von technischen Unterstützungswerkzeugen die Auslegungszeit für OS erheblich gegenüber dem manuellen Prozess verkürzen.

**Steigerung der Ergebnisqualität:** Durch die systematische Auslegung und die Möglichkeit, Lösungsalternativen zu vergleichen, soll die Qualität der OS im Vergleich zur manuellen Auslegung gesteigert werden. Es kann davon ausgegangen werden, dass die erhöhte Ergebnisqualität sich auch positiv in Form eines störungsreduzierten Betriebs der OS auswirkt.

**Digitale Wissensbasis:** Auslegungsergebnisse sollen gespeichert und wiederverwendet werden, um eine Wissensbasis zur langfristigen Nutzung des Systems aufzubauen. Zudem soll die Methode ohne manuelle Arbeitsschritte auskommen, um eine Entkopplung von personenspezifischem Wissen zu erreichen.

### 1.3 Spezifizierung des Betrachtungsbereichs

Damit die Analyse des Stands der Erkenntnisse zielgerichtet erfolgen und der Bereich potenziell gültiger Lösungen eingegrenzt werden kann, wird in diesem Abschnitt der Betrachtungsbereich näher spezifiziert. Die in dieser Arbeit entwickelte Methode zur automatisierten Generierung von OS nutzt als ein zentrales Werkzeug das Lernparadigma des Reinforcement Learning (RL)<sup>4</sup>, um Zusammenhänge zwischen der Gestalt von OS und dem Verhalten des zu orientierenden Fördergut-Teils zu erlernen. Im Folgenden soll die Wahl von RL als Werkzeug begründet werden.

LO & DICK (1990) stellen in ihrer Veröffentlichung zum Thema Modellierung von OS fest, dass die explizite Formulierung genau dieser Zusammenhänge zwischen der Gestalt von OS und dem Verhalten des zu orientierenden Fördergut-Teils für ein optimales OS-Auslegungsergebnis notwendig, bis dato aber unmöglich ist. Eine ungefähre Zuordnung kann mittels Erfahrung und Fachwissen erfolgen. Wie im Abschnitt 2.3 noch deutlich wird, fußen zudem zahlreiche Ansätze zur Unterstützung der Auslegung von OS auf regelbasierten (und damit stark simplifizierenden) digitalen Expertensystemen. Die Zielsetzung dieser Ansätze ist dabei klar auf die Schaffung eines lernfähigen Systems ausgerichtet. Ein solches konnte jedoch mit den zur jeweiligen Zeit der Veröffentlichung verfügbaren Werkzeugen nicht erreicht werden.

Zur Schaffung eines lernfähigen Systems eignet sich nach dem heutigen Stand der Erkenntnisse die Anwendung von RL besonders gut, weil diese Programme komplexe Zusammenhänge selbstständig erlernen und zur Problemlösung anwenden können. Dabei strebt ein sog. Agent ohne Vorwissen ein vorgegebenes Ziel an, indem er durch Versuch und Irrtum Zusammenhänge erlernt. Auf diese Weise kann ein RL-Agent bspw. lernen, wie die Gestalt von OS das Verhalten von Fördergut-Teilen auf der OS beeinflusst. Durch den Lernprozess baut der Agent Erfahrung und digitale Expertise auf, um nach bestimmten Vorgaben bzw. zu erreichender Zielsetzung OS optimal auszulegen. Da die vorliegende Forschungsarbeit explizit diesen lernenden Ansatz fokussiert, werden andere (nicht lernende) Ansätze wie bspw. evolutionäre Algorithmen oder Heuristiken

---

<sup>4</sup>Zu ausführlichen Erläuterungen s. Abschnitt 2.5.

## 1 Einleitung

---

nicht in Betracht gezogen, obwohl sie grundsätzlich zur Generierung von OS geeignet sind, wie z. B. EDWARDS (2004) und MATHIESEN & ELLEKILDE (2017) bereits gezeigt haben.

In der industriellen Praxis werden rein mechanische OS aufgrund des einfachen Aufbaus, der hohen Verschleißfestigkeit sowie geringen Systemkomplexität bevorzugt (HESSE 2000b, S. 87; HOFMANN 2014, S. 18–20). Für die vorliegende Arbeit sollen daher ausschließlich mechanische OS ohne zusätzliche Hilfsmittel wie Lichtschranken, Druckluftdüsen oder externe Kinematiken betrachtet werden.

Abschließend wird das Selbstverständnis der vorliegenden Arbeit betrachtet. Die wissenschaftliche Zielsetzung ist die Schaffung einer durchgängigen Methode zur automatisierten Generierung von OS für VWF, da eine solche bislang nicht existiert (vgl. Abschnitt 1.1). Diese im Rahmen dieser Arbeit entwickelte Methode versteht sich darüber hinaus als Handlungsanleitung für die Implementierung eines Software-Systems zur Auslegung von OS. Der Schwerpunkt der Arbeit liegt dabei auf den Elementen der Methode, die nach dem Stand der Erkenntnisse bisher noch nicht adressiert bzw. gelöst wurden. Da das Ziel einer durchgängigen Methode verfolgt wird, werden Teilschritte, für die bereits erfolgreich valide Ansätze vorgeschlagen wurden, aufgezeigt, jedoch nicht umfassend im Detail beschrieben und nicht erneut validiert. Zudem werden weitergehende organisationsbezogene Aspekte, wie z. B. unternehmensindividuelle Software- und Organisationsstrukturen, nicht betrachtet.

### 1.4 Wissenschaftstheoretische Einordnung

Zur Einordnung der Vorgehensweise und Ergebnisse dieser Arbeit werden diese im folgenden Abschnitt wissenschaftstheoretisch betrachtet. Dazu wird zunächst eine kurze Einführung in die zugrundeliegende Theorie gegeben und anschließend der Brückenschlag zur vorliegenden Forschungsarbeit vollzogen. Dies soll ermöglichen, die Forschungsergebnisse einzuordnen, zu beurteilen und die Grenzen der Arbeit zu verstehen.

Die Wissenschaftssystematik unterscheidet als zwei grundlegend verschiedene Kategorien zwischen den Formal- und Realwissenschaften (vgl. Abbildung 1.1).



## 1.4 Wissenschaftstheoretische Einordnung

Erstere, wie die Philosophie, Logik und Mathematik, befassen sich mit real nicht existierenden Objekten, weshalb sie nur auf ihre logische Wahrheit hin überprüfbar sind. Zweitere hingegen, bspw. die Natur- und Ingenieurwissenschaften, widmen sich der Beschreibung, Erklärung und Gestaltung der empirisch wahrnehmbaren Erfahrungswirklichkeit und müssen daher anhand ihrer faktischen Wahrheit überprüft werden. Ohne Formalwissenschaften kann es keine Realwissenschaften geben (P. ULRICH & W. HILL 1976, S. 305 f.; DÖRING & BORTZ 2016, S. 12 f.).

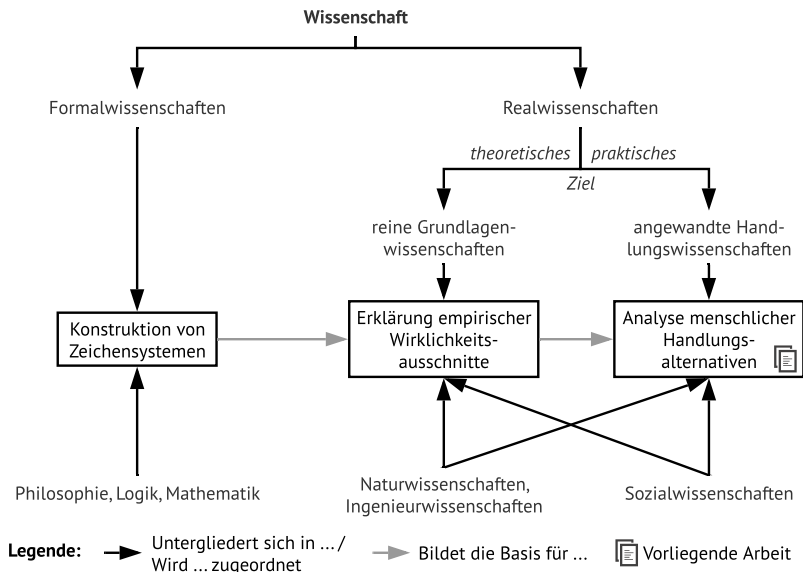


Abbildung 1.1: Wissenschaftssystematik nach P. ULRICH & W. HILL (1976, S. 305) und DÖRING & BORTZ (2016, S. 12 f.)

Innerhalb der Realwissenschaften ist weiterhin die Unterscheidung nach deren Zielsetzung möglich. Die reinen Grundlagenwissenschaften verfolgen ein theoretisches Ziel, nämlich die Bildung von Erklärungsmodellen für empirische Wirklichkeitsausschnitte und dadurch die Erweiterung des wissenschaftlichen Kenntnisstands. Denen gegenüber stehen die angewandten Handlungswissenschaften (auch: Anwendungsforschung), deren praktisches Ziel die Analyse menschlicher Handlungsalternativen zum Zweck der Entwicklung von Ent-

## 1 Einleitung

---

scheidungsmodellen darstellt. Diese sollen zur Gestaltung sozialer und technischer Systeme unter Beantwortung praktischer Fragestellungen dienen. Die Anwendungsforschung hat daher die elementare Aufgabe, mögliche Einsatzgebiete aufzuzeigen sowie ihre Ergebnisse anhand von praktischen Problemen zu evaluieren (P. ULRICH & W. HILL 1976, S. 305 f.; DÖRING & BORTZ 2016, S. 17–19).

Die Ingenieurwissenschaften können je nach Schwerpunkt beiden Ausprägungen der Realwissenschaften zugeordnet werden. Die vorliegende Forschungsarbeit versteht sich als Teil der Anwendungsforschung und legt daher besonderen Wert auf Anwendbarkeit und Validierung der Ergebnisse (s. Kapitel 7). Deshalb werden Probleme aus der Praxis betrachtet (s. Kapitel 2), mit dem Ziel, für den Anwender nutzbare Ergebnisse zu erzielen (s. Kapitel 3).

Zur besseren Einordnung der Vorgehensweise und Ergebnisse dieser Arbeit werden die zum Teil sehr unterschiedlichen Herangehensweisen der Grundlagen- bzw. Handlungswissenschaften näher beleuchtet und abgegrenzt (s. Tabelle 1.1). Insbesondere die Lösungsfindung für Probleme aus der Praxis und deren Bewertung anhand derselben bedingt innerhalb der Anwendungsforschung Vorgehensweisen, die sich von den Grundlagenwissenschaften deutlich abheben.

Nach P. ULRICH & W. HILL (1976, S. 306) unterliegt realwissenschaftliches Denken zwei Grundproblemen: Das Subjektivitätsproblem besagt, dass jegliche Wahrnehmung stets subjektiv ist, da ein Individuum die Wirklichkeit anhand erworbener Kenntnisse, Erfahrungen und Werte sowohl bewusst als auch unbewusst interpretiert (s. a. POPPER 2017, S. 221). Das Kommunikationsproblem rührt daher, dass die sprachliche Formulierung von Erkenntnissen stets Unschärfe erzeugt. Um diesen Problemen zu begegnen schlagen P. ULRICH & W. HILL (1976, S. 306 f.) die Berücksichtigung von drei Aspekten der Forschung vor, nämlich den *Entdeckungs-*, den *Begründungs-* und den *Verwendungszusammenhang* (s. dazu auch DÖRING & BORTZ 2016, S. 45 f.).

Der *Entdeckungszusammenhang* umfasst die Bildung von Theorien und beinhaltet den gedanklichen Bezugsrahmen für die erfolgreiche theoretische Konzeption einer Arbeit. Diese erfordert die Abgrenzung des Objektbereichs, die konkrete Problemstellung, die relevanten Grundbegriffe und Variablen zur Erfassung

## 1.4 Wissenschaftstheoretische Einordnung

Tabelle 1.1: Merkmale theoretischer und anwendungsorientierter Wissenschaften im Vergleich (H. ULRICH 1998, S. 163; TESCHEMACHER 2019, S. 11)

Merkmale	Grundlagenwissenschaften	Handlungswissenschaften
<b>Entstehung der Probleme</b>	in der Wissenschaft	in der Praxis
<b>Abgrenzung der Probleme</b>	durch Theoriezusammenhang	durch Praxiszusammenhang
<b>Forschungsziele</b>	Theorieentwicklung und -prüfung, Erklärung der bestehenden Wirklichkeit	Entwerfen möglicher Wirklichkeiten
<b>Angestrebte Aussagen</b>	deskriptiv, wertfrei	normativ, wertend
<b>Forschungsregulativ</b>	Wahrheit	Nützlichkeit
<b>Fortschrittskriterien</b>	Allgemeingültigkeit, Bestätigungsgrad, Erklärungskraft, Prognosekraft von Theorien	Praktische Problemlösungskraft von Modellen und Methoden
<b>Vorgehensweise zur Lösungsfindung</b>	Sequentiell: Erhebung, Auswertung, Interpretation	Zyklischer, iterativer Prozess: Problem und Ziele bestimmen, bestehende Ansätze analysieren, Handlungsplan aufstellen, Handlung realisieren, evaluieren und Handlungsplan ggf. modifizieren (vgl. BLESSING & CHAKRABARTI 2009, S. 15)

des Objektbereichs sowie die Aufstellung und Systematisierung der Arbeits-hypothesen. Um diesen Rahmen adäquat zu schaffen, werden in Kapitel 2 die bestehenden Herausforderungen analysiert und davon in Kapitel 3 die zu erfüllenden Anforderungen sowie die Zielstellung bzw. der Lösungsansatz abgeleitet. Auf höherer Ebene wurden Zielsetzung und Betrachtungsbereich bereits in Abschnitt 1.2 und 1.3 vorgestellt.

Der *Begründungszusammenhang* dient der Prüfung der zuvor gebildeten Theorien, unter welchen Bedingungen von singulären Beobachtungen auf allgemeingültige Zusammenhänge geschlossen werden kann. Deduktive Schlüsse, insb. die logische Kombination vorhandener (An-)Sätze, werden daher sowohl beim Aufstellen der Anforderungen (vgl. Kapitel 3) als auch bei der Evaluation der Forschungsergebnisse gezogen (vgl. Kapitel 7).

## 1 Einleitung

---

Zuletzt beantwortet der *Verwendungszusammenhang* die Frage nach der praktischen Anwendung der Theorien und damit der Relevanz der Forschungsergebnisse (s. a. POPPER 2017, S. 222). Die interessensgebundene Verwendung der wissenschaftlichen Aussagen wird daher in Kapitel 7 nach dem Nutzenkriterium bewertet, indem die in Kapitel 3 aufgestellten Anforderungen mit den Ergebnissen verglichen werden.

POPPER (2017, S. 222) fordert zudem als Antwort auf das Subjektivitätsproblem die kritische Diskussion von Forschungsergebnissen sowie deren Relevanz und ihrer Bedeutung für die behandelten Probleme. Dazu braucht es die wissenschaftsgemeinschaftliche Infragestellung über die Objektivitätsbemühungen des einzelnen Wissenschaftlers hinaus (POPPER 2017, S. 217). Das Wissenschaftssystem ermöglicht die Bewertung wissenschaftlicher Qualität durch unterschiedliche wissenschaftliche Anspruchsgruppen, wie z. B. Fachkolleginnen und -kollegen, welche die Qualität von eingereichten Veröffentlichungen im Zuge von Peer-Review-Verfahren bewerten (DÖRING & BORTZ 2016, S. 84). Daher wurden die zentralen Inhalte der vorliegenden Arbeit nach Durchlaufen von Peer-Review-Prozessen auf nationalen und internationalen Kongressen zur Diskussion gestellt sowie in Journals veröffentlicht<sup>5</sup>.

### 1.5 Methodische Vorgehensweise und Aufbau der Arbeit

Das Vorgehen des beschriebenen Forschungsvorhabens stützt sich auf die „Design Research Methodology (DRM)“ nach BLESSING & CHAKRABARTI (2009). Diese Forschungsmethodik ist eine der jüngsten durchgängigen ingenieurwissenschaftlichen Methoden, die in einer breiten Community wie der CIRP<sup>6</sup> anerkannt ist. Bereits in den 1990er Jahren im Kontext der CIRP International Design Conference sowie der International Conference on Engineering Design der internationalen Design Society entwickelt, wird die Methode seitdem aufgrund ihrer systematischen Struktur angewandt und weiterentwickelt (BLESSING ET AL. 1995; BLESSING & CHAKRABARTI 2009).

---

<sup>5</sup>STOCKER & REINHART 2015, 2016; STOCKER ET AL. 2017, 2019.

<sup>6</sup>CIRP: Collège International pour la Recherche en Productique.

## 1.5 Methodische Vorgehensweise und Aufbau der Arbeit

Die DRM besteht aus vier Stufen mit sog. Werkzeugen als deren Eingangs- und Ergebnissen als Ausgangsgrößen. Während des Durchlaufens der einzelnen Stufen werden frühere Schritte wiederholt, um Anpassungen vorzunehmen und neue Erkenntnisse zu berücksichtigen. Das Vorgehen in Anlehnung an die DRM spiegelt sich im Aufbau der vorliegenden Arbeit wider (vgl. Abbildung 1.2).

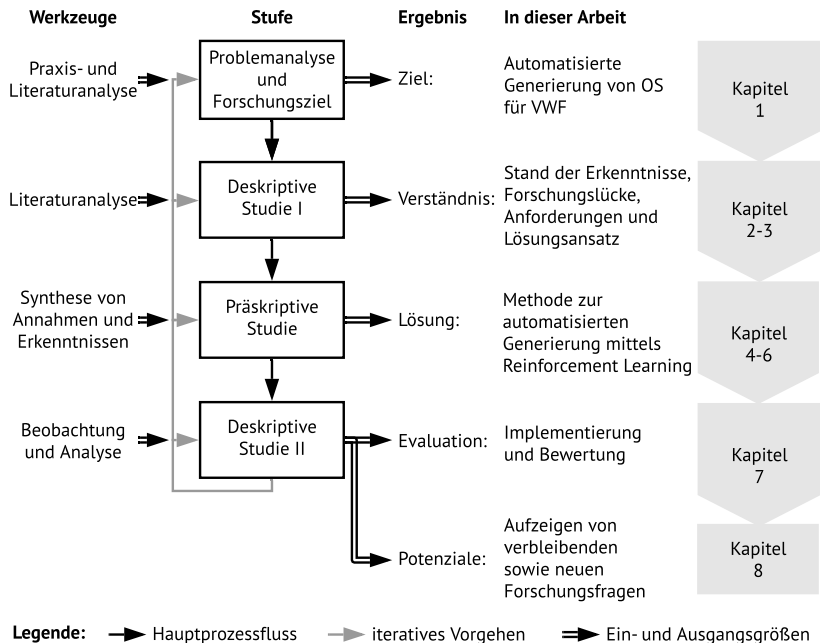


Abbildung 1.2: Dieser Arbeit zugrunde gelegte Forschungsmethodik in Anlehnung an BLESSING & CHAKRABARTI (2009, S. 15)

Im Rahmen dieser Arbeit wird zunächst eine *Problemanalyse* zur Auslegung von OS für VWF durchgeführt, die sich aus den Bestandteilen Praxis- und Literaturanalyse zusammensetzt. Ergebnis ist, dass diese ineffizient vonstatten geht und daher verbessert werden soll. Konkret sind der manuelle Aufwand und somit die dafür benötigte Zeit hoch und die Qualität der Ergebnisse oft mangelhaft. Ausgehend von dieser Beobachtung und einer ersten, breit gefächerten Literaturanalyse wird das *Forschungsziel* der automatisierten Generierung von OS für

## 1 Einleitung

---

VWF erarbeitet und der Betrachtungsrahmen abgesteckt. Die Ergebnisse der ersten Stufe in Anlehnung an die DRM sind in Kapitel 1 (Abschnitt 1.1 und 1.2) dokumentiert.

Ziel der *deskriptiven*<sup>7</sup> *Studie*<sup>8</sup> I ist das detaillierte Verständnis des Forschungsfeldes und der bestehenden Defizite. Dazu werden der aktuelle Stand der Erkenntnisse sowie existierende Lösungsansätze grundlegend analysiert und die Forschungslücke herausgearbeitet. Auf dieser Basis werden Anforderungen definiert und daraus der Lösungsansatz in Bezug auf die Defizite abgeleitet. In Kapitel 2 und 3 sind die Ergebnisse der *deskriptiven Studie I* dargelegt.

Die dritte Stufe der DRM, die *präskriptive*<sup>9</sup> *Studie*, zielt auf die Ausarbeitung einer konkreten Lösung für das zuvor formulierte Ziel und die beschriebenen Defizite ab. Daher wird durch Synthese der bisherigen Annahmen aus den ersten beiden Stufen die Methode zur automatisierten Generierung von OS mittels Reinforcement Learning ausgearbeitet. In der vorliegenden Arbeit sind das Vorgehen und die Ergebnisse der *präskriptiven Studie* in Kapitel 4 bis 6 aufbereitet.

In der vierten und letzten Stufe der DRM, der *deskriptiven Studie II*, wird die in Stufe drei erarbeitete Lösung evaluiert. Dazu wird die Methode aus dem vorherigen Schritt implementiert und durch Beobachtung und Analyse in Hinblick auf das Forschungsziel bewertet. Zudem erfolgt die Betrachtung der Gültigkeit sowie Wirtschaftlichkeit der erarbeiteten Methode. Die Evaluation ist in Kapitel 7 dokumentiert. Als Ausblick werden abschließend verbleibende sowie neue Potenziale und Forschungsfragen aufgezeigt (Kapitel 8).

---

<sup>7</sup>„Deskriptiv“: beschreibend (DUDENREDAKTION o. J.[a]). Eine deskriptive Aussage beschreibt Gegebenheiten aus der Realität oder aus diesen abgeleitete Merkmale (HELFRICH 2016, S. 40, S. 187).

<sup>8</sup>Der Begriff „Studie“ ist im DRM allgemein als wissenschaftliche Untersuchung anhand von wissenschaftlichen Methoden (z. B. systematische Befragung) zu verstehen. Diese umfasst nicht notwendigerweise die Erhebung und statistische Analyse von Daten, wie z. B. bei empirischen oder medizinischen Studien (DUDENREDAKTION o. J.[c]).

<sup>9</sup>„Präskriptiv“: Normen festlegend (DUDENREDAKTION o. J.[b]). Es werden Soll-Vorgaben erarbeitet, die vorschreibend wirken (HELFRICH 2016, S. 177).

## 2 Stand der Erkenntnisse

### 2.1 Begriffsdefinitionen

Zum besseren Verständnis der Zusammenhänge und zur einheitlichen Begriffsverwendung sind im Folgenden einige Fachbegriffe aus der Vibrationsförder-technik beschrieben<sup>1</sup>. Bei Auswahl der jeweiligen Definitionen wurde sorgfältig geprüft und verglichen, ob diese im Kontext der vorliegenden Arbeit gültig, sinnvoll und ausreichend differenziert sind. Insbesondere die Definitionen des VDI unterliegen vor ihrer Veröffentlichung einem umfassenden fachlichen Disput. Ebenso können Definitionen von Peers<sup>2</sup> aus dem Bereich der Zuführtechnik als geeignet für die vorliegende Arbeit gesehen und daher herangezogen werden.

**Freiheitsgrad:** Die Anzahl der Freiheitsgrade beschreibt die voneinander unabhängigen Bewegungsmöglichkeiten eines starren Körpers gegenüber einem Bezugskordinatensystem. Mit drei translatorischen (Verschiebungen) und drei rotatorischen (Drehungen) besitzt ein Körper maximal sechs Freiheitsgrade (VDI 1988, S. 2).

**Orientierung:** Die Orientierung eines Körpers beschreibt dessen Drehung in Relation zu einem Bezugskordinatensystem (VDI 1990, S. 2). Durch

---

<sup>1</sup>Hinweis: Die VDI-Richtlinie 2860 (Montage- und Handhabungstechnik; VDI 1990) wurde 2016 zurückgezogen (VDI 2016), da sie laut dem Verein Deutscher Ingenieure (VDI) nicht mehr dem Stand der Technik genügt. Zum Zeitpunkt der Erstellung dieser Arbeit war die Überarbeitung noch nicht abgeschlossen und für 2023 angekündigt (VDI 2019). Die Richtlinie zählt jedoch zu den wichtigsten Nachschlagewerken auf dem Themengebiet der Montage- und Handhabungstechnik. Daher wurde sie dennoch als Quelle herangezogen, insbesondere für Grundlagen, deren Gültigkeit nicht vom aktuellsten Stand der Technik abhängig ist. Selbiges gilt für die VDI-Richtlinie 2861, Blatt 1 (VDI 1988).

<sup>2</sup>Dt. etwa auf dem Gebiet führende Fachkollegen/Fachkolleginnen.

## 2 Stand der Erkenntnisse

---

Angaben bzgl. aller drei rotatorischen Freiheitsgrade ist die Orientierung vollständig bestimmt (HESSE 2010, S. 18).

**Position:** Die räumliche Lage eines Körpers in Relation zu einem Bezugskordinatensystem wird als dessen Position bezeichnet (VDI 1990, S. 2). Diese ist durch Angaben bzgl. aller drei translatorischen Freiheitsgrade bestimmt (HESSE 2010, S. 18).

**Lage:** Die Lage eines Körpers (bspw. eines Werkstücks) beschreibt dessen Orientierung und Position. Somit sind alle sechs Freiheitsgrade bestimmt. Häufig kommen Begriffszusammensetzungen wie Werkstück- oder Vorzugslage zum Einsatz (VDI 1990, S. 2).

**Ordnungszustand (OZ):** Dieser beschreibt, in wie vielen Freiheitsgraden die räumliche Anordnung eines Körpers bestimmt ist. Er setzt sich aus den Kennzahlen Orientierungsgrad ( $OG \in [0, 1, 2, 3]$ ) und Positionierungsgrad ( $PG \in [0, 1, 2, 3]$ ) zusammen, als Schreibweise ist definiert:  $OZ = OG/PG$ . Ein Körper mit  $OZ = 3/3$  ist völlig geordnet, mit  $OZ = 0/0$  völlig ungeordnet. Körper mit  $0/0 < OZ < 3/3$  bezeichnet man als teilgeordnet (VDI 1990, S. 3).

**Handhabung:** Die Schaffung, definierte Veränderung oder vorübergehende Aufrechterhaltung einer definierten Position und/oder Orientierung eines geometrisch bestimmten Körpers in einem Bezugskordinatensystem wird als Handhabung bezeichnet (VDI 1990, S. 2). Werkstückhandhabung im Speziellen beschreibt Materialflüsse im Bereich einzelner Arbeitsplätze bis hin zu ganzen Produktionsbereichen als Teil des innerbetrieblichen Transports (HESSE 2010, S. 2).

**Zuführung:** Die Teilmenge der Handhabungsvorgänge, die erforderlich sind, um Werkstücke von einer Schnittstelle des innerbetrieblichen Transports in die Wirkzone eines technologischen Vorgangs (bspw. eine Station/Anlage) zu bringen, bezeichnet man als Zuführung (HESSE 2000a, S. 154).

**Haufwerk:** Eine Ansammlung von Teilen, die in zufälligem, ungeordnetem Zustand vorliegen, wird als Haufwerk bezeichnet (HESSE 2010, S. 57).

**Ordnen:** Nach dem VDI (1990, S. 6) ist Ordnen das Bewegen von Körpern aus einer unbestimmten in eine bestimmte Orientierung und Position oder



Bewegungsrichtung. Man unterscheidet aktives Ordnen (auch: Gleichrichten), bei dem Teile in Falschlagen, meist mit mechanischen Mitteln, in die gewünschte Orientierung gebracht werden. Dagegen werden beim passiven Ordnen (auch: Auslesen) Teile in Falschlagen von den zufällig richtig orientierten getrennt, indem sie abgewiesen werden und zurück ins Haufwerk wandern (HESSE 2010, S. 150; LOTTER & WIENDAHL 2012, S. 252).

**Fördern:** Als Fördern wird das Bewegen von Körpern aus einer Position in eine beliebige andere definiert. Im Gegensatz zur Handhabung ist hierbei lediglich die Position des Körpers bestimmt, nicht dessen Orientierung (VDI 1990, S. 3, S. 6).

**Fördergut:** In dieser Arbeit werden als Fördergut Kleinteile im Haufwerk, sog. Stück- oder Schüttgut, verstanden (HESSE 2010, S. 193, S. 308).

**Orientierungswahrscheinlichkeit:** Diese bezeichnet die Häufigkeitsverteilung der stabilen Orientierungen, die ein Werkstück in Folge seiner Geometrie und Schwerpunktage einnimmt, wenn es nur unter Einfluss der Schwerkraft auf eine ebene Fläche fällt. Sie wird experimentell oder simulativ bestimmt. Dazu werden Fall-Versuche einer großen Anzahl an Werkstücken auf eine ebene Fläche durchgeführt und anschließend die resultierenden Orientierungen unter Berücksichtigung von Symmetrien<sup>3</sup> ausgezählt (WEISS 1983, S. 36; MATHIESEN & ELLEKILDE 2017).

**Vorzugsorientierung oder -lage:** Laut HESSE (2010, S. 41) und WEISS (1983, S. 36) ist dies die stabile Orientierung bzw. Lage, die ein Werkstück bei den Versuchen zur Orientierungswahrscheinlichkeit am häufigsten annimmt.

**Vibrationswendelförderer (VWF):** VWF zählen zu den elektromagnetischen Schwingförderern und dienen der automatisierten Zuführung von Kleinteilen aus dem Haufwerk (HESSE 2000b, S. 18).

**Ordnungsschikane (OS):** Elemente zur Orientierung und Vereinzelung des Förderguts entlang einer Förderstrecke im VWF werden in dieser Arbeit

---

<sup>3</sup>Zur Berücksichtigung von Symmetrien, bspw. der Rotationssymmetrie einer Schraube, sei auf detaillierte Erläuterungen in den Arbeiten von HOFMANN (2014, S. 65–67) und MATHIESEN & ELLEKILDE (2017) verwiesen.

## 2 Stand der Erkenntnisse

---

als Ordnungsschikane (OS) bezeichnet (HESSE 2000b, S. 39; LOY 2010, S. 6).

**Schikanenelement (SE):** Mehrere Schikanenelemente (SE) bilden, sequentiell und/oder parallel zusammengesetzt, eine OS (HESSE 2000b, S. 39; MATHIESEN 2017, S. 13). Häufige Beispiele hierfür sind (Kipp-)Stufen, Höhenabweiser oder Profilschienen (HESSE 2000b, S. 40–53).

**Zuführleistung:** Diese gibt an, wie viele Teile pro Minute richtig orientiert den VWF bzw. die OS verlassen (LOY 2010, S. 7).

**Förderleistung:** Im Gegensatz zur Zuführleistung beschreibt die Förderleistung die gesamte Anzahl der Teile pro Minute, die aus dem VWF bzw. der OS austreten, unabhängig von deren Orientierung (LOY 2010, S. 7).

**Mittlere Fördergeschwindigkeit  $\bar{v}$ :** Dies ist die durchschnittliche Geschwindigkeit, mit der Fördergut-Teile im VWF vorwärts bewegt werden (HESSE 2010, S. 186).

**Effizienz  $\epsilon$ :** Im Kontext eines VWF ist dies das Verhältnis der Anzahl an Fördergut-Teilen, die eine OS richtig orientiert verlassen, zur Gesamtanzahl der in die OS eingetretenen Teile (BOOTHROYD 2005, S. 58; HESSE 2010, S. 155). Diese Kennzahl wird auch als Wirkungsgrad der OS bezeichnet (HOFMANN 2014, S. 7).

### 2.2 Vibrationswendelförderer

Da die im Rahmen dieser Arbeit entwickelte Methode die Auslegung von OS für VWF fokussiert, wird zunächst genauer auf diese Handhabungstechnologie eingegangen.

Abschnitt 2.2.1 zeigt Bedeutung und Einsatzgebiete von VWF auf, gestützt von einer Praxisanalyse in Form einer strukturierten Umfrage, ergänzt durch ein Experteninterview. Zum grundlegenden Verständnis des technischen Hintergrunds werden in den nachfolgenden Abschnitten 2.2.2 und 2.2.3 zudem Aufbau und Funktionsweise von VWF sowie relevante Aspekte zu Fördergut und Ordnungsschikanen erläutert.

### 2.2.1 Produktionstechnische Bedeutung und Einsatzgebiete

VWF dienen in der automatisierten Produktion dazu, Kleinteile im Haufwerk in einen definierten Ordnungszustand zu bringen, um sie anschließend einer nachgelagerten automatisierten Montagestation zuzuführen (HESSE 2010, S. 149). Sie zählen zu den elektromechanischen Schwingförderern und werden unter diesen am häufigsten für diese Aufgabe eingesetzt (BOOTHROYD 2005, S. 29; SCHMID 2006, S. 23 f.). Trotz der bereits genannten Nachteile wie der geringen Anpassflexibilität und des hohen Entwicklungsaufwands (vgl. Abschnitt 1.1) sind VWF aus der automatisierten Montage nicht wegzudenken, da mit ihnen 80% der Kleinteile im Haufwerk theoretisch zuführbar sind (HESSE 2000b, S. 5). Für 70% bis 80% der geeigneten zu fördernden Teile werden auch tatsächlich VWF für die Zuführaufgabe eingesetzt, neben Lösungen wie Magazinen zur geordneten Bereitstellung oder Zuführsystemen mit externer Bildverarbeitung (ROCKLAND 1995, S. 30; SCHMID 2006, S. 23 f.).

Beispielanwendungen für VWF sind automatisierte Zufühhrlösungen für Kleinteile wie Schrauben, Muttern, Deckel, Stopfen oder Kontakte (HESSE 2000b, S. 9). Der Anwendungsbereich erstreckt sich dabei über die verschiedensten Branchen wie z. B. Automobil-, Pharma- und Konsumgüterindustrie sowie Sondermaschinen- und Anlagenbau (HESSE 2000b, S. 14; FELDMANN ET AL. 2014, S. 371). Prinzipiell hat der Einsatz von intelligenten Systemen, die unter dem Begriff „Griff in die Kiste“<sup>4</sup> bekannt sind, in den letzten Jahren zugenommen. Für die klassischen Anwendungsfälle von VWF sind die Spezifikationen jedoch aktuell immer noch ungenügend, insb. hinsichtlich der erreichbaren Taktzeiten und des Platzbedarfs im Vergleich zu VWF (IPA 2021).

Die weite Verbreitung von VWF lässt sich auf folgende Gründe zurückführen: VWF zählen zu den wartungsfreien Fördergeräten, da die darin enthaltenen Bauelemente (u. a. Federelemente, Elektromagnet) keine gleitende oder reibende Bauteile beinhalten und daher nur geringe Verschleißerscheinungen aufweisen. Durch ihren einfachen Aufbau (vgl. Abbildung 2.1) sind sie robust und vergleichsweise kostengünstig in der Anschaffung (HESSE 2000b, S. 18;

---

<sup>4</sup>Als „Griff in die Kiste“ werden Handhabungsvorgänge bezeichnet, bei denen eine Handhabungseinrichtung (z. B. ein Roboter) ein Teil mithilfe von Sensorik im ungeordneten Vorrat erkennt, selbstständig greift und definiert ablegt (HESSE 2010, S. 301).

## 2 Stand der Erkenntnisse

FELDMANN ET AL. 2014, S. 309, 360). Die Förderleistung lässt sich durch einfache elektrotechnische Bauteile regeln, und der Energieverbrauch von VWF ist gering, da sie im resonanznahen Bereich arbeiten (HESSE 2010, S. 186; LOTTER & WIENDAHL 2012, S. 246 f.).

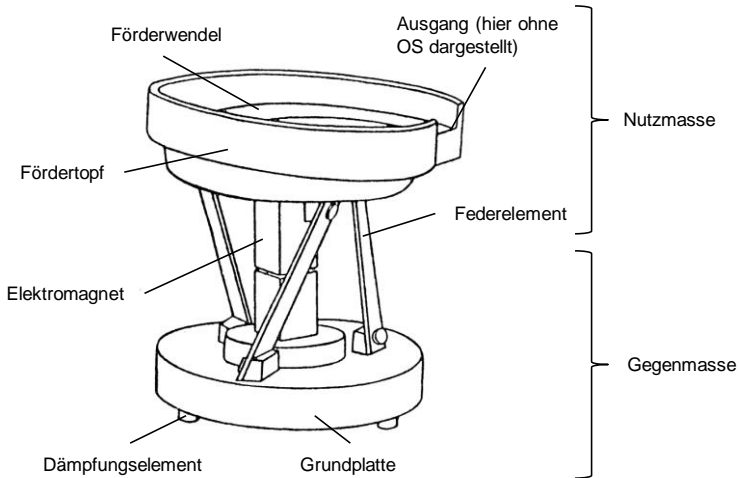


Abbildung 2.1: Schematischer Aufbau eines VWF (nach VDI 1990, S. 11)

Aufgrund der durch komplexe Wechselwirkungen gekennzeichneten Wirkweise ist der Betrieb von VWF, speziell der Ordnungsprozess, jedoch häufig störanfällig (s. hierzu Abschnitt 2.2.2). Laut FRÄDRICH & NYHUIS (2009, S. 5-4) sind VWF regelmäßig für Störungen in automatischen Montageanlagen verantwortlich.

Zudem verursacht die Zuführtechnik einen erheblichen Teil der Montageanlagenkosten (SCHMID 2006, S. 16; FRÄDRICH & NYHUIS 2009, S. 5-4), da die Entwicklung aufwändig und zeitintensiv ist (HESSE 2000b, S. 40). Dies liegt v. a. an der sehr geringen Umrüst- und Anpassflexibilität von VWF bzw. OS hinsichtlich verschiedener zuzuführender Teile, wodurch der aufwändige Entwicklungsprozess für jedes neue Fördergut erneut durchlaufen werden muss (vgl. hierzu Abschnitt 2.3; HESSE 2000b, S. 86; FELDMANN ET AL. 2014, S. 309).

Zur Überprüfung der Ergebnisse der Literaturanalyse zur produktionstechnischen Bedeutung von VWF wurde im Rahmen dieser Arbeit eine Praxisanalyse

angeschlossen. Hierzu wurde eine strukturierte Umfrage unter Zuführtechnikherstellern durchgeführt, ergänzt durch ein Experteninterview mit einem leitenden Angestellten eines großen Sondermaschinenbau-Unternehmens<sup>5</sup>.

Ziel der Umfrage war die Erhebung aktueller Hintergründe, Vorgehensweisen und Herausforderungen bei der Auslegung und Fertigung von VWF. Zur methodischen Erarbeitung des Fragebogens wurden anhand der Zielsetzung verschiedene Interessensgebiete wie Unternehmensstruktur und -größe, Auftragsabwicklung (z. B. Kundenstamm bzw. Branchen, Durchlaufzeiten und -kosten) und technische Aspekte (z. B. verwendete Materialien und Fertigungsprozesse, Technologien zur Unterstützung der Auslegung und Fertigung) definiert und anhand spezifischer Fragen abgebildet (KRAUEL 2020, S. 35–47). Mit einer Rückläuferquote von 25% kann die Umfrage als relativ erfolgreich eingestuft werden, wobei sie mit elf teilnehmenden Unternehmen als nicht uneingeschränkt repräsentativ zu bewerten ist. Auf Basis der Umfrageergebnisse wurde anschließend das Experteninterview konzipiert und durchgeführt, welches die Ergebnisse durchgehend bestätigen konnte (KRAUEL 2020, S. 53–68).

Im Folgenden werden die für diese Arbeit relevantesten Ergebnisse der Praxisanalyse zusammengefasst. Für die vollständigen Ergebnisse aus der Umfrage sowie des Experteninterviews sei auf die Arbeit von KRAUEL (2020) verwiesen.

Wie bereits eingangs ausgeführt (vgl. Abschnitt 1.1), sind die meisten Zuführtechnikhersteller den KMUs zuzurechnen. Zur Auslegung von VWF stützen die Hersteller sich hauptsächlich auf die Expertise ihrer Mitarbeitenden, technische Lösungen wie CAD-Systeme kommen nur zur Fertigungsvorbereitung zum Einsatz (z. B. zur Erstellung der zuvor manuell ausgelegten OS zur spanenden oder additiven Fertigung). Simulationsprogramme werden nicht genutzt, da auf dem Markt keine entsprechend angepassten Lösungen verfügbar sind. Laut Umfrageergebnis wären solche Programme zur daten- und zahlenbasierten Unterstützung der Auslegung allerdings wünschenswert. Die Erprobung und Anpassung von OS erfolgt stattdessen durch praktische Förderversuche und iterative Optimierung durch manuelle Anpassung und kann sich langwierig

---

<sup>5</sup>Diese Umfrage und Experteninterview sind im Rahmen einer Studienarbeit entstanden (KRAUEL 2020, siehe hierzu Abschnitt „Betreute Studienarbeiten“).

## 2 Stand der Erkenntnisse

---

gestalten. Hierbei können Durchlaufzeiten von bis zu drei Monaten entstehen. OS werden weitgehend bauteilspezifisch angefertigt, flexible oder umrüstbare Systeme sind nicht verbreitet.

Die Praxisanalyse bestätigt zudem den Einsatz von VWF als bevorzugte Zuführlösung für Kleinteile im Automotive- und Sondermaschinenbereich, insb. hinsichtlich der erreichbaren Förderleistung und Robustheit der Systeme im praktischen Einsatz.

### 2.2.2 Aufbau und Funktionsweise

In Abbildung 2.1 ist der schematische Aufbau eines VWF dargestellt. VWF können als periodisch erregte Feder-Masse-Systeme beschrieben werden, die nach den Gesetzen der Schwingungsmechanik arbeiten. Auf einer Grundplatte sind mehrere schräg gestellte Blattfedern befestigt, welche diese mit dem Fördertopf des VWF verbinden (HESSE 2010, S. 186–188; LOTTER & WIENDAHL 2012, S. 246–248). Die Federn bestehen bspw. aus mehreren übereinander gelegten Stahlblechen, oder aus einem Federelement aus Faserverbundwerkstoff (LOY 2010, S. 50). Die Grundplatte ist von der Umgebung über mehrere Dämpfungselemente aus Gummi entkoppelt, was essentiell für die störungsfreie Funktion des VWF ist (HESSE 2000b, S. 21). Den oberen Teil, bestehend aus Fördertopf und Teilen der Federbeine sowie des Elektromagneten, bezeichnet man auch als Nutzmasse. Die unteren Teile des Magneten und der Federn sowie die Grundplatte zählen zur sogenannten Gegenmasse (BOOTHROYD 2005, S. 29; HESSE 2010, S. 186–188). Der spezifische Aufbau von VWF ist in verschiedenen Bauarten möglich (HESSE 2000b, S. 23, 2010, S. 186).

Durch einen oder mehrere Elektromagnete wird das System zum Schwingen angeregt, was durch die schräg gerichtete Rückstellkraft der Federn in einer helixförmigen Oszillation des Fördertopfs resultiert. Durch die Ansteuerung mit Wechselstrom schwingen VWF mit 100 bzw. 120 Hz, entsprechend der doppelten Netzfrequenz (BOOTHROYD 2005, S. 29; FELDMANN ET AL. 2014, S. 357). Die Schwingfrequenz lässt sich durch die Verwendung entsprechender Steuergeräte allerdings auch variieren. Durch Spannungsveränderung wird die Größe der Schwingungsamplitude angepasst, die sich auf die Antriebskraft und somit die Laufgeschwindigkeit des Förderguts auswirkt (HESSE 2000b, S. 21).

Fördertöpfe existieren in verschiedenen Formen, so z. B. zylindrisch, stufen- oder kegelförmig (FELDMANN ET AL. 2014, S. 364). Ihnen gemein ist die an der Innenfläche entlang schräg nach oben verlaufende Förderwendel, die am oberen Ende des Fördertopfs in den Ausgang mündet. Dort sind OS angebracht, die das Fördergut vereinzeln und in die gewünschte Orientierung bringen (vgl. Abbildung 2.2; HESSE 2000b, S. 25, 2010, S. 6). OS werden in Abschnitt 2.2.3 vertieft behandelt.

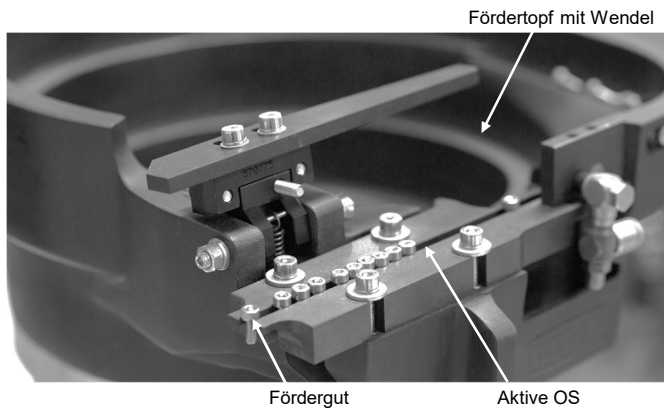


Abbildung 2.2: VWF aus der Praxis mit aktiver OS (geschlitzte Bahn) und Schrauben als Fördergut (vgl. Abschnitt 2.2.3)

Innerhalb des VWF werden die handhabungstechnischen Funktionen „Fördern“ und „Ordnen“ vereint, da die Werkstücke aus dem Haufwerk vereinzelt und orientiert werden. Durch die Kombination mit den Funktionen „Bevorraten“ und „Weitergeben“ kann eine weitgehende Autonomie von menschlichem Eingriff über längere Zeiträume erreicht werden. Hierzu werden Bunkersysteme zur Befüllung des VWF vorgelagert und Fördersysteme zum Weitergeben in die Montagestation nachgeschaltet (HESSE 2000b, S. 23, 2010, S. 192 f.).

Ist der Fördertopf mit den zu vereinzeln den Werkstücken befüllt, werden diese die Wendel nach oben gefördert. Die Förderung der Werkstücke erfolgt zumeist nach dem Prinzip des sog. Mikrowurfs (BOOTHROYD 2005, S. 38–42). Über die Beschleunigung durch den Fördertopf heben die Werkstücke von der Oberfläche ab und werden aufgrund der Schrägstellung der Federn ein Stück weit in

Förderrichtung geworfen<sup>6</sup> (HESSE 2010, S. 186 f.). Nach der durch den Wurf ausgelösten Flugphase prallen die Teile wieder auf die Oberfläche, wodurch eine erhebliche Geräuschemission und mechanische Belastung der Förderwendel sowie der Werkstücke entstehen kann (HESSE 2000b, S. 81; FELDMANN ET AL. 2014, S. 359). Durch dieses Förderprinzip sind prinzipiell Fördergeschwindigkeiten von bis zu 100 m/min sowie Förderleistungen von über 400 Teilen/min realisierbar (HESSE 2000b, S. 22; HOFMANN 2014, S. 125).

Bei Förderung nach dem sog. Gleitprinzip werden die Fördergut-Teile weniger stark beschleunigt und heben dadurch nicht von der Oberfläche ab. Stattdessen gleiten sie vorwärts, wodurch die Lärmentwicklung deutlich geringer ist, dadurch verringert sich jedoch die erreichbare Fördergeschwindigkeit respektive -leistung (HESSE 2010, S. 187; LOTTER & WIENDAHL 2012, S. 248).

Durch die hochgradig nicht-linearen Eigenschaften des Schwingungssystems ist die Berechnung des dynamischen Verhaltens von VWF und der Interaktion mit dem Fördergut äußerst komplex und im Betrieb häufig instabil (FELDMANN ET AL. 2014, S. 360). Die Schwingungscharakteristik ist außerdem stark vom aktuellen Füllgrad des Fördertopfes abhängig, der die Eigenfrequenz des Systems merklich beeinflusst (BOOTHROYD 2005, S. 42). Wie in Abschnitt 2.3 weiter ausgeführt wird, sind diese Systemcharakteristika zentrale Ursachen für die Langwierigkeit und Komplexität der Auslegung von VWF und OS, die daher umfangreiches Erfahrungswissen erfordert (FELDMANN ET AL. 2014, S. 309).

### 2.2.3 Fördergut und Ordnungsschikanen

#### 2.2.3.1 Fördergut

So breit wie die Anwendungsfelder von VWF, so breit ist auch das Teilespektrum möglichen Förderguts. Allgemein wird im Haufwerk angeliefertes, stückiges Arbeitsgut zugeführt, wie bspw. Schrauben, Muttern, Scheiben, Gewindestifte, Schaltkontakte, Gehäuseteile oder Kappen (HESSE 2000b, S. 14). Prinzipiell ist auch das Zuführen von Schüttgut und Pulvern möglich (HESSE 2000b, S. 21).

---

<sup>6</sup>Gängige Sprungweiten liegen im Bereich von 0,1 bis 0,3 mm (HESSE 2010, S. 187).



Für die vorliegende Arbeit wird dieses jedoch nicht betrachtet, da Vereinzeln und Umorientieren nur für Stückgut relevant ist.

Geeignete Fördergut-Teile weisen maximale Abmessungen von ca. 150 mm Länge, 50 mm Durchmesser sowie eine maximale Masse von ca. 0,5 kg auf (HESSE 2010, S. 193). Zu fördernde Werkstücke müssen prinzipiell robust sein, da die mechanische Belastung während des Förderprozesses bei spröden Werkstoffen zu Beschädigungen führen kann (HESSE 2010, S. 187, S. 190). Für die Zuführtechnik sind v. a. geometrische Formen und das damit verbundene Werkstückverhalten von Bedeutung, da dieses die Schwierigkeit der Handhabung maßgeblich bestimmt (HESSE 2010, S. 14). Es werden verschiedene Verhaltens- oder Grundtypen von Werkstücken unterschieden, wie z. B. Pilz-, Flach- oder Wirtteile. Diese werden genutzt, um geeignete Handhabungsgeräte auszuwählen, zu denen auch OS im VWF zählen (HESSE 2010, S. 16 f., S. 25). Beim Zuführen, Vereinzeln und Ordnen im VWF werden Werkstückbewegungen unter Nutzung des Schwerkrafteinflusses ausgenutzt (HESSE 2010, S. 14).

Dazu muss das Werkstück eine oder mehrere Vorzugslagen aufweisen, was das Vorhandensein geeigneter Merkmale voraussetzt (HESSE 2010, S. 150; BOOTHROYD 2005, S. 69–78). Zur formalisierten Darstellung der Orientierungswahrscheinlichkeiten von Fördergut wird nach BOOTHROYD (2005, S. 55) der Vorzugslagen-Vektor  $\vec{v}$  angegeben:

$$\vec{v} = [p(O_1) \quad p(O_2) \quad \dots \quad p(O_n)] \quad \forall \quad O \in \mathcal{O} \quad (2.1)$$

$O_i$  ist hierbei eine Orientierung aus der Menge aller möglichen Orientierungen  $\mathcal{O}$ . Zur Bestimmung des Vektors wird eine große Anzahl an Fördergut-Teilen aus einer bestimmten Höhe auf eine ebene Fläche fallen gelassen. Anschließend wird die Orientierung  $O_i$  erfasst, in der die Teile zur Ruhe kommen (vgl. Abbildung 2.3). Die Einträge des Vektors geben die prozentuale Verteilung der End-Orientierungen an. Zur Vereinfachung werden diese diskreten Orientierungen ohne Zwischenwerte unterschieden, was ein bewährtes Vorgehen darstellt (BOOTHROYD 2005, S. 55 f.).

Auch die Zielorientierung des Werkstücks lässt sich analog als Vektor  $\vec{z}$  darstellen. Als Zielorientierung eignen sich bevorzugt solche Orientierungen, die mit

## 2 Stand der Erkenntnisse

einer hohen Wahrscheinlichkeit auftreten oder erzielt werden können (HESSE 2000b, S. 39). Für die Zielorientierung  $O_1$  ist  $\vec{z}$  beispielsweise:

$$\vec{z} = [1 \quad 0 \quad \dots \quad 0] \quad (2.2)$$

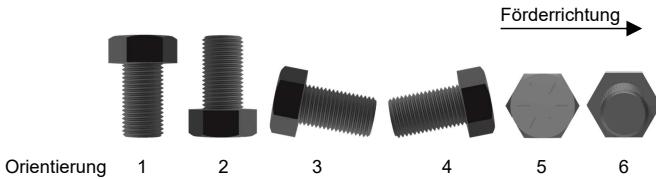


Abbildung 2.3: Beispiel: Sechs mögliche, diskrete Orientierungen einer Schraube, in Bezug auf Förderrichtung nach rechts (vgl. BOOTHROYD 2005, S. 430; LA BROOY & C. JIANG 2009)

### 2.2.3.2 Ordnungsschikanen

OS im VWF haben die Aufgabe, aus dem Fördertopf nach oben bewegtes Fördergut zu vereinzeln und in die Zielorientierung zu bringen (HESSE 2000b, S. 39). Diese ist durch den nachfolgenden Prozess vorgegeben, Beispiele hierfür sind automatische Schraubstationen oder Bestückroboter (HESSE 2000b, S. 12). Abbildung 2.4 zeigt beispielhaft eine OS, die sich aus mehreren SE zusammensetzt.

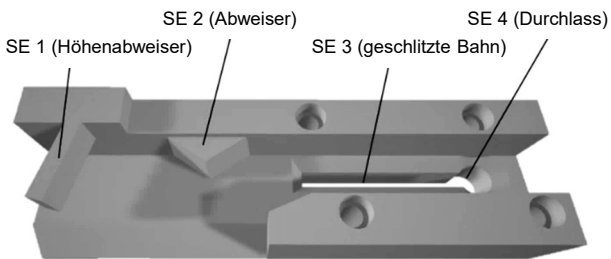


Abbildung 2.4: Beispiel für eine OS, zusammengesetzt aus mehreren SE

OS bestehen aus mechanisch wirkenden Elementen wie z. B. Höhenabweisern, Fallöffnungen, Kippstufen oder Profilschienen, deren Parameter (z. B. Stufenhöhe, Schienenbreite) auf das zu orientierende Fördergut abgestimmt werden (HESSE 2000b, S. 40–53).

Zudem existieren aerodynamische SE, die vorbeilaufendes Fördergut durch Druckluft-Impulse umorientieren oder aussortieren (FRÄDRICH & NYHUIS 2009). Innovative Ansätze optimieren die Druckluftströme mithilfe verschiedener Optimierungsalgorithmen (BLANKEMEYER ET AL. 2022; BUSCH 2016; BUSCH ET AL. 2015; KOLDITZ 2020; KOLDITZ & RAATZ 2022; KOLDITZ ET AL. 2021). Manche Ordnungslösungen werden durch Bildverarbeitungssysteme ergänzt, um die Orientierung der Teile zu erfassen und z. B. eine nachgelagerte Luftdüse entsprechend anzusteuern (LOTTER & WIENDAHL 2012, S. 249; HAUGAARD ET AL. 2022; KOLDITZ ET AL. 2022; MALIK ET AL. 2019). Für sehr spezielle Anwendungsfälle in der Mikromontage, die besondere Herausforderungen wie Adhäsionskräfte und extrem empfindliche Bauteile bewältigen müssen, existieren Fördersysteme unter Ausnutzung der Oberflächenspannung von Fluiden, wobei hier zumeist Pick&Place-Ansätze mit Sauggreifern zum Einsatz kommen (BORCHERT 2018; BURGARD ET AL. 2014). Solche Pick&Place-Anwendungen können zur Erhöhung der Zuverlässigkeit mit 3D-Bildverarbeitungslösungen zur Ermittlung der Werkstücklage im Haufwerk unterstützt werden (EFFENBERGER ET AL. 2013; LEDERMANN 2012; PALZKILL 2014; STOTZ 2011). Systeme mit angesteuerten Komponenten sowie Sensorik sind allerdings störanfällig und verursachen höhere Kosten als solche ohne (HESSE 2010, S. 162 f.).

In der industriellen Praxis werden daher aufgrund des einfacheren Aufbaus, der hohen Verschleißfestigkeit sowie geringeren Systemkomplexität mechanische OS bevorzugt eingesetzt (HESSE 2000b, S. 87; HOFMANN 2014, S. 18–20). Daher werden in dieser Arbeit ausschließlich mechanische OS behandelt (vgl. Abschnitt 1.3).

Nach dem Ordnungsprinzip werden passive und aktive SE unterschieden (vgl. Abbildung 2.5). Erstere arbeiten nach der Regel Ordnen durch Auslesen. Dabei durchlaufen nur Teile das SE, die bereits zufällig richtig orientiert sind. Falsch orientierte Fördergut-Teile werden abgewiesen und fallen zurück in den Fördertopf.

## 2 Stand der Erkenntnisse

---

Im Gegensatz dazu sind aktive SE derart konzipiert, dass sie falsch orientiertes Fördergut durch mechanische Wechselwirkung in die gewünschte Orientierung bringen, sog. Ordnen durch Gleichrichten (BOOTHROYD 2005, S. 47 f.; HESSE 2010, S. 150, S. 193). Dazu werden die Werkstückform, also Merkmale der Außenkontur, und asymmetrische Schwerpunktlagen gezielt genutzt (HESSE 2010, S. 196). Aktive SE ermöglichen durch ihr Funktionsprinzip höhere Zuführleistungen, da falsch orientierte Teile umorientiert statt in den Förderkopf zurückgeworfen werden, sind dadurch bauteilschonender und daher den passiven vorzuziehen (BOOTHROYD 2005, S. 54).

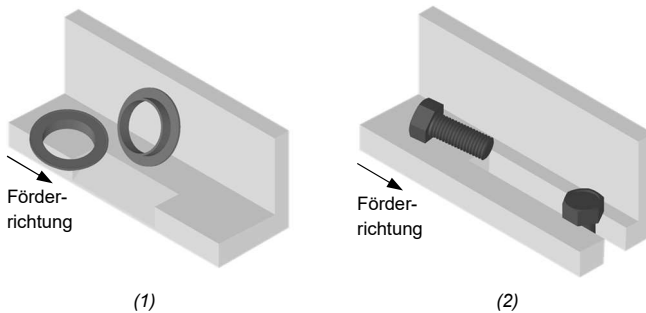


Abbildung 2.5: Wirkweisen von Schikanenelementen: (1) passiv, (2) aktiv

### 2.2.3.3 Effizienz von Ordnungsschikanen

Die Wirksamkeit von OS wird anhand der Zielgrößen Effizienz und Zuführleistung bewertet (vgl. Abschnitt 2.1; BOOTHROYD 2005, S. 58). Ein häufig verwendetes Werkzeug zur differenzierten quantitativen Beschreibung von SE bzw. OS sind sog. Transitionsmatrizen<sup>7</sup>. Dabei wird jedes SE durch eine Matrix repräsentiert, deren Zeilen und Spalten jeweils mit einer der Orientierungen des Fördergut-Teils korrespondieren. Die Elemente der Matrix geben eine statistische Verteilung an, welcher prozentuale Anteil an eintretenden Fördergut-Teilen das SE in den jeweiligen Orientierungen verlässt. Dabei ist implizit repräsen-

---

<sup>7</sup>Auch: Übergangsmatrizen.

tiert, welcher Anteil an Fördergut-Teilen, die in einer bestimmten Orientierung ankommen, durch das SE abgewiesen wird, da in diesem Fall die Summe der entsprechenden Matrix-Zeile kleiner als eins ist (vgl. Abbildung 2.6). Durch die Multiplikation mehrerer SE-Matrizen wird die Wirkung einer zusammengesetzten OS berechnet. Wird die resultierende Matrix mit den Vektoren der Vorzugslagen des Werkstücks und der Zielorientierung multipliziert, ergibt sich ein Skalar, der die Effizienz der Anordnung beschreibt (vgl. Gleichung 2.3; BOOTHROYD 2005, S. 59–61, S. 78–85).

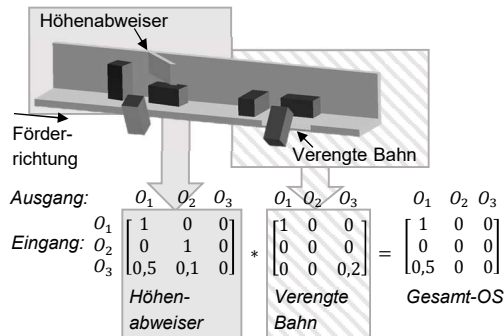


Abbildung 2.6: Beispielhafte OS zur Erläuterung des Prinzips der Transitionsmatrizen (vgl. BOOTHROYD 2005, S. 59 f.)

Dieses Konzept wird im Folgenden anhand eines Beispiels näher erläutert. Der Quader in Abbildung 2.7 kann unter Berücksichtigung seiner Symmetrieeigenschaften in drei verschiedenen Orientierungen  $O_1$ ,  $O_2$  und  $O_3$  vorliegen, wobei  $O_1$  die Zielorientierung sein soll. Die natürlichen Ruhe- oder auch Vorzugslagen drücken sich anhand des Vektors der Eingangsorientierungen  $\vec{v}$  aus. Dieser besagt, dass nach den Fall-Versuchen jeweils 40% in Orientierung  $O_1$  und  $O_2$  sowie 20% in  $O_3$  verblieben sind.

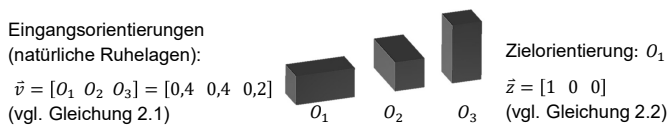


Abbildung 2.7: Darstellung von Werkstück-Orientierungen (angelehnt an BOOTHROYD 2005, S. 59 f.)

## 2 Stand der Erkenntnisse

---

Wie Abbildung 2.6 zu sehen ist, kommt als OS eine Kombination aus Höhenabweiser und verengter Bahn zum Einsatz, deren Matrizen beispielhaft angegeben sind. Teile, die in Orientierung  $O_3$  zum Höhenabweiser gefördert werden, treten im Experiment zu 50% in  $O_1$  und zu 10% in  $O_2$  aus, da sie umorientiert werden. 40% der Teile fallen in den Topf zurück, die Orientierung  $O_3$  kommt nach dem Höhenabweiser nicht mehr vor. Die verengte Bahn lässt Teile in Orientierung  $O_1$  passieren,  $O_2$  wird aussortiert und in  $O_3$  bleiben 20% der Teile übrig. Durch Multiplikation der Transitionsmatrizen beider SE kann die Gesamt-Matrix der OS errechnet werden, die nur noch Teile in  $O_1$  ausgibt.

Daraus lässt sich die Effizienz  $\epsilon$  der Anordnung unter gegebener Verteilung der Eingangsorientierungen  $\vec{v}$  sowie angestrebter Zielorientierung  $\vec{z}$  zu  $\epsilon = 0,5$  berechnen (vgl. Gleichung 2.3). Die Hälfte der Teile, welche die OS durchlaufen, sind demnach anschließend richtig orientiert.

$$\epsilon = \begin{bmatrix} 0,4 & 0,4 & 0,2 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0,5 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T = 0,5$$

Effizienz	Eingangs-	Gesamt-OS	Ziel-
der OS	orientierungen		orientierung

(2.3)

Im Betrieb treten häufig Störungen an OS auf, da sich Fördergut z. B. verklemmen kann oder eine gewünschte Umorientierung nicht wie geplant vonstatten geht (BOOTHROYD 2005, S. 49 f.). Die Ursachen hierfür sind vielfältig. So können Verschleiß oder Verschmutzung von VWF und OS durch den laufenden Betrieb, unbemerkte Veränderung der Fördergut-Teile (z. B. geändertes Maß oder Material, dadurch veränderter Reibungskoeffizient) oder auch Fehlbedienung (bspw. nicht optimales Schwingungsverhalten durch überfüllten Fördertopf) den Förderprozess beeinflussen oder sogar zum Erliegen bringen.

Besonders häufig ist die nicht optimale Auslegung der OS Ursache für Störungen im Betrieb, was auf den Auslegungsprozess zurückzuführen ist (HESSE 2000b, S. 14, 2010, S. 355). Das Vorgehen ist stark von manueller Arbeit geprägt und von Erfahrungswissen abhängig. Daher ist der Prozess langwierig,

kostenintensiv und das Ergebnis meist nicht zufriedenstellend (FELDMANN ET AL. 2014, S. 309; HOFMANN 2014, S. 20–23). Im folgenden Abschnitt 2.3 wird ausführlich auf diese Problematik eingegangen.

### 2.3 Auslegung von Ordnungsschikanen

Die vorliegende Arbeit hat zum Ziel, eine durchgängige Methode zur automatisierten Auslegung von OS für VWF zu entwickeln. Zum besseren Verständnis und zur Herleitung der Defizite werden im Folgenden bestehende Auslegungsansätze in Praxis und Wissenschaft analysiert.

Zunächst wird in Abschnitt 2.3.1 der konventionelle Auslegungsprozess für OS detailliert beschrieben. Es wird deutlich, dass dieser zwar hinsichtlich des Ablaufs und seiner Teilschritte grundsätzlich valide ist, aus verschiedenen Gründen jedoch methodischer Unterstützung bedarf.

Dazu finden sich im Stand der Forschung drei grundlegend unterschiedliche Herangehensweisen, jeweils Teile des Prozesses zu verbessern: Die erste fokussiert analytische Untersuchungen und hergeleitete Zusammenhänge zur methodischen Unterstützung der Auslegung von OS (Abschnitt 2.3.2). Das simulationsgestützte Vorgehen als zweite Herangehensweise nutzt die computerbasierte Abbildung des Förderprozesses (Abschnitt 2.3.3). Als drittes Vorgehen wird bei der automatisierten Sequenzierung die OS als Kombination von SE durch einen Algorithmus errechnet, wobei analytische und simulative Methoden als Werkzeug fungieren (Abschnitt 2.3.4). Diese drei Herangehensweisen werden nachfolgend detailliert erläutert.

#### 2.3.1 Konventionelle Auslegung

Nach WOLFSTEINER (1999, S. 1) und HOFMANN (2014, S. 20) verfolgt der konventionelle Ansatz zur Auslegung von OS eine Versuch-und-Irrtum-Strategie zur iterativen Anpassung auf Basis von Experimenten und Erfahrungswissen. Dabei erfordern Einbau, Inbetriebnahme und Funktionssicherung von OS viel Expertise und sind zeitaufwändig (HESSE 2000b, S. 40, 2010, S. 195). Dies bestätigt auch MATHIESEN (2017, S. 13, S. 31) auf Basis von aktuellen Umfragen. Eine

## 2 Stand der Erkenntnisse

---

eigens im Rahmen dieser Arbeit durchgeführte Umfrage zeigt ein ähnliches Bild unter deutschen Zuführtechnik-Herstellern (KRAUEL 2020).

Der konventionelle Auslegungsprozess ist in Abbildung 2.8 skizziert. Dieser Prozess kann als grundsätzlich sinnvolles Vorgehen angesehen werden. Dies zeigt sich daran, dass die später erklärten Methoden zu dessen Unterstützung (s. Abschnitte 2.3.2 bis 2.3.4) darauf aufbauen und jeweils einzelne Teilschritte optimieren, ohne den Prozess als Ganzes neu zu gestalten (s. Ansatzpunkte zur Unterstützung in Abbildung 2.8).

Ausgangspunkt für die Auslegung von OS ist ein zu orientierendes Fördergut und dazugehörige Anforderungen wie z. B. die Zielorientierung oder die erforderliche Zuführleistung. Der Zuführtechnikhersteller erhält diese Informationen sowie Musterteile des Förderguts vom Auftraggeber. Der iterative Prozess beginnt mit einer empirischen Identifikation der Fördergut-Charakteristika, wie z. B. Vorzugslagen anhand von Fall-Versuchen oder geometrische Merkmale mittels Sichtprüfung (HOFMANN 2014, S. 21 f.). Darauf folgt eine erfahrungsbasierte Auswahl und Zusammenstellung von SE zu möglichen Lösungen für die OS (ROCKLAND 1995, S. 26). Diese werden als Prototyp konstruiert und additiv oder spanend gefertigt (KRAUEL 2020, S. 92). Um die Funktion der OS zu überprüfen wird ein Fördertopf mit Teilen gefüllt, um damit zahlreiche Testläufe durchzuführen. Erfahrenes Entwicklungspersonal optimiert dabei schrittweise und in Handarbeit die OS, um deren Wirksamkeit zu erhöhen (MATHIESEN 2017, S. 31). Erfüllt ein Entwurf die Anforderungen, ist der Designprozess beendet, andernfalls muss der Prozess von vorne beginnen (HOFMANN 2014, S. 22 f.).

Im Allgemeinen kann mit dieser Herangehensweise keine optimale Lösung erzielt werden. Auch ist aufgrund des hohen Material- und Zeit-Aufwands für Entwurf und Fertigung der Vergleich von Lösungsalternativen nur sehr eingeschränkt möglich (HOFMANN 2014, S. 20–23; MATHIESEN 2017, S. 31). Die Entwicklungszeiten können in Extremfällen bis zu drei Monate in Anspruch nehmen (KRAUEL 2020). Zudem streuen die Anschaffungskosten für individualisierte VWF durch das unstrukturierte Vorgehen stark, möglich sind Kaufpreise von 2.000 € bis 15.000 € oder mehr (MATHIESEN 2017, S. 13).



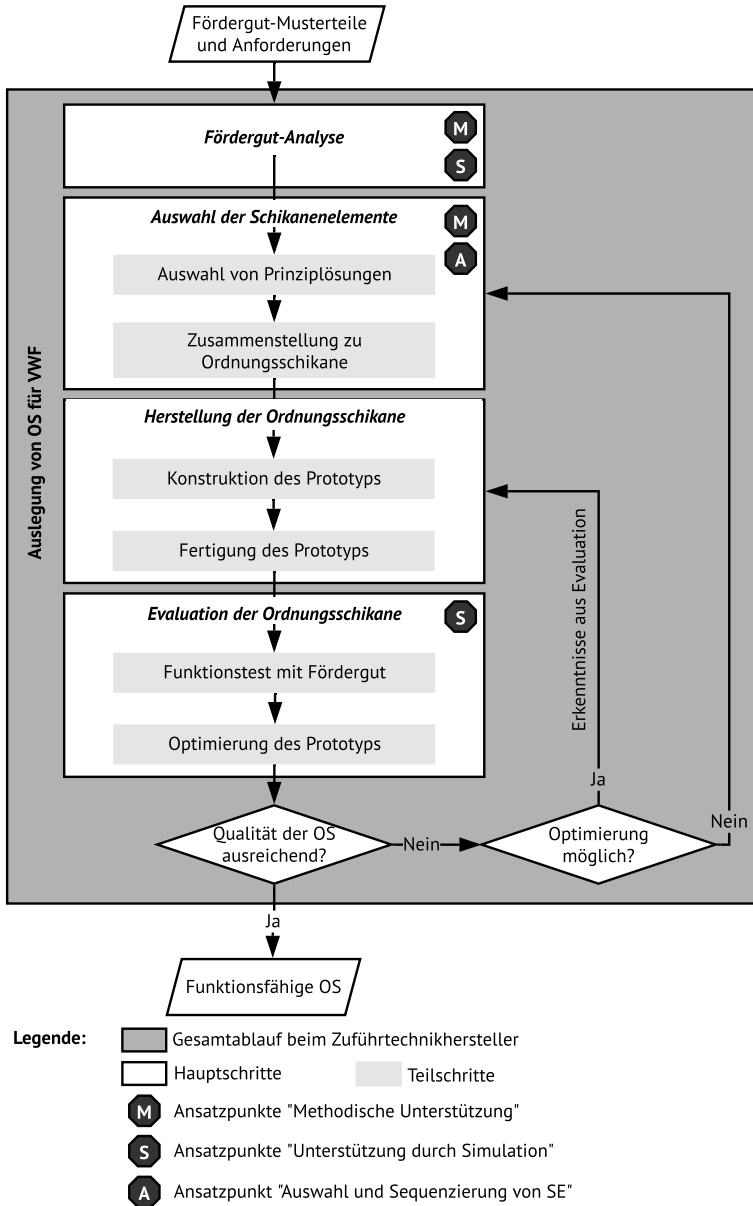


Abbildung 2.8: Ablauf der konventionellen Auslegung von OS für VWF (angelehnt an HOFMANN 2014, S. 21; MATHIESEN 2017, S. 31) und Ansatzpunkte zu dessen Unterstützung

### 2.3.2 Methodische Unterstützung bei der Auslegung

Diese Ansätze zur Unterstützung bei der Auslegung von OS bieten methodische Vorgehensweisen, um den Prozess zu vereinfachen und notwendiges Expertenwissen zu reduzieren. Insbesondere die Prozessschritte *Fördergut-Analyse* und *Auswahl der Schikanenelemente* stehen hierbei im Fokus (vgl. Abbildung 2.8, M: Ansatzpunkte „Methodische Unterstützung“).

#### 2.3.2.1 Methoden zur Unterstützung der Fördergut-Analyse

**Anzahl und Verteilung von Fördergut-Vorzugslagen:** Im Rahmen der Fördergut-Analyse ist die Bestimmung von Fördergut-Vorzugslagen eine essentielle Voraussetzung für die Auslegung von OS (BOOTHROYD 2005, S. 55, S. 70). Durch den Einsatz von unterstützenden Methoden können dazu notwendige Zeit und Aufwand reduziert werden (GOLDBERG ET AL. 1999). Zahlreiche Veröffentlichungen befassen sich daher mit analytischen Methoden zur Bestimmung von Anzahl und prozentualer Verteilung von Fördergut-Vorzugslagen ohne Experimente. Durch die Untersuchung geometrischer und physikalischer Eigenschaften des Fördergut-Teils werden mathematisch deren Zusammenhänge mit den Vorzugslagen hergeleitet (BOOTHROYD & HO 1977; NGOI & LIM 1996; NGOI ET AL. 1995c; WIEGLEY ET AL. 1992). In manchen Veröffentlichungen wurden die Ergebnisse zudem experimentell validiert, wobei partiell gute Übereinstimmungen mit der Realität erzielt werden konnten (CHUA & TAY 1998; GOLDBERG ET AL. 1995, 1999; LEE ET AL. 1996; LEVY & ROSARIO 2001; NGOI ET AL. 1995a,b, 1996, 1997; M. SURESH ET AL. 2013, 2018; UDHAYAKUMAR ET AL. 2013; VÁRKONYI 2014, 2015).

**Klassifizierung von Fördergut:** Einige Arbeiten präsentieren methodische Vorgehensweisen, um anhand von Geometriemerkmalen eine systematische Klassenzuordnung von Fördergut-Teilen durchzuführen (FRANK 1975, S. 28–41; BOOTHROYD ET AL. 1978, Abschnitt 1; BOOTHROYD 2005, S. 385–407). Dieser Ansatz wurde in weiteren Publikationen durch eine rechnergestützte Klassifizierung des Förderguts anhand physikalischer und geometrischer Merkmale

erweitert (Y. CHEN & YOUNG 1988; LA BROOY & C. JIANG 2009; LIM ET AL. 1994; P. TAN ET AL. 1995; VRANJEŠ ET AL. 1997; OU-YANG & MAUL 1993).

**Vorhersage von sekundären Kenngrößen:** Weitere Publikationen behandeln die Vorhersage von sekundären Kenngrößen wie der Fördergeschwindigkeit oder des Werkstückverhaltens sowie der Wechselwirkung mit der Dynamik des Schwingförderers. Allerdings werden Linear- und Schwerkrafft Förderer anstelle von VWF betrachtet. Die relevanten Arbeiten stellen analytische Modelle auf, die das dynamische Verhalten des Systems abbilden. Experimentelle Validierungen zeigen zum Teil hohe Modellgenauigkeiten (COMAND ET AL. 2018; KONG ET AL. 2015; KONG ET AL. 2018; KRISHNASAMY ET AL. 1996; RAMALINGAM & SAMUEL 2009). JAKSIC & MAUL (2001) schlagen ein analytisches Modell speziell für VWF vor, welches das Verhalten des Förderguts mittels Energiebilanzen charakterisiert. Es wird zudem die aktive Umorientierung der Teile mittels Druckluft berücksichtigt. In ähnlicher Weise wird in weiteren Arbeiten die analytische Prognose von Ordnungswahrscheinlichkeiten durch OS (YERALAN & CHANG 1986) sowie von Fördergut-Geschwindigkeiten (HAN & GAO 2010) untersucht. BOOTHROYD ET AL. (1978, Abschnitt 4, Blatt 2) geben als Hilfestellung Formeln zur Abschätzung von Schwingungsamplitude, Schwingwinkel und Fördergeschwindigkeit im VWF an.

Die Arbeit von RISCH (2011, S. 102, S. 107 f.) entwickelt auf Basis von analytischen Untersuchungen Berechnungsvorschriften zur Bestimmung der Fördergeschwindigkeit in nicht-geneigten Förderorganen wie Linearförderern. Dazu werden Abhängigkeiten relevanter Bewegungsparameter und aus diesen resultierende Fördergut-Reaktionen untersucht und experimentell validiert. Als praktisch anwendbares Ergebnis wird eine speziell entwickelte Software gezeigt, welche die Abschätzung der Fördergeschwindigkeit ermöglicht (RISCH 2011, S. 133–139).

**Fazit:** Bei der Analyse der vorgestellten Ansätze wird deutlich, dass diese ausnahmslos von hoher Komplexität und für die vorliegende Arbeit nicht zielführend sind. Deren Anwendung zur Fördergut-Analyse im Rahmen einer Methode zur Auslegung von OS beinhaltet folgende Hindernisse: Ein Teil der Ansätze erfordert umfangreiche Kenntnisse des Bedienpersonals in den

## 2 Stand der Erkenntnisse

---

Bereichen Mathematik und mechanischen Modellbildung. Ein zweiter Teil benötigt speziell dafür entwickelte Software. Zwar wird diese in den Veröffentlichungen teilweise erwähnt, ist aber nicht (mehr<sup>8</sup>) öffentlich zugänglich oder dokumentiert. Hervorzuheben ist jedoch die Arbeit von LA BROOY & C. JIANG (2009), die ein umfangreiches Vorgehen zur CAD-basierten Extraktion von Fördergut-Charakteristika beinhaltet. Dieses ermöglicht die automatisierte Erhebung charakterisierender Eigenschaften von Fördergut, deren Kenntnis bei der Auslegung von OS notwendig ist (vgl. HESSE 2010, S. 196).

### 2.3.2.2 Methoden zur Unterstützung der Auswahl von SE

Für die Auswahl und Zusammenstellung von SE ist umfangreiches Erfahrungswissen erforderlich (HESSE 2000b, S. 40). Daher geben zahlreiche Veröffentlichungen methodische Hilfestellung für die manuelle Auswahl von SE, indem sie diese auf Basis von Zahlenwerten systematisieren und dadurch erfahrungsbasierte Schätzungen ersetzen.

**Quantifizierung von Ordnungsvorgängen:** Eine Möglichkeit der systematischen Auswahl geeigneter SE besteht darin, deren Wirksamkeit zu quantifizieren, um diese als Auswahlkriterium zu nutzen. Dazu schlägt BOOTHROYD (2005, S. 59 f.) die experimentelle Bestimmung von Transitionsmatrizen vor (vgl. Abschnitt 2.2.3). Diese enthalten Informationen über die Wahrscheinlichkeit der Umorientierung durch das SE. Das Produkt einer Reihe von Matrizen gibt die Wirkung einer aus SE zusammengesetzten OS an. Zudem werden umfassende Tabellenwerke mit Referenzschikanen präsentiert, die zum Nachschlagen von prinzipiellen Ordnungslösungen verwendet werden können (BOOTHROYD 2005, S. 474–491).

BERKOWITZ & CANNY (1996) verfolgen einen ähnlichen Ansatz auf Basis des Markov-Modells<sup>9</sup>, das die Übergänge zwischen den SE als Zustandsänderun-

---

<sup>8</sup>In der Veröffentlichung von BERRETTY ET AL. (1999) wird bspw. die URL eines Online-SE-Konfigurators genannt, die jedoch nicht mehr erreichbar ist.

<sup>9</sup>Markov-Modelle sind die stochastische Form des Automatenmodells und beschreiben Zustandsübergänge mit Wahrscheinlichkeiten. Sie werden zur Modellierung von unsicherem Wissen und zur Prognose von zukünftigen Entwicklungen eingesetzt (DÖRN 2018, S. 219 f.).

gen eines nicht-deterministischen Automaten beschreibt. Dabei werden die Wahrscheinlichkeitsverteilungen der Orientierungen in den Transitionsmatrizen als Übergangswahrscheinlichkeit von einer Orientierung zur nächsten betrachtet und so die wahrscheinlichste Endorientierung des Fördergut-Teils nach Durchlaufen der OS ermittelt.

**Parametrierung von SE:** Bei BOOTHROYD (2005, S. 474–491) sind Regeln für die systematische Einstellung von Parametern der SE (z. B. Stufenhöhe, Länge einer Profilschiene) in Abhängigkeit der Dimensionen des Förderguts aufgeführt. Auch WEISS (1983, S. 45–63) präsentiert ähnliche, umfangreiche Untersuchungen zur Parametereinstellung der SE. WEISS (1983) bestimmt Korrelationen zwischen eigens definierten Standard-SE und dem Orientierungszustand und -verhalten des Förderguts. Hierdurch können Auswahl und Parametrierung der Schikanen systematisch durchgeführt werden. NATARAJAN (1989) erarbeitet ebenfalls auf rein analytischer Basis formelle Paradigmen für die Gestaltung einzelner Grundelemente zur Orientierung von Werkstücken.

YEONG & DE VRIES (1994) zeigen ein komplexeres Vorgehen zur Auswahl und Parametrierung von SE in drei Schritten, welches neben der analytischen Beschreibung der Fördergut-Dynamik die Methoden von BOOTHROYD (2005, S. 385–407) und LO & DICK (1990) in einem zusammenhängenden Prozess zusammenführt.

Die Parameter der SE können zudem durch geometrische Analyse bestimmt werden. Zu diesem Zweck wird die Bewegung des Förderguts als quasi-statisch und ohne Interaktion zwischen mehreren Fördergut-Teilen betrachtet und Fördergut und OS durch Polygone modelliert. Das dynamische Verhalten der Fördergut-Teile wird analytisch prognostiziert, sodass mögliche Geometrieparameter der einzelnen SE bestimmt werden können, für die das Fördergut durch die OS umorientiert wird. Basierend auf diesen Modellen werden einfache Methoden zum Gestalten von passiven SE vorgestellt (AGARWAL ET AL. 2001; BERRETTY ET AL. 1999, 2001; CAINE 1994; GOEMANS & VAN DER STAPPEN 2008; GOEMANS ET AL. 2006).

**Fazit:** Problematisch ist, dass die Konzepte der vorgestellten Arbeiten aufgrund starker Vereinfachungen nur sehr beschränkt anwendbar und übertragbar sind. Die Einteilung von Fördergut in Klassen und die Verwendung von SE-Prinziplösungen ist oftmals nicht vereinbar mit der Komplexität praktischer Ordnungsaufgaben. Eine Entkopplung vom manuellen Entwicklungsprozess wie in dieser Arbeit angestrebt ist mit diesen auf methodische Auslegung ausgerichteten Ansätzen nicht möglich, da sie lediglich auf eine Hilfestellung beim erfahrungsbasierten Prozess abzielen.

Dagegen haben sich die von BOOTHROYD (2005) vorgeschlagenen Transitionsmatrizen zur Quantifizierung von SE etabliert, da deren Ermittlung und Anwendung unkompliziert ist und die Bewertung des Förderprozesses ermöglicht. Wie weiter unten erläutert wird (vgl. Abschnitt 2.3.3), kann die Ermittlung anstelle von Experimenten mit Simulationsverfahren durchgeführt und dadurch deutlich vereinfacht werden. Die Abstimmung der SE-Parameter auf das Fördergut, wie von BOOTHROYD (2005, S. 474–491) und WEISS (1983, S. 45–63) vorgestellt, ermöglicht die systematische und zugleich relativ unkomplizierte Auswahl und Parametrierung von SE. Dadurch kann das konventionelle Vorgehen zum Teil von persönlichem Erfahrungswissen entkoppelt werden.

**Exkurs: Flexible Ordnungssysteme als Alternative zur Auswahl von SE:** In der Literatur wurden zudem Ansätze entwickelt, welche anstelle der Zusammensetzung von spezialisierten OS auf flexible Systeme abzielen. TAY ET AL. (2005) präsentieren eine programmierbare OS, die anhand von Daten einer Sensorik mit Bildverarbeitung das Umorientieren bzw. Aussortieren von Falschlagen mithilfe pneumatischer Aktorik vornimmt. LOY (2010, S. 39–60) entwickelt einen modularen, umrüstbaren VWF zur flexiblen Teilezuführung. Die OS zählen jedoch zu den produktspezifischen Komponenten des Systems, weisen somit keine Flexibilität auf und müssen für jedes neue Fördergut-Teil neu entwickelt werden (LOY 2010, S. 41). Die Herangehensweise der flexiblen Ordnungssysteme wurde sowohl in der Literatur als auch in der vorliegenden Arbeit nicht weiterverfolgt, da sie die bestehenden Probleme nicht ausreichend adressiert.

### 2.3.3 Unterstützung der Auslegung durch Simulationsverfahren

Nach der methodischen Unterstützung bei der Auslegung von OS als erste Herangehensweise basiert ein zweiter Teil der bestehenden Ansätze auf dem Einsatz von Simulationsverfahren. Diese setzen im Entwicklungsprozess bei den Schritten *Fördergut-Analyse* sowie *Evaluation der OS* an (vgl. Abbildung 2.8, S: Ansatzpunkte „Unterstützung durch Simulation“). Vor allem dem Einsatz der Simulation zur Funktionsabsicherung eines OS-Entwurfs im Anschluss an den Auslegungsprozess wird großes Potential zugesprochen, da die Simulation aufwändige physische Versuche ersetzen kann (BERKOWITZ & CANNY 1996; GOLDBERG ET AL. 1995; HESSE 2000b, S. 40).

Für die Simulation von VWF ist wie WOLFSTEINER (1999, S. 12) feststellt die Modellierung der Kontaktprozesse zwischen Fördergut und den Flächen im VWF von zentraler Bedeutung. Zur Abbildung dieser physikalischen Effekte werden in der Literatur unterschiedliche Simulationsverfahren<sup>10</sup> herangezogen. Den verschiedenen Ansätzen gemein ist dabei die Abbildung von Fördergut und VWF bzw. OS als Starrkörper und die impulsbasierte Auswertung der Interaktionen. Jede Berechnungsschleife der Simulation besteht aus drei Schritten: Im ersten werden zwischen den Simulationsobjekten auftretende Kollisionen detektiert, im zweiten berechnet und im dritten wird der Systemzustand aktualisiert, indem die resultierenden Kräfte und Impulse integriert werden (MIRTICH 1996, S. 10 f.; HAHN 1988). Die Einwirkung der Fördergutlast auf die Dynamik des Systems wird vernachlässigt. Nach BERKOWITZ & CANNY (1997) stellt dieser Ansatz der „impulsbasierten Starrkörpersimulation“ eine effiziente und schnelle Methode mit hoher Übereinstimmung mit der Realität dar.

#### 2.3.3.1 Simulationsansätze zur Unterstützung der Fördergut-Analyse

Zur Bestimmung der Vorzugslagen von Fördergut-Teilen als Teil der Fördergut-Analyse kommen in einigen Arbeiten Simulationsansätze zum Einsatz (HOFMANN 2014, S. 120 f.; GOLDBERG ET AL. 1999; HANSSON ET AL. 2016; MIRTICH

---

<sup>10</sup>Auf die Details der Verfahren und Implementierungen wird hier nicht näher eingegangen, da diese für die vorliegende Arbeit nicht relevant sind.

ET AL. 1996). Es werden Fall-Versuche simuliert, wobei Verhalten und Orientierung des Werkstücks nach einem freien Fall auf eine flache Ebene und anschließendem reibungsbehafteten Stoß ausgewertet werden. Als Erweiterung schlagen HARALDSON ET AL. (2020) ein Vorgehen vor, bei dem ein Algorithmus selbstständig anhand von Simulationsdaten ermittelt, welche Orientierungen ein Werkstück einnehmen kann, sodass diese nicht mehr zuvor angegeben werden müssen. Die Validierung der Simulationsergebnisse, die in den vorgestellten Arbeiten durchgeführt wird, zeigt sehr gute Übereinstimmungen mit experimentellen Daten.

**Fazit:** Wie in den o. g. Veröffentlichungen gezeigt wird, ist der Einsatz von Simulationsverfahren zur Bestimmung der Fördergut-Vorzugslagen ein effizientes Vorgehen, um manuelle Arbeitsschritte zu ersetzen. Im Vergleich zu experimentellen Versuchsreihen können Zeit und Ressourcen eingespart werden. Zudem wurde eine hinreichend genaue Abbildung der Realität anhand der durchgeführten Validierung nachgewiesen.

### 2.3.3.2 Simulationsansätze zur Unterstützung der Evaluation der OS

#### **Entwicklung von Simulationsverfahren zur Abbildung von Fördergut im**

**VWF:** Da die Abbildung des Förderprozesses im VWF ein komplexer Vorgang ist, wurde in vielen Arbeiten aus dem Stand der Forschung die Entwicklung bzw. Adaption von Simulationsverfahren speziell für die Abbildung von Fördergut im VWF vorangetrieben. Dazu entwickeln die Autoren ein mathematisches Modell der zu betrachtenden Größen (bspw. dynamisches Verhalten des Förderguts, erreichbare Fördergeschwindigkeit im VWF), welches anschließend als Simulation implementiert wird. Der Fokus liegt hierbei auf der Entwicklung des Modells, welches die Simulation überhaupt erst ermöglicht oder verbessert.

Zahlreiche Arbeiten beschreiben die jeweils entwickelten Modelle und Simulationsansätze, jedoch ohne deren Validierung anhand von realen Versuchen (BERKOWITZ & CANNY 1996; R. CHEN ET AL. 2011; CHUA & F. TAN 2006; DALLINGER ET AL. 2012; KAWACHI ET AL. 1997, 1998; WEIR & CIPRA 2005; YANG ET AL. 2012).



Weitere Veröffentlichungen beinhalten zusätzlich Ergebnisse experimenteller Validierung (GOLDBERG ET AL. 1995; HAN & GAO 2010; M. JIANG ET AL. 2003; MAUL & THOMAS 1997; ROCKLAND & STETTER 1994; VÁRKONYI 2014; WOLFSTEINER 1999, S. 101–104) oder des Vergleichs mit Daten aus der Literatur (ASHRAFIZADEH & ZIAEI-RAD 2013; RIMAI & CIPRA 2011). BERKOWITZ & CANNY (1997) befassen sich schwerpunktmäßig mit der Validierung von Simulationsergebnissen mit realen Experimenten.

Besonders hervorzuheben sind die Arbeiten von MIRTICH (1996). Diese basieren auf den Vorarbeiten von HAHN (1988), auf den das zum damaligen Zeitpunkt neuartige Verfahren der impulsbasierten Starrkörpersimulation zurückgeht. MIRTICH (1996, S. 10) verbessert dessen Methoden zur Kollisionserkennung und -auswertung und stellt den Bezug zum Anwendungsfeld der Produktionstechnik her, indem er das Simulationsverfahren erstmalig für ein Förderband mit zu fördernden Kleinteilen einsetzt (MIRTICH & CANNY 1995a; MIRTICH & CANNY 1995b; MIRTICH 1996, S. 209–220).

**Simulation als Werkzeug zur Evaluation von OS:** Im Gegensatz zu den eben genannten Arbeiten setzen die folgend vorgestellten bereits bestehende Simulationsansätze als Werkzeug ein, ohne sie zu adaptieren. Meist wird eine experimentelle Validierung angeschlossen. Hierbei werden unterschiedliche Zielsetzungen verfolgt, so z. B. die Bestimmung von Vorzugslagen (GOLDBERG ET AL. 1999; MIRTICH ET AL. 1996), die Erzeugung von Eingangsdaten für die Auslegung von OS (HANSSON ET AL. 2016; MATHIESEN 2017; MATHIESEN & ELLEKILDE 2017; MATHIESEN ET AL. 2018) oder die Berechnung von Leistungsmerkmalen wie der Fördergut-Geschwindigkeit im VWF (HARALDSON ET AL. 2020; HOFMANN 2014; HOFMANN ET AL. 2013; LE ET AL. 2018; REZNIK & CANNY 1997, S. 115 f.; LACOUR 2012, S. 31-40).

Ein Teil der Arbeiten entwickelt ein eigenes Modell und nutzt zu dessen Absicherung bestehende Simulationsansätze, ohne diese zu verändern (KRISHNASAMY 1996; LUIDENS & CIPRA 2008; NARANG ET AL. 2017; SHI ET AL. 2014; M. SURESH ET AL. 2018). Andere nutzen Simulationen zur Abbildung der Dynamik von VWF und Fördergut, führen aber keine experimentelle Validierung durch (HOFMANN & STIRNER 2017; KONG ET AL. 2015; REINHART & HOFMANN 2012).

**Fazit:** Die bisherigen Ansätze zur Simulation von OS ermöglichen teilweise die Verlagerung von Tests und Experimenten von der Werkstatt in die virtuelle Welt, wodurch bereits erheblich Zeit und Ressourcen eingespart werden können. Sie leisten aber keine automatisierte Auslegung und die damit verbundene Entkopplung von personenspezifischem Erfahrungswissen. Des Weiteren ist die Erreichung eines Optimums weiterhin nicht gewährleistet, da die Ansätze auf manuell erstellten Lösungen basieren.

### 2.3.4 Automatisierte Auswahl und Sequenzierung von Schikanenelementen

Neben der methodischen und der simulationsbasierten Herangehensweise zielt die dritte Art bestehender Ansätze zur Unterstützung bei der Auslegung von OS darauf ab, Algorithmen zur Auswahl und Parametrierung von SE sowie deren Kombination (auch: Sequenzierung) zu OS bereitzustellen. Im Entwicklungsprozess setzen diese bei der *Auswahl der Schikanenelemente* (vgl. Abbildung 2.8, A: Ansatzpunkt „Auswahl und Sequenzierung von SE“) an. Ähnlich zum manuellen Auslegungsprozess werden zunächst verschiedene SE-Grundtypen, wie sie bei BOOTHROYD (2005, S. 475) zu finden sind, kriterienbasiert oder zufällig ausgewählt. Die SE sind bereits oder werden noch an die Dimension des zu orientierenden Förderguts angepasst. Anhand von Merkmalen wie Orientierungswahrscheinlichkeit oder Effizienz werden die SE zu einer OS zusammengesetzt und das Resultat bewertet.

**Algorithmen zur Auswahl und Sequenzierung von SE:** Ein Teil der bestehenden Arbeiten zielt auf den Entwurf bzw. die Weiterentwicklung von Algorithmen zur Automatisierung der Auswahl und Sequenzierung von SE ab. Dazu wird zunächst ein Optimierungsproblem mathematisch formuliert, um anschließend einen geeigneten Algorithmus zur Anordnung der SE vorzuschlagen. Da der Lösungsraum hinsichtlich des Rechenzeitaufwands meist zu groß für eine exakte Lösung ist, wird er manuell beschränkt oder es kom-

men (Meta-)Heuristiken<sup>11</sup> wie z. B. genetische Algorithmen zum Einsatz. In den unterschiedlichen Veröffentlichungen erfolgt die Auswahl der SE entweder zufällig oder anhand informierter Suchverfahren. Wie durch Computerberechnungen gezeigt wird, erzeugen die gewählten Ansätze teilweise gute Lösungen, wobei eine zusätzliche experimentelle Validierung ausbleibt (CHRISTIANSEN ET AL. 1996; MATHIESEN & ELLEKILDE 2017; WIEGLEY ET AL. 1996). Bei JAUMARD ET AL. (1990, 1993) & SETHI ET AL. (1990) und MATHIESEN ET AL. (2018) ist zudem die Parametrierung der SE explizit Teil des Lösungsansatzes.

**Expertensysteme zur Auswahl und Sequenzierung von SE:** Weitere Veröffentlichungen entwerfen Expertensysteme, die regelbasiert und computergestützt OS aus SE sequenzieren. Die zugrundeliegenden Regeln enthalten z. B. die Zuordnung von Fördergut-Merkmalen zu bestimmten SE und deren Funktion (bspw. Umorientierung des Fördergut-Teils von einer bestimmten Orientierung in eine andere). Der Fokus liegt auf der Herleitung dieser Regeln, meist anhand geometrischer Analysen des Fördergut-Teils. Eine Wissensbasis dient als Gedächtnis für das digitale Expertensystem (LA BROOY & C. JIANG 2009; LIM ET AL. 1994; LO & DICK 1990; P. TAN ET AL. 1995; VRANJEŠ ET AL. 1997; OU-YANG & MAUL 1993).

**Fazit:** Die bisherigen Ansätze zur automatisierten Auswahl und Sequenzierung von SE zeigen einen vielversprechenden Lösungsweg auf, sind aber in der Praxis bislang nicht praktikabel: Durch die meist auf Zufall bzw. erschöpfender Suche basierende Auswahl der SE ist das Vorgehen wenig effizient bis hin zu rechentechnisch nicht umsetzbar. Um dieses Problem zu umgehen, wird der Lösungsraum manuell beschränkt. Dadurch entsteht das Risiko, dass ggf. zu wenige bzw. nicht die richtigen SE ausgewählt werden, um eine sinnvolle Lösung zu erzielen. Zudem bleibt die abschließende Optimierung des Gesamtergebnisses aus und es verbleiben weiterhin notwendige manuelle Arbeitsschritte.

---

<sup>11</sup>(Meta-)Heuristiken finden auch ohne die vollständige Durchsuchung des Lösungsraums sehr gute Lösungen für Optimierungsprobleme, die z. B. aufgrund zu hoher Anzahl an Eingangsdaten und daraus resultierender Rechenzeit nicht exakt gelöst werden können. Voraussetzung hierfür ist jedoch, dass die gewählte (Meta-)Heuristik zum Optimierungsproblem passt. Heuristiken gehen anhand von festgelegten Regeln vor, Meta-Heuristiken integrieren zusätzlich zufällige Entscheidungen (SCHOLZ 2018, S. 217).

Der Entwurf von Regeln für Expertensysteme weist hingegen zu große Einschränkungen hinsichtlich der Allgemeingültigkeit auf, sodass dieser Ansatz in der Literatur seit einigen Jahren nicht weiterverfolgt wurde (zuletzt durch LA BROOY & C. JIANG (2009)).

### 2.4 Forschungsdefizit

In den vorangegangenen Abschnitten 2.2 und 2.3 wurde der Stand der Erkenntnisse auf den für diese Arbeit relevanten Themengebieten untersucht. Wie in Abschnitt 1.2 ausgeführt, ist das Ziel der vorliegenden Arbeit eine durchgängige Methode zur automatisierten Generierung von OS für VWF unter Nutzung der Zukunftstechnologie RL (s. Abschnitt 1.3).

Die Analyse des Stands der Erkenntnisse in Abschnitt 2.3 ergibt, dass diese Zielsetzung bisher nicht erfüllt werden kann. Zwar können bereits einzelne Aspekte des Auslegungsprozesses effektiv unterstützt werden, eine durchgängige Methode existiert aber bislang nicht. Daher kann die angestrebte *Verkürzung der Entwicklungszeit* für OS im Vergleich zum manuellen Auslegungsprozess nicht in dem Maße erreicht werden, wie es ein durchgängiger Prozess ermöglicht. Auch die durch eine durchgängig systematische Auslegung erzielbare *Steigerung der Ergebnisqualität* ist daher nicht gegeben. Zum Teil ermöglichen bestehende Ansätze den Vergleich von Lösungsalternativen, sodass im Zuge der folgenden Methodenentwicklung partiell auf diese zurückgegriffen werden kann. Ausgewählte Ansätze aus dem Stand der Erkenntnisse erzielen bereits teilautomatisiert erzeugte OS, eine Weiterverwendung der errechneten Lösungen im Sinne einer *digitalen Wissensbasis* erfolgt jedoch nicht. Die vollständige Entkopplung von manuellen Anteilen und somit von personenspezifischem Erfahrungswissen ist bei keinem der bisherigen Ansätze gegeben. Insbesondere der Einsatz von RL zu diesem Zweck, wie im Rahmen dieser Arbeit angestrebt, ist in den analysierten Arbeiten nicht abgedeckt.

Über die Defizite hinsichtlich der in Abschnitt 1.2 abgeleiteten Teilziele hinaus wurden noch weitere festgestellt: So basieren einige der analysierten Methoden auf derart starken Vereinfachungen, dass deren *Allgemeingültigkeit* unzureichend ist. Dem gegenüber sind wiederum andere von derartiger Komplexität

bzw. erfordern so hohen Rechenaufwand, dass deren *effizienter Einsatz* nicht möglich ist.

Auf Basis der eben beschriebenen Defizite werden im nachfolgenden Kapitel 3 die Anforderungen aufgestellt, welchen die zu entwickelnde Methode zur automatisierten Generierung von OS genügen muss (zur kritischen Betrachtung der Ergebnisse und deren Abgleich mit den Anforderungen vgl. Abschnitt 7.3).

Zunächst wird jedoch genauer auf die Grundlagen zum Thema Machine Learning (ML) im Allgemeinen und RL im Speziellen eingegangen, um das notwendige Verständnis für die erarbeitete Methode zu schaffen.

### 2.5 Reinforcement Learning als Verständnisgrundlage

Im Rahmen der vorliegenden Forschungsarbeit wird Reinforcement Learning (RL) eingesetzt, um ein lernfähiges System zu schaffen, das eine Wissensbasis aus früheren OS-Auslegungsvorgängen aufbaut und nachfolgend nutzt (vgl. Abschnitt 1.3 sowie Kapitel 6).

Zur Abgrenzung der Begrifflichkeiten gibt Abschnitt 2.5.1 zunächst einen Überblick über die Domänen künstliche Intelligenz (KI)<sup>12</sup> und maschinelles Lernen (ML)<sup>13</sup> sowie dessen Sub-Kategorien überwachtes<sup>14</sup>, unüberwachtes<sup>15</sup> und bestärkendes Lernen<sup>16</sup>. Letzteres hat in den vergangenen Jahren stark an Bedeutung gewonnen (SUTTON & BARTO 2018, S. xiii, S. 13 ff.) und steht im Rahmen dieser Arbeit im Vordergrund. Daher wird in Abschnitt 2.5.2 die Thematik des bestärkenden Lernens (auch und im Folgenden: Reinforcement Learning) im Detail beleuchtet. Abbildung 2.9 zeigt einen Überblick über die Domänen Künstliche Intelligenz (KI), Machine Learning (ML) sowie verschiedene Lernparadigmen und deren Anwendungsfelder. In dieser Arbeit kommt ML zur Klassifizierung und zum sequentiellen Entscheiden mittels RL zum Einsatz.

---

<sup>12</sup>Engl. *artificial intelligence* (AI).

<sup>13</sup>Engl. *machine learning* (ML).

<sup>14</sup>Engl. *supervised learning* (SL).

<sup>15</sup>Engl. *unsupervised learning* (UL).

<sup>16</sup>Engl. *reinforcement learning* (RL).

## 2 Stand der Erkenntnisse

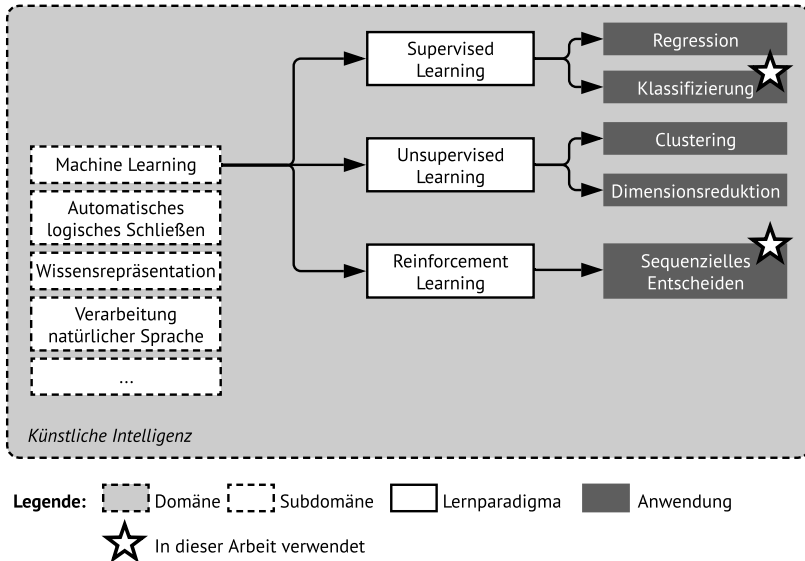


Abbildung 2.9: Übersicht über die Domänen KI, ML sowie Lernparadigmen und Anwendungsgebiete (in Anlehnung an DÖBEL ET AL. 2018, S. 10; RUSSELL & NORVIG 2012, S. 22 ff.)

Im Folgenden kommen aufgrund der allgemeinen Verbreitung meist die englischsprachigen Begriffe zur Anwendung. Des Weiteren werden relevante deutsche Fachbegriffe mit der englischen Übersetzung als Fußnote ergänzt, um die Nachvollziehbarkeit anhand der Quellen zu gewährleisten, die zu diesem Themengebiet hauptsächlich englischsprachig vorliegen.

### 2.5.1 Künstliche Intelligenz und maschinelles Lernen

KI zielt darauf ab, Computersysteme zu rationalem und intelligentem Handeln zu befähigen. Das wissenschaftliche Ziel der KI ist das Verständnis von Prinzipien der Wissensrepräsentation, welche intelligentes Verhalten ermöglichen. Auf Seiten der Anwendung resultiert dies in der Schaffung von Computerprogrammen (auch als (Computer-)Agenten bezeichnet), die reale Probleme effektiver und effizienter lösen als menschliche Individuen. KI bietet dazu eine

## 2.5 Reinforcement Learning als Verständnisgrundlage

---

Vielzahl an Werkzeugen in verschiedenen Sub-Domänen, wie z. B. Verarbeitung natürlicher Sprache, Wissensrepräsentation, automatisches logisches Schließen und ML (RUSSELL & NORVIG 2012, S. 22 ff.).

Das Ziel des ML ist es, einen Computeragenten aus anwendungsspezifischen Daten Wissen generieren, also lernen zu lassen (SCHUH ET AL. 2019). Durch die Analyse von (historischen) Daten durch spezielle Algorithmen leitet der Agent Vorhersagen ab, die mit einer gewissen Wahrscheinlichkeit eintreten bzw. korrekt sind (VDMA 2018, S. 4). Dies geschieht über die Approximation der Daten durch mathematische Funktionen, die als Modell die reale Problemstellung innerhalb eines bestimmten Gültigkeitsbereichs abbilden (ERTEL 2013, S. 173). Mit dem gewonnenen Wissen soll der Agent eine vorgegebenen Zielgröße optimieren oder eine bestimmte Aufgabe erledigen (SCHUH ET AL. 2019). Einige Aufgaben wie z. B. die Bild- oder Spracherkennung, die sich mittels konventioneller Programmiermethoden nicht oder nur mit sehr großem Aufwand lösen lassen, können mit ML effizient und umfassend gelöst werden. Dies liegt darin begründet, dass beim Einsatz von ML keine expliziten Handlungsanweisungen gegeben werden müssen, da die eingesetzten Agenten durch das Analysieren von Daten und das Erkennen von Mustern und Strukturen darin eigenständig Lösungsansätze entwickeln (GÜNTHER 2018, S. 32).

ML-Agenten werden mittels stochastischer und mathematischer Algorithmen implementiert (REBALA ET AL. 2019, S. 1 f.). Anhand von Parametern<sup>17</sup> werden diese Algorithmen speziell auf die jeweilige Problemstellung zugeschnitten, um zuverlässige Ergebnisse zu erzielen (CLAESEN & DE MOOR 2015; GÜNTHER 2018, S. 31 f.).

Das Feld des ML bedient sich dreier grundlegend verschiedener Lernparadigmen, nach denen ML-Agenten aus Daten lernen. Diese werden als Supervised, Unsupervised und Reinforcement Learning bezeichnet und im Folgenden näher erläutert (REBALA ET AL. 2019, S. 18).

---

<sup>17</sup>Parameter dienen der Konfiguration der Architektur des mathematischen Modells, das für die ML-Aufgabe gewählt wurde. Beispiele sind die Anzahl der Ebenen sowie die Anzahl der Knoten in einer Ebene eines neuronalen Netzes oder der Polynomgrad der gewählten Funktion bei Regressionsaufgaben (PAASS & HECKER 2020, S. 72).

**Supervised Learning (SL):** Beim SL hat der Computeragent das Ziel, eine Klassen-Zuordnung oder Funktions-Annäherung zu erlernen. Dazu erhält er eine große Menge an gelabelten<sup>18</sup> Daten<sup>19</sup> (GÜNTHER 2018, S. 32). Diese sog. Trainingsdaten sollen eine repräsentative Stichprobe für die zu lösende Aufgabe darstellen (ERTEL 2013, S. 181). Durch Extraktion von Merkmalen<sup>20</sup> und deren Zuordnung zu den Labels erfolgt die Generalisierung der in den gelabelten Daten vorgegebenen Klassenzuordnungen in ein Modell. Dieser Vorgang wird als Training bezeichnet. Dabei werden die Parameter des Algorithmus mit dem Ziel angepasst, den Fehler<sup>21</sup> bzw. die Abweichung zwischen den Ergebnissen des Agenten und den Labels im Trainings-Datenset zu minimieren (RUSSELL & NORVIG 2012, S. 812; GOODFELLOW ET AL. 2016, S. 97).

Je länger das Training fortgesetzt wird, desto kleiner wird der Trainings-Fehler. Dabei kann eine Überanpassung<sup>22</sup> auftreten: Der Agent passt sein Modell zu stark an in den Trainingsdaten vorhandenes Rauschen an, wodurch die Generalisierbarkeit auf unbekannte Daten reduziert wird (GOODFELLOW ET AL. 2016, S. 108–114). Daher wird der Agent parallel zum Training mittels eines Test-Datensets, welches neue und ungelabelte Daten beinhaltet, erprobt und wiederum der Fehler<sup>23</sup> berechnet. Dieser wird jedoch nicht zur Anpassung der Parameter herangezogen, sondern lediglich zur Evaluierung des Trainingserfolgs und zur Festlegung des Trainingsendes.

Dieses Vorgehen von parallelem Training und Test wird auch als Kreuz-Validierung<sup>24</sup> bezeichnet (ERTEL 2013, S. 218). Bei Überanpassung sinkt der Trainings-Fehler, während der Test-Fehler steigt. Das Training sollte bei minimalem Test-Fehler beendet werden (WOERNLE 2008, S. 21).

---

<sup>18</sup>Gelabelte Daten enthalten Angaben (bzw. Markierungen, engl. *label*) über deren Inhalt in Bezug auf die Aufgabe des Agenten (REBALA ET AL. 2019, S. 19 f.). Ein Label steht für eine zu unterscheidende Klasse (RUSSELL & NORVIG 2012, S. 848).

<sup>19</sup>Solche Datenmengen werden häufig als „Datenset“ (engl. *data set*) bezeichnet (GOODFELLOW ET AL. 2016, S. 103).

<sup>20</sup>Merkmale (engl. *features*) sind in diesem Zusammenhang Eigenschaften, die verschiedene Klassen voneinander unterscheiden. Ein Beispiel ist die Unterscheidung der Größe und Farbe von Äpfeln zu deren automatischer Einteilung in Handelsklassen durch eine Sortieranlage (ERTEL 2013, S. 178 f.).

<sup>21</sup>Engl. *training error*.

<sup>22</sup>Engl. *overfitting*.

<sup>23</sup>Engl. *test error*.

<sup>24</sup>Engl. *cross validation*.



## 2.5 Reinforcement Learning als Verständnisgrundlage

---

Bei hohem Trainings-Fehler kann zudem auch eine Unteranpassung<sup>25</sup> vorliegen. In diesem Fall verfügt der Agent nicht über ausreichende Fähigkeiten (z. B. bei Wahl einer zu geringen Größe eines neuronalen Netzes), um alle relevanten Informationen zu erfassen, die zur Lösung seiner Aufgabe nötig sind (GOOD-FELLOW ET AL. 2016, S. 109 f.). Im Optimalfall sind Trainings- und Test-Fehler nach dem Training minimal (GÜNTHER 2018, S. 32).

Um den trainierten Agenten optimal zu evaluieren, erhält dieser zuletzt ein drittes, sog. Validierungs-Datenset mit ungelabelten Daten. Dieses dient lediglich der Überprüfung des Trainingserfolgs und der Leistungsfähigkeit des Agenten (WOERNLE 2008, S. 21 f.). Nach abgeschlossenem Training, Test und Validierung des Agenten können unbekannte Daten ohne Markierung zugeordnet werden (SUTTON & BARTO 2018, S. 2).

SL wird hauptsächlich für die Problemstellungen Klassifizierung und Regression eingesetzt (RUSSELL & NORVIG 2012, S. 811 f.): Klassifikationsalgorithmen kommen häufig zur Bilderkennung zum Einsatz, z. B. in Suchmaschinen. Dabei enthalten die Trainingsdaten große Mengen an Bildern, die vorgegebenen Klassen zugeordnet sind. Aufgabe des Agenten ist es, nach dem Training unbekannte Bilder den Klassen zuzuordnen, um dadurch Suchanfragen zufriedenstellend beantworten zu können. Bei Regressionsaufgaben soll der Algorithmus eine zugrunde liegende Funktion annähern, um bspw. den Verlauf von Aktienkursen an der Börse zu prognostizieren (REBALA ET AL. 2019, S. 20; ERTEL 2013, S. 282 f.).

Nach dem SL-Prinzip lernende Algorithmen sind u. a. Support Vector Machines (REBALA ET AL. 2019, S. 63–66), die k-Nearest-Neighbour-Methode (REBALA ET AL. 2019, S. 72–76) oder (künstliche) neuronale Netze (REBALA ET AL. 2019, S. 103–116). Da im Rahmen dieser Arbeit Convolutional Neural Networks (CNN) zur Ähnlichkeit-Analyse zum Einsatz kommen (vgl. Abschnitt 4.2.2), wird im Folgenden näher auf Neuronale Netze (NN) im Allgemeinen und CNN im Speziellen eingegangen.

NN bestehen nach dem biologischen Vorbild aus einer Anordnung von Neuronen, die zu einem Netz verknüpft sind. Anhand der Eingangsdaten des Netzes

---

<sup>25</sup>Engl. *underfitting*.

## 2 Stand der Erkenntnisse

lernen sie schrittweise, lineare und nicht-lineare Trends zu erkennen, um Aussagen für unbekannte Situationen zu treffen (SAMARASINGHE 2007, S. 12 f.). Sie werden meistens als SL-Modelle zur Klassifizierung eingesetzt (REBALA ET AL. 2019, S. 103). Die Basis von NN bilden einzelne (künstliche) Neuronen<sup>26</sup>. Abbildung 2.10 zeigt die Elemente eines künstlichen Neurons.

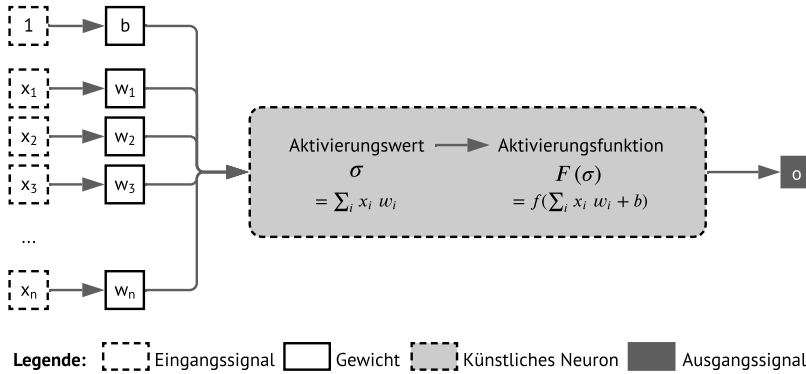


Abbildung 2.10: Elemente eines künstlichen Neurons nach VASILEV ET AL. (2019, S. 38 f.)

Künstliche Neuronen verarbeiten Eingangssignale  $x_i$ , welche über  $w_i$  gewichtet werden, zu einem Ausgangssignal  $o$ . Dieses kann z. B. die Zuordnung zu einer bestimmten Klasse bei Klassifizierungsaufgaben darstellen. Die Aktivierungsfunktion<sup>27</sup>  $F(\sigma)$  wird im Neuron auf den Aktivierungswert<sup>28</sup>, die gewichtete Summe  $\sigma$ , angewendet (ERTEL 2013, S. 250 f.; RUSSELL & NORVIG 2012, S. 846 f.):

$$F(\sigma) = f\left(\sum_i x_i w_i + b\right) \quad (2.4)$$

Während des Lernprozesses werden die Gewichtungen<sup>29</sup>  $w_i$  und  $b$  sukzessive angepasst, um das Ergebnis zu optimieren. Das Gewicht  $b$  wird als neuronnen-

<sup>26</sup>Engl. *neurons*, auch als lineare Perzeptrons bezeichnet.

<sup>27</sup>Engl. *activation function*, auch als Transferfunktion (engl. *transfer function*) bezeichnet.

<sup>28</sup>Engl. *activation value*.

<sup>29</sup>Engl. *weights*.

## 2.5 Reinforcement Learning als Verständnisgrundlage

spezifische Verzerrung<sup>30</sup> bezeichnet, deren Eingangssignal stets 1 ist (VASILEV ET AL. 2019, S. 38 f.).

Die Anwendung von nicht-linearen Aktivierungsfunktionen ermöglicht, mit künstlichen Neuronen nicht-lineare Zusammenhänge abzubilden<sup>31</sup>. Häufig verwendet werden hierbei Schwellenwert-, stückweise lineare und Sigmoid-Funktionen (für Details zu den genannten Funktionen s. PAASS & HECKER 2020, S. 83). Besonders bei Deep-Learning-Modellen (s. nächster Absatz) kommen sog. Rectified Linear Units (ReLU)<sup>32</sup> zum Einsatz, die als Positiv-Teil ihres Arguments definiert sind (s. Abbildung 2.11; VASILEV ET AL. 2019, S. 44 f.).

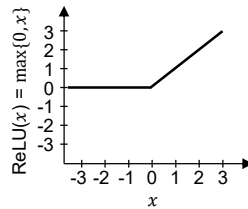


Abbildung 2.11: Aktivierungsfunktion ReLU nach PAASS & HECKER (2020, S. 83) und GOODFELLOW ET AL. (2016, S. 170)

Zum Aufbau von NN wird eine größere Anzahl an Neuronen nebeneinander zu einer Schicht<sup>33</sup> angeordnet, die wiederum mit mehreren weiteren Schichten hintereinandergeschaltet und verknüpft wird<sup>34</sup>. Es gibt jeweils eine oder mehrere Ein- und Ausgabeschichten als Start- bzw. Endpunkte des Informationsflusses in einem NN. Dazwischen liegt mindestens eine, meist jedoch eine Vielzahl an Zwischen- oder Aktivitätsschichten<sup>35</sup> (VASILEV ET AL. 2019, S. 42 f.).

<sup>30</sup>Engl. *bias*. Die Verzerrung dient mathematisch gesprochen dazu, die Aktivierungsfunktion auf der y-Achse zu verschieben und ermöglicht somit eine größere Flexibilität bei der Annäherung von Funktionen durch das Neuron (PAASS & HECKER 2020, S. 59).

<sup>31</sup>Die gewichtete Summe  $\sigma = \sum_i x_i w_i$  entspricht einer linearen Funktion. Somit würde ein NN, welches nur mit dieser arbeitet, lediglich als simples lineares Regressionsmodell fungieren. Durch die Anwendung von nicht-linearen Aktivierungsfunktionen sind komplexere, nicht-lineare Abbildungen möglich (VASILEV ET AL. 2019, S. 44).

<sup>32</sup>Auch: *rectifier*, dt. Gleichrichter.

<sup>33</sup>Engl. *layer*.

<sup>34</sup>Je nach Art und Dichte der Verknüpfung spricht man von unterschiedlichen Typen von NN, bspw. vollständig verbunden (engl. *dense/fully connected*), dünn besetzt (engl. *sparse*) oder rekurrent (engl. *recurrent*; VASILEV ET AL. 2019, S. 42 f., S. 74).

<sup>35</sup>Auch: verborgene Schicht, engl. *hidden layer*.

## 2 Stand der Erkenntnisse

Diese mehrschichtigen Netze werden als Deep Neural Networks (DNN) bezeichnet (GOODFELLOW ET AL. 2016, S. 165). Die Anzahl der Schichten kann theoretisch unbegrenzt sein, jedoch steigt mit jeder hinzukommenden Schicht die zum Betrieb des Netzes erforderliche Rechenleistung (KRUSE ET AL. 2015, S. 76 f.).

Convolutional Neural Networks (CNN) sind eine spezielle Art von DNN, die Input in Form von Matrizen verarbeiten können und deren Haupteinsatzgebiet in der Mustererkennung und Klassifizierung von Bilddaten liegt (GOODFELLOW ET AL. 2016, S. 164). Ihre Funktionsweise basiert auf dem mathematischen Prinzip der Faltung<sup>36</sup> (GOODFELLOW ET AL. 2016, S. 327–329). CNN bestehen aus sich abwechselnd wiederholenden Schichten aus Filtern<sup>37</sup> zur Faltung sowie Aggregations-Schichten<sup>38</sup> zur Verdichtung des Eingangssignals. Zuletzt schließt ein vollständig verbundenes, als Klassifizierer bezeichnetes NN das CNN ab (VASILEV ET AL. 2019, S. 112 f.). Abbildung 2.12 zeigt schematisch den Aufbau eines CNN, der im Folgenden näher erläutert wird.

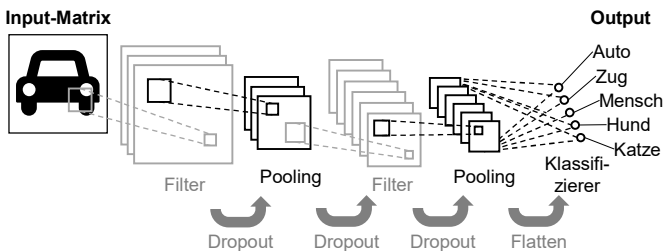


Abbildung 2.12: Schematischer Aufbau eines CNN nach VASILEV ET AL. (2019, S. 112)

Aufgabe der Filter-Schichten im CNN ist, die relevanten Informationen aus den matrixförmigen Eingangsdaten zu exzerpieren, bei der Mustererkennung z. B. Linien oder Kanten im Bildmaterial. Der Matrix-Input wird zunächst von einer festgelegten Anzahl an Filtern mit einer festen Größe<sup>39</sup> analysiert. Der Filter

<sup>36</sup>Engl. *convolution*.

<sup>37</sup>Engl. *convolutional layer*.

<sup>38</sup>Engl. *pooling layer*.

<sup>39</sup>Engl. *kernel size*; z. B. 2x2 oder 3x3 Pixel, in jedem Fall kleiner als die Dimension der Input-Matrix.

## 2.5 Reinforcement Learning als Verständnisgrundlage

scannt dabei die gesamte Matrix ab, indem er nach jeder Operation mit einer definierten Schrittweite<sup>40</sup> weiter vorrückt (vgl. Abbildung 2.13).

Aus den Pixelwerten im aktuellen Sichtfenster und seinen festgelegten Gewichten bestimmt der Filter eine Ergebnismatrix, indem er die Werte miteinander verrechnet bzw. faltet und auf diese Weise Muster in den Daten erkennt. Dabei wird die Größe der Input-Matrix um ein bestimmtes Verhältnis verringert (vgl. Abbildung 2.14; VASILEV ET AL. 2019, S. 97–98). Auf die Filter-Schicht folgt eine Aggregations-Schicht (als Pooling-Layer bezeichnet), welche überflüssige Informationen aus den Daten entfernt. Dazu wird die Ergebnismatrix der Filters aggregiert, indem nur die jeweils stärkste Information weitergegeben wird (z. B. höchster Wert in der Matrix). Dadurch wird die Repräsentation des Inhalts abstrahiert und die Dimension der weiterzugebenden Daten reduziert (VASILEV ET AL. 2019, S. 110 f.). In CNN folgen stets mehrere Filter-Pooling-Kombinationen hintereinander, wie in Abbildung 2.12 schematisch dargestellt.

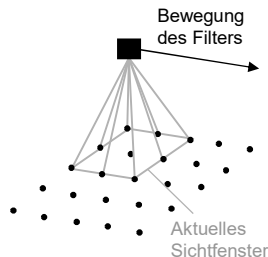


Abbildung 2.13: Schematische Darstellung des Filters im CNN, in Anlehnung an VASILEV ET AL. (2019, S. 97–99)

Des Weiteren kommen sog. Dropout<sup>41</sup>-Layer als Regularisierungsmethode zur Reduktion von Overfitting zum Einsatz. Dazu wird beim Training des Netzes bei jedem Durchlauf eine bestimmte Anzahl Neuronen in jeder Schicht des NN zufällig ausgewählt und deaktiviert. Dadurch wird sichergestellt, dass jedes Neuron einzeln ausreichend trainiert wird und sich nicht zu sehr an die

<sup>40</sup>Engl. *stride*.

<sup>41</sup>Engl. *dropout*: aussetzen.

## 2 Stand der Erkenntnisse

---

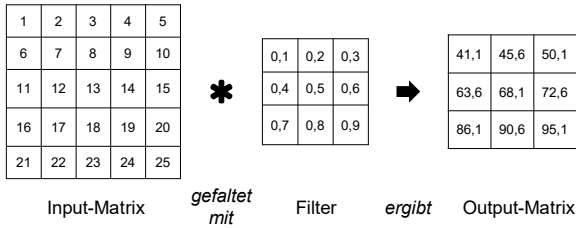


Abbildung 2.14: Beispielhafte Darstellung der Faltung durch den Filter, in Anlehnung an VASILEV ET AL. (2019, S. 97–99)

benachbarten Neuronen angleicht. So werden zu große Unterschiede im Trainingsniveau der Neuronen vermieden (SRIVASTAVA ET AL. 2014; VASILEV ET AL. 2019, S. 117 f.). Als letzte Schicht folgt im CNN ein reguläres, voll verbundenes NN, in diesem Kontext auch als Klassifizierer bezeichnet. Der mehrdimensionale Matrix-Output aus den Filtern und Pooling-Layern wird an dieser Stelle auf eine Dimension abgewickelt<sup>42</sup>, da ein voll verbundenes NN nur eindimensionale Vektoren verarbeiten kann. Für jede zu unterscheidende Klasse findet sich ein Neuron als Ausgang<sup>43</sup> des CNN (VASILEV ET AL. 2019, S. 113 f.).

**Unsupervised Learning (UL):** Das UL erfolgt datengetrieben, da der Agent aus einer großen Menge nicht gelabelter Daten selbstständig Ähnlichkeiten erkennen soll. Das Ziel ist die Identifikation von Strukturen oder Häufungen in den Daten und die Bildung von Clustern daraus (ERTEL 2013, S. 230). Haupt Einsatzgebiete für UL-Algorithmen sind das Clustering und damit verbunden die Anomalie-Erkennung sowie die Dimensionsreduktion (DÖBEL ET AL. 2018, S. 10). Verbreitete Algorithmen sind z. B. k-means (GÜNTHER 2018, S. 35), der Nearest-Neighbour-Algorithmus (ERTEL 2013, S. 240) oder die Kernel Principal Component Analysis (PCA; DÖBEL ET AL. 2018, S. 10). Die Lernmethode des UL ist für die vorliegende Arbeit nicht relevant und wird daher an dieser Stelle nicht weiter erläutert.

---

<sup>42</sup>Engl. *flatten*; diese Schichten fallen unter den Oberbegriff *reshape layer* (etwa: Umformungsschicht).

<sup>43</sup>Engl. *output layer*.

**Reinforcement Learning (RL):** Dieses Lernparadigma unterscheidet sich grundlegend von SL und UL, da keine Trainingsdaten verfügbar sind (ERTEL 2013, S. 287). Stattdessen fokussiert RL auf zielgerichtetes, selbstständiges Lernen und sequenzielles Treffen von Entscheidungen in Hinblick auf ein gegebenes Ziel. Dazu interagiert der Agent mit seiner Umgebung, die ihm für jede Aktion eine Belohnung als Rückmeldung gibt (SUTTON & BARTO 2018, S. 1). Aufgabe des Agenten ist, durch Versuch und Irrtum<sup>44</sup> herauszufinden, welche Aktionen in welcher Situation vorteilhaft sind und welche nicht (ERTEL 2013, S. 287). Da das Prinzip des RL für die vorliegende Arbeit von zentraler Bedeutung ist, wird im Folgenden ausführlich auf die relevanten Grundlagen eingegangen.

### 2.5.2 Reinforcement Learning im Detail

#### 2.5.2.1 Aufbau und Funktionsweise

**Markov-Entscheidungsprozess (MEP):** RL setzt voraus, dass die zu lösende Problemstellung als Markov-Entscheidungsprozess (MEP)<sup>45</sup> formuliert werden kann (GÜNTHER 2018, S. 44). Der MEP ist ein weit verbreiteter Formalismus, um sequentielle Entscheidungsprobleme in stochastischen Umgebungen, wie sie bei RL-Anwendungen vorliegen, mathematisch zu beschreiben (BISCHOFF 2015, S. 18).

Ein MEP wird durch das Tupel  $M = \{S, \mathcal{A}, \mathcal{R}, \mathcal{P}\}$  charakterisiert.  $S$  bezeichnet die Menge der möglichen Zustände<sup>46</sup>,  $\mathcal{A}$  die Menge der zulässigen Aktionen<sup>47</sup>,  $\mathcal{R}: S \times \mathcal{A} \times S \rightarrow \mathcal{R} \subset \mathbb{R}$ <sup>48</sup> die durch einen Übergang erzielten Belohnungen und  $\mathcal{P}: S \times \mathcal{A} \times S \rightarrow [0, 1]$ <sup>49</sup> die Übergangswahrscheinlichkeiten zwischen den Zuständen (vgl. Abschnitt 2.5.2.2; FEINBERG & SHWARTZ

---

<sup>44</sup>Engl. *trial and error*.

<sup>45</sup>Engl. *Markov Decision Process (MDP)*.

<sup>46</sup>Zustandsraum, engl. *state space*.

<sup>47</sup>Aktionsraum, engl. *action space*.

<sup>48</sup>Bedeutung: Die Menge der Belohnungen  $\mathcal{R}$ , die für den Übergang von einem Zustand  $S$  durch die Aktion  $\mathcal{A}$  in einen anderen Zustand  $S$  erzielt werden, besteht aus reellen Zahlen  $\mathcal{R}$  und ist damit eine Teilmenge von  $\mathbb{R}$ .

<sup>49</sup>Bedeutung: Die Menge der Übergangswahrscheinlichkeiten  $\mathcal{P}$ , von einem Zustand  $S$  durch die Aktion  $\mathcal{A}$  in einen anderen Zustand  $S$  zu gelangen, liegt im geschlossenen Intervall von 0 bis 1.

## 2 Stand der Erkenntnisse

2002, S. 3; GÜNTHER 2018, S. 44). Meist werden sog. finite MEP mit einer endlichen Anzahl an Zuständen und Aktionen aufgestellt (BISCHOFF 2015, S. 18). Man bezeichnet einen MEP als diskret, wenn die Zustands- und Aktionsräume nicht kontinuierlich sind, sondern diskrete Elemente enthalten (FEINBERG & SHWARTZ 2002, S. 3). Ein MEP gilt als (vollständig) beobachtbar, wenn dem Agenten alle relevanten Zustandseigenschaften zugänglich sind, andernfalls gilt der Prozess als nicht oder nur teilweise beobachtbar<sup>50</sup> (SUTTON & BARTO 2018, S. 466).

**Agent und Umgebung:** Abbildung 2.15 zeigt den Aufbau eines RL-Systems. Der Agent strebt ein vorgegebenes Ziel an, indem er durch Versuch und Irrtum lernt, was zu tun ist. Zu Beginn hat der Agent kein Vorwissen, wie er sein Ziel erreichen kann und es gibt keine Trainingsdaten, von denen er lernen kann. Stattdessen beobachtet, interagiert und verändert der Agent seine Umgebung mit der Zeit, indem er in jedem Zeitschritt  $t \in \{0, \dots, T\}$ <sup>51</sup> eine oder mehrere Aktionen  $a_t \in \mathcal{A}$  ausführt.

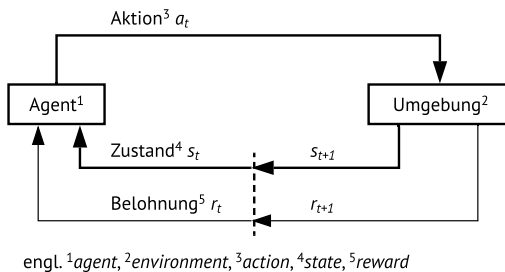


Abbildung 2.15: Aufbau eines RL-Systems nach SUTTON & BARTO (2018, S. 48)

Die Umgebung kann durch den Agenten, aber auch ohne dessen Zutun von außerhalb modifiziert werden. Der Agent nimmt seine Umgebung als Folge von Beobachtungen wahr, welche den Zustand  $s_t \in \mathcal{S}$  der Umgebung zum Zeitpunkt  $t$  beschreiben (SUTTON & BARTO 2018, S. 48; REBALA ET AL. 2019,

<sup>50</sup>Engl. *Partially Observable Markov Decision Process (POMDP)*.

<sup>51</sup> $T$ : finaler Zeitschritt.



S. 195–197). Für jede Aktion  $a_t$  erhält der Agent im nächsten Zeitschritt eine Belohnung  $r_{t+1} \in \mathbb{R}$  und findet sich im nächsten Zustand  $s_{t+1}$  wieder. Auf Basis der erzielten Belohnung wählt er die nächste Aktion  $a_{t+1}$  aus (ERTEL 2013, S. 290; SUTTON & BARTO 2018, S. 48). Dabei folgt er einer Strategie  $\pi: \mathcal{S} \rightarrow \mathcal{A}$ <sup>52</sup>, um die langfristige Belohnung zu maximieren (vgl. Abschnitt 2.5.2.2; GÜNTHER 2018, S. 44).

Die Wahrscheinlichkeit  $p$ , mit welcher der Agent durch eine gewählte Aktion  $a_t$  in einen möglichen Folgezustand  $s_{t+1}$  gelangt, hängen nur vom vorhergehenden Zustand  $s_t$  und der Aktion  $a_t$  ab, und nicht von früheren Zuständen des Agenten (sog. „Markovsche Eigenschaft“ eines MEP, ERTEL 2013, S. 291; KRAMER 2009, S. 103). Der momentane Zustand beinhaltet also alle Informationen, die zur Zukunftsprognose und somit zur Entscheidungsfindung notwendig sind (MURPHY 2012, S. 589). Diese Abhängigkeit gilt auch für die Belohnung  $r_t$  (ERTEL 2013, S. 291).

### 2.5.2.2 Lernprozess

**Belohnung:** Das Hauptziel des Agenten ist die Maximierung der Belohnung, welche er von der Umgebung erhält. Der Fokus liegt dabei nicht auf der direkten Belohnung<sup>53</sup> nach Ausführung einer Aktion, sondern auf der kumulierten Belohnung<sup>54</sup> über die Zeit, also nach einer längeren Folge von Aktionen (REBALA ET AL. 2019, S. 197 f.). Dazu wird versucht, die erwartete Summe der zukünftigen Belohnungen, den sog. Gewinn  $R_t$  zu maximieren (auch als kumulierte/diskontierte Belohnung bezeichnet<sup>55</sup>, vgl. Gleichung 2.5). Die zukünftigen Belohnungen werden dabei anhand von Prognosen und Erfahrungswerten abgeschätzt.

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \quad (2.5)$$

<sup>52</sup>  $\pi$  gibt die Wahrscheinlichkeit an, mit der sich der Agent in einem Zustand aus der Menge  $\mathcal{S}$  für eine bestimmte Aktion aus der Menge  $\mathcal{A}$  entscheidet.

<sup>53</sup> Engl. *immediate reward*.

<sup>54</sup> Engl. *delayed* oder *cumulated reward*.

<sup>55</sup> Engl. *return*.

## 2 Stand der Erkenntnisse

---

Der Diskontierungsfaktor  $\gamma$  gewichtet sofortige gegen zukünftige Belohnungen und bestimmt die Weitsicht des Agenten. Geht  $\gamma$  gegen 1, agiert dieser weit-sichtig unter hoher Einbeziehung zukünftiger Belohnungen. Bei einem Wert nahe 0 für  $\gamma$  handelt der Agent kurzsichtig und berücksichtigt verstärkt direkte Belohnungen (SUTTON & BARTO 2018, S. 57).

In deterministischen Umgebungen wird bei Ausführung einer bestimmten Ak-tion reproduzierbar derselbe nächste Zustand und damit auch die zugehörige Belohnung erreicht. Dagegen können für stochastische Umgebungen nur Über-gangswahrscheinlichkeiten in nächste Zustände angegeben werden (REBALA ET AL. 2019, S. 200–202).

Die Interaktion mit der Umgebung und damit verbundene Belohnungen wer-den als episodisch bezeichnet, wenn die Handlungsabfolgen in sich wiederholende Zyklen unterteilt sind und es jeweils einen finalen Zustand gibt, der eine Episode beendet<sup>56</sup>. Dem gegenüber stehen kontinuierliche Interaktionen<sup>57</sup> (SUT-TON & BARTO 2018, S. 54 f.).

In der Praxis birgt die Ausgestaltung der Belohnung<sup>58</sup> häufig große Herausfor-derungen. Dies liegt daran, dass die Ziele des übergeordneten Optimierungs-problems oft nicht direkt als Lernziele für den Agenten genutzt werden können. Im ungünstigsten Fall lernt der Agent Zusammenhänge, die nicht zur Lösung des eigentlichen Problems beitragen oder sogar kontraproduktiv sind. Die Belohnung muss also so gestaltet werden, dass der Agent sein Verhalten in der Art anpasst, wodurch die übergeordnete Zielsetzung erreicht wird (SUTTON & BARTO 2018, S. 469 f.; ESCHMANN 2021).

**Wertefunktion und Strategie:** Da die Umgebung im RL dem Agenten keine Information bereitstellt, welche Folgen seine Aktionen haben werden bzw. wie vorteilhaft diese für die Zielerreichung sind, muss dieser diese Zusammenhänge erlernen (REBALA ET AL. 2019, S. 197 f.). Stets in Hinblick auf sein Ziel

---

<sup>56</sup>Beispiel: Das Gewinnen oder Verlieren einer Partie Schach. Die nächste Partie beginnt unabhängig vom Ausgang der letzten und bildet somit eine neue Episode.

<sup>57</sup>Beispiel: Die Bewegungssteuerung eines Laufroboters. Ziel ist, den Roboter kontinuierlich am Laufen zu halten, es gibt kein definiertes Ende.

<sup>58</sup>Engl. *reward design*.

bewertet der Agent die Vorteilhaftigkeit eines Zustands bzw. der Durchführung einer Aktion zur Erreichung dieses Zustands (Zustands-Aktions-Paar) anhand der sog. Wertefunktion<sup>59</sup> auf Basis der zu erwartenden kumulierten Belohnungen  $R_t$ . Die Zuordnung von getätigten Aktionen zu Zuständen und Belohnungen wird als Strategie  $\pi$  bezeichnet<sup>60</sup> (GÜNTHER 2018, S. 43). Ziel ist hierbei die Entwicklung einer Strategie mit der höchsten erwarteten kumulierten Belohnung (SUTTON & BARTO 2018, S. 58). Verfolgt der Agent eine Strategie  $\pi$ , beschreibt  $\pi(a|s)$  die Wahrscheinlichkeit, dass dieser in Zustand  $s$  die Aktion  $a$  wählt.

Die Lösung eines RL-Problems ist gleichbedeutend mit dem Auffinden einer optimalen Strategie im MEP. Diese entspricht einer Sequenz von Entscheidungen, die zur höchsten erwarteten kumulierten Belohnung führt. Daher ist das Ziel der verschiedenen RL-Methoden die Bestimmung und Optimierung der Strategie des Agenten<sup>61</sup> (DAMMANN 2020, S. 2; SUTTON & BARTO 2018, S. 58). Hierbei wird unterschieden, ob der Agent eine explizite Zuordnung von Aktionen zu Zuständen als Strategie vornimmt und während des Lernprozesses zur Auswahl neuer Aktionen anwendet<sup>62</sup> oder lediglich eine Wertefunktion speichert<sup>63</sup>. Die Strategie ist hierin implizit repräsentiert, indem die Aktion mit dem höchsten Wert gewählt wird. Es existieren auch Methoden, die beide Techniken kombinieren<sup>64</sup> (BISCHOFF 2015, S. 21 f., S. 31).

**Umgebungsmodell:** Hinsichtlich der Bewertung von möglichen Nachfolgezuständen unterscheidet man RL-Agenten nach Einsatz modellbasierter und modellfreier Methoden. Erstere nutzen ein Modell des MEP bzw. der Umgebung, das entweder zuvor bekannt ist oder aus Datensätzen mittels Funktionsapproximation erlernt wurde. Werden zur Funktionsapproximation DNN eingesetzt, spricht man von Deep RL (SUTTON & BARTO 2018, S. 475). Zudem

---

<sup>59</sup>Zustandswertefunktion  $v_\pi(s)$ , engl. *state-value function*, Aktionswertefunktion  $q_\pi(s, a)$ , engl. *action-value function*.

<sup>60</sup>Engl. *policy*.

<sup>61</sup>Eine detaillierte Vorstellung der Vielzahl an Methoden und Algorithmen würde den Rahmen dieser Arbeit sprengen. BISCHOFF (2015, S. 21 ff.) gibt einen kompakten Überblick, für detaillierte Erläuterungen sei auf die entsprechenden Kapitel bei SUTTON & BARTO (2018) verwiesen.

<sup>62</sup>Engl. *policy based*.

<sup>63</sup>Engl. *value function based*.

<sup>64</sup>Engl. *actor-critic*.

## 2 Stand der Erkenntnisse

---

kann das Modell als Simulations- bzw. Trainingsumgebung genutzt werden, um die Strategie zu evaluieren und zu verbessern (BISCHOFF 2015, S. 21). Meist nutzen modellbasierte RL-Methoden die vorhandenen Daten effizient und erfordern wenig Interaktion mit der Umgebung, um eine Strategie zu erlernen. Jedoch kann das Modell Fehler in sich oder gegenüber der Realität aufweisen, die den Lernerfolg stark beeinträchtigen können (SUTTON & BARTO 2018, S. 6 f., S. 159 f.).

Modellfreie Algorithmen lernen eine Strategie direkt aus Daten, ohne das System zu modellieren. Dies kann von Vorteil sein, wenn das Systemmodell komplex und dessen Annäherung, insbesondere der Übergangswahrscheinlichkeiten, schwierig ist (BISCHOFF 2015, S. 21). Jedoch erfordern modellfreie Methoden sehr umfangreiche Interaktion mit der Umgebung, bevor eine Strategie gelernt wird (SUTTON & BARTO 2018, S. 6 f., S. 159 f.).

**Erkunden und Verwerten:** Ein weiterer wichtiger Aspekt im RL ist die Balance zwischen dem Verwerten<sup>65</sup> der Strategie und dem Erkunden<sup>66</sup> bislang nicht bekannter ggf. vorteilhafterer Zustände (SUTTON & BARTO 2018, S. 3, S. 27 f.; DAMMANN 2020, S. 7 f.). Dabei entsteht das Dilemma, dass der Agent einerseits optimale Aktionen erlernen soll, andererseits muss er sich aber nicht-optimal verhalten, um den Aktionsraum möglichst vollständig zu erkunden mit dem Ziel, ebendiese Aktionen zu finden (SUTTON & BARTO 2018, S. 103).

Zu diesem Zweck wurden verschiedene Arten von Lernprozessen entwickelt. Bei den strategiesynchronen<sup>67</sup> Lernprozessen werden Erfahrungen genutzt, die durch Anwendung einer bekannten Strategie<sup>68</sup> erzeugt wurden. Die strategiesynchronen<sup>69</sup> Lernprozesse hingegen erzeugen diese Erfahrungen durch eine zusätzliche zweite Strategie<sup>70</sup>, wodurch ein größerer Anteil an Exploration gewährleistet ist (BISCHOFF 2015, S. 36; SUTTON & BARTO 2018, S. 103, S. 129, S. 197).

---

<sup>65</sup>Engl. *exploitation*.

<sup>66</sup>Engl. *exploration*.

<sup>67</sup>Engl. *on-policy*.

<sup>68</sup>Engl. *target policy*.

<sup>69</sup>Engl. *off-policy*.

<sup>70</sup>Engl. *behaviour policy*.

## 3 Anforderungsanalyse und Lösungsansatz

### 3.1 Anforderungen an die Methode

Die Darstellung des Stands der Erkenntnisse im vorhergehenden Kapitel zeigt eine Vielzahl an Einzelbausteinen als Lösungsansätze für Teilprobleme des Auslegungsprozesses für OS. Wie jedoch in Abschnitt 1.2 beschrieben, ist das übergeordnete Ziel dieser Arbeit die Schaffung einer durchgängigen Methode zur automatisierten Generierung von OS. Diese muss als Gesamtmethode bestimmte Anforderungen erfüllen, die im folgenden Kapitel aufgestellt werden. Die einzelnen Bausteine werden nicht jeweils separat mit diesen Anforderungen abgeglichen, da diese ausschließlich für die Gesamtmethode relevant und sinnvoll zu betrachten sind.

#### 3.1.1 Allgemeine Anforderungen

Um die praktische Anwendbarkeit der Methode zu gewährleisten, müssen folgende vier grundlegende, allgemeine Anforderungen an die zu erarbeitende Methode erfüllt werden, die für die anwendungsnahe Forschung gelten (vgl. Abschnitt 1.4):

**Richtigkeit der Ergebnisse:** Als grundlegende Anforderung müssen die durch die Methode ermittelten Ergebnisse richtig sein. Dies wird maßgeblich durch die Wahl geeigneter Werkzeuge innerhalb der Methode bestimmt. Der Ansatz selbst muss zudem unabhängig von den gewählten Werkzeugen allgemeingültig richtig und anwendbar sein.

### 3 Anforderungsanalyse und Lösungsansatz

---

**Anwendbarkeit im industriellen Kontext:** Die Methode muss praktisch anwendbar sein. Dazu ist z. B. darauf zu achten, dass die vorgeschlagenen Methodenbausteine Eingangsdaten verwenden, die in der industriellen Praxis mit vertretbarem Aufwand verfügbar sind.

**Wirtschaftlichkeit:** Die Auslegungsmethode muss dem anwendenden Unternehmen in der Praxis einen wirtschaftlichen Mehrwert generieren. Dazu müssen die Kosten für die Einführung des neuen Prozesses kompensiert und die des bisherigen Vorgehens in der laufenden Anwendung unterschritten werden.

**Allgemeingültigkeit:** Die Methode muss allgemeingültig sein. Das bedeutet, dass sie mit geringen Anpassungen für verschiedene Anwendungsfälle anwendbar und gültig sein muss.

#### 3.1.2 Spezifische Anforderungen

Im Folgenden werden auf Basis der in Abschnitt 3.1.1 genannten allgemeinen Anforderungen spezifische definiert, welche für die konkrete Problemstellung dieser Arbeit (vgl. Abschnitt 1.1) relevant sind. Dabei stehen die in Abschnitt 1.2 erarbeiteten Teilzielsetzungen sowie die korrespondierenden Defizite des bisherigen Stands der Erkenntnisse (s. Abschnitt 2.4) im Vordergrund:

**Korrektheit:** Die mithilfe der Methode erzeugten OS müssen das Fördergut, für welches die Auslegung erfolgt ist, korrekt orientieren und die prognostizierte Effizienz erzielen.

**Langfristige Nutzbarkeit:** Um Wirtschaftlichkeit und industrielle Anwendbarkeit dauerhaft sicherzustellen, muss die Methode langfristig nutzbar sein. Dies gilt einerseits für die Struktur der Methode, welche dazu modular und somit erweiterbar aufgebaut sein muss. Andererseits muss das System sukzessive Erfahrungswissen aufbauen und wiederverwenden, z. B. mithilfe einer Wissensbasis zur Speicherung der errechneten OS.

**Erweiterbarkeit:** Wie in der vorhergehenden Anforderung hergeleitet muss die Methode erweiterbar sein, um industriell und langfristig anwendbar zu sein. Insbesondere die Erweiterbarkeit für neues Fördergut ist von zentraler Notwendigkeit.

**Rechenzeit:** Um die effiziente Anwendung der Methode zu gewährleisten, muss diese innerhalb kurzer Rechenzeiten zu guten Ergebnissen kommen. Im Vergleich mit bisherigen Entwicklungszeiten ist die Erreichung einer zeitlichen Größenordnung von einem bis mehreren Tagen Rechenzeit tragbar. Die exakte Bezifferung ergibt sich aus den Ressourcen (insb. verfügbare Rechenkapazität) und Anforderungen des konkreten Anwendungsfalls. Eine geringe Rechenzeit ermöglicht zudem die Skalierbarkeit für komplexe Auslegungsvorgänge und somit die langfristige Nutzbarkeit des Ansatzes, ohne an mögliche Systemgrenzen zu stoßen.

**Ergebnisqualität:** Die OS, welche durch die Methode automatisiert generiert wurden, sollen größtmögliche Qualität aufweisen, wobei die Kriterien zur Bewertung im Zuge der Methodenentwicklung zu überprüfen sind. Das Erreichen des Optimums ist je nach zugrundeliegendem Lösungsverfahren ggf. möglich und auch wünschenswert, muss aber hinsichtlich der Einhaltung der Anforderung der Rechenzeiteffizienz betrachtet werden, um die Anwendbarkeit im industriellen Kontext zu sichern.

**Unabhängigkeit von personenspezifischer Erfahrung:** Die Methode muss ohne die Notwendigkeit manueller Arbeitsschritte jederzeit gleichbleibend hochqualitative und reproduzierbare Ergebnisse erzielen. Diese müssen dabei unabhängig von Erfahrung und Tagesform der anwendenden Fachkraft sein.

Tabelle 3.1 zeigt die Zuordnung der spezifischen zu den allgemeinen Anforderungen. Zur Erfüllung der übergeordneten Anforderungen müssen die jeweils untergeordneten ausreichend erfüllt werden.

Basierend auf diesen Anforderungen wird im folgenden Abschnitt der Lösungsansatz für die Methode zur automatisierten Generierung von OS vorgestellt. Die

### 3 Anforderungsanalyse und Lösungsansatz

Erfüllung dieser Anforderungen durch den Lösungsansatz wird in Abschnitt 7.3 anhand der kritischen Bewertung der Forschungsergebnisse evaluiert.

Tabelle 3.1: Zuordnung der spezifischen zu den allgemeinen Anforderungen

Spezifische Anforderungen	Allgemeine Anforderungen			
	Richtigkeit der Ergebnisse	Anwendbarkeit im industriellen Kontext	Wirtschaftlichkeit	Allgemeingültigkeit
Korrektheit	●	●	○	●
Langfristige Nutzbarkeit	○	●	●	○
Erweiterbarkeit	○	●	○	●
Rechenzeit	○	●	●	○
Ergebnisqualität	●	●	●	○
Unabhängigkeit von personenspez. Erfahrung	○	●	●	○

**Legende:**

- Anforderung ohne Einfluss auf Erfüllung der allgemeinen Anforderung
- Anforderung ist zur Erfüllung der Hauptanforderung essenziell

### 3.2 Lösungsansatz: Methode zur automatisierten Generierung von Ordnungsschikanen

Wie in Kapitel 2 erläutert existieren bislang lediglich Ansätze zur Unterstützung von Teilaspekten der Auslegung von OS, die jedoch keine Durchgängigkeit des Prozesses und keine vollständige Entkopplung von personenspezifischem Erfahrungswissen ermöglichen. Daher wurde in der vorliegenden Arbeit eine Methode zur automatisierten Generierung von OS für VWF entwickelt. Diese adressiert die in Abschnitt 2.4 vorgestellten Defizite, indem sie die in den Abschnitten 3.1.1 und 3.1.2 aufgezeigten Anforderungen erfüllt.

Abbildung 3.1 zeigt eine Übersicht über die entwickelte Methode, die aus drei Haupt-Methodenschritten besteht und die im Folgenden genauer erläutert wird. Für die Grundstruktur der Methode wurde sich am Stand der Erkenntnisse



### 3.2 Lösungsansatz: Methode zur automatisierten Generierung von Ordnungsschikanen

---

orientiert (vgl. HANSSON ET AL. 2016; HILGENBÖCKER 1985, S. 88 f.; LO & DICK 1990; MATHIESEN 2017, S. 13; WOLFSTEINER 1999, S. 101 f.).

Als Eingangsdaten benötigt die Methode ein CAD-Modell eines zu orientierenden Förderguts. Im ersten Methodenschritt *Datenaufbereitung* werden durch eine Fördergut-Analyse relevante Fördergut-Charakteristika wie z. B. Geometrie, Masse, Material, Vorzugslagen und Fördergut-Klasse (auch bezeichnet als Fördergut-Typ) extrahiert. Anschließend unterscheidet sich der Methodenablauf abhängig von der Klasse des zu orientierenden Förderguts.

Entspricht das zu orientierende Fördergut einer Klasse, welche dem System bekannt ist, wird als nächstes der Förderprozess simuliert. Ziel dieses zweiten Methodenschritts ist die *Bewertung der SE* mithilfe von Simulationsdaten (CAD-Daten, Transitionsmatrizen). Für die Simulation werden generische, parametrierbare SE aus einer CAD-Bibliothek geladen, die bei Implementierung der Methode in Form einer Software einmalig befüllt werden muss. Diese SE werden abhängig von den Fördergut-Charakteristika parametrisiert und in die Physiksimulation geladen, um den Förderprozess mit Fördergut zu berechnen. Anhand des Verhaltens des simulierten Fördergut-Teils werden die SE nach der Erzielung einer gewünschten Orientierung bewertet.

Gehört das zu orientierende Fördergut zu einer unbekanntem Klasse, müssen Eingangsdaten in Form von Transitionsmatrizen berechnet werden, um später im dritten Methodenschritt das RL-System neu zu trainieren. Als Voraussetzung zur Berechnung dieser Eingangsdaten wird zunächst eine repräsentative Anzahl an CAD-Daten der neuen Fördergut-Klasse importiert, die vorher bereitstellen sind. Für diese Teile werden anhand der Fördergut-Charakteristika SE erzeugt. Anschließend erfolgt die Simulation und Bewertung im nachfolgenden Methodenschritt 2 analog zum oben bereits beschriebenen Ablauf für bekanntes Fördergut.

Zur *Generierung der OS* im dritten Methodenschritt kommt ein Reinforcement-Learning-System zum Einsatz, welches mit einer Auswahl an Fördergut-Klassen trainiert wurde, um für diese eine OS vorzuschlagen. Dabei hat der RL-Agent gelernt, wie Fördergut-Klasse und -Parameter mit verschiedenen SE interagieren und welche Kombinationen zu einer hohen Effizienz der OS führen. Zu Beginn

### 3 Anforderungsanalyse und Lösungsansatz

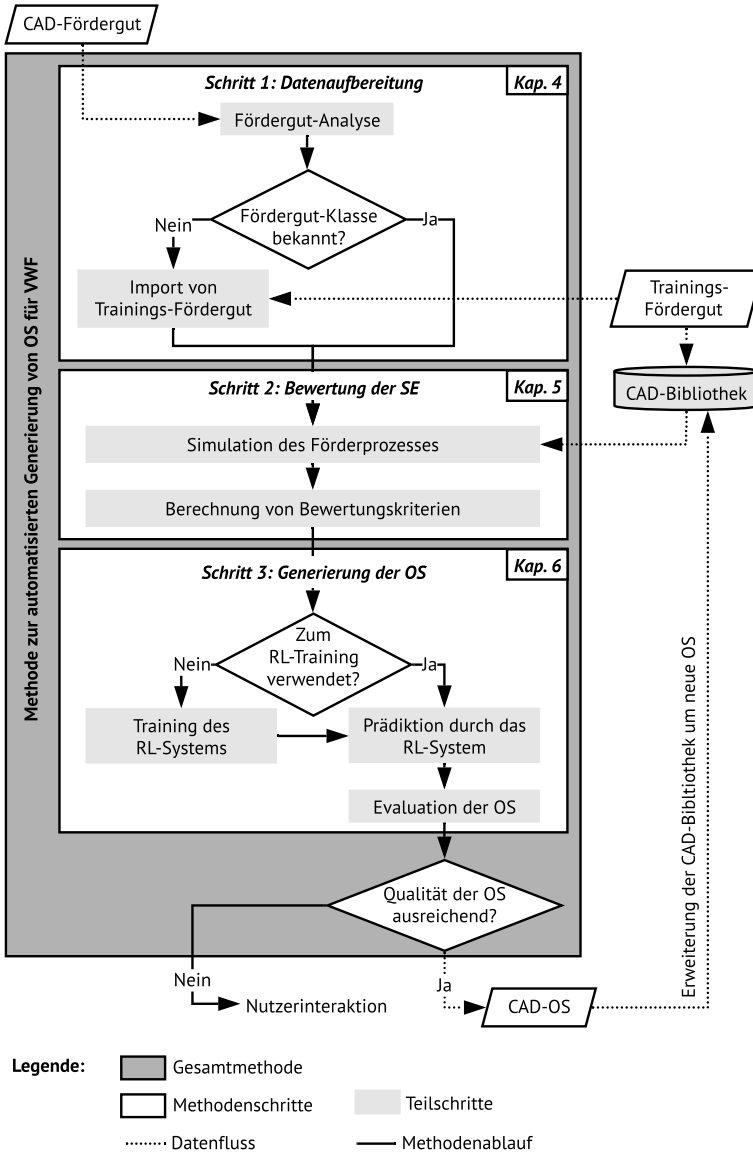


Abbildung 3.1: Methode zur automatisierten Generierung von OS für VWF (zu den erarbeiteten Bausteinen s. Kapitel 4 bis 6)

### 3.2 Lösungsansatz: Methode zur automatisierten Generierung von Ordnungsschikanen

---

des Methodenschritts wird geprüft, ob bereits alle SE-Fördergut-Kombinationen zum Training des RL-Systems verwendet wurden.

Falls ja, gehört das zu orientierende Fördergut zu einer der bekannten Klassen und das RL-System kann einen Designvorschlag für die OS präzisieren. Bei unbekannter Fördergut-Klasse wird das RL-System mit den in Schritt 2 bewerteten SE trainiert und die Prädiktion einer OS abgeschlossen.

Zur Evaluation wird die OS mit Fördergut-Teilen in die Simulation geladen und ggf. noch optimiert. Entspricht das Orientierungsverhalten einem vorher festzulegenden Qualitätskriterium, wird der CAD-Entwurf der OS als Ergebnis ausgegeben. Ist die Qualität nicht ausreichend, muss eine Nutzerinteraktion erfolgen, um z. B. die Generierung der OS mit veränderten Parametern zu wiederholen. Nach jedem durchgeführten OS-Auslegungsvorgang werden die OS-Daten in die SE-Bibliothek eingespeist und diese dadurch erweitert.

In den nachfolgenden Kapiteln 4 bis 6 werden die Methodenschritte sukzessive erarbeitet, sowie die verwendeten und angepassten Werkzeuge erläutert.



## 4 Methodenschritt 1: Datenaufbereitung

### 4.1 Überblick

Wie bereits in Abschnitt 3.2 erläutert, zielt der erste Methodenschritt *Datenaufbereitung* darauf ab, die Eingangsdaten für die *Bewertung der Schikanenelemente* (zweiter Methodenschritt) bereitzustellen. Abbildung 4.1 zeigt die erarbeiteten Bausteine, die in den folgenden Abschnitten 4.2 und 4.3 detailliert erläutert werden.

Die Fördergut-Analyse ist der erste Teilschritt innerhalb der *Datenaufbereitung*. Die Analyse der CAD-Daten des Förderguts muss automatisiert erfolgen, um sie zu Eingangsdaten für die Simulation des Förderprozesses zu verarbeiten bzw. die für das RL-System benötigten Daten zu exzerpieren. In Abschnitt 4.2.1 werden daher relevante Daten ausgewählt sowie ein Konzept zur Automatisierung der Extraktion von Fördergut-Charakteristika erarbeitet.

Die Berechnung von neuen OS wird durch Erkennung von Fördergut-Typen vereinfacht, indem das System Erfahrungswerte aus früheren OS-Auslegungsvorgängen nutzt. Dazu wird in Abschnitt 4.2.2 eine Ähnlichkeit-Analyse entwickelt, die das eingespeiste Fördergut klassifiziert (z. B. Pflzteil/Schraube, Scheibe, Mutter etc.). Die in Abschnitt 4.2.1 und 4.2.2 beschriebenen Bausteine sind Teil des Methodenschritts *Datenaufbereitung* und werden bei jedem Methodendurchlauf ausgeführt.

Die CAD-Bibliothek beinhaltet Modelle von SE und stellt diese als Eingangsdaten für die Simulation des Förderprozesses zur Verfügung. Die Bibliothek muss initial mit diesen CAD-Daten befüllt werden. In Abschnitt 4.3.1 erfolgt daher die Auswahl und Parametrierung von generischen SE. Deren Parameter werden erst später zur Anpassung an das Fördergut mit konkreten Werten belegt (im zweiten Methodenschritt, vgl. Abschnitt 5.3.1). Liegt während eines

## 4 Methodenschritt 1: Datenaufbereitung

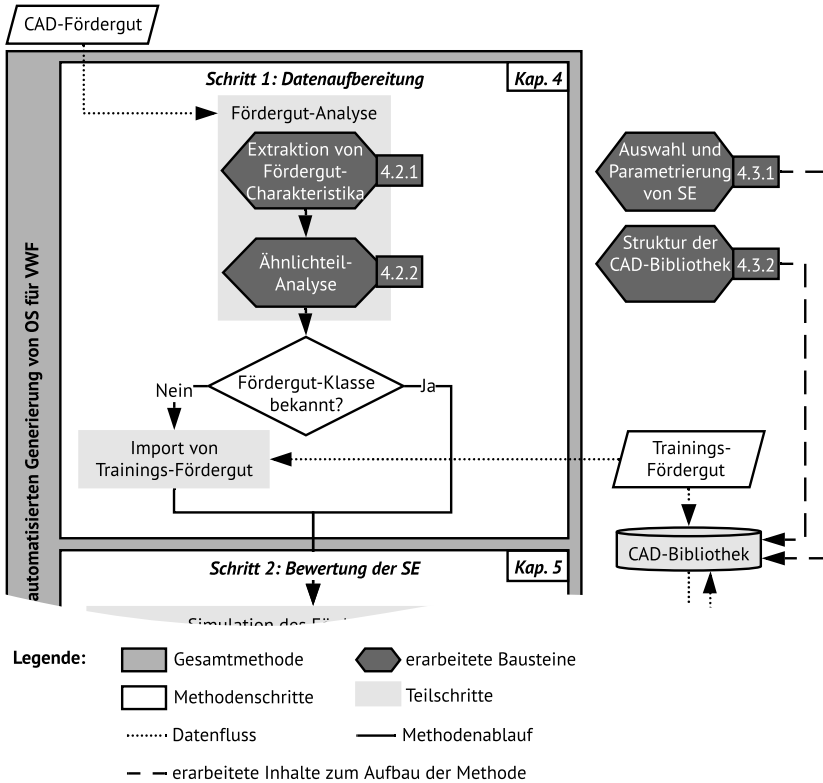


Abbildung 4.1: Erster Methodenschritt „Datenaufbereitung“ und erarbeitete Bausteine

Methodendurchläufe ein Fördergut-Teil einer unbekannt Klasse vor, werden extern bereitzustellende CAD-Daten von Trainings-Fördergut importiert. Damit erfolgt später im Methodenablauf (dritter Schritt) das erneute Training des RL-Systems.

Die CAD-Bibliothek soll zudem bei jedem Auslegungsvorgang Daten von Fördergut-Teilen und OS sowie deren Relationen speichern. Zu diesem Zweck wird in Abschnitt 4.3.2 die entsprechende Struktur der CAD-Bibliothek konzipiert. Die in Abschnitt 4.3.1 und 4.3.2 erarbeiteten Bausteine wurden im Rahmen dieser Arbeit einmalig erarbeitet und müssen nicht bei jedem Methodendurchlauf wiederholt werden.

### 4.2 Automatisierte Fördergut-Analyse

#### 4.2.1 Extraktion von Fördergut-Charakteristika

Um das Fördergut automatisiert auf Basis von dessen CAD-Daten charakterisieren zu können, müssen folgende Fragen beantwortet werden:

1. Welche Parameter sind relevant, um das Fördergut hinreichend zu beschreiben? (Abschnitt 4.2.1.1)
2. Welche Werkzeuge eignen sich zur automatisierten Erhebung der relevanten Parameter aus den CAD-Daten? (Abschnitt 4.2.1.2)

##### 4.2.1.1 Analyse der zu erhebenden Parameter

Um die erste Frage zu beantworten, wurde zunächst der Einsatzzweck der extrahierten Parameter geklärt. Davon abhängig ergeben sich eine Reihe von Anforderungen, welche die Parameter erfüllen müssen. Vor diesem Hintergrund wurde im nächsten Schritt die relevante Literatur erneut betrachtet und untersucht, welche Parameter dort zur Charakterisierung von Werkstücken herangezogen werden. Abschließend wurden diese den Anforderungen innerhalb dieser Arbeit gegenüber gestellt, um die relevanten Parameter zu identifizieren und auszuwählen (s. Tabelle 4.1 am Ende des Abschnitts).

**Klärung des Einsatzzwecks der extrahierten Parameter:** In dieser Arbeit sollen die Parameter in drei Feldern zum Einsatz kommen: Als erstes sollen sie zur Fördergut-spezifischen Parametrierung der generischen SE genutzt werden. Des Weiteren muss die Simulation des Förderprozesses mit den extrahierten Daten befüllt werden. Als drittes ermöglicht das Lernen von Zusammenhängen zwischen Fördergut-Charakteristika und erzeugter Wirkung der OS die Generierung von OS. Folgende Anforderungen an die zu definierenden Parameter wurden daher in den drei Einsatzfeldern ermittelt<sup>1</sup>:

---

<sup>1</sup>Hierfür wurde sich auf die Ausführungen von BOOTHROYD (2005, S. 55–87) sowie auf Erfahrungswerte aus der Praxis gestützt.

- **Fördergut-spezifische Parametrierung der generischen SE:** Die Einstellung der SE-Parameter muss abhängig von den Abmessungen des Förderguts erfolgen. So müssen Stufen beispielsweise eine Mindesthöhe aufweisen, damit darüber laufende Teile einen Kipp-Effekt erfahren, dürfen jedoch eine Maximalhöhe nicht überschreiten, da dies chaotisches Verhalten auslösen und damit den Förderprozess stören würde. Das Orientierungsverhalten der Werkstücke – abhängig von der Lage des Schwerpunkts und eventuell vorhandenen Symmetrie-Achsen – bestimmt maßgeblich die Wirksamkeit der SE. Zusätzlich ist das Vorhandensein von kennzeichnenden Formelementen meist eine Voraussetzung für die Möglichkeit zur Umorientierung.
- **Simulation des Förderprozesses:** Zur realitätsgetreuen Abbildung des Förderguts in der Simulation müssen dessen physikalische Eigenschaften wie Abmessungen, Masse, Lage des Schwerpunkts und Massenträgheitsmoment bekannt sein. Diese Größen bestimmen das dynamische Orientierungsverhalten des Förderguts, das korrekt simuliert werden soll. Die Wechselwirkung mit den Kontaktpartnern in der Simulation hängt von den Materialpaarungen ab. Auch Vorzugslagen und Symmetrieachsen haben großen Einfluss auf den Erfolg des Ordnungsvorgangs und sind daher relevante Größen.
- **Lernen von Zusammenhängen:** Der RL-Agent muss Zusammenhänge zwischen Fördergut-Typ, -Parametern und -Verhalten erlernen, um effektive OS für unbekanntes Fördergut prognostizieren zu können. Daher sind Verhaltens-Typ und Abmessungen essentielle Eingangsdaten für das RL-System.

**Werkstückparameter in der Literatur:** Auf Basis der eben genannten Anforderungen wurde die bereits in Abschnitt 2.2 vorgestellte Literatur herangezogen. Im Themenfeld der Handhabungstechnik beleuchten BOOTHROYD (2005)<sup>2</sup> und HESSE (2010) Werkstückparameter in diesem Kontext. Deren Ausführungen werden im Folgenden näher beschrieben. Zahlreiche Publikationen behandeln

---

<sup>2</sup>Basierend auf früheren Arbeiten des Autors (u. a. BOOTHROYD & HO 1977; BOOTHROYD ET AL. 1978).



die Thematik nur am Rande und referenzieren auf eine der beiden Monographien bzw. deren Vorarbeiten oder frühere Auflagen (z. B. HANSSON ET AL. 2016; HOFMANN 2014; LA BROOY & C. JIANG 2009; LIM ET AL. 1994; LO & DICK 1990; OU-YANG & MAUL 1993). Weitere Einzelveröffentlichungen präsentieren ähnliche Überlegungen, die in der Literatur jedoch nur vereinzelt referenziert werden (WEISS 1983, S. 30 f.; HILGENBÖCKER 1985, S. 88–90).

BOOTHROYD (2005, S. 385-407) wählt einen Ansatz zur schrittweisen Beschreibung von Werkstücken. Das Ziel seiner Ausführungen ist, Werkstücke in Gruppen mit ähnlichen Orientierungseigenschaften zu unterteilen. Die Wahl passender Orientierungseinrichtungen kann anhand dieser Eigenschaften erfolgen. Über ein Codierungs-System in mehreren Schritten erfolgt die Zuordnung von Teilen zu diesen Gruppen:

- **Grundform:** Diese wird anhand der Hüllgeometrie des Werkstücks bestimmt. Dazu wird der kleinste Zylinder oder das kleinstmögliche, reguläre oder rechtwinklige Prisma bestimmt, welcher das Teil vollständig umhüllt. Beispiele für die Grundform sind Zylinder, Quader oder flache Teile.
- **Kennzeichnende Eigenschaften:** Es werden geometrische und nicht geometrische Kennzeichen unterschieden: Erstere können Absätze, Fasen, Löcher oder Nuten sein, zweitere z. B. die Oberflächenbeschaffenheit. Es werden nur Eigenschaften erfasst, die mit dem Handhabungssystem interagieren können. Aus diesem Grund werden die geometrische Form des Werkstücks sowie Position und Ausprägung seiner Kennzeichen als wichtigste Faktoren bewertet.
- **Symmetrien:** Durch das Vorhandensein von Rotationssymmetrie können mehrere Orientierungen eines Werkstücks als gleichwertig gelten. Daher wird im dritten Schritt bestimmt, ob und wenn ja um wie viele und welche Achsen Rotationssymmetrien vorliegen.

Den geometrischen Eigenschaften des Werkstücks wird hohe Bedeutung beimessen, da diese bestimmen, wie gut sich ein Teil zuführen und orientieren lässt.

## 4 Methodenschritt 1: Datenaufbereitung

---

HESSE (2010, S. 14-17) widmet sich allgemein den Merkmalen von Teilen, die im Produktionsprozess verarbeitet werden. Den geometrischen Formen wird auch hier besonders hohe Bedeutung zugesprochen, da diese das Werkstückverhalten maßgeblich bestimmen. Dieses soll zur Handhabung ausgenutzt werden, vor allem für freie Werkstückbewegungen unter Schwerkrafteinfluss, wozu auch die Förderung im VWF zählt. Für den praktischen Einsatz leistet HESSE einen Beitrag zur allgemeinen Verhaltenslehre von Werkstücken, indem er Beschreibungsgrößen für Handhabungsobjekte strukturiert. Dazu werden die Werkstückmerkmale in sechs Kategorien unterteilt:

- **Geometrische Werkstückdaten:** Diese Kategorie beinhaltet Größen wie Abmessungen, Symmetrien und Seitenverhältnisse.
- **Kennzeichnende Formelemente:** Bohrungen, Absätze, Nuten, Schlitzlöcher oder Fasen spezifizieren die geometrische Form des Werkstücks weiter.
- **Physikalische Eigenschaften:** Werkstoff, Schwerpunkt, die damit verbundene Trägheit, Masse bzw. Gewicht oder Oberflächenbeschaffenheit charakterisieren das Fördergut über dessen Form hinaus.
- **Ruheverhalten:** Als Folge von geometrischen und physikalischen Eigenschaften ergeben sich z. B. Lagestabilität, Ordnungswahrscheinlichkeit und Vorzugslagen in ruhendem Zustand des Werkstücks.
- **Bewegungsverhalten:** In Bewegung zeigen sich Gleit-, Roll- oder Schüttfähigkeit sowie die Richtungsstabilität eines Handhabungsobjekts.
- **Verhaltenstyp:** Die Summe der bisher beschriebenen Werkstückmerkmale führt zum spezifischen Verhalten des Fördergut-Teils. Als Typen werden z. B. Wirt-, Flach-, Zylinder-, Kegel- oder Pilzteile unterschieden.

Auf der Basis dieser Beschreibungsgrößen können Funktionsträger für die Handhabung der Werkstücke ausgewählt werden. Besonderen Fokus legt HESSE (2010, S. 16 f.) auf die Einteilung der Verhaltenstypen (auch: Werkstückgrundform), da diese die Zuordnung von geeigneten Handhabungsgeräten erlauben. Anhand von Werkzeugen wie Entscheidungstabellen kann die Auswahl kriterienbasiert und systematisch erfolgen. Die Zugrundelegung von Verhaltenstypen für die Auslegung von OS wird explizit genannt, da die Werkstückgrundform maßgeblich das Handhabungsverhalten beeinflusst (HESSE 2000b, S. 91).

### **Abgleich und Auswahl der für diese Arbeit relevanten Fördergut-Parameter:**

Tabelle 4.1 zeigt eine Übersicht der betrachteten Parameter bei BOOTHROYD (2005) und HESSE (2010) im Vergleich mit den zu Beginn dieses Abschnitts erhobenen Anforderungen. Ein Großteil der Kenngrößen ist für die Parametrierung der SE, die Simulation des Förderprozesses und das Lernen von Zusammenhängen ebenfalls relevant und wird daher im Folgenden berücksichtigt (s. Spalte „Fazit“ in Tabelle 4.1).

Lediglich die bei BOOTHROYD (2005, S. 386 f.) betrachtete Hüllgeometrie bietet keine Anknüpfungspunkte im Rahmen dieser Arbeit und ist daher nicht relevant. Innerhalb des dort beschriebenen Coding-Systems dient sie als Hilfestellung, den Grundtyp eines Werkstücks zu bestimmen, was innerhalb der vorliegenden Arbeit anhand einer Ähnlichkeit-Analyse realisiert wird (vgl. Abschnitt 4.2.2).

Indirekt relevant ist das Massenträgheitsmoment eines Fördergut-Teils, das innerhalb der Simulation genutzt wird, um das dynamische Fördergut-Verhalten korrekt abzubilden. Es wird jedoch aus den geometrischen Eigenschaften und der Masse des Werkstücks berechnet und muss daher nicht extrahiert werden. Ähnlich dazu werden die Lage- und Richtungsstabilität des Förderguts und damit verbunden dessen Orientierungswahrscheinlichkeit durch die Simulation errechnet. Dieses gilt auch für die Gleit-, Roll- und Schütffähigkeit nach HESSE (2010, S. 16).

### **4.2.1.2 Werkzeuge zur automatischen Datenerhebung**

Tabelle 4.2 zeigt eine Übersicht über die Werkzeuge, welche zur Erhebung der erforderlichen Parameter identifiziert wurden. Wie daraus hervorgeht, ermöglicht die CAD-Repräsentation des Fördergut-Teils den Zugriff auf einen Großteil der erforderlichen Daten, z. B. mithilfe eines CAD-Programms oder einer Physiksimulation. So können sämtliche Abmessungen, Informationen zu Formelementen, Lage des Schwerpunkts (in Millimetern) und Symmetrieachsen (in Relation zu den Hauptachsen in Grad oder Radiant), Masse (in Kilogramm),

## 4 Methodenschritt 1: Datenaufbereitung

Table 4.1: Vergleich der betrachteten Fördergut-Parameter in der Literatur und der Anforderungen innerhalb dieser Arbeit

Parameter	...in der Literatur		...innerhalb dieser Arbeit			
	BOOTH- ROYD 2005	HESSE 2010	Parametrie- rung der SE	Simulation des Förder- prozesses	Lernen von Zu- sammen- hängen	Fa- zit
Hüllgeometrie	●	○	○	○	○	○
Abmessungen	○	●	●	●	●	●
Seitenverhältnisse	●	●	●	○	◐	●
Symmetrien	●	●	●	○	◐	●
Kennzeichnende Formelemente	●	●	●	○	◐	●
Werkstoff	○	●	○	●	○	●
Schwerpunkt	○	●	○	●	○	●
Masse	○	●	○	●	○	●
Massenträgheit	○	●	○	◐	○	◐
Oberflächen- beschaffenheit	○	●	○	●	○	●
Vorzugslagen	○	●	●	●	●	●
Ordnungs- wahrscheinlichkeit	○	●	○	◐	○	◐
Lage-/ Richtungs- stabilität	○	●	○	◐	○	◐
Gleit- und Roll- fähigkeit	○	●	○	◐	○	◐
Schüttfähigkeit	○	●	○	◐	○	◐
Werkstück- grundform/ Verhaltenstyp	●	●	●	○	●	●

**Legende:**

- betrachtet/relevant
- nicht betrachtet/relevant
- ◐ indirekt betrachtet/relevant

*Tabelle 4.2: Relevante Fördergut-Parameter und Werkzeuge zur Extraktion*

Parameter	Einheit	Werkzeug
Abmessungen	<i>mm</i>	} Feature-Extraktion mittels CAD-Programm bzw. in einer Physiksimulation
Seitenverhältnisse	-	
Lage und Dimension von kennzeichnenden Formelementen	<i>mm</i>	
Lage der Symmetrie- Achsen	<i>deg/rad</i>	
Lage des Schwerpunkts	<i>mm (x-y-z-Koordinaten)</i>	
Masse	<i>kg</i>	
Material	- ( <i>Werkstoffangabe</i> )	
Oberflächenbeschaffenheit	- ( <i>Rauheitswert; Reib-/Stoßkoeffizienten</i> )	CAD-Programm oder Tabellen zu Material- eigenschaften
Vorzugslagen	- ( <i>Zuordnung in %</i> )	Fall-Versuche in einer Physiksimulation
Verhaltenstyp	- ( <i>Zuordnung</i> )	Ähnlichteil-Analyse

Material sowie Oberflächenbeschaffenheit extrahiert werden<sup>3</sup>. Die anzuwendenden Reib- und Stoßkoeffizienten sind in gängigen CAD-Programmen hinterlegt oder können ergänzend aus entsprechenden Tabellen über die Materialpaarung zwischen Fördergut und SE bestimmt werden.

Zur Ermittlung der Vorzugslagen sind nach BOOTHROYD (2005, S. 55) sowie HESSE (2010, S. 41) Versuche durchzuführen, bei denen das Werkstück wiederholt auf eine horizontale Fläche fallen gelassen und die Orientierung aufgezeichnet wird, in der es zum Liegen kommt. Nach einer ausreichenden Zahl von Wiederholungen stellt sich eine statistische, prozentuale Verteilung über alle möglichen Orientierungen ein. Diese Fall-Versuche werden in dieser Arbeit mithilfe einer Physiksimulation implementiert, um die Vorzugslagen zu bestimmen, in Anlehnung an verschiedene jüngere Forschungsarbeiten (z. B. HARALDSON ET AL. 2020; MATHIESEN & ELLEKILDE 2016; REINHART & HOF-

<sup>3</sup>Zur automatisierten Extraktion von Parametern aus 3D-CAD-Daten sei auf die Veröffentlichung von LA BROOY & C. JIANG (2009) verwiesen, welche hierzu umfassende Vorarbeiten präsentiert haben.

## 4 Methodenschritt 1: Datenaufbereitung

---

MANN 2012). Anhand des Vorgehens von HARALDSON ET AL. (2020) kann eine automatisierte Clusterung von simulierten Orientierungen vorgenommen werden, sodass deren experimentelle Bestimmung entfällt.

Zur Bestimmung des Werkstück- bzw. Verhaltenstyps wurde eine Ähnlichkeit-Analyse mithilfe eines künstlichen neuronalen Netzes entwickelt, welche in Abschnitt 4.2.2 ausführlich erläutert wird.

### 4.2.2 Ähnlichkeit-Analyse

Wie in Abschnitt 4.2.1 beschrieben, ist der Verhaltenstyp eines Fördergut-Teils eine der notwendigen Eingangsinformationen für das RL-System zur Generierung der OS (vgl. Kapitel 6). Um die Klassifizierung des Werkstücks zu automatisieren, wurde eine Ähnlichkeit-Analyse entwickelt<sup>4</sup>. Dazu wurde folgendes Vorgehen angewendet:

1. Analyse potentiell geeigneter Ansätze zur Ähnlichkeit-Analyse (Abschnitt 4.2.2.1)
2. Erarbeitung eines Datensets und der CNN-Architektur<sup>5</sup> (Abschnitt 4.2.2.2)
3. Training und Evaluation des CNN (Abschnitt 4.2.2.3)

#### 4.2.2.1 Analyse potentiell geeigneter Ansätze zur Ähnlichkeit-Analyse

Zunächst wurden bestehende Ansätze im Themenfeld der Werkstück- und Ähnlichkeit-Analyse untersucht. BOOTHROYD (2005, S. 385-407) stellt eine regelbasierte Richtlinie zur Klassifizierung von Werkstücken vor, die als Basis für die Auswahl von Prinziplösungen für OS dient. Die Einordnung in bestimmte Klassen erfolgt ausschließlich manuell, eine Automatisierung wird nicht adressiert. Die regelbasierte Einordnung von Werkstücken wird in weiteren Arbeiten als Werkzeug angewandt (z. B. Y. CHEN & YOUNG 1988; FRANK 1975, S. 28-38;

---

<sup>4</sup>Die Inhalte dieses Abschnitts sind zum Teil im Rahmen einer Studienarbeit entstanden (HONEIN 2020, s. hierzu Abschnitt „Betreute Studienarbeiten“).

<sup>5</sup>Zu den Grundlagen s. Abschnitt 2.5.1.

HOFMANN 2014, S. 115; HANSSON ET AL. 2016; NIESS 1983, S. 45 f.), basiert jedoch jeweils auf der Einschätzung von Fachpersonal und steht meist nicht im Fokus der Forschungsarbeiten. Eine Veröffentlichung von LA BROOY & C. JIANG (2009), in der ein Expertensystem zur Auslegung von OS für VWF vorgestellt wird, beinhaltet die manuelle Zuweisung von geometrischen Merkmalen zu zweidimensionalen CAD-Daten<sup>6</sup>. Die Autoren KAPARTHI & N. SURESH (1991) stellen ein System zur Klassifizierung von rotationssymmetrischen Teilen vor, welches mithilfe von NN zweidimensionale Bilder von Werkstücken in Klassen kategorisiert. Dieser Ansatz ist nur sehr eingeschränkt funktionsfähig und übertragbar.

Allgemeiner gefasste Ansätze zur 3D-CAD-Daten-Analyse stützen sich ebenfalls auf die Anwendung von Machine Learning (z. B. HOU ET AL. 2005; LEE & FISCHER 1999; MATURANA & SCHERER 2015; MOU 2018). Für Klassifizierungsaufgaben kommen meist Deep-Learning-Implementierungen unter Verwendung von NN zum Einsatz (vgl. Abschnitt 2.5.1; QIN ET AL. 2014; M. WU & JEN 1996). Dazu werden die 3D-Daten zu Eingangsdaten für einen Algorithmus verarbeitet, der in einem überwachten Trainingsprozess anhand von Datensets lernt, relevante Merkmale einer Klasse von Werkstücken zuzuordnen. Mit dem Erlernten kann der Algorithmus unbekannte Daten mit einer bestimmten Genauigkeit klassifizieren (Z. WU ET AL. 2015; XU & TODOROVIC 2016). Die Herangehensweise der ML-gestützten 3D-CAD-Analyse erfüllt die im Rahmen dieser Arbeit geforderte Funktion für die Ähnlichteil-Analyse und wurde daher weiter verfolgt. Im nächsten Schritt wurden bestehende Deep-Learning-Ansätze zur 3D-Modell-Klassifizierung verglichen mit dem Ziel, einen geeigneten Ansatz für die Ähnlichteil-Analyse zu identifizieren.

Tabelle 4.3 gibt einen Überblick bestehender Ansätze, Details zur Funktionsweise sind den referenzierten Publikationen zu entnehmen. Wichtigstes Unterscheidungskriterium der Ansätze ist die Repräsentation des Input-Modells, also in welcher Form die 3D-CAD-Daten für das jeweilige Konzept zur Klassifizierung aufbereitet werden. Zur Bewertung der Eignung des jeweiligen Konzepts wird die Genauigkeit der Ergebnisse herangezogen. Die prozentualen Werte stehen

---

<sup>6</sup>Die genaue Funktionsweise geht aus dem Zeitschriftenbeitrag nicht hervor, zudem ist weiterhin umfangreiche Nutzerinteraktion erforderlich.

## 4 Methodenschritt 1: Datenaufbereitung

für den Anteil der CAD-Modelle, die richtig den vorgegebenen Klassen zugeordnet wurden. Die Größe eines Datensets hängt von der Anzahl an Klassen ab, sowie von der Menge an Datensätzen, die innerhalb einer Klasse zum Training zur Verfügung stehen. Hierbei wird unterschieden, ob es sich um kleine oder große Datensets handelt, jedoch existiert keine einheitliche Definition dieses Begriffs (z. B. definieren XU & TODOROVIC (2016): klein: 10; groß: 40 Klassen mit jeweils rund 1.000 Datensätzen).

Tabelle 4.3: Vergleich von Deep-Learning-Ansätzen für die 3D-Modell-Klassifizierung

Quelle	Repräsentation des Input-Modells	Konzept	Genauigkeit bei...	
			kleinem Datenset	großem Datenset
WANG & LIU 1993	2D-Ansichten von 3D-Modellen	Hopfield-Netz	k. A.	k. A.
M. WU & JEN 1996	Vereinfachte 2D-Werkstück-Konturen	Back-propagation Neural Network	k. A.	k. A.
HOU ET AL. 2005	Hierarchischer Graph des Werkstück-Skeletts	Support Vector Machines	k. A.	≥88%
QIN ET AL. 2014	2D-Ansichten von 3D-Modellen mittels Lichtfeld-Projektion	Deep Neural Network	k. A.	≥98%
Z. WU ET AL. 2015	Voxelierte 3D ShapeNets	Lineare Support Vector Machine	≥77%	≥83%
SU ET AL. 2015	Gerenderte 2D-Ansichten von 3D-Modellen	Multi-view CNN	≥90%	k. A.
MATURANA & SCHERER 2015	Voxelierte 3D-CAD-Modelle	„VoxNet“: 3D-CNN	≥83%	≥92%
XU & TODOROVIC 2016	Voxelierte 3D-CAD-Modelle	Deep CNN	≥81%	≥88%

Frühe Ansätze zur 3D-Modell-Klassifizierung stützen sich auf zweidimensionale Abbildungen von 3D-Daten, die als Input für neuronale Netze in verschiedenen Ausprägungen fungieren (z. B. HOU ET AL. 2005; QIN ET AL. 2014; SU ET AL. 2015; WANG & LIU 1993; M. WU & JEN 1996). Dabei geht ein signifikanter Anteil an geometrischen Informationen verloren, vor allem innen liegende



Merkmale können nicht erkannt werden. Spätere Arbeiten nutzen voxelierete<sup>7</sup> 3D-Repräsentationen als Eingangsdaten für die CAD-Analyse (z. B. MATORANA & SCHERER 2015; Z. WU ET AL. 2015; XU & TODOROVIC 2016). Mit diesem Ansatz konnte die Leistungsfähigkeit und Robustheit der Klassifizierung selbst bei kleinen Trainingsdatensätzen erheblich gesteigert werden.

Im Rahmen dieser Arbeit wurde die Architektur eines CNN nach XU & TODOROVIC (2016) adaptiert. Zwar weisen andere Modelle höhere Genauigkeiten auf, der gewählte Ansatz bietet aber folgende Vorteile:

- **Parameter:** Die Anzahl der verwendeten Parameter<sup>8</sup> ist deutlich geringer als bei anderen Ansätzen (z. B. 150-fach geringere Anzahl als bei Verwendung von 3D ShapeNets, vgl. Z. WU ET AL. 2015). Der gewählte Ansatz wurde für die Anwendung zur 3D-Daten-Analyse optimiert, auch in Anbetracht der im Vergleich zu klassischen Bilderkennungsaufgaben geringen Anzahl an frei verfügbaren Trainingsdaten.
- **Trainingsprozess:** Für große Datensets ist die Trainingsdauer verhältnismäßig niedrig (unter 5 Stunden gegenüber ca. 12 Stunden für das Training von VoxNet, vgl. MATORANA & SCHERER 2015). Dieses Kriterium ist relevant, wenn das CNN für neue Fördergut-Klassen erneut trainiert werden muss.
- **Inputdaten:** Es werden verlustfrei 3D-Daten anstelle von projizierten 2D-Daten zur Erkennung von dreidimensionalen Merkmalen genutzt.
- **Trainingsdaten:** Der Ansatz kann für verschiedenste Problemstellungen verwendet werden, die robuste Lernvorgänge mit kleinen Trainings-Datensets erfordern.

---

<sup>7</sup>Voxel sind Gitterpunkte in einem dreidimensionalen Gitter und vergleichbar mit Pixeln im zweidimensionalen Raum (WATT 2002, S. 163).

<sup>8</sup>Je größer die Anzahl der Parameter, desto höher die Rechenintensität des neuronalen Netzes und der Bedarf an Trainingsdaten (vgl. KRUSE ET AL. 2015, S. 76 f.).

### 4.2.2.2 Erarbeitung eines Datensets und der CNN-Architektur

Die Arbeiten in obenstehender Tabelle 4.3 greifen auf Modell-Bibliotheken wie z. B. das Princeton ModelNet zu (PRINCETON UNIVERSITY 2020; Z. WU ET AL. 2015). Diese enthalten große Mengen an 3D-CAD-Daten in verschiedenen Klassen, die zum Teil aber für die industrielle Anwendung nicht relevant sind, wie bspw. Badewannen oder Stühle.

Basierend auf Literaturangaben (BOOTHROYD 2005, S. 385–407; ELLEKILDE ET AL. 2016; FRANK 1975, S. 31 f.; HESSE 2000b, S. 96, 2010, S. 14 f.; SCHMID 2006, S. 41 f.) und Aussagen aus der Praxis (KRAUEL 2020) wurden zehn Klassen ausgewählt, die ein relevantes Teilespektrum aus der Industrie repräsentieren. Das aufgebaute Datenset „TUM10“ beinhaltet 1.764 Datensätze, die aus verschiedenen Quellen zusammengeführt (bspw. JAYANTI ET AL. 2006; STRATASYS 2020; VVG-BEFESTIGUNGSTECHNIK 2020) und anschließend nach dem Vorgehen von MIN (2004–2019) und NOORUDDIN & TURK (2003) voxelisiert wurden.

Abbildung 4.2 zeigt Beispiele für Teile der zehn Klassen des Datensets. Sollte eine Erweiterung um neue Bauteilklassen erforderlich sein, müssen diese mit ausreichend Datensätzen angelegt und das CNN neu trainiert werden.

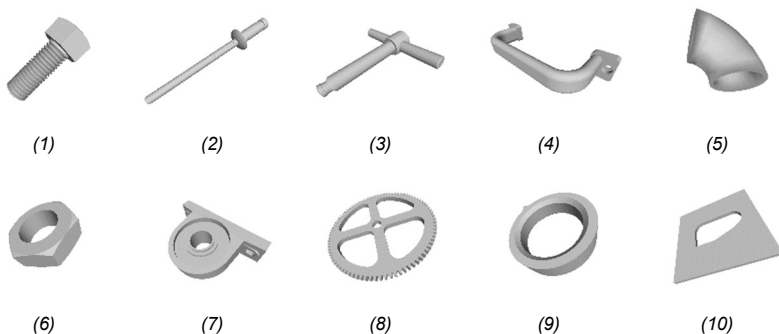


Abbildung 4.2: Beispiel-Bauteile für die zehn Klassen des Datensets TUM10:  
(1): Pilzteil, (2): Niet, (3): T-Stück, (4): Griff, (5): Winkel,  
(6): Mutter, (7): Lager, (8): Zahnrad, (9): Scheibe, (10): Platte;  
vgl. Anhang, Abschnitt A.1 (in Anlehnung an HONEIN 2020, S. 32)

Die von XU & TODOROVIC (2016) vorgestellte CNN-Architektur wurde unter Verwendung der zehn TUM10-Klassen mittels gängiger ML-Bibliotheken<sup>9</sup> umgesetzt (vgl. Abbildung 4.3; VASILEV ET AL. 2019, S. 81). In Ergänzung zu diesem Modell wurden weitere Layer hinzugefügt (vgl. Abschnitt 2.5.1). So musste die Voxel-Repräsentation mittels Reshape-Layer für die Verarbeitung mit CNN-Operationen aufbereitet werden (VASILEV ET AL. 2019, S. 112). Zudem wurden zwischen den Layern nach XU & TODOROVIC (2016) Dropout-Layer ergänzt, um Overfitting zu vermeiden. Des Weiteren wurde die Architektur um einen Optimierer erweitert, der den Lernprozess durch die Anpassung der Gewichtungen verbessert<sup>10</sup>.

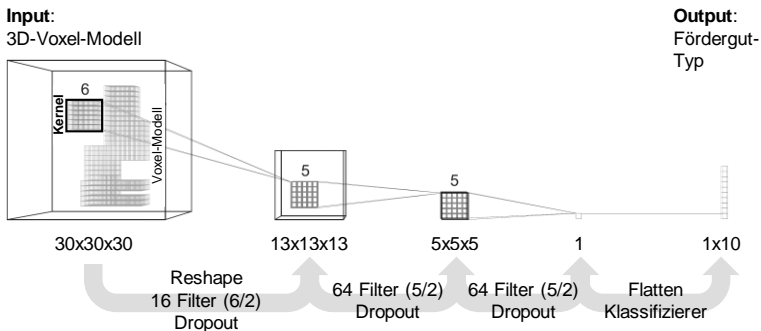


Abbildung 4.3: Architektur des adaptierten CNN (in Klammern Größe und Schrittweite der Filter; in Anlehnung an HONEIN 2020, S. 33 und XU & TODOROVIC 2016)

### 4.2.2.3 Training und Evaluation des CNN

Das Datenset TUM10 wurde in ein Trainingsset mit rund 1.400 und ein Testset mit etwa 350 Datensätzen aufgeteilt (80% der insgesamt zur Verfügung stehenden Daten zu 20%). Die Aufteilung der Daten wurde anhand des in der Literatur üblichen Vorgehens vorgenommen (REBALA ET AL. 2019, S. 95).

<sup>9</sup>TensorFlow (GOOGLE, INC. 2020) und Keras (CHOLLET 2015).

<sup>10</sup>Es wurde der Adam-Optimierer nach KINGMA & BA (2014) ergänzt. Optimierung ist ein gängiger Bestandteil von NN-Modellen (MICHELUCCI 2019, S. 161), wird von XU & TODOROVIC (2016) aber nicht behandelt, da diese sich auf die Architektur des CNN fokussieren.

## 4 Methodenschritt 1: Datenaufbereitung

Innerhalb des Trainingssets wurde ein Teil der Daten gelabelt, um das CNN anzulernen (rund 1.100 Datensätze). Zur Überprüfung der Genauigkeit und Feinjustierung der Modellparameter wurden knapp 300 Validierungsdatensätze verarbeitet. Dabei konnte eine maximale Genauigkeit von 94,7% erzielt werden, eine Anzahl von 30 Trainingsepochen hat sich hierbei als ausreichend erwiesen (vgl. Abbildung 4.4). Das Netz wurde abschließend mit den Neuronengewichten, welche die höchste Genauigkeit erzielten, final eingestellt.

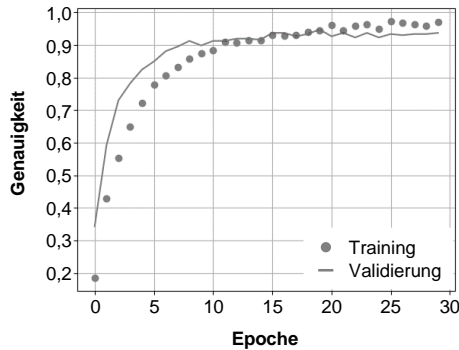


Abbildung 4.4: Genauigkeit der Klassifizierung im Verlauf von Training und Validierung (Maximum bei 0,947 in Epoche 30; in Anlehnung an HONEIN 2020, S. 44)

In der Testphase mit ungelabelten Datensätzen und unter Verwendung der fixierten Neuronengewichte konnte der Algorithmus eine Genauigkeit von 82,8% im Mittelwert über 10 Testläufe in allen Klassen mit einer Standardabweichung von 0,01 erreichen. Mittels eines  $X^2$ -Tests wurde die Annahme der Normalverteilung bestätigt (WALPOLE ET AL. 2017, S. 371). Die Genauigkeit liegt somit etwas unter der von XU & TODOROVIC (2016) beschriebenen, jedoch im Rahmen derer von ähnlichen Ansätzen (vgl. Tabelle 4.3). Die oben beschriebenen Veröffentlichungen lassen allerdings keine Aussage darüber zu, ob es sich bei den Angaben um einmalig erzielte Maximalwerte oder Durchschnittswerte über mehrere Durchläufe und Klassen handelt. Zudem wurden die Netze mit deutlich größerer Anzahl an Datensätzen trainiert (mindestens 1.000 Datensätze pro Klasse), wodurch die Genauigkeit der Klassifizierung gesteigert wird (REBALA ET AL. 2019, S. 98). Bei einem Vergleichstest des adaptierten CNN mit den

ModelNet10-Daten<sup>11</sup> wurden im Zuge der Untersuchungen 89% Genauigkeit erreicht, was die grundsätzliche Eignung des Ansatzes bekräftigt.

### 4.3 Entwurf einer erweiterbaren CAD-Bibliothek

#### 4.3.1 Auswahl und Parametrierung von Schikanenelementen

Zur Befüllung der CAD-Bibliothek mit SE für die Simulation des Förderprozesses und die Generierung der OS wurden folgende Bausteine erarbeitet:

1. Auswahl und Zusammenstellung relevanter SE (Abschnitt 4.3.1.1)
2. Ermittlung wirksamer SE-Parameter und Abstimmung auf Fördergut (Abschnitt 4.3.1.2)

##### 4.3.1.1 Auswahl und Zusammenstellung relevanter SE

Die Ausgangsbasis für die Wahl relevanter SE bildet deren funktionale Analyse hinsichtlich der zu erfüllenden Ordnungsaufgabe. Dazu werden verschiedene Kriterien wie Ordnungsprinzip (aktiv, passiv) oder Wirkprinzip (z. B. Schwerkraft, Reibung) herangezogen. Ziel ist es, ein möglichst großes Spektrum an Ordnungsaufgaben ausführen zu können (WEISS 1983, S. 45 f.).

In Hinsicht darauf wurden zunächst die in Abschnitt 2.3 aufgeführten Vorarbeiten genauer untersucht, die im Rahmen ihrer Forschungstätigkeit eine Auswahl von SE vornehmen. Diese Arbeiten stützen sich bezüglich der SE-Auswahl auf die Arbeit von BOOTHROYD (2005, S. 47–87 sowie S. 474–491)<sup>12</sup> und dessen Ausführungen zu SE. Bei BOOTHROYD (2005) werden 27 elementare Grundtypen von SE vorgestellt und hinsichtlich ihrer Parameter untersucht. MATHIESEN (2017, S. 103–105) nimmt eine Vereinfachung dieser Auswahl vor, indem jeweils

---

<sup>11</sup>Zum ModelNet10-Datensatz s. PRINCETON UNIVERSITY 2020.

<sup>12</sup>Bzw. die Vorarbeiten zu dieser Publikation (BOOTHROYD ET AL. 1978; BOOTHROYD ET AL. 2011: 1. Auflage von 1983).

## 4 Methodenschritt 1: Datenaufbereitung

---

mehrere, bei BOOTHROYD (2005) einzeln aufgeführte SE zu insgesamt zehn parametrisierten Grundelementen zusammengefasst werden. WEISS (1983, S. 45 f.) hat eine vergleichbar kompakte Darstellung mit ähnlichen Grundelementen entwickelt, wobei dieser die Elemente aus der Praxis hergeleitet hat. Nach der Auswahl geeigneter SE werden diese zu einer OS zusammengefügt, welche die vorgegebene Ordnungsaufgabe durch die Kombination der Wirkung der einzelnen SE erfüllen soll.

Im Rahmen der vorliegenden Arbeit sollen ebenfalls elementare SE ausgewählt, an das Fördergut adaptiert und mehrere SE zu OS zusammengestellt werden. Um die Komplexität des Ansatzes zu reduzieren wurden dazu die Grundelemente nach BOOTHROYD (2005) verallgemeinert bzw. nach Funktion der SE zusammengefasst, ähnlich wie MATHIESEN (2017) vorgegangen ist. Dieser schränkt seine Auswahl jedoch auf wenige sehr spezifische Fördergut-Teile ein, sodass im Rahmen der vorliegenden Arbeit SE ergänzt bzw. verallgemeinert wurden. In Anlehnung an eben beschriebene Vorarbeiten und Erfahrungswerte aus der Praxis wurden daher elf Grundelemente mit einstellbaren Parametern konzipiert. Abbildung 4.5 zeigt eine Übersicht, in Abschnitt A.3 im Anhang sind die SE und ihre Parameter ausführlich beschrieben.

Des Weiteren sollten die verfügbaren SE Fördergut-spezifisch erweitert werden. Durch sog. Boolesche Operationen wie Vereinigung oder Subtraktion können in CAD-Programmen mehrere Geometrien miteinander verschnitten werden (LO & DICK 1990). Dies wird genutzt, um ausgehend von der Geometrie eines Förderguts das Negativ dessen Zielorientierung zu errechnen. Ein solches SE eignet sich als letztes Element der OS, da es bei entsprechender Vorbereitung bzw. geeignet vorgeschalteten SE bevorzugt die Zielorientierung des Werkstücks hervorbringt (vgl. Abbildung 4.6).

### 4.3.1.2 Ermittlung wirksamer SE-Parameter und Abstimmung auf Fördergut

Aufbauend auf den o. g. Vorarbeiten wurden die Wirkflächen der ausgewählten SE identifiziert und hinsichtlich der Wirkung auf das Fördergut analysiert. Auf dieser Basis wurden für jedes SE Parameter gewählt, welche dessen Einstellung auf das zu orientierende Werkstück ermöglichen.

### 4.3 Entwurf einer erweiterbaren CAD-Bibliothek

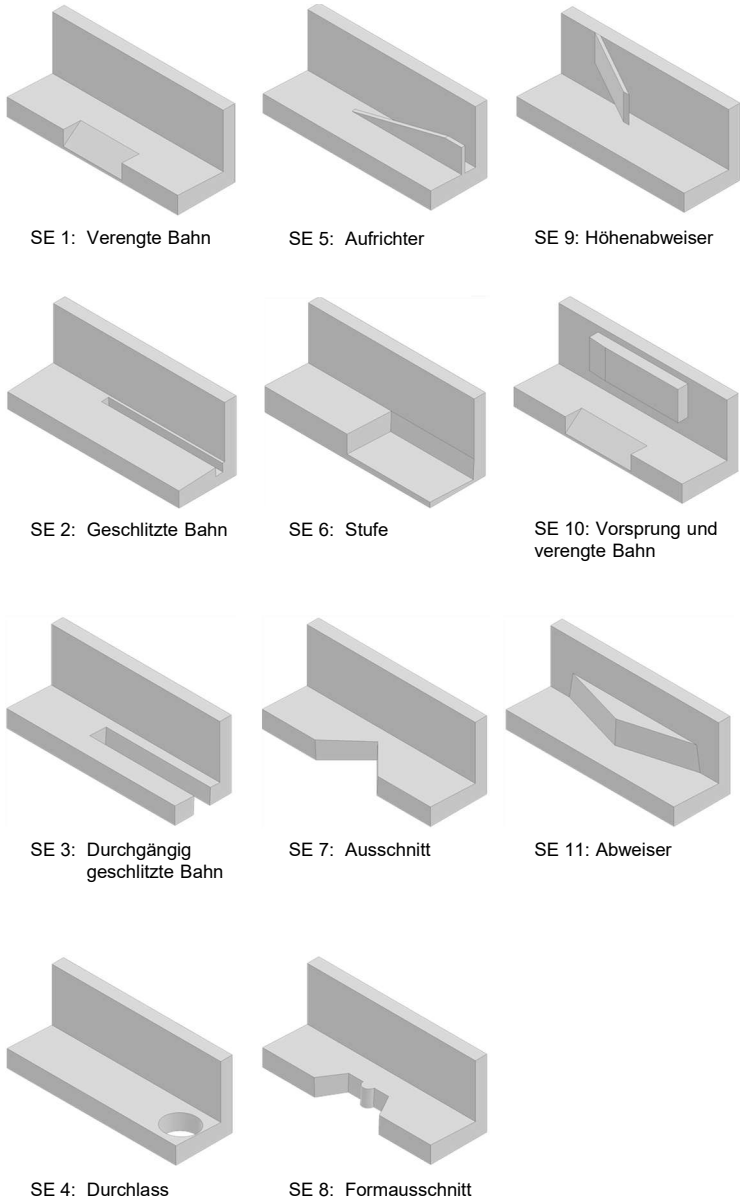


Abbildung 4.5: Im Rahmen dieser Arbeit entwickelte SE mit Bezeichnung (zu den Parametern s. Anhang, Abschnitt A.3)

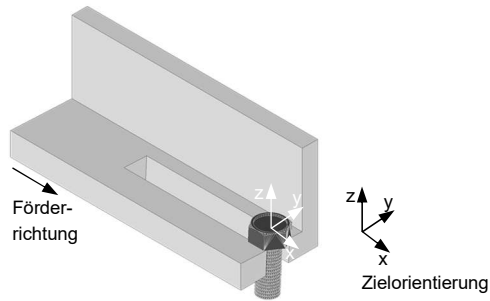


Abbildung 4.6: Beispiel für ein Förderer-gut-spezifisches SE: Die Schraube befindet sich in der Zielorientierung, das SE wurde entsprechend ausgeschnitten.

Wie bereits WEISS (1983, S. 50) ausführt, müssen diese Einstellparameter in Abhängigkeit der Werkstück-Klasse und weiterer Werkstückmerkmale (z. B. Abmessungsverhältnisse wie Länge zu Breite) variiert werden. Dazu hat WEISS diese Abhängigkeiten ausführlich analytisch hergeleitet (WEISS 1983, S. 50–63) sowie praktisch validiert (WEISS 1983, S. 94–107).

Des Weiteren hat BOOTHROYD (2005, S. 47–87 sowie S. 474–491) ähnliche sowie ergänzende Vorarbeiten präsentiert. Auf Basis von formelmäßigen Zusammenhängen können die konkreten Zahlenwerte, mit denen die SE-Parameter abhängig von den Förderer-gut-Parametern einzustellen sind, berechnet werden (vgl. Abschnitt 5.3).

Im Rahmen der vorliegenden Arbeit wurden die Vorarbeiten von WEISS (1983) und BOOTHROYD (2005) adaptiert und die Parameter der generischen SE exemplarisch für die Förderer-gut-Klasse Pilzteil als Formeln in Abhängigkeit der dafür relevanten Förderer-gut-Parameter definiert (s. Anhang, Abschnitt A.3). Für detaillierte Ausführungen dazu sei auf die referenzierten Arbeiten verwiesen (WEISS 1983, S. 50–63; BOOTHROYD 2005, S. 47–87 sowie S. 474–491).

Zur Erweiterung auf die übrigen TUM10-Förderer-gut-Klassen müssen die Angaben bei WEISS und BOOTHROYD entsprechend übertragen und ggf. erweitert werden. Selbiges gilt auch für den Methodendurchlauf mit einer unbekanntem Förderer-gut-Klasse.



### 4.3.2 Struktur der CAD-Bibliothek

Um die verschiedenartigen Daten von OS, SE und Fördergut effizient nutzbar zu machen, wurde eine Struktur zu deren Speicherung und Verknüpfung entwickelt. Dazu wurden zunächst anhand des Methodenablaufs (vgl. Abbildung 3.1) und der übergeordneten Zielsetzung (vgl. Abschnitt 1.2) Anforderungen an die CAD-Bibliothek erarbeitet und jeweils korrespondierende Lösungsansätze hergeleitet:

- **Aufnahme von CAD- und weiteren Daten zu OS, SE und Fördergut:** Die CAD-Bibliothek soll sowohl als Datenquelle als auch -senke für die Simulation des Förderprozesses, das RL-System sowie die Evaluation der OS dienen. So werden z. B. CAD-Daten daraus geladen und Simulationsergebnisse abgespeichert. Hierunter fallen bspw. die Transitionsmatrizen zu den SE.
- **Darstellung der Relationen zwischen den Objekten:** Für jedes SE in der CAD-Bibliothek muss gespeichert werden, welches Fördergut-Teil im Förderprozess simuliert wurde. Es sind auch mehrere Simulationsreihen mit verschiedenen Werkstücken möglich. Berechnete OS sollen als Zusammensetzung von mehreren SE abgelegt werden und als Ergebnisspeicher dienen.
- **Laufende Erweiterung:** Nach anfänglicher Befüllung der CAD-Bibliothek mit Daten sollen im Laufe der Nutzung die berechneten OS, zur Auslegung geladenes Fördergut, aber auch ggf. neue SE eingespeist werden.

Ausgehend von diesen Anforderungen wurden zunächst bestehende Ansätze der strukturierten Speicherung im Themenfeld der Auslegung von OS recherchiert (vgl. Abschnitt 2.3). LO & DICK (1990) stellen eine Methode zur regelbasierten Anordnung von SE vor. Hier wird eine hierarchische Strukturierung von Bauteil- und SE-Merkmalen beschrieben, die einer Klassendefinition in der objektorientierten Programmierung ähnelt. Zur Verbindung der Klassen untereinander werden keine Angaben gemacht. P. TAN ET AL. (1995) beschreiben ein wissensbasiertes Assistenzsystem zur Auswahl und Sequenzierung von SE und ein ähnliches Konzept zur hierarchischen Speicherung der Daten. HANSSON ET AL. (2016, S. 151 f.) erwähnen eine „knowledge base“ (dt.: Wissensbasis),

## 4 Methodenschritt 1: Datenaufbereitung

---

welche die Transitionsmatrizen anhand eines ontologischen Modells speichert. Es wird angekündigt, diese zukünftig zur Erfassung heuristischen Wissens zu verwenden, welches im Auslegungsprozess genutzt werden kann. Bis dato sind jedoch keine Veröffentlichungen bekannt, in welchen dies näher beschrieben wird. In Anlehnung an diese bisherigen Ansätze wurde in der vorliegenden Arbeit die Struktur der CAD-Bibliothek entworfen (s. Abbildung 4.7).

Zur Beschreibung der zu verarbeitenden Daten und der Modellierung von Relationen wurde dazu ein UML<sup>13</sup>-Klassendiagramm konzipiert (VAN RANDEN ET AL. 2016a, S. 5–17). Dieses enthält Objekte in drei verschiedenen Klassen<sup>14</sup>: OS, SE und Fördergut (vgl. Abbildung 4.7). Auf Seiten der Klassen können beliebig viele Objekte<sup>15</sup> innerhalb der Bibliothek erzeugt werden.

OS, SE und Fördergut wurden als übergeordnete Klassen konzipiert. Die beiden letzteren dienen als Oberklassen für die Unterklassen SE-Typ und Fördergut-Typ. In den Oberklassen werden Attribute, also beschreibende Merkmale der Objekte<sup>16</sup> definiert, die unabhängig von den SE- bzw. Fördergut-Typen sind, wie z. B. Masse oder Material. Die Unterklassen erben<sup>17</sup> diese Eigenschaften und verfügen je nach Typ über weitere spezifische Attribute wie bspw. Parameter oder Vorzugslagen. Ein Objekt kann nur von der Unterklasse erzeugt werden, da SE und Fördergut stets von einem bestimmten Typ sein müssen. Die SE-Typen entsprechen hierbei den in Abschnitt 4.3.1 erarbeiteten Grundelementen, die Fördergut-Typen korrespondieren mit den in Abschnitt 4.2.2 vorgestellten Klassen des TUM10-Datensets.

Zu jedem SE in der CAD-Bibliothek werden die in der Simulation mit dem Fördergut errechneten Ergebnisse wie erzielte Orientierung und Effizienz sowie die Transitionsmatrizen abgelegt. Jedem SE und Fördergut-Teil werden jeweils die zugehörigen Simulationsergebnisse zugeordnet. Handelt es sich um

---

<sup>13</sup>Unified Modeling Language (VAN RANDEN ET AL. 2016b, S. vii).

<sup>14</sup>Eine Klasse steht für eine Kategorie von Objekten, die als Instanzen mit konkreten Werten belegt werden und somit bspw. ein spezielles Fördergut-Teil repräsentieren. In Abbildung 4.7 sind sie als grau hinterlegte Rechtecke dargestellt und beinhalten den Namen der Klasse. Die schwarze Raute steht für die Beziehung der Komposition, die besagt, dass genau eine Bibliothek übergeordnet existiert.

<sup>15</sup>Gekennzeichnet durch einen Stern.

<sup>16</sup>Im weiß hinterlegten Rechteck zu finden.

<sup>17</sup>Dargestellt durch ein Dreieck.



#### 4 Methodenschritt 1: Datenaufbereitung

---

ein neues SE oder Fördergut-Typ, sind zu Beginn keine Ergebnisse abgelegt, im weiteren Verlauf können beliebig viele zugeordnet werden<sup>18</sup>. Diese Zuordnung dient auch der Speicherung der Erfahrungswerte, welche SE für welche Fördergut-Typen geeignet sind.

Durch den Algorithmus berechnete OS sind jeweils eine Aggregation einer bestimmten Anzahl an SE<sup>19</sup> und werden in der Bibliothek abgespeichert.

Auf Basis der vorgestellten Struktur kann bspw. eine relationale Datenbank aufgebaut werden, welche die notwendigen Informationen und Pfade zu den Speicherorten der CAD-Daten enthält und beim Methodendurchlauf bereitstellt. Durch die Möglichkeit des Erzeugens von neuen Objekten der jeweiligen Klasse ist die Erweiterbarkeit gewährleistet. Die Bibliothek dient hierdurch auch als wachsende Wissensbasis für alle durchgeführten Auslegungsvorgänge, da neben dem Ergebnis, also der OS, auch die beteiligten Elemente und deren Relationen verknüpft abgelegt sind. Somit können alle eingangs definierten Anforderungen an die CAD-Bibliothek erfüllt werden.

---

<sup>18</sup>Gekennzeichnet durch die Angabe 0..\*.

<sup>19</sup>Dargestellt durch die nicht ausgefüllte Raute.

## 5 Methodenschritt 2: Bewertung der Schikanenelemente

### 5.1 Überblick

Der zweite Methodenschritt, *Bewertung der Schikanenelemente*, hat zum Ziel, die Eingangsdaten für den Algorithmus im nachfolgenden Schritt, der *Generierung der OS*, zu errechnen. Abbildung 5.1 zeigt die erarbeiteten Bausteine, die in den folgenden Abschnitten 5.2 und 5.3 im Detail erläutert werden.

Im wissenschaftlichen Erarbeitungsprozess der vorliegenden Arbeit wurde zuerst die Bewertungsmethode konzipiert (Abschnitt 5.2), um anschließend die vorbereitenden Schritte für die Simulation des Förderprozesses passend zu gestalten (Abschnitt 5.3), während diese Schritte in der Auslegungsmethode umgekehrt zum Einsatz kommen.

Zur *Bewertung der Schikanenelemente* erfolgt zunächst die Simulation des Förderprozesses, um relevante Daten über die Interaktion von Fördergut-Teil und SE zu generieren. Dazu müssen deren CAD-Daten automatisiert in die Physiksimulation geladen und die erforderlichen Simulationsläufe durchgeführt werden. In Abschnitt 5.3.1 wird daher eine Versuchsplanung erarbeitet, welche anhand der Parameter von SE und Fördergut-Teil die notwendigen Umfänge an Parameterabstufungen für die Simulation festlegt. Da es sich bei den Simulationsergebnissen um statistische Verteilungen handelt (vgl. Abschnitt 5.2), muss sichergestellt sein, dass die Ergebnisse der Simulation stabile Werte erreichen. Dazu wird in Abschnitt 5.3.2 eine Konvergenzanalyse entwickelt, die eine ausreichende Anzahl an Versuchen gewährleistet.

Die in Abschnitt 5.3.1 und 5.3.2 erarbeiteten Bausteine sind Teil des Methodenschritts *Bewertung der Schikanenelemente* und werden bei jedem Methodendurchlauf ausgeführt.

## 5 Methodenschritt 2: Bewertung der Schikanenelemente

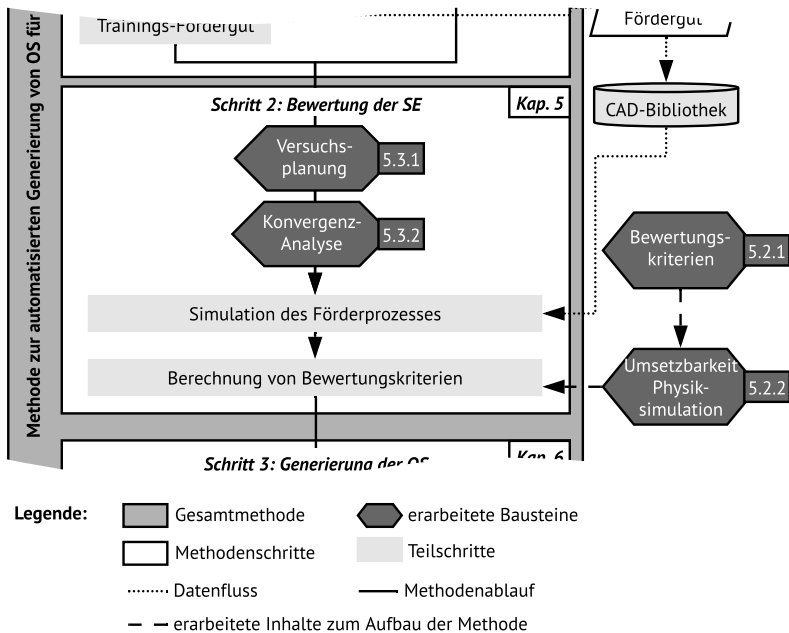


Abbildung 5.1: Methodenschritt „Bewertung der Schikanenelemente“ und erarbeitete Bausteine

Aus den Ergebnissen der Simulation werden Bewertungskriterien berechnet, welche den Förderprozess quantifizierbar machen. Dazu zeigt Abschnitt 5.2.1 eine Bewertungsmethode auf, welche die Wirksamkeit der SE für bestimmtes Fördergut quantifiziert. Anschließend zeigt Abschnitt 5.2.2 die Umsetzbarkeit mit der Physiksimulation.

Die in Abschnitt 5.2.1 und 5.2.2 erarbeiteten Bausteine wurden im Rahmen der vorliegenden Arbeit einmalig erarbeitet und ihre Ergebnisse stehen bei Durchlaufen der Auslegungsmethode zur Anwendung bereit.

### 5.2 Simulationsbasierte Bewertung von Schikanenelementen

#### 5.2.1 Bewertung von Schikanenelementen

Ziel der Bewertung ist, dem RL-System für die Generierung von OS geeignete Daten zur Verfügung zu stellen. Der Algorithmus soll kriterienbasiert SE auswählen und diese zielgerichtet kombinieren. Bei der Simulation des Förderprozesses fallen verschiedene Daten an, die zu diesem Zweck potenziell genutzt werden können. Daher wurde folgendes Vorgehen angewendet, um daraus geeignete Kriterien auszuwählen:

1. Erhebung der Anforderungen an die Bewertungskriterien (Abschnitt 5.2.1.1)
2. Analyse potentiell geeigneter Bewertungskriterien (Abschnitt 5.2.1.2)
3. Auswahl auf Basis der Anforderungen (Abschnitt 5.2.1.3)

##### 5.2.1.1 Erhebung der Anforderungen an die Bewertungskriterien

Zur Erreichung o. g. Zielsetzung wurden folgende Anforderungen definiert<sup>1</sup>:

- **Kriterienbasierte Auswahl von SE:** Die verwendeten Bewertungskriterien müssen eine eindeutige Aussage über die Eignung der gespeicherten SE für die gestellte Orientierungsaufgabe und das gewählte Fördergut zulassen. Des Weiteren müssen aus den Informationen zum einzelnen SE weiterführende Schlüsse auf die aus mehreren SE generierte OS möglich sein.
- **Nutzung der Physiksimulation als Werkzeug:** Die zu erhebenden Daten müssen mit der Physiksimulation erfassbar sein, da diese als Werkzeug für die virtuelle Evaluation der SE und OS zum Einsatz kommt (vgl. Abschnitt 2.3.3).

---

<sup>1</sup>Neben der direkten Ableitung aus der Zielsetzung in Abschnitt 5.2.1 wurde sich hierfür auf die Ausführungen von BOOTHROYD (2005, S. 58–69), DALLINGER (2017, S. 20–23), HOFMANN (2014, S. 61 f.) sowie auf Erfahrungswerte aus der Praxis gestützt.

## 5 Methodenschritt 2: Bewertung der Schikanenelemente

---

- **Automatisierter Ablauf:** Die Erhebung und Nutzung der Daten zur Bewertung der SE muss ohne manuelle Interaktion stattfinden, um die Entkopplung von Erfahrungswissen zu gewährleisten.

### 5.2.1.2 Analyse potentiell geeigneter Bewertungskriterien

Auf Basis der im vorigen Abschnitt beschriebenen Anforderungen wurde die bereits in Abschnitt 2.3 vorgestellte Literatur herangezogen, um potenziell geeignete Methoden zu analysieren. Es existieren wenige verschiedene Ansätze zur quantitativen Bewertung des Förder- und Orientierungsprozesses, der an den SE im VWF stattfindet, bei denen folgende Kenngrößen und Werkzeuge zum Einsatz kommen<sup>2</sup> (s. a. HOFMANN 2014, S. 61):

- **Mittlere Fördergeschwindigkeit  $\bar{v}$ :** Diese Kenngröße wird in verschiedenen Arbeiten genutzt, um die Einstellung der Schwingungsparameter des VWF zu bewerten, da diese Parameter sich direkt auf die Fördergeschwindigkeit der Teile im VWF und der OS auswirken (ASHRAFIZADEH & ZIAEI-RAD 2013; BOOTHROYD 2005, S. 39; DALLINGER 2017, S. 91; HOFMANN 2014, S. 62–64).
- **Effizienz  $\epsilon$ :** Laut BOOTHROYD (2005, S. 58) ist die Effizienz das wichtigste Bewertungskriterium für OS (vgl. Abschnitt 2.2.3). Dabei soll durch Kombination möglichst effizienter SE-Abschnitte die größtmögliche Gesamteffizienz erreicht werden. Weitere Arbeiten verwenden diese ebenfalls als Verhältnis der Anzahl richtig orientierter zu der insgesamt geförderter Teile (z. B. BERKOWITZ & CANNY 1996; BERRETTY ET AL. 1999; CAINE 1994; CHRISTIANSEN ET AL. 1996; EDWARDS 2004; HOFMANN 2014, S. 62–70; SETHI ET AL. 1990). BERKOWITZ & CANNY (1996) schreiben der Umorientierung von Teilen, ausgedrückt durch die Effizienz der OS, höhere Bedeutung zu als deren Fördergeschwindigkeit.
- **Übergangswahrscheinlichkeiten:** Diese eignen sich zur Quantifizierung der Wirkung von SE bzw. OS indem bewertet wird, mit welcher

---

<sup>2</sup>Zu den Grundlagen vgl. Abschnitte 2.1 und 2.2.3.



Wahrscheinlichkeit ein SE die Umorientierung eines Fördergut-Teils bewirkt. Das Grundprinzip wurde von BOOTHROYD (2005)<sup>3</sup> vorgestellt und seitdem wiederholt eingesetzt (z. B. BERKOWITZ & CANNY 1996; HANSSON ET AL. 2016; JAUMARD ET AL. 1993; MATHIESEN & ELLEKILDE 2016; OU-YANG & MAUL 1993; YERALAN & CHANG 1986). Eine allgemein gebräuchliche Formalisierung von Übergangswahrscheinlichkeiten sind Transitionsmatrizen, die für jede Eingangsorientierung eine Verteilung über die Orientierungen nach Durchlaufen eines SE angeben (s. nächster Punkt).

- **Transitionsmatrizen:** Als Werkzeug zur Quantifizierung der Funktion von SE bzw. OS eignen sich Transitionsmatrizen, aus denen die Effizienz der Anordnung errechnet wird (BOOTHROYD 2005, S. 59–61, S. 78–85). Auf diese Technik stützen sich, meist mit Verweis auf BOOTHROYD (1992 bzw. 2005), zahlreiche Arbeiten (z. B. CHRISTIANSEN ET AL. 1996; EDWARDS 2004; HARALDSON ET AL. 2020; LA BROOY & C. JIANG 2009; MATHIESEN & ELLEKILDE 2017; MATHIESEN ET AL. 2018; SETHI ET AL. 1990; YEONG & DE VRIES 1994).

Laut der Ausführungen von HOFMANN (2014, S. 62–64) hängt die mittlere Fördergeschwindigkeit  $\bar{v}$  maßgeblich von der Schwingungsamplitude des VWF ab, die Effizienz  $\epsilon$  zusätzlich von den Parametern der SE. HOFMANN definiert  $\bar{v}$  und  $\epsilon$  als Zielfunktionen für das dort behandelte Optimierungsproblem, in dem beide Größen jeweils maximiert werden sollen. Zur Reduzierung der Komplexität des Optimierungsproblems wird eine Entkopplung dieser beiden Zielfunktionen erarbeitet. Aufgrund gegensätzlicher Verläufe der Funktionen existiert eine optimale Amplitude, welche als Ergebnis eine bestimmte, bestmögliche Effizienz ermöglicht. Die Fördergeschwindigkeit resultiert aus dem Zusammenhang. In der vorliegenden Arbeit wird vorausgesetzt, dass die Schwingungsparameter in der Physiks simulation nach den Vorarbeiten von HOFMANN (2014) optimal eingestellt sind. Aufgrund des eben beschriebenen Zusammenhangs entfällt damit eine gesonderte Betrachtung der Fördergeschwindigkeit  $\bar{v}$  im Rahmen dieser Arbeit.

---

<sup>3</sup>Relevant ist hier insb. die Erstausgabe aus dem Jahr 1992 (BOOTHROYD 1992).

### 5.2.1.3 Auswahl der Bewertungskriterien

Abgeleitet von den o. g. Veröffentlichungen und basierend auf den in Abschnitt 5.2.1.1 erarbeiteten Anforderungen werden in der vorliegenden Arbeit Transitionsmatrizen zur Berechnung der Effizienz  $\epsilon$  herangezogen. Da die Nutzung der Matrizen implizit auch aussortiertes Fördergut beinhaltet (vgl. Abschnitt 2.2.3), eignen sich diese für eine umfassende und eindeutige Aussage über die Eignung der SE für die gestellte Ordnungsaufgabe. Durch die Multiplikation mehrerer Transitionsmatrizen einzelner SE ist auch der Schluss auf die zusammengesetzte OS möglich. Die Matrizen lassen sich mithilfe der Physiksimulation automatisiert erzeugen und können über spezialisierte Bibliotheken für Matrix-Operationen rechenzeiteffizient weiterverarbeitet werden. Des Weiteren ist dazu keine manuelle Interaktion vonnöten, da alle notwendigen Informationen und Daten vorhanden sind und die erforderlichen Schritte als Abläufe im Simulationsprogramm hinterlegt werden können. Somit sind alle Anforderungen aus Abschnitt 5.2.1.1 erfüllt.

### 5.2.2 Umsetzbarkeit mit der Physiksimulation

Die Eignung der in Abschnitt 5.2.1 erarbeiteten Bewertungskriterien muss hinsichtlich der Umsetzbarkeit mit der Physiksimulation zudem in Hinblick auf die Validierung und damit die praktische Anwendbarkeit betrachtet werden: Zwar können Transitionsmatrizen und Effizienz mithilfe der Simulation erzeugt bzw. berechnet werden, die Ergebnisse müssen aber die Realität hinreichend genau abbilden. Dies ist insofern an dieser Stelle zentral, da die Qualität der Ergebnisse der automatisierten Generierung einer OS (s. Kapitel 6) nur so hoch sein kann wie die dafür bereitgestellten Eingangsdaten, welche durch die Physiksimulation erzeugt wurden.

Im Hinblick darauf wurden die bereits in Abschnitt 2.3.3 vorgestellten Arbeiten nochmals mit Fokus auf die darin unternommenen Realversuche zur Validierung analysiert. Insbesondere die Arbeiten von HOFMANN (2014, S. 53) und MATHIESEN (2017, S. 40-44) bestätigen die Eignung der Physiksimulation als Werkzeug zur realitätsgetreuen Abbildung des Förderprozesses sowie der Errechnung von Transitionsmatrizen zur anschließenden Weiterverarbeitung.

### 5.3 Berechnungsvorschriften für Simulationsläufe

#### 5.3.1 Automatisierte Versuchsplanung

In Abschnitt 4.3.1 wurde die Auswahl relevanter Parameter von SE und deren formelmäßige Abstimmung auf Fördergut behandelt. Zur Durchführung von Simulationsläufen mit den SE-CAD-Daten müssen deren Parameter, abhängig von den Dimensionen des Förderguts, mit konkreten Werten belegt werden. Es wurden zwei Schritte identifiziert, die hierzu erforderlich sind:

1. Wahl des Parameterbereichs und der Schrittweite darin (Abschnitt 5.3.1.1)
2. Planung der Simulationsversuche (Abschnitt 5.3.1.2)

##### 5.3.1.1 Wahl des Parameterbereichs

Zunächst muss der Bereich, innerhalb dessen die SE-Parameter sich bewegen sollen, identifiziert werden, wie in relevanter Literatur ausgeführt wird. So erläutert bspw. BOOTHROYD (2005, S. 61 f.) das Vorgehen zur Anpassung der Stufenhöhe eines Stufen-SEs an ein bestimmtes Werkstück: Es werden Versuchsreihen mit verschiedenen Höhen durchgeführt und die jeweils resultierenden Effizienzwerte verglichen, um das Optimum zu identifizieren. Des Weiteren werden umfassende Empfehlungen in Form von Wertebereichen zur Abstimmung der Parameter auf das Fördergut gegeben (BOOTHROYD 2005, S. 476–491). Dazu sind Diagramme mit Verläufen von Effizienz und Orientierungsverhalten bei Variation der SE-Parameter aufgeführt, die auf praktischen Versuchen basieren.

Darauf aufbauend wenden MATHIESEN ET AL. (2018) die Empfehlungen von BOOTHROYD (2005) an und erweitern diese um dort ausgesparte Parameter. Allerdings ist aus den Erläuterungen nicht nachvollziehbar, wie die Auswahl der Werte und Schrittweiten vorgenommen wurde, da die Dimensionen des als Beispiel verwendeten Werkstücks nicht angegeben sind. In einer Veröffentlichung von HESSE (2000b, S. 38 ff.) wird zwar nicht explizit auf Wertebereiche eingegangen, sondern lediglich auf qualitative Zusammenhänge zwischen

## 5 Methodenschritt 2: Bewertung der Schikanenelemente

---

Werkstückgeometrie und daraus resultierendem Verhalten im SE (z. B. Kippwahrscheinlichkeit an einer Aussparung aufgrund der Lage des Werkstück-Schwerpunkts). Die Notwendigkeit der Abstimmung auf das Werkstück geht aber dennoch aus den Ausführungen hervor.

Auf Basis dieser Vorarbeiten wurde die Einstellung der SE-Parameter jeweils abhängig von den maximalen Abmessungen des Werkstücks vorgenommen. Als Obergrenze kann der größte Wert der Fördergut-Abmaße herangezogen werden, da sich die SE-Parameterwerte weit unterhalb davon bewegen. So bewirkt bspw. die Einstellung der Höhe einer Stufe auf mehr als die Hälfte der längsten Werkstückkante bereits chaotisches Förderverhalten (BOOTHROYD 2005, S. 58). Als Untergrenze hat sich die Einstellung im Bereich der kleinsten Werkstück-Abmessungen bewährt, da zu kleine Parametereinstellungen oft keinen Effekt bewirken. Innerhalb dieses Bereichs sollte mindestens ein Zwischenwert verwendet werden, um einen hinreichend großen Unterschied zwischen den Stufen und damit beobachtbare Effekte zu erwirken (SIEBERTZ ET AL. 2017, S. 6). Im Anhang unter Abschnitt A.3 werden zur Veranschaulichung jeweils für konkrete SE Minimal- und Maximalwerte aufgezeigt, die den Parameterbereich eingrenzen. Die dort angegebenen Prozentwerte beziehen sich auf den formelmäßig errechneten Wert für die Einstellung der jeweiligen SE-Parameter.

Die Einstellung der Parameterstufen auf Optimalwerte ist an dieser Stelle nicht unbedingt erforderlich, da später im Prozess noch eine Parameteroptimierung vorgenommen wird (vgl. Abschnitt 6.4).

### 5.3.1.2 Planung der Simulationsversuche

Für die gewählten Parameterwerte wurde im Rahmen dieser Arbeit eine automatisierte Versuchsplanung erarbeitet. Um zu jeder Parameterkombination Ergebnisse in Form einer Transitionsmatrix zu erzeugen, wird aus den Parameterwerten ein vollfaktorieller Versuchsplan aufgestellt (SIEBERTZ ET AL. 2017, S. 6 f.). Dazu werden alle Parameter nacheinander mit allen zuvor festgelegten Werten belegt und die Liste der Kombinationen abgespeichert. Anhand dieses Versuchsplans werden die CAD-Modelle der SE parametrisiert und mit

dem Fördergut in die Simulation geladen. Somit werden automatisiert für alle Parameterkombinationen nacheinander die benötigten Transitionsmatrizen berechnet und in der CAD-Bibliothek abgespeichert.

### 5.3.2 Konvergenz der Simulationsergebnisse

Die zur Bewertung der SE errechneten Transitionsmatrizen unterliegen einer stochastischen Verteilung, da sie Orientierungswahrscheinlichkeiten beinhalten. Je mehr Versuche durchgeführt werden, desto mehr nähern sich die Einträge einer Matrix einem bestimmten Wert an. Um zu gewährleisten, dass die Ergebnisse hinreichende Genauigkeit aufweisen, sind Analysen der Konvergenz dieser Annäherung erforderlich.

Bei BOOTHROYD (2005, S. 56) finden sich entsprechende Untersuchungen zur Ermittlung der Vorzugslagen eines Werkstücks. Hier wird ein Konfidenzintervall von 95% mit einer Abweichung von weniger als  $\pm 0,05$  aus 250 Fall-Versuchen auf einen ebenen Untergrund angegeben. In MATHIESEN ET AL. 2018 werden 2.000 Simulationsdurchläufe zur Bestimmung der Vorzugslagen durchgeführt, allerdings ohne die Angabe der dabei erreichten Genauigkeit. Eine andere Veröffentlichung von MATHIESEN & ELLEKILDE (2017) untersucht die Konvergenz während des Förderprozesses in der Simulation und gibt eine verbleibende Änderung der Verteilung von  $< 0,001$  nach 4.000 Simulationsläufen über ein SE an.

Die Auswertungen der Simulationsergebnisse im Rahmen dieser Arbeit analog zum Vorgehen in MATHIESEN & ELLEKILDE (2017) haben gezeigt, dass in den meisten Fällen eine verbleibende Änderung der Verteilung von  $< 0,001$  nach 250 Simulationsläufen erreicht ist (vgl. Abbildung 5.2). Daher wurde dies als erforderliche Anzahl an Durchläufen festgelegt<sup>4</sup>.

Die Abweichungen zu den Konvergenzgeschwindigkeiten in der Literatur (MATHIESEN & ELLEKILDE 2017; MATHIESEN ET AL. 2018) konnten aufgrund der spärlichen Informationen zu den dortigen Versuchsaufbauten nicht vollständig erklärt werden. Es liegt die Vermutung nahe, dass die Verwendung von

---

<sup>4</sup>Zu Konvergenzanalysen für statistische Matrizen vgl. REBALA ET AL. 2019, S. 11.

## 5 Methodenschritt 2: Bewertung der Schikanenelemente

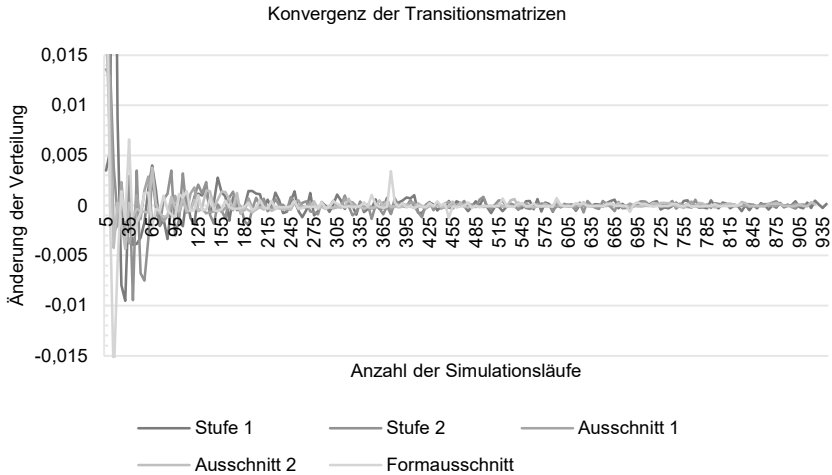


Abbildung 5.2: Konvergenz der Transitionsmatrizen für fünf repräsentative Simulationsläufe

unterschiedlichen Simulationsbibliotheken einen maßgeblichen Einfluss auf die Konvergenzentwicklung der Transitionsmatrizen hat.

Für die Methode zur automatisierten Auslegung von OS muss nach jeder Änderung der verwendeten Physik-Engine erneut untersucht werden, wie sich die Annäherung der Verteilungen in den Transitionsmatrizen verhält, um die notwendige Anzahl an Simulationsläufen zu bestimmen. Dies kann durch automatisierte Testläufe durchgeführt werden.

## 6 Methodenschritt 3: Generierung der Ordnungsschikane

### 6.1 Überblick

Im dritten Methodenschritt *Generierung der Ordnungsschikane* wird auf Basis der Ergebnisse aus dem zweiten Methodenschritt ein OS-Designvorschlag zur Orientierung des betrachteten Fördergut-Teils errechnet. Abbildung 6.1 zeigt die erarbeiteten Bausteine, die in den folgenden Abschnitten 6.2 bis 6.4 erläutert werden.

Die *Generierung der OS* wird durch ein RL-System realisiert. Um dessen Aufbau zu gestalten, erfolgt zunächst in Abschnitt 6.2.1 die mathematische Modellierung des zugrundeliegenden Optimierungsproblems als Markov-Entscheidungs-Prozess (MEP, vgl. Abschnitt 2.5.2.1). Auf dieser Basis werden in Abschnitt 6.2.2 die Struktur und Komponenten des RL-Systems entwickelt. Die in Abschnitt 6.2.1 und 6.2.2 erarbeiteten Bausteine wurden im Rahmen dieser Arbeit einmalig entworfen und müssen nicht bei jedem Methodendurchlauf wiederholt werden.

Für den Aufbau des RL-Systems werden zudem in Abschnitt 6.3.1 Lernprozess und Training konzipiert, damit das System mit den in Abschnitt 4.2.2 erarbeiteten Fördergut-Typen angelernt werden kann. Im Rahmen dieser Arbeit wurden die grundlegenden Mechanismen entworfen, sodass das RL-System vor dem ersten Einsatz der Methode trainiert werden kann. Bei bekannter Fördergut-Klasse, deren Daten bereits zum Training verwendet wurden, muss der in Abschnitt 6.3.1 beschriebene Lernprozess innerhalb eines Methodendurchlaufs somit nicht erneut durchgeführt werden. Enthalten die Eingangsdaten eine unbekannte Fördergut-Klasse, muss das RL-System dahingehend neu trainiert werden.

## 6 Methodenschritt 3: Generierung der Ordnungsschikane

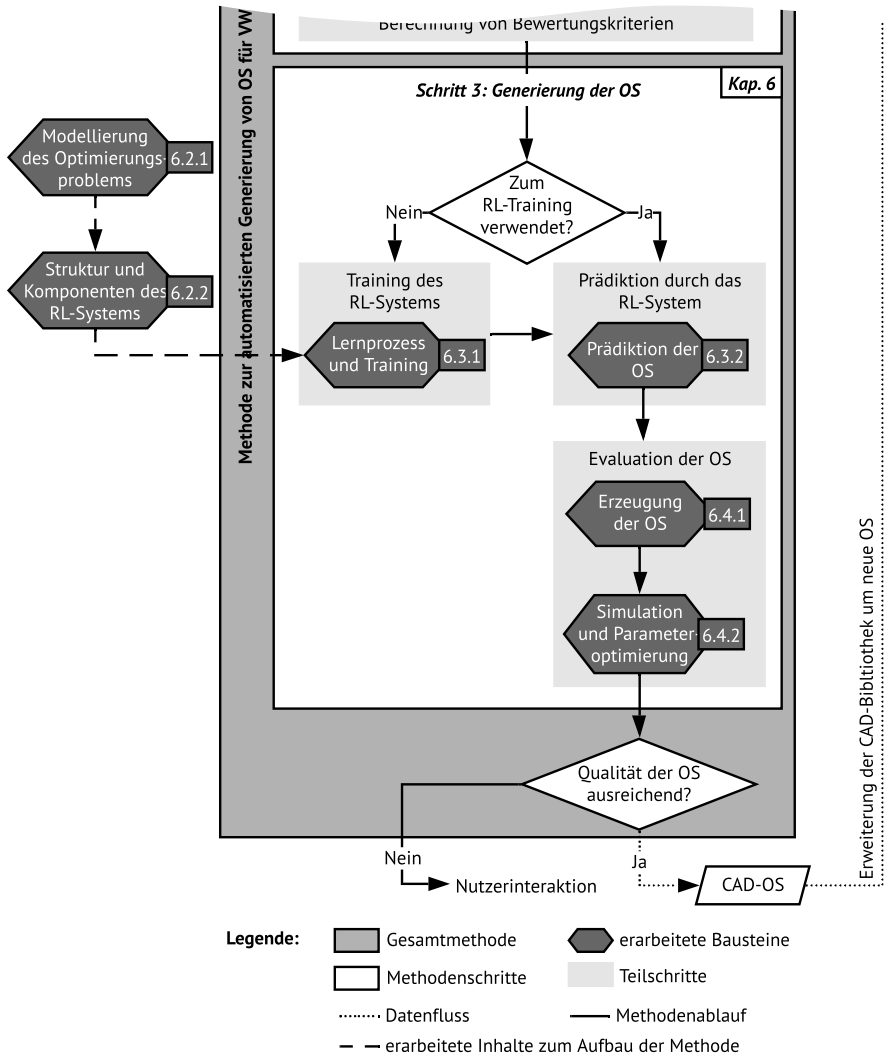


Abbildung 6.1: Methodenschritt „Generierung der Ordnungsschikane“ und erarbeitete Bausteine



Anschließend soll das trainierte System auf Basis seiner Kenntnisse die Prädiktion von OS zur Orientierung des betrachteten Fördergut-Teils ausführen (Abschnitt 6.3.2). Diese wird bei jedem Methodendurchlauf ausgeführt.

Der Designvorschlag für die OS liegt nach der Prädiktion als Liste an einzelnen SE vor. Damit die simulative Evaluation erfolgen kann, wird diese zunächst in ein durchgängiges CAD-Modell überführt. Das Vorgehen dazu wird in Abschnitt 6.4.1 erarbeitet. Auf dieser Basis kann der Designvorschlag anschließend mit Fördergut simuliert und bewertet werden. Sofern das erzielte Ergebnis noch nicht dem angestrebten Ziel einer bestimmten Effizienz entspricht, kommen Optimierungsverfahren zum Einsatz, um die Parameter der zur OS zusammengeführten SE fein zu justieren. Das dazu erarbeitete Vorgehen zur Simulation der OS und Parameteroptimierung wird in Abschnitt 6.4.2 erläutert. Die in Abschnitt 6.4.1 und 6.4.2 erarbeiteten Bausteine sind Teil des Methodenschritts *Generierung der OS* und werden bei jedem Methodendurchlauf ausgeführt.

Zuletzt erfolgt bei jedem Methodendurchlauf die Prüfung, ob die Qualität der OS nach den Vorgaben des Anwendungsfalls ausreichend ist. Falls noch Optimierungsbedarf besteht, muss eine Nutzerinteraktion erfolgen, um z. B. das Training des RL-Systems anzupassen. Falls die Qualitätsanforderungen erfüllt sind, wird die OS in der CAD-Bibliothek zur späteren Nutzung abgelegt. Mit Abschluss des letzten Methodenschritts *Evaluation der Ordnungsschikane* ist die automatisierte Generierung der OS abgeschlossen.

An dieser Stelle sei darauf verwiesen, dass das sog. „Automated Design“<sup>1</sup> als eigenes Forschungsfeld Teil der Produktentwicklungsforschung ist. Im Rahmen dieser Arbeit wird jedoch bewusst sehr speziell RL zur Generierung der OS angewandt, um ein lernfähiges System zu erzeugen. Zudem stellt dieser Aspekt lediglich einen sehr kleinen Teil der durchgängigen Gesamtmethode dar, die hier erarbeitet wird. Aus diesen Gründen soll hier nicht weiter auf das Thema Automated Design eingegangen werden, stattdessen sei auf einschlägige Publikationen der Design Society (DESIGN SOCIETY 2022) sowie der CIRP-Fachgruppe für Design (CIRP 2022) verwiesen.

---

<sup>1</sup>Dt. etwa *automatisierte Auslegung*.

### 6.2 Aufbau des Reinforcement-Learning-Systems

#### 6.2.1 Modellierung des zugrundeliegenden Optimierungsproblems

RL erfordert das Zugrundeliegen eines MEP und die Sequenzierung von SE kann als solcher formalisiert werden. Mathematisch ist das zugrundeliegende Modell ein MEP, der durch das 4-Tupel  $M = \{S, \mathcal{A}, \mathcal{R}, \mathcal{P}\}$  beschrieben wird:

- **$S$  (Menge der Zustände  $s$ ):** In Rahmen dieser Arbeit wird  $s$  durch Fördergut-Charakteristika, Ziel-Orientierung des Fördergut-Teils und die durch den Agenten erzielte Effizienz beschrieben. Die Bestandteile des Zustands sind mit Ausnahme der Effizienz statisch, ändern sich also nicht bei Ausführung einer Aktion durch den Agenten. Es ist dennoch erforderlich, diese Größen als Teil des Zustands an den Agenten zu übergeben, da dieser die Zusammenhänge zwischen den statischen Charakteristika und der erzielten Effizienz erlernen soll.
- **$\mathcal{A}$  (Menge der Aktionen  $a$ ):** Eine Aktion  $a$  beinhaltet im Kontext dieser Arbeit die Anordnung von  $k$  SE zu einer OS.
- **$\mathcal{R}$  (Menge der Belohnungen  $r$ )** für Zustand  $s$  und Aktion  $a$ : Im Rahmen der SE-Sequenzierung in der vorliegenden Arbeit erhält der Agent für das Erreichen einer bestimmten Effizienz der OS  $\epsilon$  eine positive oder negative Belohnung.
- **$\mathcal{P}$  (Menge der Übergangswahrscheinlichkeiten  $p$ )** von einem Zustand  $s_t$  in einen anderen Zustand durch die Aktion  $a$ : Im Kontext dieser Arbeit gilt stets  $p(s, a) = 1^2$ , da der Übergang von  $s_t$  nach  $s_{t+1}$  deterministisch ist: Das Setzen derselben  $k$  SE resultiert wiederholbar im selben nachfolgenden Zustand, da die Transitionsmatrizen nicht veränderlich sind und somit die berechnete Effizienz  $\epsilon$  einer gleichen SE-Anordnung stets gleich bleibt.

---

<sup>2</sup>Bedeutung: Die Wahrscheinlichkeit, durch eine bestimmte Aktion  $a$  in genau einen bestimmten Zustand  $s$  zu gelangen, liegt bei 100%. Der Übergang ist somit deterministisch.

Die Markov-Eigenschaft<sup>3</sup> ist erfüllt, da ein Zustand  $s_{t+1}$  nur jeweils vom vorhergehenden Zustand  $s_t$  und der durchgeführten Aktion  $a_t$  abhängt. Der MEP ist diskret und finit, da die Menge der Zustände und Aktionen begrenzt ist und diskrete Elemente  $s_i$  bzw.  $a_i$  enthält. Zudem ist der MEP vollständig beobachtbar, da die Zustände der Umgebung diese vollständig beschreiben.

### 6.2.2 Struktur und Komponenten des Reinforcement-Learning-Systems

Auf Basis des in Abschnitt 6.2.1 beschriebenen Modells wird das RL-System zur automatisierten Generierung von OS aufgebaut (s. Abbildung 6.2). Die Struktur orientiert sich am gängigen Grundaufbau von RL-Systemen (vgl. Abschnitt 2.5.2.1). Zur Definition der einzelnen Komponenten in Abbildung 6.2 wurden aktuelle Forschungsarbeiten herangezogen (vgl. z. B. BISCHOFF 2015; GARYCHL 2018; HAARNOJA ET AL. 2018b; A. HILL ET AL. 2018; LAPAN 2020; SUTTON & BARTO 2018) sowie Zielsetzung und Anforderungen an das angestrebte RL-Systems einbezogen (vgl. Abschnitte 1.2, 3.1 und 3.2). Nachfolgend werden die einzelnen Komponenten detailliert beschrieben, deren Funktionsweise und Interaktion wird anschließend in Abschnitt 6.3.1 erläutert.

Ziel ist, dass der Agent die Zusammenhänge zwischen Fördergut-Charakteristika und der Effizienz einer Anordnung von SE erlernt. Dies geschieht durch die Interaktion des Agenten mit der Umgebung anhand von Fördergut- und SE-Daten. Der trainierte Agent kann anschließend für unbekanntes Fördergut bekannten Typs eine Prognose für eine funktionsfähige OS abgeben.

#### 6.2.2.1 Agent

Der Agent ist die lernende Komponente des RL-Systems, welches modellbasiert konzipiert ist. Ziel des Agenten ist, die Effizienz und daraus folgend seine Beloh-

---

<sup>3</sup>Die Markov-Eigenschaft besagt, dass ein Zustand alle Informationen zur Bewertung der Folgen einer Aktion enthält und es somit ermöglicht, die optimale Aktion zu wählen (BISCHOFF 2015, S. 18; SUTTON & BARTO 2018, S. 49).

## 6 Methodenschritt 3: Generierung der Ordnungsschikane

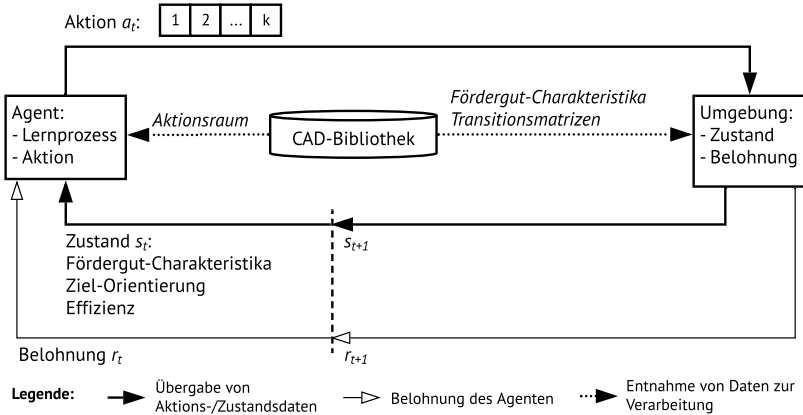


Abbildung 6.2: Übersicht über das RL-System und dessen Komponenten

nung zu maximieren, indem er eine entsprechende SE-Kombination ermittelt. Der Agent wählt seine Aktionen entsprechend seiner Strategie aus dem ihm zur Verfügung stehenden Aktionsraum. Dieser setzt sich aus den jeweils für das aktuell betrachtete Fördergut zur Verfügung stehenden SE zusammen und wird von der CAD-Bibliothek bereitgestellt.

Der Agent basiert auf Methoden des Deep RL, sodass während des Lernens im Hintergrund neuronale Netze zur Funktionsapproximation genutzt werden. Hierbei werden die Komponenten des Zustands, den der Agent von der Umgebung erhalten hat, mit der vorausgehenden Aktion verknüpft. Der Lerneffekt wird über die Anpassung der Gewichtungen innerhalb der NN erzielt, die nach dem Lernprozess in einem Modell abgelegt werden.

### 6.2.2.2 Umgebung

Der Agent provoziert durch seine Aktion eine Zustandsänderung. Aufgabe der Umgebung ist es dabei, den geänderten Zustand zu berechnen und dem Agenten zusammen mit einer daraus resultierenden Belohnung zurückzugeben. Dazu verfügt die Umgebung über eine Schnittstelle zu den benötigten Eingangsdaten in der CAD-Bibliothek. Konkret sind dies relevante Charakteristika des ak-

tuell betrachteten Förderguts (Fördergut-Typ, -Parameter, Vorzugslagen-Vektor) sowie die mit der Simulation erzeugten Transitionsmatrizen (vgl. Abschnitt 5.2). Aus diesen Informationen wird der nächste Zustand zusammengestellt sowie die Belohnungen für die Aktionen des Agenten errechnet.

### 6.2.2.3 Aktion und Zustand

Als Aktion, die der Agent durchführt, wird das Setzen verschiedener SE für ein bestimmtes Fördergut-Teil auf eine bestimmte Anzahl von Plätzen festgelegt. Der Aktionsraum besteht somit aus allen SE, die in der CAD-Bibliothek für das jeweilige Fördergut abgelegt sind. Der Agent handelt hierbei in Episoden: Er führt seine Aktionen solange aus, bis ein bestimmtes Kriterium erreicht ist. Anschließend beginnt eine neue Episode, für die ein neues Fördergut-Teil ausgewählt wird.

Die Informationen in der Zustandsbeschreibung, welche an den Agenten kommuniziert wird, bestimmen auch dessen Lernprozess (s. Abschnitt 6.3.1). Ausgehend vom Lernziel des Agenten (s. Abschnitt 6.2.2) wurden folgende Daten als relevant bewertet, um den Zustand der Umgebung zu beschreiben:

- **Fördergut-Charakteristika:** Als Basis für diese Auswahl wurde auf die Analysen in Abschnitt 4.2 zurückgegriffen. Teil der Zustandsbeschreibung sind daher die Fördergut-Klasse aus den Ergebnissen der Ähnlichkeit-Analyse (vgl. Abschnitt 4.2.2), die geometrischen Fördergut-Parameter und der Vorzugslagen-Vektor (vgl. Abschnitt 4.2.1). Diese Größen charakterisieren das Fördergut so weit, dass der Agent sie im Lernprozess mit den erzielten Belohnungen bzw. Effizienzwerten verknüpfen kann.
- **Ziel-Orientierung:** Relevant für die Auslegung einer OS und damit auch für den Agenten ist zudem, in welcher Orientierung das Fördergut am Ende vorliegen soll.
- **Effizienz:** Zentrales Kriterium für die Bewertung der Aktionen und somit den Lernerfolg des Agenten ist die Effizienz der SE-Anordnung. Sie wird nach dem in Abschnitt 5.2 vorgestellten Vorgehen berechnet.

### 6.2.2.4 Belohnung

Um die Vorteilhaftigkeit seiner Aktionen zu bewerten, muss der Agent Rückmeldung erhalten. Wie in den Grundlagen zu RL bereits beschrieben, ist der Lernerfolg nicht unbedingt garantiert, wenn die Zielgröße des Agenten mit der übergeordneten übereinstimmt (vgl. Abschnitt 2.5.2.2 und SUTTON & BARTO 2018, S. 469). Im Zuge der Untersuchungen zur vorliegenden Arbeit hat sich gezeigt, dass die Nutzung der Effizienz als Belohnungssignal keine zufriedenstellenden Lernergebnisse erzeugt<sup>4</sup>. Zur Gestaltung der Belohnung wurde daher folgendes Konzept entwickelt, wie Abbildung 6.3 als Ablaufdiagramm zeigt. Dabei wurde sich an gängigen und zum Teil langjährig bewährten RL-Benchmarks (z. B. GADGIL ET AL. 2020; KUMAR 2020; SUTTON 1996) orientiert.

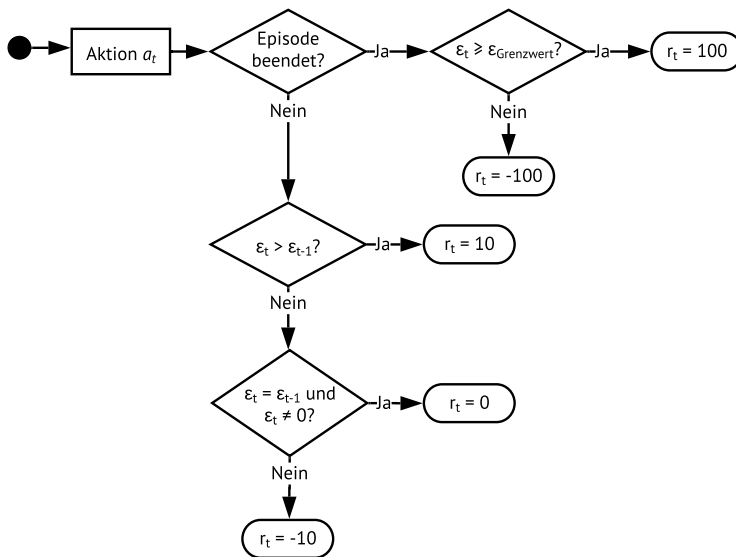


Abbildung 6.3: Belohnungskonzept für den RL-Agenten

Der Agent erhält zu zwei verschiedenen Zeitpunkten Belohnungen für seine

<sup>4</sup>Vgl. hierzu die Erläuterungen zur Ausgestaltung der Belohnung in Abschnitt 2.5.2.2.

Aktionen (s. Abbildung 6.3):

- **Innerhalb einer Episode zu jedem Zeitschritt  $t$ :** Ist die aktuelle Episode noch nicht beendet, wird jede Aktion entsprechend der erzielten Effizienz belohnt bzw. bestraft: Wenn die Effizienz zum Zeitpunkt  $t$  im Vergleich zum vorhergehenden Zeitpunkt  $t - 1$  gesteigert werden konnte ( $\epsilon_t > \epsilon_{t-1}$ ), erhält der Agent eine positive Belohnung ( $r_t = 10$ ). Gelingt keine Steigerung ( $\epsilon_t = \epsilon_{t-1}$ ), die Effizienz ist jedoch ungleich Null ( $\epsilon_t \neq 0$ ), wird keine Belohnung vergeben ( $r_t = 0$ ). Für alle anderen Fälle erhält der Agent eine negative Belohnung bzw. eine Bestrafung ( $r_t = -10$ ). Dies geschieht insb. auch, wenn die erzielte Effizienz Null ist.
- **Nach Beendigung einer Episode:** Ist die aktuelle Episode beendet, wird zudem eine übergeordnete Belohnung vergeben, welche den Lernerfolg der ganzen Episode quantifiziert: Wenn bei Beendigung einer Episode ein bestimmter minimaler Effizienz-Grenzwert erreicht oder übertroffen wurde ( $\epsilon_t \geq \epsilon_{Grenzwert}$ ), erhält der Agent eine abschließende Belohnung, die erheblich größer ausfällt als diejenigen nach den einzelnen Aktionen innerhalb der Episode ( $r_t = 100$ ). Kann der Grenzwert innerhalb der maximal zulässigen Anzahl an Zeitschritten pro Episode  $t_{max}$  nicht erreicht werden, wird der Agent dagegen empfindlich bestraft ( $r_t = -100$ ).

Auf Basis der eben beschriebenen Komponenten und Abläufe wurde das RL-System zur automatisierten Generierung von OS befähigt. Dazu werden im folgenden Abschnitt Lernprozess und Training des Agenten entwickelt und zur Prädiktion von OS angewandt.

## 6.3 Reinforcement Learning zur automatisierten Generierung von Ordnungsschikanen

### 6.3.1 Lernprozess und Training

Im Zuge des Lernprozesses nutzt der Agent die Interaktion mit der Umgebung, um ein Modell zu generieren. Dieses wird nachfolgend abgerufen, um OS für unbekannte Fördergut-Teile zu präzisieren. Der Ablauf des Trainings ist

## 6 Methodenschritt 3: Generierung der Ordnungsschikane

---

schematisch in Abbildung 6.4 zu sehen. Als Basis für das Anlernen des Agenten dienen SE- und Fördergut-Daten aus der CAD-Bibliothek. Hierbei handelt es sich um Datensätze, die für verschiedene Fördergut-Teile jeweils nach den Vorgaben aus den Abschnitten 4.2.1, 4.2.2, 4.3.1, 5.2 und 5.3 ausgewählte und erzeugte Daten wie Parameter und Transitionsmatrizen enthalten.

Vor Beginn des Trainings erfolgt die Initialisierung von Umgebung und Agent. Für letzteren werden dazu anzuwendender Algorithmus, Strategie und Parameter<sup>5</sup> spezifiziert. Die Umgebung erhält Informationen wie die maximale Anzahl an Episoden und Zeitschritten innerhalb einer Episode, sowie einen Wert für die Effizienz, die durch den Agenten mindestens erreicht werden soll. Um die Lernergebnisse des Agenten abzuspeichern, wird ein Modell angelegt, das die Gewichtungen des DNN nach dem Training ablegt.

Anschließend beginnt das Training. Für ein Fördergut-Teil aus der CAD-Bibliothek wird aus dem Aktionsraum eine initiale Aktion zusammengestellt. Der Aktionsraum besteht aus einer Liste aller in der CAD-Bibliothek für das gewählte Fördergut abgelegten SE, aus denen der Agent in Folge wählen kann. Die Umgebung erstellt als nächstes den initialen Zustand, der sich aus den in Abschnitt 6.2.2.3 beschriebenen Informationen zusammensetzt. Insbesondere wird dazu auch die Effizienz der initialen SE-Anordnung berechnet, sowie darauf basierend die initiale Belohnung.

Im nächsten Schritt beginnt der eigentliche Lernprozess des Agenten. Dieser erhält von der Umgebung den zuvor erstellten Zustand sowie die zugehörige Belohnung für die vorhergehende Aktion. Mit diesen Informationen erfolgt die Aktualisierung der Gewichtungen in den NN und die entsprechende Anpassung der Strategie. Dieser folgend wählt der Agent anschließend eine neue Aktion aus dem Aktionsraum und übergibt sie der Umgebung. Diese lädt für die Elemente der Aktion, die jeweils einem SE aus dem Aktionsraum entsprechen, die zugehörigen Transitionsmatrizen aus der CAD-Bibliothek und berechnet daraus die Effizienz, welche sich aus der Anordnung der SE ergibt. So wie auch zu Anfang wird damit der Zustand aktualisiert und die Belohnung auf Basis

---

<sup>5</sup>In diesem Zusammenhang z. B. Diskontierungsfaktor  $\gamma$  sowie die zur Parametrierung des zugrundeliegenden DNN verwendeten Parameter (vgl. Anhang, Abschnitt A.6.)



### 6.3 Reinforcement Learning zur automatisierten Generierung von Ordnungsschikanen

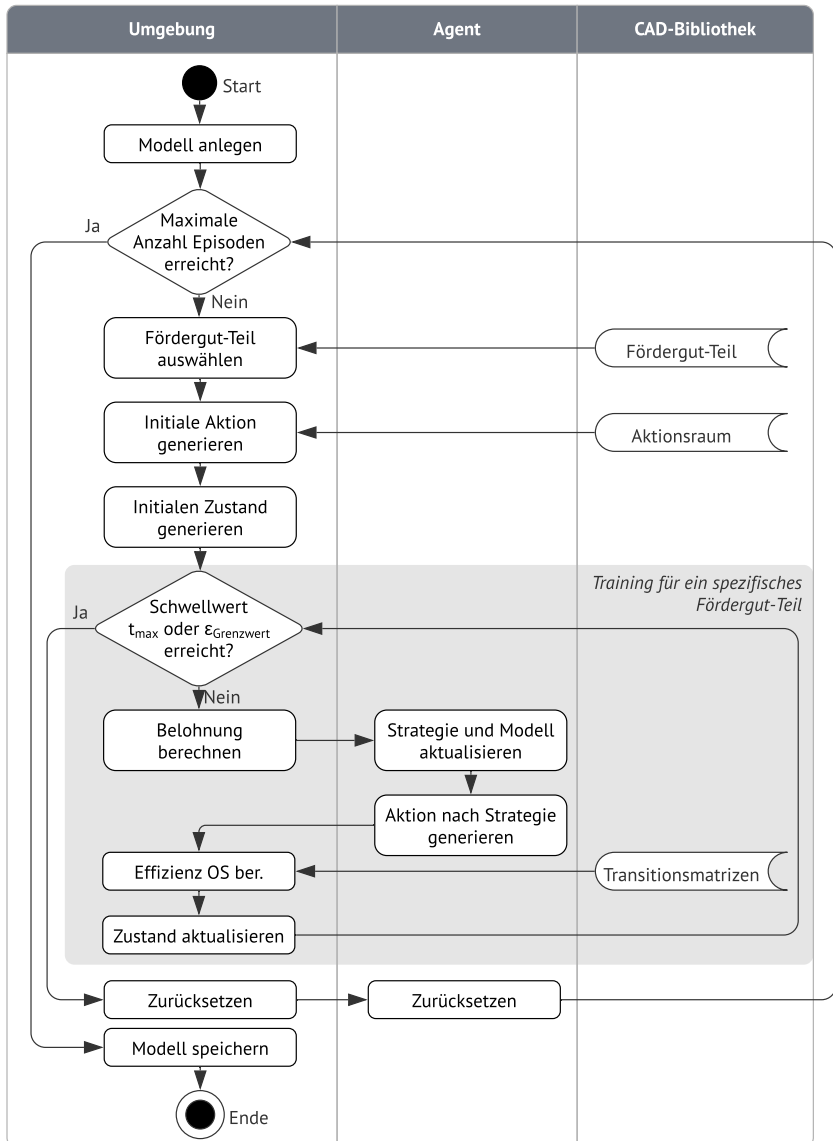


Abbildung 6.4: Training des RL-Agenten

## 6 Methodenschritt 3: Generierung der Ordnungsschikane

---

der Effizienz ermittelt. Anschließend startet die Schleife von neuem, indem Zustand und Belohnung wieder an den Agenten übermittelt werden.

Der eben beschriebene Ablauf wird solange ausgeführt, bis die Episode beendet ist. Kriterien hierfür sind entweder das Erreichen der maximalen Anzahl an Zeitschritten innerhalb einer Episode oder der mindestens erwünschten Effizienz. Nach Beendigung einer Episode werden Agent und Umgebung zurückgesetzt und ein neues Fördergut-Teil aus der CAD-Bibliothek gewählt. Wenn die maximale Anzahl an Episoden erreicht wurde, ist das Training beendet. Das finale Modell wird abgespeichert, um es später zur Prädiktion nutzen zu können.

### 6.3.2 Prädiktion von Ordnungsschikanen

Das Modell, welches nach dem Training des Agenten gespeichert wird (vgl. Abbildung 6.4), dient als Wissenspeicher für die Prädiktion von OS. Der Abruf dieses Wissens erfolgt nach einem ähnlichen Schema wie das Training des Agenten, wie Abbildung 6.5 zeigt.

Nach vorhergehender Initialisierung werden das zuvor erlernte Modell sowie das Fördergut-Teil geladen, für das eine OS prädiziert werden soll. Anschließend beginnt die Prädiktion. Für das geladene Fördergut-Teil wird aus dem Aktionsraum eine zufällige<sup>6</sup> initiale Aktion gewählt. Daraus erstellt die Umgebung den initialen Zustand und übergibt die Belohnung an den Agenten. Dieser nutzt an der Stelle das gelernte Modell, um eine Aktion abzuleiten und zurückzugeben. Dabei geht der Agent rein deterministisch vor, es erfolgt also, anders als beim Training (vgl. Abschnitt 6.3.1), keine weitere Exploration des Lösungsraums und somit kein Lernvorgang. Analog zum Training berechnet die Umgebung als nächstes die aus der Aktion resultierende Effizienz. Damit ist die Prädiktion als solches abgeschlossen.

---

<sup>6</sup>In der aktuellen Konzeption des RL-Systems wird die initiale Aktion zur OS-Prädiktion zufällig gewählt. Im Zuge der Erarbeitung hat sich gezeigt, dass die initiale Aktion in Hinblick auf Qualität und Reproduzierbarkeit der Prädiktionsergebnisse aus dem erlernten Modell generiert werden sollte (vgl. Diskussion der Ergebnisse in Abschnitt 7.3.1). Eine dementsprechende nachträgliche Anpassung hätte jedoch den Umfang der vorliegenden Arbeit gesprengt.

### 6.3 Reinforcement Learning zur automatisierten Generierung von Ordnungsschikanen

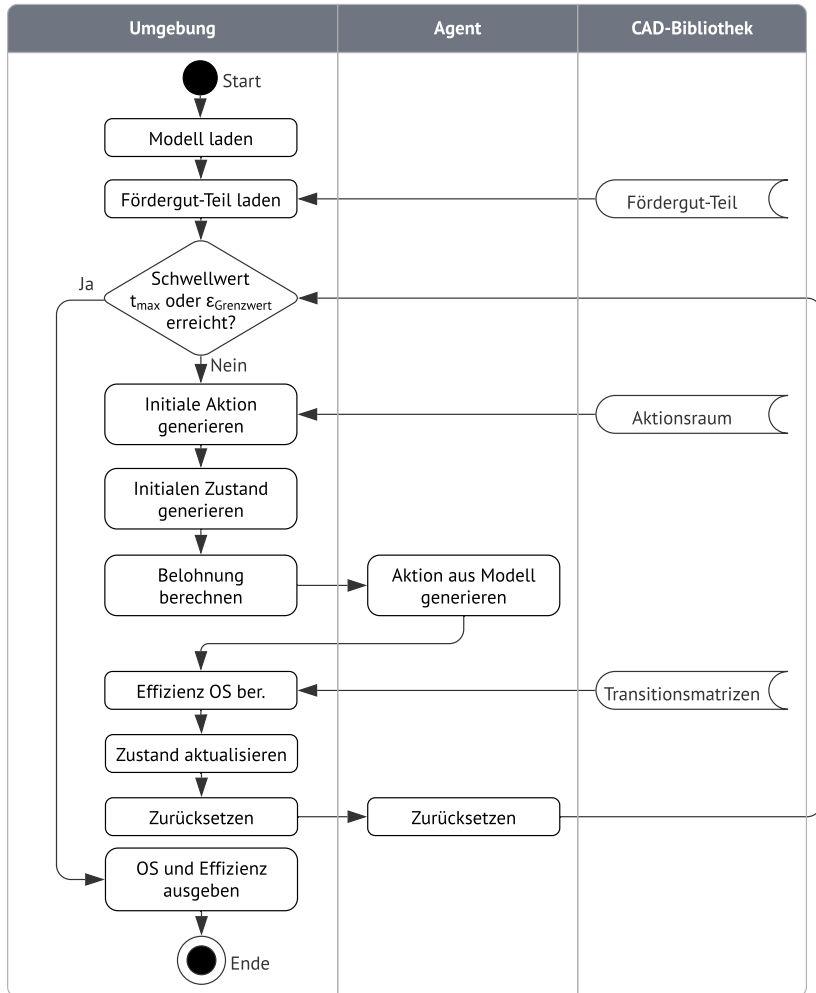


Abbildung 6.5: Prädiktion einer OS

Diese Schleife muss mehrfach wiederholt werden, da die Ergebnisse von RL-Algorithmen systemimmanent aufgrund der zufälligen Wahl der initialen Aktion nicht zu einem hohen Grad reproduzierbar sind. Abhängig von der Qualität der Anfangsaktion kann der Agent Ergebnisse von sehr unterschiedlicher Qualität erreichen. Somit ergibt eine einmalige Prädiktion meist kein auf Anhieb zufriedenstellendes Ergebnis. Daher wird die Prädiktion mehrfach wiederholt und schließlich bei Erreichen einer geforderten Mindesteffizienz oder einer maximalen Zeitspanne beendet. Abschließend werden die beste ermittelte OS sowie deren Effizienz als Ergebnis ausgegeben.

### 6.4 Simulative Evaluation und Optimierung

#### 6.4.1 Erzeugung der Ordnungsschikane aus dem Designvorschlag

Ziel dieses Schrittes ist die Erzeugung eines durchgängigen Modells der OS aus den SE, welche das RL-System als geeignet vorgeschlagen hat. Um ein adäquates Werkzeug zur Erzeugung der OS zu ermitteln, wurden zunächst die Anforderungen hergeleitet, welche das durchgängige Modell der OS erfüllen muss. Dazu wurden die praktische Eignung, insb. im Kontext der Einbettung in die Gesamtmethode und der Weiterverwendung der OS nach Durchlaufen der Methode, sowie vergleichbare Ansätze in der Literatur herangezogen (z. B. DALLINGER 2017; HOFMANN 2014; MATHIESEN 2017):

- **Simulierbarkeit des Förderprozesses:** Zur Evaluation der OS muss der Förderprozess mit dem Fördergut, für welches die OS errechnet wurde, simuliert und bewertet werden. Kriterium hierfür ist die Effizienz der OS für die gewünschte Zielorientierung des Fördergut-Teils.
- **Optimierbarkeit der SE-Parameter:** Zur Feinjustage müssen die Parameter der einzelnen SE, aus welchen der Designvorschlag der OS besteht, separat einstellbar sein (vgl. Abschnitt 6.4.2).
- **Exportmöglichkeit für das CAD-Modell:** Das Modell der OS muss für die nachfolgende Fertigung exportiert werden. Diese sollte bestenfalls additiv durchgeführt werden, da diese Art Fertigungsverfahren die größtmögliche Entkopplung von manuellen Arbeitsschritten ermöglicht (vgl.

KLEMP & POTTEBAUM 2015, S. 1), sodass ggf. spezielle Anforderungen zu berücksichtigen sind.

- **Automatisierbarkeit des Vorgangs:** Die Erzeugung der OS aus dem Designvorschlag muss automatisiert ablaufen, um die Durchgängigkeit der Methode nicht zu unterbrechen.

Anhand dieser Gesichtspunkte wurde als geeignete Lösungsmöglichkeit die Nutzung einer Physiksimulation identifiziert und im Zuge der Implementierung umgesetzt (s. Abschnitt 7.2.4).

### 6.4.2 Simulation der Ordnungsschikane und Parameteroptimierung

Nach der Erstellung einer durchgängigen OS muss diese auf ihre Funktionalität hin überprüft werden. Wie gerade in Abschnitt 6.4.1 hergeleitet, kommt als Werkzeug hierzu (wie auch schon in Abschnitt 5.2) die Physiksimulation zum Einsatz. Es werden sukzessive Fördergut-Teile in die Simulation geladen und deren Förderverhalten auf der OS simuliert. Als Bewertungskriterium wird die Effizienz der OS herangezogen.

Wie BOOTHROYD (2005, S. 61 f.) sowie HOFMANN (2014, S. 59) detailliert darstellen, ist anzunehmen, dass eine Parameteroptimierung der OS erforderlich ist, um möglichst nahe an deren optimale Funktionsweise heranzukommen. Hierzu haben HOFMANN ET AL. (2013) ein Verfahren zur automatisierten Optimierung von SE-Parametern erarbeitet und validiert. Dieses soll zur iterativen Variation und Optimierung der in Abschnitt 4.3 erarbeiteten Parameter zum Einsatz kommen. Da dieses Vorgehen in der genannten Veröffentlichung ausführlich dargestellt wird, sei auf ebendiese verwiesen. Im Rahmen der vorliegenden Arbeit wird daher nicht näher auf diesen Arbeitsschritt eingegangen.



## 7 Implementierung und Evaluierung der Auslegungsmethode

Das folgende Kapitel beinhaltet die Umsetzung der entwickelten Auslegungsmethode mit dem Ziel der Evaluierung und Validierung. Zunächst wird die erarbeitete Methode praktisch validiert (vgl. VDI 2018, S. 35), indem anhand der prototypischen Umsetzung überprüft wird, ob sie die in Kapitel 1 beschriebene Problemstellung und Zielsetzung hinreichend adressiert. Des Weiteren wird evaluiert, ob die Methode die in Kapitel 3 gestellten Anforderungen erfüllt.

Um hierbei ein methodisches Vorgehen zu gewährleisten, wurden folgende Techniken zur Validierung in Produktion und Logistik nach RABE ET AL. (2008, S. 93–111) herangezogen:

- **Test von Teilmodellen:** Systematische Überprüfung von einzelnen Bestandteilen des Konzepts zum Rückschluss auf das Gesamtkonzept (RABE ET AL. 2008, S. 106 f.)
- **Ereignisvaliditätstest:** Bewertung des Modellverhaltens anhand von Ereignissen, die aufgrund der Eingangsdaten zu erwarten sind (RABE ET AL. 2008, S. 98 f.)
- **Animation und Monitoring:** Grafische Darstellung und Beobachtung zur Überprüfung des Modellverhaltens auf Konsistenz (RABE ET AL. 2008, S. 95 f., S. 101 f.)
- **Statistische Techniken:** Bewertung, mit welcher Wahrscheinlichkeit die angenommenen (statistischen) Verteilungen eines Modells mit denen der Realität übereinstimmen (RABE ET AL. 2008, S. 103 f.)
- **Sensitivitätsanalyse:** Veränderung von Eingabeparametern des Modells und Beobachtung der Auswirkungen auf die Ergebnisgrößen (RABE ET AL. 2008, S. 102 f.)

## 7 Implementierung und Evaluierung der Auslegungsmethode

---

- **Selbsttest und Validierung im Dialog:** Sorgfältige Überprüfung des Konzepts sowie Diskussion mit Fachpersonal (RABE ET AL. 2008, S. 102, S. 109)

Abschnitt 7.1 erläutert die Rahmenbedingungen, unter denen die prototypische Implementierung der Methode durchgeführt wurde.

In Abschnitt 7.2 wird die Umsetzung der Methode anhand der automatisierten Generierung einer OS beschrieben. Mittels des *Tests von Teilmodellen* wurde die Methode in Einzelschritten auf ihre Sinnhaftigkeit überprüft. Zu diesem Zweck wurden die Methodenteilschritte anhand der o. g. Verfahren evaluiert. So wurden *Ereignisvaliditätstest* sowie *Animation und Monitoring* mithilfe der Physiksimulation und deren grafischer Darstellung im Rahmen der Datenaufbereitung, der Bewertung von SE für Fördergut sowie zur Evaluation der OS eingesetzt. Die Nutzung der *statistischen Technik* des  $X^2$ -Tests zur Evaluation der Ähnlichkeit-Analyse wurde bereits in Abschnitt 4.2.2 erläutert. Die *Sensitivitätsanalyse* kam zur Parametereinstellung der Physiksimulation sowie des RL-Systems bei der automatisierten Generierung der OS zum Einsatz. Hierzu wurden systematisch Parameter der jeweiligen Systeme variiert, die Auswirkung auf die Ergebnisse beobachtet und in Relation dazu bewertet.

Sämtliche Einzelbestandteile der Methode sowie deren Gesamtheit wurden laufend anhand des *Selbsttests und der Validierung im Dialog* evaluiert. Dazu wurden regelmäßig Gespräche mit Fachpersonal aus den relevanten Fachgebieten (z. B. Montage-/Zuführtechnik, ML, RL) geführt sowie (Teil-)Ergebnisse und -Erkenntnisse laufend hinterfragt, aus verschiedenen Blickwinkeln beleuchtet und bewertet. Auch deren Publikation<sup>1</sup> nach Durchlaufen von Peer-Review-Prozessen auf nationalen und internationalen Kongressen sowie in Journals war Teil der Evaluation.

Abschließend erfolgt in Abschnitt 7.3 die kritische Betrachtung der Forschungsergebnisse. Dazu wird die Erfüllung der Anforderungen an die Methode (vgl. Abschnitt 3.1) bewertet und mögliche Ursachen verbleibender Abweichungen erarbeitet.

---

<sup>1</sup>STOCKER & REINHART 2015, 2016; STOCKER ET AL. 2017, 2019.



### 7.1 Rahmenbedingungen

Mit dem Ziel der Validierung wurden die Teilschritte der Methode unter der Nutzung verschiedener Werkzeuge als gesonderte Teilbausteine implementiert. Um die Durchgängigkeit der Methode sicherzustellen, wurde auf eine lückenlose Gestaltung der Schnittstellen (insb. der übergebenen Daten) zwischen den Bausteinen geachtet.

Als Anwendungsfall für die Validierung wurde die Zuführung einer Schraube gewählt, da das Pilzteil nach den Analysen von LA BROOY & C. JIANG (2009), HOFMANN (2014, S. 119, in Anlehnung an FRANK 1975) sowie MATHIESEN & ELLEKILDE (2016) eine repräsentative Fördergut-Klasse darstellt, welche einen Großteil an Werkstück-Typen zur Handhabung in der automatisierten Produktion abdeckt.

Die Erläuterungen bzgl. der Umsetzung enthalten im Folgenden jeweils Angaben zu den verwendeten Werkzeugen (z. B. Software-Programme, -Bibliotheken, Programmiersprachen, zu den jeweiligen Versionen s. A.7 im Anhang). Prinzipiell ist die Umsetzung der in dieser Arbeit konzipierten Methode auch mit anderen Werkzeugen möglich. Für die Berechnungen zu Methodenbaustein 1 wurde ein handelsüblicher PC<sup>2</sup> verwendet. Für die aufwändigeren Berechnungen zu den Methodenschritten 2 und 3 kam eine leistungsstärkere Workstation<sup>3</sup> zum Einsatz.

### 7.2 Automatisierte Generierung einer Ordnungsschikane

#### 7.2.1 Datenaufbereitung

Als erster Schritt in der Methode wurde die *Datenaufbereitung* (vgl. Kapitel 4) umgesetzt. Dieser Schritt stellt die Eingangsdaten für die *Bewertung der Schikanelemente* im zweiten Methodenschritt und das Training des RL-Systems im dritten Methodenschritt *Generierung der OS* bereit.

---

<sup>2</sup>Betriebssystem: Windows 10 64-bit, CPU: Intel®Core™i5-8250U, Arbeitsspeicher: 8GB RAM.

<sup>3</sup>Betriebssystem: Windows 10 64-bit, CPU: Intel®Core™i7-4790K, Arbeitsspeicher: 16GB RAM.

### 7.2.1.1 Fördergut-Analyse

Das zu orientierende Fördergut-Teil, eine Innensechskant-Schraube, ist in Abbildung 7.1 zu sehen. Zielorientierung ist O1 zur Bereitstellung für einen dem VWF nachgelagerten Schraubautomaten.

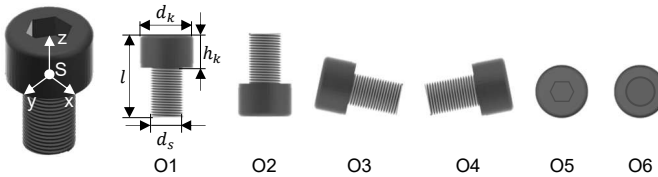


Abbildung 7.1: Für die Implementierung betrachtete Schraube: In isometrischer Ansicht (ganz links, vergrößert, mit Koordinatensystem und Schwerpunkt) sowie in den Orientierungen O1 (mit Darstellung der Fördergut-Parameter) bis O6 (vgl. BOOTHROYD 2005, S. 430)

Tabelle 7.1 zeigt die nach Abschnitt 4.2.1 relevanten Fördergut-Charakteristika. Zur Extraktion der geometrischen Eigenschaften sowie von Masse und Materialeigenschaften wurde das CAD-Programm „Autodesk Inventor Professional 2021“ (im Folgenden kurz: Inventor) eingesetzt.

Die Parameter zur Oberflächenbeschaffenheit in Tabelle 7.1 sind unter Berücksichtigung der Physiksimulation zu interpretieren: Aufgrund auf deren erfolgreicher Validierung durch HOFMANN (2014, S. 53 f.) wurde zur Simulation die Software-Bibliothek „Bullet Physics“ eingesetzt. Diese nimmt für Reib- und Stoßkoeffizienten Werte von 0 (minimale Reibung, elastischer Stoß) bis 1 (maximale Reibung, kein Rückprall bei Stoß) an. In Anlehnung an HOFMANN (2014), zudem abgeglichen mit den Materialeigenschaften des Förderguts und unter Beobachtung der Simulation hinsichtlich der realitätsnahen Abbildung des Förderverhaltens, wurden daher Werte von 0,5 für beide Koeffizienten angenommen.

Zur Ermittlung des Vektors der Vorzugslagen-Wahrscheinlichkeiten wurde dieselbe Simulationsumgebung für Fall-Versuche eingesetzt. Analog zu den Untersuchungen zur Stabilität der Simulationsergebnisse (vgl. Abschnitt 5.3.2) wurden 250 Wiederholungen ausgeführt.

## 7.2 Automatisierte Generierung einer Ordnungsschikane

Tabelle 7.1: Fördergut-Charakteristika der zu orientierenden Schraube

Charakteristikum	Parameter	Wert/Angabe	Werkzeug
Abmessungen	Gesamtlänge $l$	19,9 mm	<div style="display: flex; align-items: center;"> <div style="font-size: 3em; margin-right: 10px;">}</div> <div> <p>Feature-Extraktion mittels CAD-Programm</p> <p>Tabellen und Physik-simulation</p> </div> </div>
	Schraubenkopfdurchmesser $d_k$	12,8 mm	
	Schaftdurchmesser $d_s$	7,7 mm	
	Kopfhöhe $h_k$	7,9 mm	
Seitenverhältnisse	$l : d_k$	1,55 : 1	
Lage und Dimension von Formelementen	$h_k$	s. o.	
	$d_s$	s. o.	
Lage Symmetrie-Achsen	Relation zur z-Achse (vertikal)	0 deg (parallel zur z-Achse)	
Lage des Schwerpunkts	x-y-z-Koordinaten	$S = (0, 0, 0)$	
	<i>Anschaulich:</i> Abstand zum Kopfe	7,7 mm	
Masse	Masse $m$	11 g	
Material	Werkstoff	Legierter Stahl	
Oberflächenbeschaffenheit	Reibkoeffizient	0,5	
	Stoßkoeffizient	0,5	
Vorzugslagen-Wahrscheinlichkeit	$\vec{v} = [0,00616; 0,04517; 0,2423; 0,2423; 0,2608; 0,2033]$		Physik-simulation
Verhaltenstyp		Pilzteil	Ähnlichkeit-Analyse

Der Verhaltenstyp des Förderguts wurde mithilfe der Ähnlichkeit-Analyse ermittelt (vgl. Abschnitt 4.2.2). Evaluation und statistische Bewertung des Ansatzes wurden dort bereits vorgestellt. Das CNN-System identifiziert die zu orientierende Schraube mit einer Wahrscheinlichkeit von 0,999 als Pilzteil, wie Abbildung 7.2 zeigt. Somit entspricht das Fördergut einem Typ, welcher dem System bekannt ist.

## 7 Implementierung und Evaluierung der Auslegungsmethode

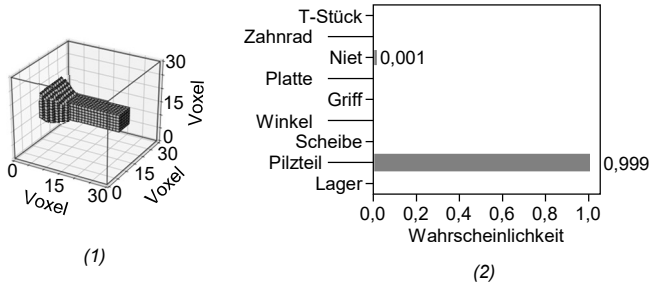


Abbildung 7.2: Ergebnisse des CNN-Systems zur Ähnlichkeit-Analyse: (1): Voxel-Annäherung der Schraube, (2): Wahrscheinlichkeit des Ergebnisses

### 7.2.1.2 Auswahl und Parametrierung von Schikanenelementen

Im Rahmen der prototypischen Umsetzung wurden zwei Sätze an CAD-Daten von SE erzeugt: Zum einen wurde die CAD-Bibliothek mit SE befüllt, um damit das RL-System initial trainieren zu können<sup>4</sup>. Diese SE wurden für die Fördergut-Teile aus dem Pilzteil-Trainingsdatensatz (s. Anhang A.2) parametrisiert und in der CAD-Bibliothek abgelegt. Zum anderen wurden SE für die zu orientierende Schraube (s. Abbildung 7.1) parametrisiert. Mit diesen erfolgt im nächsten Methodenschritt die Simulation des Förderprozesses, wie im nachfolgende Abschnitt 7.2.2 beschrieben wird.

Die SE wurden nach dem in Abschnitt 4.3.1 beschriebenen Vorgehen erstellt und jeweils Fördergut-spezifisch eingestellt. Dazu wurde zunächst für jedes Fördergut-Teil mithilfe eines Python-Skripts<sup>5</sup> die entsprechenden SE-Parameterabstufungen berechnet und tabellarisch aufgelistet. Um daraus die parametrisierten SE-Modelle zu erzeugen und strukturiert abzuspeichern, wurde erneut die CAD-Software Inventor genutzt. Inventor bietet die Möglichkeit, die Parametrierung anhand der tabellarischen Aufstellung automatisiert vorzuneh-

<sup>4</sup>Der Trainingsvorgang des RL-Systems wird im nachfolgenden Abschnitt 7.2.3 beschrieben. Im Zuge der prototypischen Umsetzung der entwickelten Methode zur Generierung von OS wurde das RL-System exemplarisch ausschließlich mit Pilzteilen trainiert.

<sup>5</sup>Python ist eine höhere Programmiersprache, die insbesondere im wissenschaftlichen Umfeld seit einigen Jahren sehr weit verbreitet ist.

men. Prinzipiell ist jedoch jedes beliebige CAD-Programm mit dieser speziellen Funktionalität zu diesem Zweck nutzbar (vgl. z. B. MATHIESEN ET AL. 2018).

Eine Übersicht über die konstruierten generischen SE sowie eine exemplarische Aufstellung der für die zu orientierende Schraube berechneten SE-Parameterwerte sind im Anhang in Abschnitt A.3 und A.4 zu finden.

### 7.2.2 Bewertung der Schikanenelemente

Nach erfolgreicher *Datenaufbereitung* im vorherigen Schritt wurden mithilfe der Physiksimulation die Transitionsmatrizen berechnet, die zur *Bewertung der Schikanenelemente* dienen. Auch hier wurden wieder zwei Sätze an Ergebnissen berechnet: Zum einen wurde der Förderprozess mit den in der CAD-Bibliothek abgelegten SE und den Fördergut-Teilen aus dem Pilzteil-Trainingsdatensatz simuliert und daraus die Transitionsmatrizen errechnet, die im nächsten Schritt für das Training des RL-Systems genutzt werden. Zum anderen wurde der Förderprozess für die zu orientierende Schraube mit den zugehörigen SE simuliert und daraus die Transitionsmatrizen berechnet.

Für die Simulationsläufe wurde aus den Parametern der Fördergut-Teile sowie den Parametern der jeweils angepassten SE ein vollfaktorieller Versuchsplan erstellt (vgl. Abschnitt 5.3.1). Die CAD-Daten der SE sowie der Fördergut-Teile wurden anhand dieses Plans automatisiert in die Physiksimulation geladen und der Förderprozess simuliert. Abbildung 7.3 zeigt ein Beispiel aus der Simulation einer geschlitzten Bahn mit einer Schraube.

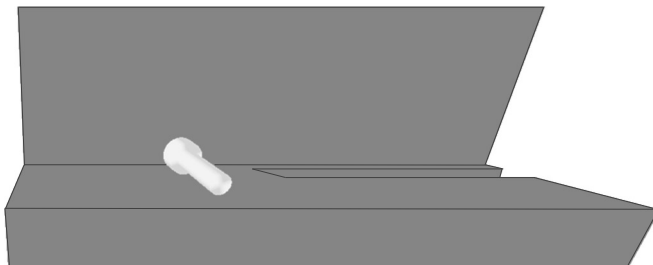


Abbildung 7.3: Simulation einer Schraube auf einer geschlitzten Bahn

## 7 Implementierung und Evaluierung der Auslegungsmethode

---

Die Anzahl der Parameter sowie der in Abschnitt 5.3.2 erarbeitete Umfang an notwendigen Simulationsläufen ergaben pro Fördergut-Teil 231 Simulationspaarungen, die mit jeweils 250 Durchläufen berechnet wurden. Als Ergebnisse zur Weiterverarbeitung wurden die Transitionsmatrizen aus dem Förderprozess strukturiert unter Nutzung des Datenformats JSON<sup>6</sup> abgespeichert. Die Rechenzeit betrug hierbei durchschnittlich 39,5 Stunden pro Fördergut-Teil. Eine graphische Auswertung der Rechenzeiten findet sich im Anhang unter A.5.

### 7.2.3 Generierung der Ordnungsschikane

Um die automatisierte *Generierung der OS* durchzuführen und zu evaluieren wurde zunächst das RL-System auf Basis des in Abschnitt 6.2 beschriebenen Aufbaus umgesetzt und wie in Abschnitt 6.3 erläutert trainiert. Das Training erfolgte mit den im vorherigen Schritt *Bewertung der SE* erzeugten Transitionsmatrizen sowie den Fördergut-Daten aus dem Schritt *Datenaufbereitung*.

#### 7.2.3.1 Aufbau des Reinforcement-Learning-Systems

Die Implementierung des RL-Systems erfolgte mit einem Python-Programm, welches die benötigten Funktionalitäten bereitstellt. Dabei wurden verschiedene, spezialisierte Software-Bibliotheken genutzt, eine Übersicht zeigt Abschnitt A.7 im Anhang.

Für die Umsetzung des Agenten wurde der Algorithmus „Soft Actor Critic (SAC)“ ausgewählt<sup>7</sup>. Hierbei handelt es sich um einen Actor-Critic Off-Policy-Algorithmus (vgl. Abschnitt 2.5.2.2), der durch Maximierung der Entropie<sup>8</sup> ein praktikables Gleichgewicht zwischen der Exploration des Suchraums und der Optimierung seiner Strategie erzielt. Im Vergleich zu anderen RL-Algorithmen weist SAC einige Vorteile auf, die für das entworfene RL-System relevant sind:

---

<sup>6</sup>Die JavaScript Object Notation (JSON) ist ein Datenformat, welche das kompakte und strukturierte Speichern von Daten in Textform ermöglicht. Da es in Python effizient eingelesen werden kann, eignet es sich sehr gut für die Verarbeitung der Transitionsmatrizen (ECMA INTERNATIONAL 2017).

<sup>7</sup>Zu den theoretischen Hintergründen s. COLAS ET AL. 2019; MAZYAVKINA ET AL. 2020; NAIR ET AL. 2020.

<sup>8</sup>„Entropie“ ist hier im wahrscheinlichkeitstheoretischen Sinne als Maß an Unsicherheit zu verstehen, vgl. hierzu z. B. DULLER 2016.

So reagieren RL-Algorithmen allgemein sehr empfindlich auf die Einstellung der Parameter, wodurch auch die Qualität der Ergebnisse beeinflusst wird. Für SAC hingegen wurde mehrfach nachgewiesen, dass dieser Algorithmus aufgrund der zugrundeliegenden mathematischen Konzepte in dieser Hinsicht relativ unempfindlich ist.

Des Weiteren benötigen viele RL-Agenten eine sehr große Menge an Lerndaten. Auch hier hebt sich der SAC-Ansatz positiv durch eine hohe Daten-Effizienz ab, da er deutlich weniger Lerndaten benötigt, um gute Ergebnisse zu erzielen (HAARNOJA ET AL. 2018a,b). Zudem zeigt die Analyse von Anwendungsbeispielen für verschiedene RL-Algorithmen, dass Strategie-basierte Algorithmen bevorzugt für kombinatorische Optimierungsprobleme zum Einsatz kommen. Diese ähneln hinsichtlich ihrer Struktur der Optimierungsaufgabe bei der Generierung einer OS aus einzelnen SE, wodurch die Entscheidung für SAC zur Lösung der vorliegenden Problemstellung bekräftigt wird (MAZYAVKINA ET AL. 2020).

Der Aktionsraum des Agenten wurde bei der Implementierung auf eine Größe von drei festgelegt, da in der industriellen Praxis meist maximal drei SE zu einer OS zusammengelegt werden (vgl. BOOTHROYD 2005, S. 48; HOFMANN 2014, S. 123; KRAUEL 2020; MATHIESEN ET AL. 2018).

Mit drei Plätzen, auf die jeweils eines der 231 SE gesetzt werden kann, ergibt sich zudem eine maximale Anzahl an Kombinationsmöglichkeiten von 12.326.391. Im Zuge der Evaluation wurde anhand der simulierten Transitionsmatrizen die größte rechnerisch erzielbare Effizienz<sup>9</sup> berechnet, indem alle Kombinationen ausgewertet wurden. So konnte die Qualität der durch den Algorithmus generierten OS mit der maximal möglichen verglichen werden. Für einen Aktionsraum mit einer Größe von vier ergibt sich bereits eine Zahl von 2.847.396.321 möglichen Kombinationen, für die eine solche Berechnung nicht mehr in sinnvoller Rechenzeit durchgeführt werden könnte.

---

<sup>9</sup>Anmerkung: Die maximal erzielbare Effizienz der OS liegt nur theoretisch bei 100% (d. h. alle Teile durchlaufen die OS und verlassen diese in gewünschter Orientierung). Praktisch fallen immer wieder Teile zurück in den Fördertopf oder verlassen die OS in einer unerwünschten Orientierung. Werden dann die Transitionsmatrizen von mehreren SE multipliziert, ergibt sich eine Gesamteffizienz von weniger als 100%.

### 7.2.3.2 Training des RL-Agenten und Prädiktion einer OS

Für das Training wurde der Agent mit den im Anhang unter Abschnitt A.6 aufgeführten Parametern eingestellt, die zuvor optimiert wurden (zur genutzten Software s. Anhang A.7). Das Training wurde mit den Transitionsmatrizen durchgeführt, die mittels der Physiksimulation für die Trainings-Fördergut-Teile berechnet wurden (vgl. Abschnitt 7.2.2 und A.2 im Anhang).

Zur Beobachtung des Trainingsfortschritts wird im RL der Verlauf der kumulierten durchschnittlichen Belohnung herangezogen. Steigt diese über den Trainingsverlauf an, steigert der Agent die erzielten Belohnungen, sodass ein Lernerfolg angenommen werden kann. Abbildung 7.4 zeigt ebendiese Beobachtung aus dem Training des Agenten, das nach 1,8 Stunden beendet war.

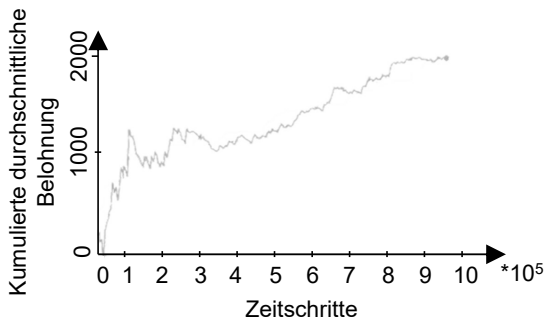


Abbildung 7.4: Verlauf der kumulierten durchschnittlichen Belohnung über den Trainingsfortschritt

Um den Agenten zu evaluieren, wurde das gespeicherte Modell nach Abschluss des Trainings für die Prädiktion einer OS für die Beispiel-Schraube aus Abbildung 7.1 genutzt. Der beste Vorschlag des Agenten erzielte eine berechnete Effizienz von 86,4% und wurde innerhalb von 5 Minuten ermittelt.

Wie Abbildung 7.5 zeigt, wurden ein SE vom Typ Durchlass und zwei verschiedene geschlitzte Bahnen vorgeschlagen. Die rechnerisch maximal mögliche Effizienz für diese drei SE in Kombination liegt bei 96,1% (zur kritischen Diskussion der Ergebnisse s. Abschnitt 7.3).



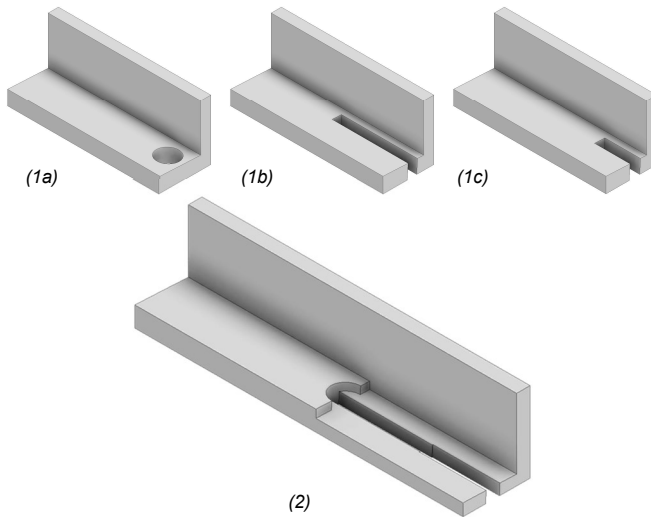


Abbildung 7.5: Ergebnis aus der OS-Prädiktion: (1a-c): Durch den RL-Algorithmus ausgewählte SE: (1a): Durchlass, (1b): Geschlitzte Bahn 1, (1c): Geschlitzte Bahn 2; (2): Zusammengefügte OS

### 7.2.4 Evaluation der Ordnungsschikane

Abschließend wurde im letzten Methodenschritt das Prädiktionsergebnis des RL-Algorithmus aus dem vorherigen Schritt *Generierung der OS* evaluiert. Dazu wurden die einzelnen SE zu einer OS zusammengefügt (Abbildung 7.5).

Dieses Modell wurde im Anschluss anhand von Simulationsläufen mit dem Fördergut-Teil validiert. Dabei wurde eine maximale Effizienz von 80,0% und somit weniger als die prognostizierte erreicht, da die Abbildung des Förderprozesses in der Simulation einer Unschärfe unterliegt. Es ist jedoch zu beachten, dass bei der praktischen Anwendung der Auslegungsmethode an dieser Stelle die Optimierung der OS nach HOFMANN (2014, S. 59) anzuschließen ist, um das Ergebnis zu finalisieren. Wie bei HOFMANN (2014, S. 123) gezeigt wird, sind durch diese Optimierungsmaßnahmen noch erhebliche weitere Effizienzsteigerungen möglich.

### 7.3 Kritische Bewertung der Forschungsergebnisse

Die Ergebnisse der prototypischen Umsetzung und Evaluation werden in den folgenden Abschnitten 7.3.1 bis 7.3.5 anhand der allgemeinen (s. Abschnitt 3.1.1) sowie der daraus abgeleiteten *problemspezifischen*<sup>10</sup> Anforderungen (s. Abschnitt 3.1.2) bewertet (vgl. hierzu Tabelle 3.1). Auf dieser Basis wird abschließend in Abschnitt 7.3.6 der Zielerfüllungsgrad der Ergebnisse unter Berücksichtigung der Zielsetzung dieser Arbeit (s. Abschnitt 1.2) ermittelt.

#### 7.3.1 Richtigkeit der Ergebnisse

Die allgemeine Anforderung der Richtigkeit der Ergebnisse kann als zum Teil erfüllt angesehen werden. Prinzipiell wurden für einen Großteil der Teilschritte bereits validierte Werkzeuge gewählt, um die Richtigkeit der Ergebnisse sicherzustellen. Bisher nicht validierte Werkzeuge wurden im Rahmen der Arbeit evaluiert und deren Eignung nachgewiesen. Der Ansatz selbst kann auch unabhängig von den gewählten, spezifischen Werkzeugen in anderer Weise umgesetzt werden, was dessen allgemeine Richtigkeit und Anwendbarkeit unterstreicht.

Zudem ist zum einen die *Korrektheit* der Lösung weitgehend gegeben, da die durch das RL-System prädiizierte OS das betrachtete Fördergut-Teil bei der Evaluation zum Großteil (86,7% bzw. 80,0%) korrekt orientiert, also in Zielorientierung hervorbringt. Dabei wird die prognostizierte Effizienz mit geringer Abweichung erzielt, die durch den Einsatz von Optimierungsmethoden jedoch voraussichtlich zu eliminieren ist.

Zum anderen konnte die gewünschte *Ergebnisqualität* nicht vollständig erreicht werden. Zentraler Punkt ist hierbei die Prädiktion der OS durch das RL-System. Im Vergleich zur rechnerisch maximal erzielbaren Effizienz erreicht das beste Ergebnis des Agenten nur knapp 90% des Optimalwerts (86,4% anstelle von 96,1%). Im Zuge der Entwicklungsarbeiten wurde die grundsätzliche Eignung des Ansatzes durch systematische Experteninterviews auf dem Gebiet des RL

---

<sup>10</sup>Im Folgenden *kursiv* hervorgehoben.

sichergestellt. Zudem wurden mögliche Ursachen für die suboptimalen Ergebnisse identifiziert, die im Folgenden erklärt werden. Diese sollten zukünftig eingehend untersucht werden, was jedoch den Rahmen dieser Arbeit sprengen würde.

Zum ersten ist die in Abschnitt 6.2 formulierte Struktur nicht typisch für RL-Anwendungen, wie sie in der Literatur zu finden sind. In der Regel zeichnen sich diese durch hochgradig mehrdimensionale Zustände aus. Im Rahmen dieser Arbeit besteht der Zustandsraum lediglich aus wenigen Fördergut-Charakteristika<sup>11</sup>, Zielorientierung und Effizienz. Dagegen ist der Aktionsraum klassischer RL-Ansätze i. d. R. von kleinerer Dimension als in der vorliegenden Arbeit. Diese verfügt mit drei Plätzen, auf die jeweils 231 SE gesetzt werden können, über einen sehr großen Aktionsraum. Um das in dieser Arbeit konzipierte RL-System im Kontext klassischer RL-Ansätze einordnen zu können, soll ein gängiger RL-Benchmark zum Vergleich herangezogen werden, der sog. „Lunar Lander“<sup>12</sup>: Hierbei stehen dem Agenten nur vier verschiedene Aktionen zur Verfügung. Der Zustandsraum des Lunar Lander besteht dagegen aus acht Parametern mit theoretisch unendlichem Wertebereich (z. B. x-/y-Koordinaten des Raumschiffs).

Des Weiteren ist auch die Struktur des Lernprozesses eher ungewöhnlich für typische RL-Anwendungen. Ebenfalls am Beispiel des Lunar Lander wird deutlich, dass der Lerneffekt beim RL i. d. R. kontinuierlicher Natur ist: Der Agent verbessert in jeder Trainingsepisode sukzessive die Strategie zur Wahl seiner Aktionen, da er das langfristige Ziel verfolgt, das Raumschiff stabil zu landen. Von Episode zu Episode wendet er die Strategie an, um die Vorteilhaftigkeit seiner Aktionen über mehrere Zeitschritte hinweg zu prognostizieren. Dies ist nur möglich, da die Aufgabe in jeder Episode gleich bleibt. Bei der Anordnung von SE sind die Episoden voneinander unabhängig, da im Sinne der Genera-

---

<sup>11</sup>Im Zuge der Implementierung zur Konzeptvalidierung wurden hierfür vier Parameter eingesetzt, vgl. Abschnitt 6.2.2.3.

<sup>12</sup>In diesem im RL-Bereich gängigen Benchmark-Problem hat der Agent die Aufgabe, ein Raumschiff sicher auf einem festen Untergrund zu landen. Dazu stehen ihm vier diskrete Aktionen zur Verfügung (Steuerdüse links ansteuern, Steuerdüse rechts ansteuern, Antriebsdüse ansteuern, keine Aktion). Als Zustand werden u. a. die x-/y-Koordinaten des Raumschiffs, die horizontale und vertikale Geschwindigkeit sowie die Orientierung im Raum zurückgegeben. Der Agent erhält eine Belohnung für das erfolgreiche Landen innerhalb eines bestimmten Areals (GADGIL ET AL. 2020; OPENAI 2019).

## 7 Implementierung und Evaluierung der Auslegungsmethode

---

lisierbarkeit des Lerneffekts innerhalb der Fördergut-Typen in jeder Episode mit einem neuen Fördergut-Teil trainiert wurde. Somit kann vom aktuellen Zustand aus nicht sinnvoll die Vorteilhaftigkeit nachfolgender Aktionen geschlossen werden. Diese Annahme wird auch dadurch bestärkt, dass mit einem Diskontierungsfaktor  $\gamma = 0$  die besten Lernergebnisse erzielt wurden<sup>13</sup>.

Zum dritten hat die Wahl der Initial-Lösung für die SE-Anordnung zu Beginn des Trainingsprozesses möglicherweise einen maßgeblichen Einfluss auf das Ergebnis. Es ist bekannt, dass RL-Systeme mitunter sehr empfindlich auf eine günstige bzw. ungünstige Wahl der initialen Aktion reagieren. Je nach Ausprägung führt diese zum zuverlässigen Erreichen oder auch Nicht-Erreichen einer sehr guten oder sogar optimalen Lösung. Die Initial-Aktion wird bei RL-Anwendungen häufig zufällig gewählt, so auch im Rahmen dieser Forschungsarbeit. Es ist anzunehmen, dass die Ergebnisse des RL-Systems reproduzierbar besser ausfallen, wenn eine geeignete Start-Lösung (bspw. aus früheren Auslegungsvorgängen mit ähnlichen Fördergut-Teilen) vorgegeben wird (BISCHOFF 2015, S. 28; SUTTON & BARTO 2018, S. 34).

Unabhängig von der Struktur des RL-Systems ist als letzter relevanter Aspekt die formelmäßige Beschreibung des Zusammenhangs zwischen den Parametern des Förderguts und der SE zu betrachten. Diese wurde, wie in Abschnitt 4.3.1 erläutert, auf Basis von validierten Vorarbeiten analytisch und anhand von Regeln bestimmt. Dieses Vorgehen ist zwar uneingeschränkt valide, birgt aber sehr wahrscheinlich eine hohe Unschärfe, da die verschiedenen Fördergut-Teile eines Typs allesamt mit derselben Formel abgebildet werden. Als möglicher Ausblick ist es denkbar, diese Zusammenhänge durch ein zusätzliches RL-System erlernen zu lassen, sodass diese für jedes Fördergut-Teil differenzierter bzw. individueller abgebildet werden.

### 7.3.2 Anwendbarkeit im industriellen Kontext

Wie durch die prototypische Umsetzung der entwickelten Methode in Abschnitt 7.2 gezeigt, ist deren Anwendbarkeit im industriellen Kontext gegeben.

---

<sup>13</sup>Mit  $\gamma = 0$  blickt der Agent minimal weit in die Zukunft und versucht nur, die aktuell nächste Belohnung zu maximieren (vgl. Abschnitt 2.5.2.2).

Es wurden Werkzeuge und Daten genutzt, die in der industriellen Praxis mit vertretbarem Aufwand verfügbar sind. Des Weiteren wurde darauf Wert gelegt, dass alle Methodenschritte prinzipiell auch mit anderen Werkzeugen umgesetzt werden können, um eine möglichst breite Anwendbarkeit zu gewährleisten.

Neben den in Abschnitt 7.3.1 bereits beleuchteten Kriterien *Korrektheit* und *Ergebnisqualität* ist die *langfristige Nutzbarkeit* für die Anforderung der Anwendbarkeit relevant. Diese wird durch die modulare und somit *erweiterbare* Struktur der Auslegungsmethode erzielt. Des Weiteren wurde eine Wissensbasis zur Speicherung und Wiederverwendung der errechneten OS konzipiert. Durch den Einsatz von RL kann das System laufend für neue Fördergut-Teile angelernt und erweitert werden, wodurch die *langfristige Nutzbarkeit* ebenfalls gesteigert wird.

Die *Rechenzeiten* im Zuge der praktischen Umsetzung erwiesen sich als unproblematisch, da für die Simulationen jeweils 1-2 Tage, für die übrigen Rechenoperationen wenige Stunden benötigt wurden (vgl. Anhang A.5). Insbesondere in Hinblick auf die Entwicklungszeiten von mehreren Monaten bei der bisherigen, konventionellen Auslegung wurde hiermit bereits eine erhebliche Verringerung des Aufwands und Einsparung von manuellen Anteilen erreicht. Hier besteht jedoch weiteres Potential, die Rechenzeit zu verringern, wenn die Methode professionell in eine durchgängige Software implementiert wird.

Da die Methode ohne die Notwendigkeit manuellen Eingreifens konzipiert wurde, sind ihre Ergebnisse *unabhängig von personenspezifischem Erfahrungswissen*. Das Bedienpersonal muss lediglich über grundlegende Fähigkeiten im Umgang mit Computern zur Einspielung des Fördergut-Teils als CAD-Modell verfügen.

Die praktische Umsetzung der Methode hat zudem gezeigt, dass die erzielbare Reproduzierbarkeit der Ergebnisse ein nicht zu vernachlässigender Aspekt ist. Diese ist ebenfalls höchst relevant für die Anwendbarkeit der Methode. Typisch für RL-Ansätze ist, dass die Reproduzierbarkeit ihrer Ergebnisse abhängig von verschiedenen Faktoren u. U. unzureichend sein kann. Gründe hierfür können z. B. die Anteile zufälliger Exploration während des Trainings, die zufällige Wahl von Start-Zuständen oder die Undurchsichtigkeit der Vorgänge innerhalb

## 7 Implementierung und Evaluierung der Auslegungsmethode

---

der zugrunde liegenden NN beim Deep RL sein. Insbesondere die Wahl der Parameter wirkt sich mitunter stark auf die Ergebnisse aus (CLAESSEN & DE MOOR 2015; HENDERSON ET AL. 2017). Um diesen Einflussfaktor zu minimieren, wurde im Rahmen der vorliegenden Arbeit systematische Parameteroptimierung betrieben (vgl. AKIBA ET AL. 2019).

In Hinblick auf die industrielle Anwendbarkeit der Methode ist des Weiteren die Simulation von Fördergut zur Berechnung der Transitionsmatrizen kritisch zu beleuchten. Wie BOOTHROYD (2005, S. 37) feststellt, kann die Wechselwirkung der geförderten Teile in der Praxis zur fehlerhaften Funktion einer OS führen. In der Praxis werden übereinanderliegende Werkstücke im Förderprozess durch konstruktive Anpassungen an der Förderstrecke auseinandergezogen<sup>14</sup>, bspw. über einen Steigungsknick im Wendelhochlauf (BOOTHROYD 2005, S. 78–85; HESSE 2000b, S. 60 f.). Im Vorgehen wie in Abschnitt 5.2 beschrieben werden jeweils nur einzelne Fördergut-Teile sequenziell simuliert. Die Gültigkeit der Ergebnisse bzw. der errechneten Transitionsmatrizen ist somit in Hinblick auf die Praxis gegeben. Nichtsdestotrotz sollten in zukünftigen Untersuchungen Szenarien mit mehreren zeitgleich simulierten Fördergut-Teilen betrachtet werden, um mögliche Wechselwirkungen zu analysieren.

Ein weiterer praktisch relevanter Aspekt ist die Abhängigkeit des Förderverhaltens im VWF vom Füllgrad des Fördertopfs. Durch den sukzessive abnehmendem Füllgrad bei der Förderung des eingefüllten Förderguts verschiebt sich die Eigenfrequenz des Systems, wodurch sich die Förderleistung reduziert. Dies kann jedoch durch stetiges intelligentes Nachfüllen behoben werden und tangiert somit nicht die im Rahmen dieser Arbeit erzielten Ergebnisse (BOOTHROYD 2005, S. 42–46).

### 7.3.3 Wirtschaftlichkeit

Um die Wirtschaftlichkeit der Methode sicherzustellen, müssen Aufwand und Nutzen im richtigen Verhältnis zueinander stehen. An zusätzlichen Kosten für ein Unternehmen, welches die entwickelte Methode einsetzen möchte, sind die

---

<sup>14</sup>Sogenanntes „Differenzialvereinzeln“ (HESSE 2010, S. 193 f.).

Lizenzkosten für das Software-Tool<sup>15</sup>, ein leistungsstarker Computer für die notwendigen Berechnungen sowie Personalkosten für das Einpflegen meist bereits vorhandener CAD-Daten in die CAD-Bibliothek anzusetzen. Die genauen Kosten hängen hierbei stark von unternehmensinternen Kostensätzen sowie der Verfügbarkeit von internen<sup>16</sup> und externen<sup>17</sup> Kapazitäten ab. Da der Methodenablauf ohne manuelle Eingriffe und dadurch wartungsarm konzipiert wurde, sind keine weiteren laufenden Kosten zu erwarten.

Der Nutzen der Auslegungsmethode zeigt sich insb. durch Betrachtung der Zeit- und Aufwandsersparnis, die durch den Einsatz erreicht werden kann. Die für die Durchführung der Methode erforderliche *Rechenzeit* bewegt sich in der Größenordnung einiger Tage, während bislang zum Teil mehrere Monate benötigt wurden. Da die Auslegung mithilfe der Methode zukünftig automatisiert und *unabhängig von personenspezifischem Erfahrungswissen* erfolgen kann, sind keine manuellen Arbeitsschritte mehr erforderlich.

Die Kompensation des anfänglichen Aufwands wird zudem durch eine langfristige Kostenersparnis ergänzt, da die automatisierte Auslegungsmethode auch *langfristig genutzt* werden kann. Die *Qualität der Ergebnisse* als Teilanforderung für die Wirtschaftlichkeit der Methode weist allerdings noch Verbesserungspotential auf, wie in Abschnitt 7.3.1 bereits ausführlich erläutert.

Tabelle 7.2 zeigt eine Aufstellung der Wirtschaftlichkeitsbetrachtung mit beispielhaften Kostensätzen. Für die Abschätzung wurde sich am aktuell erhobenen durchschnittlichen Bruttoentgelt für Beschäftigte im Maschinenbau orientiert (StMWI 2020, S. 49). Es zeigt sich unter den getroffenen Annahmen, dass die Kosten für die Einrichtung des Software-Systems zur automatisierten Auslegung von OS bereits nach 1,5 Auslegungsvorgängen durch die daraus hervorgehende Einsparung kompensiert werden können.

---

<sup>15</sup>Die Teilschritte der konzipierten Methode müssen zuvor in ein durchgängiges Programm integriert werden. Da die Entwicklung dieser Software nicht Teil der vorliegenden Forschungsarbeit ist, können hierfür keine realistischen Kostensätze angenommen werden.

<sup>16</sup>Z. B. für das Einpflegen der Daten.

<sup>17</sup>Z. B. bei Beauftragung der Entwicklung des Software-Tools.

## 7 Implementierung und Evaluierung der Auslegungsmethode

---

Tabelle 7.2: Wirtschaftlichkeitsbetrachtung mit beispielhaften Kostensätzen

<b>Einmalige Kosten für die Einrichtung des Systems</b>	
Lizenzkosten	10.000 €
Leistungsstarker Computer	5.000 €
Personalkosten zum Einpflegen der CAD-Daten	6.000 €
<b>Summe</b>	<b>21.000 €</b>

<b>Einsparung pro Auslegungsvorgang (durchschnittliche Dauer: 1,5 Monate)</b>	
Personalkosten	9.000 €
Materialkosten	5.000 €
<b>Summe</b>	<b>14.000 €</b>

<b>Gewinnschwelle</b>	
Kosten / Einsparung	1,5 Auslegungsvorgänge

### 7.3.4 Allgemeingültigkeit

Die Auslegungsmethode ist durch das Einlernen von neuen Fördergut-Typen ohne Anpassungen für neue Anwendungsfälle bei der OS-Generierung anwendbar und somit innerhalb dieser Grenze allgemeingültig.

Durch die modulare und *erweiterbare* Konzeption der Methode ist diese prinzipiell auch auf andere Anwendungsfälle übertragbar, solange die Struktur des zu lösenden Optimierungsproblems ähnlich der des hier formulierten Problems zur Sequenzierung von SE ist. Vor allem können die entwickelten Bewertungskriterien im Sinne einer Zielfunktion beliebig adaptiert und erweitert werden, um eventuell notwendige Anpassungen vorzunehmen. Dadurch wird eine Allgemeingültigkeit auch außerhalb der Anwendung zur OS-Generierung erreicht. Ein möglicher Anwendungsfall ist z. B. die Anordnung von Montagestationen (LEIBER & REINHART 2020; für weitere Beispiele sei auf MAZYAVKINA ET AL. 2020 verwiesen).

Durch die Zugrundelegung allgemeingültiger Annahmen und Voraussetzungen wurde des Weiteren die übergeordnete Allgemeingültigkeit sichergestellt.



Es wurden keine anwendungsspezifischen Einschränkungen getroffen. Die *Korrektheit* der Ergebnisse wurde durch die Wahl von validierten, im Sinne der Allgemeingültigkeit zudem prinzipiell austauschbaren Werkzeugen gewährleistet.

### 7.3.5 Zusammenfassung

Zusammengefasst erfüllt die entwickelte Methode die in Abschnitt 3.1 gestellten Anforderungen zum Teil vollständig. Hier sind *langfristige Nutzbarkeit*, *Erweiterbarkeit* und *Unabhängigkeit von personenspezifischem Erfahrungswissen* zu nennen. Den übrigen Anforderungen *Korrektheit* und *Rechenzeit* werden die Ergebnisse für den praktischen Einsatz ausreichend gerecht. Hinsichtlich der *Ergebnisqualität* sind weitere Forschungsarbeiten erforderlich. Tabelle 7.3 zeigt die Erfüllung der Anforderungen in der Übersicht.

Table 7.3: Erfüllungsgrad der an die Methode gestellten Anforderungen

	Erfüllung der Anforderung
Korrektheit	●
Langfristige Nutzbarkeit	●
Erweiterbarkeit	●
Rechenzeit	●
Ergebnisqualität	⦿
Unabhängigkeit von personenspez. Erfahrung	●

**Legende:**

- Anforderung erfüllt
- ⦿ Anforderung teilweise erfüllt, ausreichend für den praktischen Einsatz
- ⦿ Anforderung mäßig erfüllt, weitere Forschungsarbeiten erforderlich

### 7.3.6 Bewertung der Zielerreichung

Wie in der übergeordneten wissenschaftlichen Zielsetzung formuliert (vgl. Abschnitt 1.2), wurde im Zuge dieser Arbeit eine **durchgängige Methode zur automatisierten Generierung von OS** entworfen und umgesetzt. Das wissenschaftliche Ziel wurde somit erreicht. Dazu wurden bereits vorhandene Teilbausteine analysiert, angepasst und zu einer durchgängigen Methode zusammengeführt, wobei verbleibende Lücken geschlossen und geglättet wurden.

## 7 Implementierung und Evaluierung der Auslegungsmethode

---

Die ebenfalls in Abschnitt 1.2 definierten Teilziele wurden wie folgt erreicht:

**Verkürzte Entwicklungszeit für OS:** Trotz der genannten Einschränkungen insb. hinsichtlich der Ergebnisqualität ist eine Effizienzsteigerung in Hinblick auf den Gesamtprozess der Auslegung von OS gegeben. Zwar kann die automatisiert generierte OS nicht unmittelbar produktiv eingesetzt werden, da der theoretisch mögliche Effizienzwert nicht erreicht wurde, dient jedoch als wertvolle Ausgangslösung für die Finalisierung durch menschliche Expertise und verkürzt somit den aufwändigen Auslegungsprozess sowie die erforderliche Entwicklungszeit. Dies wird erreicht durch die durchgängige Struktur der Methode, die Brüche und somit Zeit- und Informationsverluste vermeidet, sowie die Nutzung von technischen Unterstützungswerkzeugen. Somit kann dieses Teilziel als weitgehend erreicht betrachtet werden.

**Steigerung der Ergebnisqualität:** Durch die Verankerung des systematischen Vorgehens sowie der Möglichkeit zum Vergleich von Lösungsalternativen in der Methode sind die Voraussetzungen für die Steigerung der Ergebnisqualität geschaffen. Wie bereits zuvor erläutert, wurde der theoretisch mögliche Effizienzwert nicht erreicht, die generierte OS dient jedoch als wertvolle Ausgangslösung zur nachfolgenden Steigerung der Ergebnisqualität. Dieses Teilziel kann daher als weitgehend erreicht betrachtet werden.

**Digitale Wissensbasis:** Das Rückgrat der entwickelten Methode bildet eine digitale Wissensbasis, in welche Daten und Auslegungsergebnisse laufend eingespeist, ausgewertet und erweitert werden. Damit ist der Grundstein für die geforderte langfristige Nutzung des Systems gelegt. Die durchgängige Gesamtmethode kommt zudem ohne manuelle Arbeitsschritte aus und ermöglicht so eine Entkopplung von personenspezifischem Wissen. Das gestellte Teilziel wurde somit vollständig erreicht.

Abschließend kann festgestellt werden, dass somit alle in Abschnitt 1.2 aufgeführten Zielsetzungen erreicht wurden, wie nachfolgende Tabelle 7.4 zeigt:

### 7.3 Kritische Bewertung der Forschungsergebnisse

---

Tabelle 7.4: Bewertung der Zielerreichung

	Erreichung des Ziels
<i>Übergeordnetes wissenschaftliches Ziel:</i>	
Durchgängige Methode	●
<i>Teilziele:</i>	
Verkürzte Entwicklungszeit	◐
Steigerung der Ergebnisqualität	◑
Digitale Wissensbasis	●

**Legende:**

- Ziel erreicht
- ◐ Ziel weitgehend erreicht



## 8 Zusammenfassung und Ausblick

### 8.1 Zusammenfassung

Bedingt durch Faktoren wie den verstärkten internationalen Wettbewerb, die steigende Bedeutung von robusten Lieferketten und der damit verbundenen Rückverlagerung von Produktionskapazitäten in Hochlohnländer wie Deutschland müssen Unternehmen zunehmend die Digitalisierung und Automatisierung von Produktionssystemen vorantreiben, um wirtschaftlich produzieren zu können. Dem steht der träge und langwierige Auslegungsprozess von OS für VWF zur automatisierten Zuführung von Schüttgut als notwendiger Bestandteil automatisierter Montagesysteme gegenüber. In Konsequenz muss dieser Prozess schneller und effizienter werden, um den wachsenden Anforderungen genügen zu können. Die im Rahmen dieser Arbeit entwickelte Methode verkürzt die notwendige Entwicklungszeit, liefert einen Beitrag zur Steigerung der Ergebnisqualität und entkoppelt den Prozess von manuellen Arbeitsschritten und personenspezifischem Fachwissen. Damit wird die automatisierte Generierung von OS als übergeordnete Zielsetzung erreicht.

Die strukturierte Analyse des Stands der Erkenntnisse in Praxis und Wissenschaft in Kapitel 2 zeigt in Hinblick auf die in Kapitel 3 erarbeiteten Anforderungen, dass die o. g. Zielsetzung bisher nicht erreicht werden konnte. Zwar können einzelne Teilschritte im Auslegungsprozess bereits effektiv unterstützt werden, es existiert jedoch bislang keine durchgängige Methode zur automatisierten Generierung von OS.

Daher wurde in den Kapiteln 4 bis 6 eine Methode entwickelt, um diese Defizite zu adressieren. Der erste Methodenschritt (Kapitel 4) bereitet die Eingangsdaten auf, welche in einem zweiten Methodenschritt (Kapitel 5) zur Bewertung von SE mittels einer Physiksimulation genutzt werden. Die bewerteten SE nutzt der Algorithmus zur automatisierten Generierung der OS (Kapitel 6) im dritten

## 8 Zusammenfassung und Ausblick

---

Methodenschritt, um mithilfe eines RL-Systems eine OS zusammenzustellen. Zuletzt wird die OS simulativ evaluiert und optimiert.

Im Zuge der Validierung wurde in Kapitel 7 gezeigt, dass die Anwendung der Methode für die Auslegung von OS sowohl richtige Lösungen erzeugt, als auch eine signifikante Effizienzsteigerung des Prozesses ermöglicht. Die Ergebnisse wurden anhand der in Kapitel 3 aufgestellten Anforderungen sowie der übergeordneten Zielsetzung (vgl. Abschnitt 1.2) bewertet. Die meisten Anforderungen konnten durch die entwickelte Methode in hohem bis sehr hohem Maße erfüllt werden, wobei sich dennoch weiterer Forschungsbedarf gezeigt hat.

Der wissenschaftliche Mehrwert dieser Arbeit liegt in der Durchgängigkeit der Methode zur automatisierten Generierung von OS, die bislang so nicht existiert. Sie basiert auf der systematischen Analyse, Anpassung und Zusammenführung von bereits vorhandenen Teilbausteinen. Verbleibende Lücken wurden in diesem Zuge mit eigens erarbeiteten Lösungsbausteinen geschlossen und geglättet, um die Durchgängigkeit der Methode sicherzustellen.

### 8.2 Ausblick

An die Ergebnisse der vorliegenden Arbeit können weitere Forschungsvorhaben angeschlossen werden. Im Folgenden sind Themengebiete skizziert, die als Grundlage für nachfolgende Arbeiten dienen können.

**Weiterentwicklung des RL-Systems:** Wie bereits in Abschnitt 7.3 ausführlich erläutert, greifen der Aufbau des RL-Systems und die formulierte Problemstruktur nicht optimal ineinander und sollten dahingehend weiterentwickelt werden. So könnte die Ergebnisqualität mithilfe von Anpassungen bei der Systemstrukturierung, des RL-Lernprozesses sowie der Wahl der Initial-Lösung gesteigert werden. Des Weiteren ist es denkbar, insbesondere nach einer Umgestaltung der Systemstruktur, dass andere Algorithmen aus dem RL (bzw. dem ML allgemein) effizienter und reproduzierbarer bessere Lösungen finden. Dadurch würde v. a. die Ergebnisqualität gesteigert, was wiederum die praktische Anwendbarkeit und die Wirtschaftlichkeit der Methode erhöhen würde.

**Parameterabstimmung von Fördergut und SE:** Die Zusammenhänge zwischen den Parametern von Fördergut und SE werden nach derzeitigem Stand analytisch und anhand von Regeln bestimmt. Um die dabei entstehende Unschärfe zu reduzieren, sollte das Ziel folgender Forschungsarbeiten sein, diese Abhängigkeiten z. B. durch Kopplung eines lernenden Systems mit der Physiksimulation interaktiv zu ermitteln.

Des Weiteren birgt die Parameteroptimierung der prädizierten OS nach HOFMANN ET AL. (2013) weiteres Potential zur Erhöhung der Ergebnisqualität. So stellt z. B. MATHIESEN (2017, S. 71) ein Verfahren zur Annäherung des besten Parametersatzes für SE auf Basis Bayesscher Optimierung vor. Durch wiederholte Anwendung dieses Verfahrens auf die durch das RL-System ermittelte OS könnten relativ schnell sehr gute Parametersätze ermittelt werden.

**Clustering für unbekannte Fördergut-Typen:** Das im Rahmen dieser Arbeit entwickelte Konzept zur Ähnlichkeit-Analyse setzt die vorherige Definition von Klassen zur Differenzierung verschiedener Fördergut-Typen voraus. Um die Flexibilität und somit die langfristige Nutzbarkeit der Methode weiter zu erhöhen, sollte die Klassifizierung durch ML-Ansätze zum Clustering automatisiert werden. Dabei könnte ein Algorithmus selbstständig relevante Eigenschaften von Objekten analysieren und diese anhand der erkannten Unterschiede verschiedenen Klassen zuordnen (vgl. REBALA ET AL. 2019, S. 4; HARALDSON ET AL. 2020).

Die hier vorgestellten Ansätze zur Weiterführung der Forschungsarbeiten auf dem Gebiet der automatisierten Generierung von OS sind nicht als vollständige Aufzählung zu verstehen. Sie sollen Anhaltspunkte geben, wie das vorgestellte Konzept sinnvoll weiterentwickelt werden kann. Insbesondere durch die rapiden Entwicklungen auf dem Gebiet des RL bzw. des ML im Allgemeinen sind laufend weitere neue Potentiale und mögliche Ansatzpunkte zu erwarten.





## Literatur

ABADI ET AL. 2015

Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Goodfellow, I.; Harp, A.; Irving, G.; Isard, M.; Jozefowicz, R.; Jia, Y.; Kaiser, L.; Kudlur, M.; Levenberg, J.; Mané, D.; Schuster, M.; Monga, R.; Moore, S.; Murray, D.; Olah, C.; Shlens, J.; Steiner, B.; Sutskever, I.; Talwar, K.; Tucker, P.; Vanhoucke, V.; Vasudevan, V.; Viégas, F.; Vinyals, O.; Warden, P.; Wattenberg, M.; Wicke, M.; Yu, Y.; Zheng, X.: TensorFlow: Large-scale machine learning on heterogeneous systems: Version 1.15.4. URL: <https://www.tensorflow.org/> (besucht am 16. 03. 2021).

ABELE & REINHART 2011

Abele, E.; Reinhart, G.: *Zukunft der Produktion: Herausforderungen, Forschungsfelder, Chancen*. München: Hanser. 2011. ISBN: 978-3-446-42595-8.

AGARWAL ET AL. 2001

Agarwal, P.; Collins, A.; Harer, J.: Minimal trap design. In: *Proceedings of the 2001 IEEE International Conference on Robotics and Automation (ICRA)*. Piscataway: IEEE Operations Center. 2001, S. 2243–2248. ISBN: 0-7803-6576-3.

AKIBA ET AL. 2019

Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M.: Optuna: A Next-generation Hyperparameter Optimization Framework. URL: <https://arxiv.org/pdf/1907.10902> (besucht am 15. 04. 2021).

AMARJYOTI 2017

Amarjyoti, S.: Deep Reinforcement Learning for Robotic Manipulation – The state of the art. URL: <https://arxiv.org/pdf/1701.08878> (besucht am 15. 04. 2021).

ASHRAFIZADEH & ZIAEI-RAD 2013

Ashrafizadeh, H.; Ziaei-Rad, S.: A numerical 2D simulation of part motion in vibratory bowl feeders by discrete element method. *Journal of Sound and Vibration* Band 332 (2013) 13, S. 3303–3314.

AUTODESK, INC. 2020

Autodesk, Inc.: Autodesk Inventor Professional: Build 183, Release: 2021.

BAUER ET AL. 2017

Bauer, H.; Breuer, P.; Richter, G.; Wüllenweber, J.: Smartening up with Artificial Intelligence (AI): What's in it for Germany and its Industrial Sector? Hrsg. von McKinsey & Company. URL: <https://www.mckinsey.com/~media/McKinsey/Industries/Semiconductors/Our%20Insights/Smartening%20up%20with%20artificial%20intelligence/Smartening-up-with-artificial-intelligence.ashx> (besucht am 24. 09. 2020).

BAUR & BLASIUS 2014

Baur, N.; Blasius, J., Hrsg. (2014): *Handbuch Methoden der empirischen Sozialforschung*. Wiesbaden: Springer VS. 2014. ISBN: 978-3-531-18939-0.

BERKOWITZ & CANNY 1996

Berkowitz, D.; Canny, J.: Designing parts feeders using dynamic simulation. In: *Proceedings of the 1996 IEEE International Conference on Robotics and Automation (ICRA)*. Piscataway: IEEE. 1996, S. 1127–1132. ISBN: 0-7803-2988-0.

BERKOWITZ & CANNY 1997

Berkowitz, D.; Canny, J.: A comparison of real and simulated designs for vibratory parts feeding. In: *Proceedings of the 1997 IEEE International Conference on Robotics and Automation (ICRA)*. Piscataway: IEEE. 1997, S. 2377–2382. ISBN: 0-7803-3612-7.

BERRETTY ET AL. 1999

Berretty, R.-P.; Goldberg, K.; Cheung, L.; Overmars, M.; Smith, G.; van der Stappen, A.: Trap design for vibratory bowl feeders. In: *Proceedings of the 1999 IEEE International Conference on Robotics and Automation (ICRA)*. Piscataway: IEEE Service Center. 1999, S. 2558–2563. ISBN: 0-7803-5180-0.

BERRETTY ET AL. 2001

Berretty, R.-P.; Goldberg, K.; Overmars, M.; van der Stappen, A.: Trap Design for Vibratory Bowl Feeders. *The International Journal of Robotics Research* Band 20 (2001) 11, S. 891–908.

BISCHOFF 2015

Bischoff, B.: *Reinforcement learning for industrial applications*. Diss. Technische Universität München (2015). Berichte aus der Informatik. Aachen: Shaker. 2015. ISBN: 978-3-84403-448-6.

BLANKEMEYER ET AL. 2022

Blankemeyer, S.; Kolditz, T.; Busch, J.; Seitz, M.; Nyhuis, P.; Raatz, A.: Adaptive aerodynamic part feeding enabled by genetic algorithm. *Production Engineering* Band 16 (2022) 1, S. 1–8.

BLESSING & CHAKRABARTI 2009

Blessing, L.; Chakrabarti, A.: *DRM, a Design Research Methodology*. London: Springer. 2009. ISBN: 978-1-84882-586-4.

BLESSING ET AL. 1995

Blessing, L.; Chakrabarti, A.; Wallace, K.: A design research methodology. In: *Proceedings of ICED 95: The 10th International Conference on Engineering Design*. Hrsg. von Hubka, V. 1995. ISBN: 3-85693-028-0. URL: [https://cpdm.iisc.ac.in/cpdm/ideaslab/paper\\_scans/uid\\_151.pdf](https://cpdm.iisc.ac.in/cpdm/ideaslab/paper_scans/uid_151.pdf) (besucht am 19.04.2022).

BMW I 2019a

BMW i, Hrsg. (2019a): *Industriestrategie 2030: Leitlinien für eine deutsche und europäische Industriepolitik*. URL: [https://www.bmw.de/Redaktion/DE/Publikationen/Industrie/industriestrategie-2030.pdf?\\_\\_blob=publicationFile&v=20](https://www.bmw.de/Redaktion/DE/Publikationen/Industrie/industriestrategie-2030.pdf?__blob=publicationFile&v=20) (besucht am 12.12.2021).

BMW I 2019b

BMW i, Hrsg. (2019b): *Wirtschaft in einer digitalisierten Zukunft – ein Strategischer Vorausschauprozess: Monatsbericht 08-2019*. URL: <https://www.bmw.de/Redaktion/DE/Schlaglichter-der-Wirtschaftspolitik/>

## Literatur

---

2019/08/kapitel-1-3-wirtschaft-in-einer-digitalisierten-zukunft.html (besucht am 12. 12. 2021).

BOOTHROYD & HO 1977

Boothroyd, G.; Ho, C.: Natural Resting Aspects of Parts for Automatic Handling. *Journal of Engineering for Industry* Band 99 (1977) 2, S. 314–317.

BOOTHROYD ET AL. 1978

Boothroyd, G.; Poli, C.; Murch, L.: *Handbook of feeding and orienting techniques for small parts*. Amherst: University of Massachusetts. 1978.

BOOTHROYD 1992

Boothroyd, G.: *Assembly automation and product design*. Band 37. Manufacturing Engineering and Materials Processing. New York: Dekker. 1992. ISBN: 0-82478-547-9.

BOOTHROYD 2005

Boothroyd, G.: *Assembly Automation and Product Design*. Manufacturing Engineering and Materials Processing. Hoboken: CRC Press. 2005. ISBN: 1-57444-643-6.

BOOTHROYD ET AL. 2011

Boothroyd, G.; Dewhurst, P.; Knight, W. A.: *Product design for manufacture and assembly*. 3. Band 74. Manufacturing Engineering and Materials Processing. Boca Raton: CRC Press. 2011. ISBN: 978-1-42008-927-1.

BORCHERT 2018

Borchert, G.: *Unteraktuierte Handhabungssysteme*. Diss. Leibniz Universität Hannover (2018). Band 2018, Band 1. Berichte aus dem match. Garbsen: TEWISS-Verlag. 2018. ISBN: 978-3-95900-202-8.

BROCKMAN ET AL. 2016

Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; Zaremba, W.: OpenAI Gym. URL: <https://arxiv.org/pdf/1606.01540> (besucht am 22. 03. 2021).

BURGARD ET AL. 2014

Burgard, M.; Othman, N.; Mai, U.; Schlenker, D.; Verl, A.: Feeding of Small Components Using the Surface Tension of Fluids. In: *Precision Assembly Technologies*

*and Systems*. Hrsg. von Turner, A. Heidelberg: Springer. 2014, S. 47–51. ISBN: 978-3-662-45585-2.

BUSCH 2016

Busch, J.: *Entwicklung einer intelligenten aerodynamischen Zuführanlage für die Hochleistungsmontage*. Diss. Gottfried Wilhelm Leibniz Universität Hannover (2016). Band 2016, Band 7. Berichte aus dem IFA. Garbsen: PZH Verlag. 2016. ISBN: 978-3-9590-00987.

BUSCH ET AL. 2015

Busch, J.; Quirico, M.; Richter, L.; Schmidt, M.; Raatz, A.; Nyhuis, P.: A genetic algorithm for a self-learning parameterization of an aerodynamic part feeding system for high-speed assembly. *CIRP Annals* Band 64 (2015) 1, S. 5–8.

CAINE 1994

Caine, M.: The design of shape interactions using motion constraints. In: *Proceedings of the 1994 IEEE International Conference on Robotics and Automation (ICRA)*. Band 1. 1994, S. 366–371.

CEARLEY 2020

Cearley, D.: Top 10 Strategic Technology Trends for 2020. Hrsg. von Gartner, Inc. URL: <https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2020/> (besucht am 24.09.2020).

R. CHEN ET AL. 2011

Chen, R.; Chen, L.; Wang, X.; Chen, X.: Dynamic design and simulation of a vibratory hopper. In: *Proceedings of the 2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC 2011)*. Hrsg. von Stokes, P. Piscataway: IEEE. 2011, S. 3935–3938. ISBN: 978-1-4577-0536-6.

Y. CHEN & YOUNG 1988

Chen, Y.; Young, R.: PACIES - A PArt Code Identification Expert System. *IIE Transactions* Band 20 (1988) 2, S. 132–136.

## Literatur

---

CHOLLET 2015

Chollet, F.: Keras: The Python Deep Learning library. URL: <https://keras.io/> (besucht am 07. 05. 2020).

CHRISTIANSEN ET AL. 1996

Christiansen, A.; Edwards, A.; Coello, C.: Automated design of part feeders using a genetic algorithm. In: *Proceedings of the 1996 IEEE International Conference on Robotics and Automation (ICRA)*. Band 1. Piscataway: IEEE. 1996, S. 846–851. ISBN: 0-7803-2988-0.

CHUA & F. TAN 2006

Chua, P.; Tan, F.: Dynamic Computer Simulation of Parts feeding on a Vibratory Bowl Feeder. *Journal - The Institution of Engineers, Malaysia* Band 67 (2006) 2, S. 55–60.

CHUA & TAY 1998

Chua, P.; Tay, M.: Modelling the Natural Resting Aspect of Small Regular Shaped Parts. *Journal of Manufacturing Science and Engineering* Band 120 (1998) 3, S. 540–546.

CIOFFI ET AL. 2020

Cioffi, R.; Travaglioni, M.; Piscitelli, G.; Petrillo, A.; Parmentola, A.: Smart Manufacturing Systems and Applied Industrial Technologies for a Sustainable Industry: A Systematic Literature Review. *Applied Sciences* Band 10 (2020) 8, S. 2897–2920.

CIRP 2022

CIRP: STC Dn - Design. URL: <https://www.cirp.net/scientific-groups/stc-scientific-technical-committees/stc-dn-design.html> (besucht am 20. 04. 2022).

CLAESEN & DE MOOR 2015

Claesen, M.; De Moor, B.: Hyperparameter Search in Machine Learning. URL: <http://arxiv.org/pdf/1502.02127v2> (besucht am 02. 07. 2020).

COHEN ET AL. 2019

Cohen, Y.; Faccio, M.; Pilati, F.; Yao, X.: Design and management of digital

manufacturing and assembly systems in the Industry 4.0 era. *The International Journal of Advanced Manufacturing Technology* Band 105 (2019) 9, S. 3565–3577.

COLAS ET AL. 2019

Colas, C.; Sigaud, O.; Oudeyer, P.-Y.: A Hitchhiker’s Guide to Statistical Comparisons of Reinforcement Learning Algorithms. URL: <https://arxiv.org/pdf/1904.06979> (besucht am 04.05.2021).

COMAND ET AL. 2018

Comand, N.; Boschetti, G.; Rosati, G.: Vibratory Feeding of Cylindrical Parts: A Dynamic Model. In: *Proceedings of the Second International Conference of IFToMM Italy*. Hrsg. von IFToMM ITALY. Cham: Springer. 2018, S. 203–210.

COUMANS 2019

Coumans, E.: Bullet Physics: Bullet Real-Time Physics Simulation: Version 2.83, Release: 2019. URL: <https://pybullet.org/wordpress/> (besucht am 16.03.2021).

DALLINGER ET AL. 2012

Dallinger, N.; Risch, T.; Nendel, K.: Simulation von Förderprozessen bei Vibrationsförderanlagen. *Logistics Journal* (2012) 2012. URL: <https://www.logistics-journal.de/proceedings/2012/3426> (besucht am 15.04.2021).

DALLINGER 2017

Dallinger, N.: Die Diskrete Elemente Methode als Simulationsmethode in der Vibrationsfördertechnik. Diss. Technische Universität Chemnitz. 2017. URL: <https://nbn-resolving.org/urn:nbn:de:bsz:chl-qucosa-229364> (besucht am 15.04.2021).

DAMMANN 2020

Dammann, P.: Einführung in das Reinforcement Learning. URL: [https://hci.iwr.uni-heidelberg.de/system/files/private/downloads/541645681/dammann\\_reinforcement-learning-report.pdf](https://hci.iwr.uni-heidelberg.de/system/files/private/downloads/541645681/dammann_reinforcement-learning-report.pdf) (besucht am 30.06.2020).

## Literatur

---

DESIGN SOCIETY 2022

Design Society: The Design Society - a worldwide community. URL: <https://www.designsociety.org/> (besucht am 20. 04. 2022).

DÖBEL ET AL. 2018

Döbel, I.; Leis, M.; Vogelsang, M.; Neustroev, D.; Petzka, H.; Riemer, A.; Rüping, S.; Voss, A.; Wegele, M.; Welz, J.: *Maschinelles Lernen: Eine Analyse zu Kompetenzen, Forschung und Anwendung*. Hrsg. von Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V. München. URL: [https://www.bigdata.fraunhofer.de/content/dam/bigdata/de/documents/Publikationen/Fraunhofer\\_Studie\\_ML\\_201809.pdf](https://www.bigdata.fraunhofer.de/content/dam/bigdata/de/documents/Publikationen/Fraunhofer_Studie_ML_201809.pdf) (besucht am 05.10.2020).

DÖRING & BORTZ 2016

Döring, N.; Bortz, J.: *Forschungsmethoden und Evaluation in den Sozial- und Humanwissenschaften*. 5. vollständig überarbeitete, aktualisierte und erweiterte Aufl. Springer-Lehrbuch. Berlin: Springer. 2016. ISBN: 978-3-642-41089-5.

DÖRN 2018

Dörn, S.: Markov-Modelle. In: *Programmieren für Ingenieure und Naturwissenschaftler*. Hrsg. von Dörn, S. eXamen.press. Berlin: Springer Vieweg. 2018, S. 219–290. ISBN: 978-3-662-54304-7.

DUDENREDAKTION o. J.(a)

Dudenredaktion: deskriptiv auf Duden online. URL: <https://www.duden.de/node/134043/revision/134079> (besucht am 26. 05. 2020).

DUDENREDAKTION o. J.(b)

Dudenredaktion: präskriptiv auf Duden online. URL: <https://www.duden.de/node/156737/revision/156773> (besucht am 26. 05. 2020).

DUDENREDAKTION o. J.(c)

Dudenredaktion: Studie auf Duden online. URL: <https://www.duden.de/node/175744/revision/175780> (besucht am 20. 05. 2020).



## DULLER 2016

Duller, C.: Die Maximum-Entropie-Methode zur Bestimmung von Mischanteilen. *Austrian Journal of Statistics* Band 32 (2016) 3, S. 183–200.

## ECMA INTERNATIONAL 2017

Ecma International: The JSON data interchange syntax: ECMA-404. URL: <https://www.ecma-international.org/publications-and-standards/standards/ecma-404/> (besucht am 19.03.2021).

## EDWARDS 2004

Edwards, A.: Using a parallel genetic algorithm to design vibratory bowl feeders. In: *Proceedings of the 42nd annual Southeast regional conference*. Hrsg. von Yoo, S.-M.; Etkorn, L. New York: ACM. 2004, S. 320–325. ISBN: 1-58113-870-9.

## EFFENBERGER ET AL. 2013

Effenberger, I.; Kühnle, J.; Verl, A.: Fast and flexible 3D object recognition solutions for machine vision applications. In: *Proceedings Volume 8661, Image Processing: Machine Vision Applications VI*. Hrsg. von Bingham, P.; Lam, E. SPIE Proceedings. Washington: SPIE. 2013, 86610N1–86610N10.

## ELLEKILDE ET AL. 2016

Ellekilde, L.-P.; Buch, J.; Iversen, T.; Laursen, J.; Mathiesen, S.; Sørensen, L.; Kraft, D.; Savarimuthu, T.; Petersen, H.; Chrysostomou, D.; Hansson, M.; Schou, C.: Industrial Assembly Cases. URL: <https://portal.findresearcher.sdu.dk/en/publications/industrial-assembly-cases> (besucht am 15.05.2020).

## ERTEL 2013

Ertel, W.: *Grundkurs Künstliche Intelligenz: Eine praxisorientierte Einführung*. 3. Aufl. Wiesbaden: Springer Vieweg. 2013. ISBN: 978-3-8348-2157-7.

## ESCHMANN 2021

Eschmann, J.: Reward Function Design in Reinforcement Learning. In: *Reinforcement Learning Algorithms: Analysis and Applications*. Springer, Cham. 2021, S. 25–33.

### EU 2003

EU, Hrsg. (2003): Empfehlung der Kommission vom 6. Mai 2003 betreffend die Definition der Kleinstunternehmen sowie der kleinen und mittleren Unternehmen: Amtsblatt der Europäischen Union. Brüssel. URL: <https://eur-lex.europa.eu/legal-content/DE/TXT/PDF/?uri=CELEX:32003H0361&from=EN> (besucht am 16.06.2021).

### FEINBERG & SHWARTZ 2002

Feinberg, E. A.; Shwartz, A., Hrsg. (2002): *Handbook of Markov Decision Processes: Methods and Applications*. Band 40. International Series in Operations Research & Management Science. New York: Springer Science+Business Media. 2002. ISBN: 978-1-4615-0805-2.

### FELDMANN ET AL. 2004

Feldmann, K.; Slama, S.; Gergs, H.-J.; Wirth, U.: *Montage strategisch ausrichten – Praxisbeispiele marktorientierter Prozesse und Strukturen*. VDI-Buch. Berlin: Springer. 2004. ISBN: 978-3-64218-742-1.

### FELDMANN ET AL. 2014

Feldmann, K.; Schöppner, V.; Spur, G., Hrsg. (2014): *Handbuch Fügen, Handhaben, Montieren. 2.*, vollständig neu bearbeitete Aufl. Edition Handbuch der Fertigungstechnik. München: Hanser. 2014. ISBN: 978-3-44643-656-5.

### FRÄDRICH & NYHUIS 2009

Frädrich, T.; Nyhuis, P.: Aerodynamische Zentrifuge – Vereinzeln und Orientieren mit Luft. In: *Zuführtechnik*. Hrsg. von Zäh, M. Band 95. IWB Seminarberichte. München: Utz. 2009, S. 5-1-5-21. ISBN: 978-3-89675-095-2.

### FRANK 1975

Frank, H.: *Handhabungseinrichtungen: Grundlagen, Übersicht, Beispiele*. Band 13. Produktionstechnik heute. Mainz: Krausskopf. 1975. ISBN: 3-78300-101-3.

### GADGIL ET AL. 2020

Gadgil, S.; Xin, Y.; Xu, C.: Solving The Lunar Lander Problem under Uncertainty using Reinforcement Learning. URL: <https://arxiv.org/pdf/2011.11850> (besucht am 30.03.2021).

GARTNER, INC. 2020

Gartner, Inc.: 5 Trends Drive the Gartner Hype Cycle for Emerging Technologies, 2020. URL: <https://www.gartner.com/smarterwithgartner/5-trends-drive-the-gartner-hype-cycle-for-emerging-technologies-2020/> (besucht am 24.09.2020).

GARTNER, INC. 2021a

Gartner, Inc.: Gartner Hype Cycle Research Methodology. URL: <https://www.gartner.com/en/research/methodologies/gartner-hype-cycle> (besucht am 27.04.2021).

GARTNER, INC. 2021b

Gartner, Inc.: Gartner Top Strategic Technology Trends for 2021. URL: <https://www.gartner.com/smarterwithgartner/gartner-top-strategic-technology-trends-for-2021/> (besucht am 27.04.2021).

GARYCHL 2018

garychl: Applications of Reinforcement Learning in Real World. URL: <https://towardsdatascience.com/applications-of-reinforcement-learning-in-real-world-1a94955bcd12> (besucht am 22.07.2020).

GOEMANS & VAN DER STAPPEN 2008

Goemans, O.; van der Stappen, A.: On the design of traps for feeding 3D parts on vibratory tracks. In: *Proceedings of the 2008 IEEE International Conference on Robotics and Automation (ICRA)*. Piscataway: IEEE Service Center. 2008, S. 385–392. ISBN: 978-1-4244-1646-2.

GOEMANS ET AL. 2006

Goemans, O.; Goldberg, K.; van der Stappen, A.: Blades: A New Class of Geometric Primitives for Feeding 3D Parts on Vibratory Tracks. In: *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA)*. Piscataway: IEEE. 2006, S. 1730–1736. ISBN: 0-7803-9505-0.

GOLDBERG ET AL. 1995

Goldberg, K.; Craig, J.; Carlisle, B.; Zanutta, R.: Estimating throughput for a flexible part feeder. In: *Experimental Robotics IV*. Hrsg. von Khatib, O.; Salisbury,

## Literatur

---

J. Lecture Notes in Control and Information Sciences. Berlin: Springer. 1995, S. 486–497. ISBN: 978-3-540-76133-4.

GOLDBERG ET AL. 1999

Goldberg, K.; Mirtich, B.; Zhuang, Y.; Craig, J.; Carlisle, B.; Canny, J.: Part Pose Statistics: Estimators and Experiments. *IEEE Transactions on Robotics and Automation* Band 15 (1999) 5, S. 849–857.

GOODFELLOW ET AL. 2016

Goodfellow, I.; Bengio, Y.; Courville, A.: *Deep learning*. Cambridge: MIT Press. 2016. ISBN: 978-0-26203-561-3.

GOOGLE, INC. 2020

Google, Inc.: TensorFlow: Why TensorFlow. URL: <https://www.tensorflow.org/> (besucht am 07. 05. 2020).

GÜNTHER 2018

Günther, J.: Machine intelligence for adaptable closed loop and open loop production engineering systems. Diss. Technische Universität München. 2018. URL: <http://mediatum.ub.tum.de/?id=1381895> (besucht am 03. 06. 2020).

HAARNOJA ET AL. 2018a

Haarnoja, T.; Zhou, A.; Hartikainen, K.; Tucker, G.; Ha, S.; Tan, J.; Kumar, V.; Zhu, H.; Gupta, A.; Abbeel, P.; Levine, S.: Soft Actor-Critic Algorithms and Applications. URL: <https://arxiv.org/pdf/1812.05905> (besucht am 25. 01. 2021).

HAARNOJA ET AL. 2018b

Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S.: Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. URL: <https://arxiv.org/pdf/1801.01290> (besucht am 25. 01. 2021).

HAHN 1988

Hahn, J.: Realistic animation of rigid bodies. *ACM SIGGRAPH Computer Graphics* Band 22 (1988) 4, S. 299–308.

HAN & GAO 2010

Han, L.; Gao, J.: A Study on the Modelling and Simulation of Part Motion in

Vibratory Feeding. *Applied Mechanics and Materials* Band 34-35 (2010), S. 2006–2010.

HANSSON ET AL. 2016

Hansson, M.; Mathiesen, S.; Ellekilde, L.-P.; Madsen, O.: Configuration system for simulation based design of vibratory bowl feeders. In: *Proceedings of the 2016 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAN)*. Piscataway: IEEE. 2016, S. 147–154. ISBN: 978-1-5090-4616-4.

HARALDSON ET AL. 2020

Haraldson, D.; Sørensen, L.; Mathiesen, S.: Free-form Trap Design for Vibratory Feeders using a Genetic Algorithm and Dynamic Simulation. In: *Proceedings of the 17th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*. Hrsg. von Gusikhin, O.; Madani, K.; Zaytoon, J. Band 1. Setúbal: SCITEPRESS. 2020, S. 294–304. ISBN: 978-989-758-442-8.

HARRIS ET AL. 2020

Harris, C.; Millman, K.; van der Walt, S.; Gommers, R.; Virtanen, P.; Cournapeau, D.; Wieser, E.; Taylor, J.; Berg, S.; Smith, N.; Kern, R.; Picus, M.; Hoyer, S.; van Kerkwijk, M.; Brett, M.; Haldane, A.; Del Río, J.; Wiebe, M.; Peterson, P.; Gérard-Marchant, P.; Sheppard, K.; Reddy, T.; Weckesser, W.; Abbasi, H.; Gohlke, C.; Oliphant, T.: Array programming with NumPy. *Nature* Band 585 (2020) 7825, S. 357–362.

HASTIE ET AL. 2017

Hastie, T.; Tibshirani, R.; Friedman, J.: *The elements of statistical learning: Data mining, inference, and prediction*. 2. Aufl. Springer series in statistics. New York: Springer. 2017. ISBN: 978-0-38784-858-7.

HAUGAARD ET AL. 2022

Haugaard, R.; Iversen, T.; Buch, A.; Kramberger, A.; Mathiesen, S.: A Flexible and Robust Vision Trap for Automated Part Feeder Design. URL: <http://arxiv.org/pdf/2206.00373v1> (besucht am 04. 06. 2022).

HECKER ET AL. 2017

Hecker, D.; Döbel, I.; Petersen, U.; Rauschert, André, Schmitz, Velina; Voss, A.:

## Literatur

---

Zukunftsmarkt Künstliche Intelligenz: Potenziale und Anwendungen. Hrsg. von Fraunhofer-Allianz Big Data. URL: [https://www.bigdata.fraunhofer.de/content/dam/bigdata/de/documents/Publikationen/KI-Potenzialanalyse\\_2017.pdf](https://www.bigdata.fraunhofer.de/content/dam/bigdata/de/documents/Publikationen/KI-Potenzialanalyse_2017.pdf) (besucht am 24. 09. 2020).

HELFRICH 2016

Helfrich, H.: *Wissenschaftstheorie für Betriebswirtschaftler*. Wiesbaden: Springer Gabler. 2016. ISBN: 978-3-658-07035-9.

HENDERSON ET AL. 2017

Henderson, P.; Islam, R.; Bachman, P.; Pineau, J.; Precup, D.; Meger, D.: Deep Reinforcement Learning that Matters. URL: <https://arxiv.org/pdf/1709.06560> (besucht am 31. 03. 2102).

HESSE 2000a

Hesse, S.: *Fertigungsautomatisierung: Automatisierungsmittel, Gestaltung und Funktion*. Wiesbaden: Vieweg+Teubner. 2000. ISBN: 978-3-52803-914-1.

HESSE 2000b

Hesse, S.: Rationalisierung der Kleinteilezuführung: Ordnen, Sortieren, Prüfen und Zuführen. Hrsg. von Festo AG. Esslingen.

HESSE 2010

Hesse, S.: *Grundlagen der Handhabungstechnik. 2.*, aktualisierte und erweiterte Aufl. München: Hanser. 2010. ISBN: 978-3-44641-872-1.

HESSE & MALISA 2016

Hesse, S.; Malisa, V., Hrsg. (2016): *Taschenbuch Robotik, Montage, Handhabung: Mit 34 Tabellen. 2.*, neu bearbeitete Aufl. München: Fachbuchverlag Leipzig im Carl Hanser Verlag. 2016. ISBN: 978-3-44644-365-5.

HILGENBÖCKER 1985

Hilgenböcker, H.: *Methodische Entwicklung von Zuführsystemen*. Diss. Universität Hannover (1985). Band 97. Fortschritt-Berichte VDI/Reihe 2, Fertigungstechnik. Düsseldorf: VDI-Verlag. 1985. ISBN: 3-18149-702-9.

A. HILL ET AL. 2018

Hill, A.; Raffin, A.; Ernestus, M.; Gleave, A.; Kanervisto, A.; Traore, R.; Dhariwal,

P.; Hesse, C.; Klimov, O.; Nichol, A.; Plappert, M.; Radford, A.; Schulman, J.; Sidor, S.; Wu, Y.: Stable Baselines: Version 2.10.1, Release: 2018. URL: <https://github.com/hill-a/stable-baselines> (besucht am 01.02.2021).

HOFMANN 2014

Hofmann, D.: Simulationsgestützte Auslegung von Ordnungsschikanen in Vibrationswendelförderern. Diss. Technische Universität München. 2014. URL: <http://mediatum.ub.tum.de/node?id=1216825> (besucht am 15.04.2021).

HOFMANN & STIRNER 2017

Hofmann, D.; Stirner, F.: Digitale Zwillinge in der Zuführtechnik. Hrsg. von Hofmann & Stirner Zuführsysteme GmbH. Ingolstadt. URL: <https://www.hofmannstirner.de/app/download/11389881397/White+Paper+Digitale+Zwillinge+in+der+Zuf%C3%BChrtechnik.pdf?t=1533136105> (besucht am 23.06.2019).

HOFMANN ET AL. 2013

Hofmann, D.; Huang, H.; Reinhart, G.: Automated Shape Optimization of Orienting Devices for Vibratory Bowl Feeders. *Journal of Manufacturing Science and Engineering* Band 135 (2013) 5.

HONEIN 2020

Honein, J.: *Classification of 3D CAD Models for Vibratory Bowl Feeders Using Convolutional Neural Network*. Bachelorarbeit (Nr. 51994). Technische Universität München (2020). München. 2020.

HOU ET AL. 2005

Hou, S.; Lou, K.; Ramani, K.: SVM-based Semantic Clustering and Retrieval of a 3D Model Database. *Computer-Aided Design and Applications* Band 2 (2005) 1-4, S. 155–164.

HUNTER 2007

Hunter, J.: Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering* Band 9 (2007) 3, S. 90–95.

### IFW 2020

IfW, Hrsg. (2020): Analyse der industrierelevanten wirtschaftlichen Rahmenbedingungen in Deutschland im internationalen Vergleich: Endbericht an das Bundesministerium für Wirtschaft und Energie, Referat I C 4. Kiel. URL: [https://www.ifw-kiel.de/fileadmin/Dateiverwaltung/IfW-Publications/Dirk\\_Christian\\_Dohse/Analyse\\_der\\_industrie\\_\\_relevanten\\_wirtschaftlichen\\_Rahmenbedingungen\\_in\\_Deutschland\\_im\\_internationalen\\_Vergleich/Finale\\_23.06.2020\\_komprimiert.pdf](https://www.ifw-kiel.de/fileadmin/Dateiverwaltung/IfW-Publications/Dirk_Christian_Dohse/Analyse_der_industrie__relevanten_wirtschaftlichen_Rahmenbedingungen_in_Deutschland_im_internationalen_Vergleich/Finale_23.06.2020_komprimiert.pdf) (besucht am 12. 12. 2021).

### IPA 2021

IPA, Hrsg. (2021): Griff-in-die-Kiste-Software BP3: Roboterlösung für die Bauteilzuführung. Stuttgart. URL: [https://www.ipa.fraunhofer.de/content/dam/ipa/de/documents/Kompetenzen/Roboter--und-Assistenzsysteme/Produktblatt\\_Griff-in-die-Kiste.pdf](https://www.ipa.fraunhofer.de/content/dam/ipa/de/documents/Kompetenzen/Roboter--und-Assistenzsysteme/Produktblatt_Griff-in-die-Kiste.pdf) (besucht am 07. 07. 2021).

### JAKSIC & MAUL 2001

Jaksic, N.; Maul, G.: Development of a model for part reorientation in vibratory bowl feeders with active air jet tooling. *Robotics and Computer-Integrated Manufacturing* Band 17 (2001) 1-2, S. 145–149.

### JAUMARD ET AL. 1990

Jaumard, B.; Lu, S.-H.; Sriskandarajah, C.: Design parameters selection and ordering of part-orienting devices. *International Journal of Production Research* Band 28 (1990) 3, S. 459–476.

### JAUMARD ET AL. 1993

Jaumard, B.; Lou, S.; Lu, S.-H.; Sethi, S.; Sriskandarajah, C.: Heuristics for the design of part-orienting systems. *International Journal of Flexible Manufacturing Systems* Band 5 (1993) 3, S. 167–185.

### JAYANTI ET AL. 2006

Jayanti, S.; Kalyanaraman, Y.; Iyer, N.; Ramani, K.: Developing an engineering shape benchmark for CAD models. *Computer-Aided Design* Band 38 (2006) 9, S. 939–953.



M. JIANG ET AL. 2003

Jiang, M.; Chua, P.; Tan, F.: Simulation software for parts feeding in a vibratory bowl feeder. *International Journal of Production Research* Band 41 (2003) 9, S. 2037–2055.

KAPARTHI & N. SURESH 1991

Kaparthi, S.; Suresh, N.: A neural network system for shape-based classification and coding of rotational parts. *International Journal of Production Research* Band 29 (1991) 9, S. 1771–1784.

KARST 1992

Karst, M.: *Methodische Entwicklung von Expertensystemen*. Diss. Universität Saarbrücken (1991). DUV Wirtschaftswissenschaft. Wiesbaden: Deutscher Universitätsverlag. 1992. ISBN: 978-3-663-14584-4.

KAWACHI ET AL. 1997

Kawachi, K.; Suzuki, H.; Kimura, F.: Simulation of rigid body motion with impulsive friction force. In: *Proceedings of the 1997 IEEE International Symposium on Assembly and Task Planning (ISATP'97)*. Hrsg. von Lee, S. Band SESSION FM3: MOTION PLANNING II. Piscataway: IEEE. 1997, S. 182–187. ISBN: 0-7803-3820-0.

KAWACHI ET AL. 1998

Kawachi, K.; Suzuki, H.; Kimura, F.: Technical issues on simulating impulse and friction in three dimensional rigid body dynamics. In: *Proceedings Computer Animation '98*. Piscataway: IEEE Service Center. 1998, S. 170–176. ISBN: 0-8186-8541-7.

KINGMA & BA 2014

Kingma, D.; Ba, J.: Adam: A Method for Stochastic Optimization. URL: <http://arxiv.org/pdf/1412.6980v9> (besucht am 06.05.2020).

KLEMP & POTTEBAUM 2015

Klemp, E.; Pottebaum, J.: Additive Fertigungsverfahren im Kontext von Industrie 4.0. In: *Handbuch Industrie 4.0*. Hrsg. von Vogel-Heuser, B.; Bauernhansl, T.; Hompel, M. ten. Berlin: Springer. 2015, S. 1–21. ISBN: 978-3-662-45537-1.

## Literatur

---

KOLDITZ 2020

Kolditz, T.: Orientieren mit Luft: Aerodynamische Zuführtechnik. Hrsg. von match/IPA. Hannover. URL: <https://phi-hannover.de/orientieren-mit-luft-aerodynamische-zufuehrtechnik/> (besucht am 17.05.2022).

KOLDITZ ET AL. 2022

Kolditz, T.; Müller, P.; Bansmann, D.; Raatz, A.: Flexible Aerodynamic Part Feeding Using High-Speed Image Processing. In: *Production at the Leading Edge of Technology*. Hrsg. von Behrens, B.-A.; Brosius, A.; Drossel, W.-G.; Hintze, W.; Ihlenfeldt, S.; Nyhuis, P. Lecture Notes in Production Engineering. Cham: Springer. 2022, S. 403–411. ISBN: 978-3-030-78423-2.

KOLDITZ & RAATZ 2022

Kolditz, T.; Raatz, A.: Aerodynamic Feeding 4.0: A New Concept for Flexible Part Feeding. *Procedia CIRP* Band 106 (2022), S. 45–50.

KOLDITZ ET AL. 2021

Kolditz, T.; Rochow, N.; Nyhuis, P.; Raatz, A.: Batch time optimization for an aerodynamic feeding system under changing ambient conditions. *Procedia CIRP* Band 97 (2021), S. 278–283.

KONG ET AL. 2015

Kong, X.; Zhang, X.; Li, Q.; Wen, B.: Dynamical analysis of vibratory feeder and feeding parts considering interactions by an improved increment harmonic balance method. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* Band 229 (2015) 6, S. 1029–1040.

KONG ET AL. 2018

Kong, X.; Chen, C.; Wen, B.: Dynamic and stability analysis of the vibratory feeder and parts considering interactions in the hop and the hop-sliding regimes. *Nonlinear Dynamics* Band 93 (2018) 4, S. 2213–2232.

KRAMER 2009

Kramer, O.: *Computational Intelligence: Eine Einführung*. Informatik im Fokus. Berlin: Springer. 2009. ISBN: 978-3-540-79739-5.

KRAUEL 2020

Krauel, M.: *Potentialanalyse für Simulationstools in der Auslegung von Vibrationswendelförderern*. Bachelorarbeit (Nr. 48176). Technische Universität München (2020). München. 2020.

KRISHNASAMY 1996

Krishnasamy, J.: *Mechanics of Entrapment with Applications to Design of Industrial Parts Feeders*. Diss. Cambridge: Massachusetts Institute of Technology. 1996. URL: <http://hdl.handle.net/1721.1/10558> (besucht am 04. 09. 2020).

KRISHNASAMY ET AL. 1996

Krishnasamy, J.; Jakiela, M. J.; Whitney, D. E.: *Mechanics of vibration-assisted entrapment with application to design*. In: *Proceedings of the 1996 IEEE International Conference on Robotics and Automation (ICRA)*. Piscataway: IEEE. 1996, S. 838–845. ISBN: 0-7803-2988-0.

KRUSE ET AL. 2015

Kruse, R.; Borgelt, C.; Braune, C.; Klawonn, F.; Moewes, C.; Steinbrecher, M.: *Computational Intelligence: Eine methodische Einführung in künstliche neuronale Netze, evolutionäre Algorithmen, Fuzzy-Systeme und Bayes-Netze*. 2., überarbeitete und erweiterte Auflage. Computational Intelligence. Wiesbaden: Springer Vieweg. 2015. ISBN: 978-3-658-10904-2.

KUMAR 2020

Kumar, S.: *Balancing a CartPole System with Reinforcement Learning – A Tutorial*. URL: <https://arxiv.org/pdf/2006.04938> (besucht am 23. 04. 2021).

LA BROOY & C. JIANG 2009

La Brooy, R.; Jiang, C.: *Expert System for Vibratory Bowl Feeder Tooling*. *New Engineer Journal* Band 12 (2009) 2, S. 13–17.

LACOUR 2012

Lacour, F.-F.: *Modellbildung für die physikbasierte virtuelle Inbetriebnahme materialflussintensiver Produktionsanlagen*. Diss. Technische Universität München (2011). Band 257. Forschungsberichte IWB. München: Utz. 2012. ISBN: 978-3-8316-4162-8.

## Literatur

---

LAPAN 2020

Lapan, M.: *Deep reinforcement learning hands-on: Apply modern RL methods to practical problems of chatbots, robotics, discrete optimization, web automation, and more.* 2. Aufl. Birmingham: Packt. 2020. ISBN: 978-1-83882-004-6.

LE ET AL. 2018

Le, G.; Nguyen, V.-M.; Dang, A.; Nguyen, V.-D.: A Method to Design Vibratory Bowl Feeder by Using Numerical Simulation Analysis. *International Journal of Scientific Engineering and Science* Band 2 (2018) 9, S. 26–30.

LEDERMANN 2012

Ledermann, T.: *Partikel-Schwarm-Optimierung zur Objektlageerkennung in Tiefendaten.* Diss. Universität Stuttgart (2012). Band 523. IPA-IAO Forschung und Praxis. Heimsheim: Jost-Jetter. 2012. ISBN: 978-3-939890-97-3.

LEE & FISCHER 1999

Lee, S.; Fischer, G.: Grouping parts based on geometrical shapes and manufacturing attributes using a neural network. *Journal of Intelligent Manufacturing* Band 10 (1999) 2, S. 199–209.

LEE ET AL. 1996

Lee, S.; Ngoi, B.; Lye, S.; Lim, L.: An analysis of the resting probabilities of an object with curved surfaces. *The International Journal of Advanced Manufacturing Technology* Band 12 (1996) 5, S. 366–369.

LEIBER & REINHART 2020

Leiber, D.; Reinhart, G.: A bi-level optimisation approach for assembly line design using a nested genetic algorithm. *International Journal of Production Research* (2020), S. 1–16.

LEVY & ROSARIO 2001

Levy, J.; Rosario, L.: Predicting The Natural Resting Aspects Probabilities For A Family Of Connectors: A Comparison Of Theoretical And Experimental Results. In: *Proceedings of the Computing Research Conference 2001 (CRC 2001)*. 2001.

LIM ET AL. 1994

Lim, L.; Ngoi, B.; Lee, S.; Lye, S.; Tan, P.: A computer-aided framework for the

selection and sequencing of orientating devices for the vibratory bowl feeder. *International Journal of Production Research* Band 32 (1994) 11, S. 2513–2524.

LO & DICK 1990

Lo, E. K.; Dick, M. D.: The solid modelling of part orienting tracks for automated assembly. *International Journal of Production Research* Band 28 (1990) 8, S. 1513–1525.

LOTTER & WIENDAHL 2012

Lotter, B.; Wiendahl, H.-P., Hrsg. (2012): *Montage in der industriellen Produktion: Ein Handbuch für die Praxis*. 2. Aufl. VDI-Buch. Berlin: Springer. 2012. ISBN: 978-3-642-29061-9.

LOY 2010

Loy, M.: *Modulare Vibrationswendelförderer zur flexiblen Teilezuführung*. Diss. Technische Universität München (2010). Band 240. Forschungsberichte IWB. München: Utz. 2010. ISBN: 978-3-8316-4027-0.

LUIDENS & CIPRA 2008

Luidens, M.; Cipra, R.: Simulated and Experimental Capture Times of Micro-Scale Pins Collected From a Vibratory Feeder. In: *Proceedings of the ASME 2007 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference (IDETC-CIE 2007)*. New York: ASME. 2008, S. 485–494. ISBN: 0-7918-4806-X.

MALIK ET AL. 2019

Malik, A.; Andersen, M.; Bilberg, A.: Advances in machine vision for flexible feeding of assembly parts. *Procedia Manufacturing* Band 38 (2019), S. 1228–1235.

MATHIESEN 2017

Mathiesen, S.: *Automatic Design of Vibratory Feeders using Dynamic Simulation*. Diss. Odense: University of Southern Denmark. 2017. URL: <https://portal.findresearcher.sdu.dk/en/publications/automatic-design-of-vibratory-feeders-using-dynamic-simulation> (besucht am 16.04.2021).

MATHIESEN & ELLEKILDE 2017

Mathiesen, S.; Ellekilde, L.-P.: Automatic Selection and Sequencing of Traps for Vibratory Feeders. In: *Proceedings of the 7th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH 2017)*. Setúbal: SCITEPRESS. 2017, S. 145–154. ISBN: 978-989-758-265-3.

MATHIESEN ET AL. 2018

Mathiesen, S.; Sorensen, L.; Kraft, D.; Ellekilde, L.-P.: Optimisation of Trap Design for Vibratory Bowl Feeders. In: *Proceedings 2018 IEEE International Conference on Robotics and Automation (ICRA)*. Hrsg. von Lynch, K. Piscataway: IEEE. 2018, S. 3467–3474. ISBN: 978-1-5386-3081-5.

MATHIESEN & ELLEKILDE 2016

Mathiesen, S.; Ellekilde, L.-P.: Configuration and validation of dynamic simulation for design of vibratory bowl feeders. In: *2016 12th IEEE International Conference on Control and Automation (ICCA)*. Piscataway: IEEE. 2016, S. 485–492. ISBN: 978-1-5090-1738-6.

MATURANA & SCHERER 2015

Maturana, D.; Scherer, S.: VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Piscataway: IEEE. 2015, S. 922–928. ISBN: 978-1-4799-9994-1.

MAUL & THOMAS 1997

Maul, G.; Thomas, M.: A systems model and simulation of the vibratory bowl feeder. *Journal of Manufacturing Systems* Band 16 (1997) 5, S. 309–314.

MAZYAVKINA ET AL. 2020

Mazyavkina, N.; Sviridov, S.; Ivanov, S.; Burnaev, E.: Reinforcement Learning for Combinatorial Optimization: A Survey. URL: <https://arxiv.org/pdf/2003.03600> (besucht am 13.09.2020).

MCKINNEY 2010

McKinney, W.: Data Structures for Statistical Computing in Python. In: *Proceedings of the 9th Python in Science Conference (SciPy 2010)*. Hrsg. von van der Walt,

S.; Millman, J. Proceedings of the Python in Science Conference. SciPy. 2010, S. 56–61.

MEHLHORN & SANDERS 2008

Mehlhorn, K.; Sanders, P.: *Algorithms and data structures: The basic toolbox*. Berlin: Springer. 2008. ISBN: 978-3-540-77977-3.

MICHELUCCI 2019

Michelucci, U.: Fundamentals of Convolutional Neural Networks. In: *Advanced applied deep learning*. Hrsg. von Michelucci, U. New York: Apress. 2019, S. 79–123. ISBN: 978-1-4842-4975-8.

MIN 2004–2019

Min, P.: Bivox: 3D mesh voxelizer. URL: <http://www.patrickmin.com/bivox> (besucht am 05. 05. 2020).

MIRTICH 1996

Mirtich, B.: Impulse-based Dynamic Simulation of Rigid Body Systems. Diss. Berkeley: University of California. 1996. URL: <https://people.eecs.berkeley.edu/~jfc/mirtich/thesis/mirtichThesis.pdf> (besucht am 16. 04. 2021).

MIRTICH & CANNY 1995a

Mirtich, B.; Canny, J.: Impulse-based Dynamic Simulation. In: *WAFR: Proceedings of the workshop on Algorithmic foundations of robotics*. Hrsg. von Goldberg, K.; Halperin, D.; Latombe, J.; Wilson, R. WAFR. San Francisco: A. K. Peters. 1995, S. 407–418. ISBN: 1-56881-045-8.

MIRTICH & CANNY 1995b

Mirtich, B.; Canny, J.: Impulse-based simulation of rigid bodies. In: *Proceedings of the 1995 symposium on Interactive 3D graphics (I3D'95)*. Hrsg. von Zyda, M. New York: ACM. 1995, S. 181–189. ISBN: 978-0-89791-736-0.

MIRTICH ET AL. 1996

Mirtich, B.; Zhuang, Y.; Goldberg, K.; Craig, J.; Zanutta, R.; Carlisle, B.; Canny, J.: Estimating pose statistics for robotic part feeders. In: *Proceedings of the 1996 IEEE*

## Literatur

---

*International Conference on Robotics and Automation (ICRA)*. Band 2. Piscataway: IEEE. 1996, S. 1140–1146. ISBN: 0-7803-2988-0.

MITCHELL 2010

Mitchell, T. M.: *Machine learning*. McGraw-Hill series in computer science. New York: McGraw-Hill. 2010. ISBN: 0-07042-807-7.

MOU 2018

Mou, L.: *Tree-based convolutional neural networks: Principles and applications*. SpringerBriefs in computer science. Singapur: Springer. 2018. ISBN: 978-981-13-1869-6.

MURPHY 2012

Murphy, K. P.: *Machine learning: A probabilistic perspective*. Adaptive computation and machine learning series. Cambridge: MIT Press. 2012. ISBN: 978-0-262-01802-9.

NAIR ET AL. 2020

Nair, A.; Gupta, A.; Dalal, M.; Levine, S.: AWAC: Accelerating Online Reinforcement Learning with Offline Datasets. URL: <https://arxiv.org/pdf/2006.09359> (besucht am 04.05.2021).

NARANG ET AL. 2017

Narang, D.; Raj, A.; Khanna, P.: Mathematical Modelling of a Vibratory Bowl Feeder for spherical Washer. *Journal of Production Engineering* Band 20 (2017) 2, S. 79–84.

NATARAJAN 1989

Natarajan, B.: Some Paradigms for the Automated Design of Parts Feeders. *The International Journal of Robotics Research* Band 8 (1989) 6, S. 98–109.

NGOI & LIM 1996

Ngoi, B.; Lim, L.: Analysing the Natural Resting Aspect of a Component for Automated Assembly using the Energy Envelope Method. *The International Journal of Advanced Manufacturing Technology* Band 12 (1996) 2, S. 132–136.

NGOI ET AL. 1995a

Ngoi, B.; Lim, L.; Lee, S.: Analysing the natural resting aspects of a complex part. *International Journal of Production Research* Band 33 (1995) 11, S. 3163–3172.



NGOI ET AL. 1995b

Ngoi, B.; Lee, S.; Lim, L.: Analysing the probabilities of the natural resting aspects of a component with a displaced centre of gravity. *International Journal of Production Research* Band 33 (1995) 9, S. 2387–2394.

NGOI ET AL. 1995c

Ngoi, B.; Lim, L.; Lee, S.: Analyzing the Natural Resting Aspects of a Component by its centroid solid Angle. *Journal of Electronics Manufacturing* Band 5 (1995) 3, S. 193–197.

NGOI ET AL. 1996

Ngoi, B.; Lye, S.; Chen, J.: Analysing the Natural Resting Aspect of a Prism on a Hard Surface for Automated Assembly. *The International Journal of Advanced Manufacturing Technology* Band 11 (1996) 6, S. 406–412.

NGOI ET AL. 1997

Ngoi, B.; Lim, L.; Ee, J.: Analysis of Natural Resting Aspects of Parts in a Vibratory Bowl Feeder – Validation of Drop Test. *The International Journal of Advanced Manufacturing Technology* Band 13 (1997) 4, S. 300–310.

NGUYEN & LA 2019

Nguyen, H.; La, H.: Review of Deep Reinforcement Learning for Robot Manipulation. In: *Proceedings of the Third IEEE International Conference on Robotic Computing (IRC)*. Hrsg. von Brugali, D.; Sheu, P.; Siciliano, B.; Tsai, J. Los Alamitos: IEEE Computer Society. 2019, S. 590–595. ISBN: 978-1-5386-9245-5.

NIESS 1983

Niess, P.: Die Auslegung von Zuführanlagen in automatisierten Montageeinrichtungen: Vorgehensweise, Kriterien und Beispiele. In: *Praxis der Montageautomatisierung*. Hrsg. von VDI. VDI-Berichte. Düsseldorf: VDI-Verlag. 1983, S. 45–52. ISBN: 3-18090-479-8.

NOORUDDIN & TURK 2003

Nooruddin, F.; Turk, G.: Simplification and Repair of Polygonal Models Using Volumetric Techniques. *IEEE Transactions on Visualization and Computer Graphics* Band 9 (2003) 2, S. 191–205.

## Literatur

---

OPENAI 2019

OpenAI: Gym: A toolkit for developing and comparing reinforcement learning algorithms: Lunar Lander v2. URL: <https://gym.openai.com/envs/LunarLander-v2/> (besucht am 30.03.2021).

PAASS & HECKER 2020

Paaß, G.; Hecker, D.: *Künstliche Intelligenz: Was steckt hinter der Technologie der Zukunft?* 1. Aufl. Wiesbaden: Springer Fachmedien. 2020. ISBN: 978-3-658-30211-5.

PALZKILL 2014

Palzkill, M.: *Heuristisches Suchverfahren zur Objektlageerkennung aus Punktwolken für industrielle Zuführsysteme*. Diss. Universität Stuttgart (2014). Band 37. Stuttgarter Beiträge zur Produktionsforschung. Stuttgart: Fraunhofer-Verlag. 2014. ISBN: 978-3-8396-0784-8.

POPPER 2017

Popper, K.: Die Logik der Sozialwissenschaften: Referat: Erstveröffentlichung: Kölner Zeitschrift für Soziologie und Sozialpsychologie 14 (1962) 2, S. 233–248. *KZfSS Kölner Zeitschrift für Soziologie und Sozialpsychologie* Band 69 (2017) Suppl 1, S. 215–228.

PRINCETON UNIVERSITY 2020

Princeton University: Princeton ModelNet: 127,915 CAD Models; 662 Object Categories; 10 Categories with Annotated Orientation. Princeton. URL: <https://modelnet.cs.princeton.edu/> (besucht am 07.05.2020).

PROFF ET AL. 2016

Proff, H.; Sandau, J.; Gönninger, F.; Bittrich, C.; Roscher, O.; Hinzen, A.: *Manufacturing 4.0: Meilenstein, Must-Have oder Millionengrab? Warum bei Manufacturing 4.0 die Integration den entscheidenden Unterschied macht*. Hrsg. von Deloitte GmbH Wirtschaftsprüfungsgesellschaft. URL: [https://www2.deloitte.com/content/dam/Deloitte/de/Documents/operations/DELO-2267\\_Manufacturing-4.0-Studie\\_s.pdf](https://www2.deloitte.com/content/dam/Deloitte/de/Documents/operations/DELO-2267_Manufacturing-4.0-Studie_s.pdf) (besucht am 24.09.2020).

PURDY & DAUGHERTY 2017

Purdy, M.; Daugherty, P.: How AI Boosts Industry Profits and Innovation. Hrsg. von Accenture. URL: [https://www.accenture.com/fr-fr/\\_acnmedia/36dc7f76eab444cab6a7f44017cc3997.pdf](https://www.accenture.com/fr-fr/_acnmedia/36dc7f76eab444cab6a7f44017cc3997.pdf) (besucht am 24. 09. 2020).

PYTHON SOFTWARE FOUNDATION 2021

Python Software Foundation: Welcome to Python.org. URL: <https://www.python.org/> (besucht am 19. 03. 2021).

QIN ET AL. 2014

Qin, F.; Li, L.; Gao, S.; Yang, X.; Chen, X.: A deep learning approach to the classification of 3D CAD models. *Journal of Zhejiang University SCIENCE C* Band 15 (2014) 2, S. 91–106.

RABE ET AL. 2008

Rabe, M.; Spieckermann, S.; Wenzel, S.: *Verifikation und Validierung für die Simulation in Produktion und Logistik: Vorgehensmodelle und Techniken*. Berlin: Springer. 2008. ISBN: 978-3-540-35282-2.

RÅDE ET AL. 2000

Råde, L.; Westergren, B.; Vachenauer, P.: *Springers Mathematische Formeln: Taschenbuch für Ingenieure, Naturwissenschaftler, Informatiker, Wirtschaftswissenschaftler*. 3., durchgesehene Aufl. Berlin & Heidelberg: Springer. 2000. ISBN: 3-54067-505-1.

RAMALINGAM & SAMUEL 2009

Ramalingam, M.; Samuel, G.: Investigation on the conveying velocity of a linear vibratory feeder while handling bulk-sized small parts. *The International Journal of Advanced Manufacturing Technology* Band 44 (2009) 3-4, S. 372–382.

REBACK ET AL. 2021

Reback, J.; McKinney, W.; Jbrockmendel; Van den Bossche, J.; Augspurger, T.; Cloud, P.; Gfyoung; Hawkins, S.; Sinhrks; Roeschke, M.; Klein, A.; Terji Petersen; Tratner, J.; She, C.; Ayd, W.; Naveh, S.; Garcia, M.; Patrick; Schendel, J.; Hayden, A.; Saxton, D.; Jancauskas, V.; Shadrach, R.; Gorelli, M.; McMaster, A.; Battiston, P.; Seabold, S.; Dong, K.; Chris-B1; H-Vetinari: pandas-dev/pandas: Pandas 1.2.4. URL: <https://github.com/pandas-dev/pandas/tree/v1.2.4> (besucht am 06. 04. 2021).

## Literatur

---

REBALA ET AL. 2019

Rebala, G.; Ravi, A.; Churiwala, S.: *An Introduction to Machine Learning*. Cham: Springer Nature Switzerland. 2019. ISBN: 978-3-030-15729-6.

REINHART & HOFMANN 2012

Reinhart, G.; Hofmann, D.: Physiksimulation in der Zuführtechnik: Effiziente Bestimmung von Leistungsmerkmalen für Vibrationswendelförderer. *wt Werkstattstechnik online* Band 102 (2012) 6, S. 435–439.

REZNIK & CANNY 1997

Reznik, D.; Canny, J.: Design Verification and Optimization of Novel Parts Feeders with the Impulse Dynamic Simulator. In: *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Hrsg. von Laugier, C. Band 1. Piscataway: IEEE Service Center. 1997, S. 421–431. ISBN: 0-7803-4119-8.

RIMAI & CIPRA 2011

Rimai, B.; Cipra, R.: On the Spatial Modeling of a Vibratory Micro-Pin Feeder Using Rigid-Body Dynamics. In: *Proceedings of the ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC-CIE 2011)*. Hrsg. von ASME. Band 4, Teil A. New York: ASME. 2011, S. 247–255. ISBN: 978-0-7918-5481-5.

RISCH 2011

Risch, T.: Zweidimensionale Bewegungsformen in der Vibrationsfördertechnik. Diss. Technische Universität Chemnitz. 2011. URL: <https://nbn-resolving.org/urn:nbn:de:bsz:ch1-qucosa-65499> (besucht am 26. 09. 2022).

ROCKLAND & STETTER 1994

Rockland, M.; Stetter, R.: Flexibles Ordnen und Zuführen: Einbinden physikalischer Effekte in die 3D-Bewegungssimulation. *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb* Band 89 (1994) 1-2, S. 55–57.

ROCKLAND 1995

Rockland, M.: *Flexibilisierung der automatischen Teilebereitstellung in Montageanlagen*. Diss. Technische Universität München (1995). Band 87. Forschungsberichte IWB. Berlin: Springer. 1995. ISBN: 978-3-540-58999-0.

RUSSELL & NORVIG 2012

Russell, S. J.; Norvig, P.: *Künstliche Intelligenz: Ein moderner Ansatz*. 3., aktualisierte Aufl. Always learning. München: Pearson. 2012. ISBN: 978-3-86326-504-5.

SAMARASINGHE 2007

Samarasinghe, S.: *Neural networks for applied sciences and engineering: From fundamentals to complex pattern recognition*. Boca Raton: Auerbach. 2007. ISBN: 978-0-849-33375-0.

SCHENK 2015

Schenk, M., Hrsg. (2015): *Produktion und Logistik mit Zukunft: Digital engineering and operation*. Berlin: Springer Vieweg. 2015. ISBN: 978-3-662-48266-7.

SCHMID 2006

Schmid, S.: *Automatisierte Ordnungs- und Kommissionierzelle zur hochflexiblen Bereitstellung von Werkstücken in der Montage*. Diss. Universität Stuttgart (2006). Band 435. IPA-IAO Forschung und Praxis. Heimsheim: Jost-Jetter. 2006. ISBN: 3-936947-84-8.

SCHOLZ 2018

Scholz, D.: Heuristiken. In: *Optimierung interaktiv*. Hrsg. von Scholz, D. Lehrbuch. Berlin: Springer Spektrum. 2018, S. 217–230. ISBN: 978-3-662-57952-7.

SCHUH 2012

Schuh, G.: *Innovationsmanagement: Handbuch Produktion und Management* 3. 2., vollständig neu bearbeitete und erweiterte Aufl. VDI-Buch. Berlin: Springer Vieweg. 2012. ISBN: 978-3-642-25050-7.

SCHUH ET AL. 2019

Schuh, G.; Reinhart, G.; Prote, J.-P.; Sauer mann, F.; Horsthofer, J.; Oppolzer, F.; Knoll, D.: Data Mining Definitions and Applications for the Management of Production Complexity. *Procedia CIRP* Band 81 (2019), S. 874–879.

SETHI ET AL. 1990

Sethi, S.; Sriskandarajah, C.; Tayi, G.; Rao, M.: Heuristic Methods for Selection and Ordering of Part-Orienting Devices. *Operations Research* Band 38 (1990) 1, S. 84–98.

## Literatur

---

SHI ET AL. 2014

Shi, D.; Civet, Y.; Perriard, Y.: Modelling and Design of Complex Geometry Parts Vibratory Conveying. In: *2014 17th International Conference on Electrical Machines and Systems (ICEMS 2014)*. Hrsg. von IEEE. Piscataway: IEEE. 2014, S. 2170–2175. ISBN: 978-1-4799-5162-8.

SIEBERTZ ET AL. 2017

Siebertz, K.; van Bebber, D.; Hochkirchen, T.: *Statistische Versuchsplanung: Design of Experiments (DoE)*. 2. Aufl. VDI-Buch. Berlin: Springer Vieweg. 2017. ISBN: 978-3-662-55743-3.

SOPRA STERIA CONSULTING 2017

Sopra Steria Consulting: Potenzialanalyse Künstliche Intelligenz: 2017. Hrsg. von Sopra Steria Consulting. Hamburg. URL: [https://www.soprasteria.de/docs/librariesprovider2/sopra-steria-de/publikationen/studien/potenzialanalyse-kuenstliche-intelligenz-2017.pdf?sfvrsn=190f45dc\\_4](https://www.soprasteria.de/docs/librariesprovider2/sopra-steria-de/publikationen/studien/potenzialanalyse-kuenstliche-intelligenz-2017.pdf?sfvrsn=190f45dc_4) (besucht am 24.09.2020).

SRIVASTAVA ET AL. 2014

Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R.: Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research* Band 15 (2014) 56, S. 1929–1958. URL: <https://jmlr.org/papers/v15/srivastava14a.html> (besucht am 12.12.2021).

STABLE BASELINES 2021

Stable Baselines, Hrsg. (2021): SAC – Stable Baselines 2.10.2a1 documentation. URL: <https://stable-baselines.readthedocs.io/en/master/modules/sac.html> (besucht am 23.03.2021).

STMWi 2020

StMWi, Hrsg. (2020): Industriebericht Bayern 2020. München. URL: [https://www.stmwi.bayern.de/fileadmin/user\\_upload/stmwi/Publikationen/2020/2020-08-31\\_Industriebericht\\_Bayern\\_2020.pdf](https://www.stmwi.bayern.de/fileadmin/user_upload/stmwi/Publikationen/2020/2020-08-31_Industriebericht_Bayern_2020.pdf) (besucht am 10.01.2021).

STOCKER & REINHART 2015

Stocker, C.; Reinhart, G.: Sensitivitätsanalyse des Fördergutverhaltens: Phy-

siksimulationsbasierte Analyse des Förderguts bezüglich der Topologie im Vibrationswendelförderer. *wt Werkstattstechnik online* Band 105 (2015) 9, S. 622–626.

STOCKER & REINHART 2016

Stocker, C.; Reinhart, G.: Sensitivity Analysis of the Dynamic Behavior of Transported Material in Vibratory Bowl Feeders Using Physics Simulation. *Applied Mechanics and Materials* Band 840 (2016), S. 107–113.

STOCKER ET AL. 2017

Stocker, C.; Hell, M.; Reisch, R.; Reinhart, G.: Automated generation of orienting devices for vibratory bowl feeders. In: *2017 IEEE International Conference on Industrial Engineering & Engineering Management (IEEM)*. Piscataway: IEEE. 2017, S. 1586–1590. ISBN: 978-1-5386-0948-4.

STOCKER ET AL. 2019

Stocker, C.; Schmid, M.; Reinhart, G.: Reinforcement learning–based design of orienting devices for vibratory bowl feeders. *The International Journal of Advanced Manufacturing Technology* Band 105 (2019) 9, S. 3631–3642.

STOTZ 2011

Stotz, M.: *Adaptive Segmentierung von Tiefenbildern für die 3-D-Objektlageerkennung auf Basis von kombinierten regelgeometrischen Elementen*. Diss. Universität Stuttgart (2011). Band 506. IPA-IAO Forschung und Praxis. Heimsheim: Jost-Jetter. 2011. ISBN: 978-3-939890-73-7.

STRATASYS 2020

Stratasys: GrabCAD: The largest online community of professional designers, engineers, manufacturers, and students. Hrsg. von Stratasys Inc. URL: <https://grabcad.com/> (besucht am 05.05.2020).

SU ET AL. 2015

Su, H.; Maji, S.; Kalogerakis, E.; Learned-Miller, E.: Multi-view Convolutional Neural Networks for 3D Shape Recognition. URL: <http://arxiv.org/pdf/1505.00880v3> (besucht am 10.04.2020).

M. SURESH ET AL. 2013

Suresh, M.; Jagadeesh, K.; Varthanan, P.: Determining the natural resting orientation of a part using drop test and theoretical methods. *Journal of Manufacturing Systems* Band 32 (2013) 1, S. 220–227.

M. SURESH ET AL. 2018

Suresh, M.; Narasimharaj, V.; Arul Navalan, G.; Chandra Bose, V.: Effect of orientations of an irregular part in vibratory part feeders. *The International Journal of Advanced Manufacturing Technology* Band 94 (2018) 5-8, S. 2689–2702.

SUTTON 1996

Sutton, R.: Generalization in Reinforcement Learning: Successful Examples Using Sparse Coarse Coding. In: *Advances in Neural Information Processing Systems 8*. Hrsg. von Touretzky, D.; Mozer, M.; Hasselmo, M. Cambridge: MIT Press. 1996, S. 1038–1044. ISBN: 978-0-262-20107-0.

SUTTON & BARTO 2018

Sutton, R.; Barto, A.: *Reinforcement learning: An introduction*. 2. Aufl. Adaptive Computation and Machine Learning. Cambridge: MIT Press. 2018. ISBN: 978-0-262-19398-6.

P. TAN ET AL. 1995

Tan, P.; Ngoi, B.; Lee, S.; Lim, L.: A knowledge-based Advisor for the Automatic Selection and Sequencing of Orienting Devices for Vibratory Feeding. *Engineering Applications of Artificial Intelligence* Band 8 (1995) 1, S. 1–13.

TAY ET AL. 2005

Tay, M.; Chua, P.; Sim, S.; Gao, Y.: Development of a flexible and programmable parts feeding system. *International Journal of Production Economics* Band 98 (2005) 2, S. 227–237.

TESCHEMACHER 2019

Teschemacher, U.: Dynamische Routenzugoptimierung bei kurzfristigen Materialabrufen. Diss. Technische Universität München. 2019. URL: <https://mediatum.ub.tum.de/node?id=1473618> (besucht am 16.04.2021).



UDHAYAKUMAR ET AL. 2013

Udhayakumar, S.; Mohanram, P.; Keerthi Anand, P.; Srinivasan, R.: Determining the most probable natural resting orientation of sector shaped parts. *Assembly Automation* Band 33 (2013) 1, S. 29–37.

H. ULRICH 1998

Ulrich, H.: Praxisbezug und wissenschaftliche Fundierung einer transdisziplinären Managementlehre. In: *Universität und Praxis*. Hrsg. von Spoun, S. Zürich: Verlag Neue Zürcher Zeitung. 1998, S. 159–168. ISBN: 3-85823-716-7.

P. ULRICH & W. HILL 1976

Ulrich, P.; Hill, W.: Wissenschaftstheoretische Grundlagen der Betriebswirtschaftslehre. *Wirtschaftswissenschaftliches Studium: Zeitschrift für Ausbildung und Hochschulkontakt* Band 5 (1976) 7+8.

VAN RANDEN ET AL. 2016a

van Randen, J.; Bercker, C.; Fieml, J.: Daten in einem Klassendiagramm abbilden. In: *Einführung in UML*. Hrsg. von van Randen, J.; Bercker, C.; Fieml, J. Wiesbaden: Springer Vieweg. 2016, S. 5–24. ISBN: 978-3-658-14411-1.

VAN RANDEN ET AL. 2016b

van Randen, J.; Bercker, C.; Fieml, J., Hrsg. (2016b): *Einführung in UML: Analyse und Entwurf von Software*. Wiesbaden: Springer Vieweg. 2016. ISBN: 978-3-658-14411-1.

VÁRKONYI 2014

Várkonyi, P.: Estimating Part Pose Statistics With Application to Industrial Parts Feeding and Shape Design: New Metrics, Algorithms, Simulation Experiments and Datasets. *IEEE Transactions on Automation Science and Engineering* Band 11 (2014) 3, S. 658–667.

VÁRKONYI 2015

Várkonyi, P.: The Secret of Gambling with Irregular Dice: Estimating the Face Statistics of Polyhedra. *Periodica Polytechnica Architecture* Band 46 (2015) 1, S. 1–5.

VASILEV ET AL. 2019

Vasilev, I.; Slater, D.; Spacagna, G.; Roelants, P.; Zocca, V.: *Python deep learning:*

## Literatur

---

*Exploring deep learning techniques and neural network architectures with PyTorch, Keras and TensorFlow*. 2. Aufl. Birmingham: Packt. 2019. ISBN: 978-1-789-34846-0.

VCI 2021

VCI, Hrsg. (2021): Industrieland Deutschland. URL: <https://www.vci.de/ergaenzende-downloads/industrieland-deutschland-daten-fakten-bedeutung-deutsche-industrie.pdf> (besucht am 12. 12. 2021).

VDI 1988

VDI: VDI 2861: Montage- und Handhabungstechnik: Kenngrößen für Industrieroboter; Achsbezeichnungen. Düsseldorf: VDI-Verlag 1988.

VDI 1990

VDI: VDI 2860: Montage- und Handhabungstechnik: Handhabungsfunktionen, Handhabungseinrichtungen; Begriffe, Definitionen, Symbole. Berlin: Beuth 1990.

VDI 2016

VDI: VDI 2860 - Montage- und Handhabungstechnik; Handhabungsfunktionen, Handhabungseinrichtungen; Begriffe, Definitionen, Symbole. Hrsg. von VDI-Gesellschaft Produktion und Logistik. URL: <https://www.vdi.de/richtlinien/details/vdi-2860-montage-und-handhabungstechnik-handhabungsfunktionen-handhabungseinrichtungen-begriffe-definitionen-symbole> (besucht am 21. 11. 2021).

VDI 2018

VDI: VDI 3633: Simulation von Logistik-, Materialfluss- und Produktionssystemen: Begriffe. Berlin: Beuth 2018.

VDI 2019

VDI: VDI 2860 - Projekt: Handhabungsfunktionen, Handhabungseinrichtungen; Begriffe, Definitionen, Symbol. Hrsg. von VDI-Gesellschaft Produktion und Logistik. URL: <https://www.vdi.de/richtlinien/details/vdi-2860-handhabungsfunktionen-handhabungseinrichtungen-begriffe-definitionen-symbol> (besucht am 16. 04. 2021).

## VDMA 2018

VDMA: Quick Guide Machine Learning im Maschinen- und Anlagenbau. Hrsg. von Claus Oetter. Frankfurt am Main. URL: [https://sud.vdma.org/documents/15012668/26742261/!VDMA%20Quick%20Guide\\_Machin e%20Learning\\_1554727571775.pdf/824248b7-7454-0357-84b4-50102ffe7491](https://sud.vdma.org/documents/15012668/26742261/!VDMA%20Quick%20Guide_Machin e%20Learning_1554727571775.pdf/824248b7-7454-0357-84b4-50102ffe7491) (besucht am 03.06.2020).

## VIRTANEN ET AL. 2020

Virtanen, P.; Gommers, R.; Oliphant, T.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; van der Walt, S.; Brett, M.; Wilson, J.; Millman, K.; Mayorov, N.; Nelson, A.; Jones, E.; Kern, R.; Larson, E.; Carey, C.; Polat, I.; Feng, Y.; Moore, E.; VanderPlas, J.; Laxalde, D.; Perktold, J.; Cimrman, R.; Henriksen, I.; Quintero, E.; Harris, C.; Archibald, A.; Ribeiro, A.; Pedregosa, F.; van Mulbregt, P.: SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods* Band 17 (2020) 3, S. 261–272.

## VRANJEŠ ET AL. 1997

Vranješ, B.; Vrtodušić, D.; Jerbić, B.: CAD integriertes System zur Auslegung der Ordnungseinrichtungen. *Österreichische Ingenieur- und Architekten-Zeitschrift (ÖIAZ)* Band 142 (1997) 11-12, S. 815–822.

## VVG-BEFESTIGUNGSTECHNIK 2020

VVG-Befestigungstechnik: Produkte. Hrsg. von VVG-Befestigungstechnik GmbH & Co. URL: <https://www.vvg.info/en/products/> (besucht am 05.05.2020).

## WALPOLE ET AL. 2017

Walpole, R.; Myers, R.; Myers, S.; Ye, K.: *Probability & Statistics for Engineers & Scientists*. 9. Aufl. Boston: Pearson. 2017. ISBN: 978-0-321-62911-1.

## WANG &amp; LIU 1993

Wang, J.; Liu, S.: Hopfield Neural Network-based Automatic Recognition for 3-D Features. In: *Proceedings of 1993 International Conference on Neural Networks (IJCNN)*. Hrsg. von IEEE. Piscataway: IEEE. 1993, S. 2121–2124. ISBN: 0-7803-1421-2.

## Literatur

---

WATT 2002

Watt, A.: *3D-Computergrafik*. 3. Aufl. i-informatik Grafik. München: Pearson Studium. 2002. ISBN: 3-82737-014-0.

WEIR & CIPRA 2005

Weir, N.; Cipra, R.: Modeling and analysis of a vibratory micro-pin feeder using impulse-based simulation. In: *Proceedings of the 2005 ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conferences (DETC2005)*. Hrsg. von Chen, W.; Chang Gea, H. Band 6. New York: ASME. 2005, S. 533–543.

WEISS 1983

Weiss, K.: *Entwicklung flexibler Ordnungssysteme für die Automatisierung der Werkstückhandhabung in der Klein- und Mittelserienfertigung*. Diss. Universität Stuttgart (1983). Band 67. IPA-IAO Forschung und Praxis. Berlin: Springer. 1983. ISBN: 3-540-12455-1.

WIEGLEY ET AL. 1992

Wiegley, J.; Rao, A.; Goldberg, K.: Computing a Statistical Distribution of Stable Poses for a Polyhedron. In: *Proceedings of the 30th Annual Allerton Conference on Communications, Control and Computing*. Urbana-Champaign: University of Illinois. 1992.

WIEGLEY ET AL. 1996

Wiegley, J.; Goldberg, K.; Peshkin, M.; Brokowski, M.: A Complete Algorithm for Designing Passive Fences to Orient Parts. In: *Proceedings of the 1996 IEEE International Conference on Robotics and Automation (ICRA)*. Piscataway: IEEE. 1996, S. 1133–1139. ISBN: 0-7803-2988-0.

WOERNLE 2008

Woernle, M.: *Anwendbarkeit künstlicher neuronaler Netze zur Untergrundbewertung in der oberflächennahen Geothermie*. Diss. Universität Fridericiana zu Karlsruhe. 2008. URL: <https://publikationen.bibliothek.kit.edu/1000009235/431188> (besucht am 12. 12. 2021).

WOLFSTEINER 1999

Wolfsteiner, P.: *Dynamik von Vibrationsförderern*. Diss. Technische Universität

München (1999). Band 511. Fortschritt-Berichte VDI/Reihe 2, Fertigungstechnik. Düsseldorf: VDI-Verlag. 1999. ISBN: 3-18-351102-9.

M. WU & JEN 1996

Wu, M.; Jen, S.: A Neural Network Approach to the Classification of 3D Prismatic Parts. *The International Journal of Advanced Manufacturing Technology* Band 11 (1996) 5, S. 325–335.

Z. WU ET AL. 2015

Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; Xiao, J.: 3D ShapeNets: A Deep Representation for Volumetric Shapes. URL: <http://arxiv.org/pdf/1406.5670v3> (besucht am 10.04.2020).

XU & TODOROVIC 2016

Xu, X.; Todorovic, S.: Beam Search for Learning a Deep Convolutional Neural Network of 3D Shapes. URL: <http://arxiv.org/pdf/1612.04774v1> (besucht am 10.04.2020).

OU-YANG & MAUL 1993

Ou-Yang, C.; Maul, G.: A computer analysis of orientation devices for vibratory bowl feeders. *International Journal of Production Research* Band 31 (1993) 3, S. 555–578.

YANG ET AL. 2012

Yang, Y.; Huang, Y.; Wu, H.: Research of Discrete Element Method in Vibratory Feeder. In: *Software Engineering and Knowledge Engineering: Theory and Practice*. Hrsg. von Zhang, W. Band 2. Advances in Intelligent and Soft Computing. Berlin: Springer. 2012, S. 699–704. ISBN: 978-3-642-29455-6.

YEONG & DE VRIES 1994

Yeong, M.; De Vries, W.: A Methodology for Part Feeder Design. *CIRP Annals* Band 43 (1994) 1, S. 19–22.

YERALAN & CHANG 1986

Yeralan, S.; Chang, C.: Models of Part-Orienting Systems. *Decision Sciences* Band 17 (1986) 3, S. 357–375.

## Literatur

---

ZHOU ET AL. 2017

Zhou, Z.; Li, X.; Zare, R.: Optimizing Chemical Reactions with Deep Reinforcement Learning. *ACS central science* Band 3 (2017) 12, S. 1337–1344.

## Betreute Studienarbeiten

Im Rahmen dieser Forschungsarbeit sind unter der wissenschaftlichen, fachlichen und inhaltlichen Anleitung der Autorin folgende studentische Arbeiten am Institut für Werkzeugmaschinen und Betriebswissenschaften (*iwb*) entstanden. Deren Ergebnisse zu verschiedenen Problemstellungen auf dem Gebiet der Auslegung und Simulation von VWFs sind zu Teilen in die vorliegende Dissertation eingeflossen. Die Autorin bedankt sich bei allen Studierenden für ihr Engagement bei der Unterstützung dieser wissenschaftlichen Arbeit.

STIRNER 2015

Stirner, Florian: Systematisierung des Entwicklungsprozesses von Vibrationswendelförderern und Entwicklung eines neuartigen Strukturaufbaus. Master-Semesterarbeit (Nr. 22869). Bearbeitungszeitraum: 01.10.2014–01.04.2015.

GERNBECK 2015

Gernbeck, Florian: Auswahl und Konzeption von Montagevorrichtungen und Zuführsystemen für die Batteriemontage. Masterarbeit (Nr. 23458). Bearbeitungszeitraum: 15.04.2015–15.10.2015.

SCHNEIDER 2015

Schneider, Lukas: Konzeption und Aufbau eines Demonstrators für den Versuchsstand Zuführtechnik. Bachelorarbeit (Nr. 23460). Bearbeitungszeitraum: 01.05.2015–30.10.2015.

MATTES 2015

Mattes, Carla: Sensitivitätsanalyse der Fördergutdynamik durch Simulationsstudien. Masterarbeit (Nr. 23459). Bearbeitungszeitraum: 04.05.2015–02.12.2015.

## Betreute Studienarbeiten

---

### THUMM 2015

Thumm, Michael: Methode zur intelligenten Manipulation von vernetzten CAD-Daten. Masterarbeit (Nr. 24113). Bearbeitungszeitraum: 01.06.2015–30.11.2015.

### BEYERLE 2015

Beyerle, Maximilian: Konzeptentwicklung zur Integration der Mensch-Roboter-Interaktion in einen Montageprozess. Master-Semesterarbeit (Nr. 27256). Bearbeitungszeitraum: 01.07.2015–31.12.2015.

### WINTER 2016

Winter, Sebastian: Entwicklung und prototypische Umsetzung einer modularen Ordnungsschikane für einen Vibrationswendelförderer. Bachelorarbeit (Nr. 31461). Bearbeitungszeitraum: 01.05.2016–31.10.2016.

### KLINGSHIRN 2016

Klingshirn, Maximilian: Risikoanalyse und Vorbereitung einer Roboterzelle zur flexiblen Montage von Batteriemodulen mit dem Fokus auf Mensch-Roboter-Kooperation. Master-Semesterarbeit (Nr. 32446). Bearbeitungszeitraum: 01.04.2016–30.09.2016.

### HERKOMMER 2016

Herkommer, Marion: Physiksimulation in der Zuführtechnik – Implementierung einer variablen Ordnungsschikane für Vibrationswendelförderer. Master-Semesterarbeit (Nr. 32609). Bearbeitungszeitraum: 15.05.2016–15.11.2016.

### DOMME 2017

Domme, Markus: Entwicklung und Erprobung eines Sensorsystems zur Befähigung bandsynchroner Mensch-Roboter-Kooperation. Master-Semesterarbeit (Nr. 32610). Bearbeitungszeitraum: 04.07.2016–04.01.2017.

### VARADY 2017

Varady, Maria: Industrie 4.0: Methodische Bewertung der Wirkung von artfremden Sensor-Technologien auf robuste Montageprozesse am Beispiel der Schraubtechnik. Masterarbeit (Nr. 36815). Bearbeitungszeitraum: 01.10.2016–31.03.2017.



---

HOGREFE 2017

Hogrefe, Georg: Konzeptentwicklung zur automatischen Generierung von Ordnungsschikanen von Vibrationswendelförderern. Master-Semesterarbeit (Nr. 34968). Bearbeitungszeitraum: 01.11.2016–30.04.2017.

OSTERMEIER 2017

Ostermeier, Heinrich Lukas: Manipulation von Schikanendaten für Vibrationswendelförderer auf Basis der Physiksimulation. Bachelorarbeit (Nr. 34837). Bearbeitungszeitraum: 01.11.2016–30.04.2017.

NAZAR 2017

Nazar, Azat: Bewertung von Fördergutorientierungen im Vibrationswendelförderer auf Basis der Physiksimulation. Bachelorarbeit (Nr. 34834). Bearbeitungszeitraum: 01.11.2016–31.05.2017.

HELL 2017

Hell, Melanie: Systematische Ansätze zur automatischen Generierung von Zuführsystemen. Bachelorarbeit (Nr. 34836). Bearbeitungszeitraum: 01.01.2017–30.06.2017.

KLINGSHIRN 2017

Klingshirn, Maximilian: Entwicklung einer Methodik zur Definition von CKD-Bausätzen am Beispiel Rohde&Schwarz. Masterarbeit (Nr. 35106). Bearbeitungszeitraum: 01.01.2017–30.06.2017.

HARTMANN 2017

Hartmann, Markus: Parameterstudie zur Optimierung der Effizienz von Ordnungsschikanen im Vibrationswendelförderer. Master-Semesterarbeit (Nr. 35702). Bearbeitungszeitraum: 01.03.2017–31.08.2017.

WEICH 2018

Weich, Daniel: Methode zum Unschärfemanagement in komplexen Prozessketten am Beispiel Montageprozessplanung. Masterarbeit (Nr. 39306). Bearbeitungszeitraum: 02.10.2017–02.04.2018.

PY 2018

Py, Berenike: Evaluierung einer Methodik zur Planung veränderungsfähiger Fabrikstrukturen durch Einsatz von Veränderungsbefähigern. Masterarbeit (Nr. 39420). Bearbeitungszeitraum: 01.11.2017–01.05.2018.

## Betreute Studienarbeiten

---

HUBER 2018

Huber, Martina: Entwicklung und Konstruktion eines Universalgreifers für die Handhabung von Kleinbauteilen. Bachelorarbeit (Nr. 40662). Bearbeitungszeitraum: 01.12.2017–31.05.2018.

SCHMID 2018

Schmid, Mark: Automatisierte Auslegung von Ordnungsschikanen mit Reinforcement Learning. Master-Semesterarbeit (Nr. 40459). Bearbeitungszeitraum: 01.12.2017–30.06.2018

LEE 2019

Lee, Edwin: Reinforcement Learning - Anwendungsfälle und Potentiale für die Industrie 4.0. Master-Semesterarbeit (Nr. 48142). Bearbeitungszeitraum: 30.03.2019–30.09.2019.

KRAUEL 2020

Krauel, Moritz: Potentialanalyse für Simulationstools in der Auslegung von Vibrationswendelförderern. Bachelorarbeit (Nr. 48176). Bearbeitungszeitraum: 18.09.2019–18.03.2020.

SIESMANN 2020

Siesmann, Paula Anna: Simulation von Ordnungsschikanen mit Hilfe von Physik-Engines in Unity. Masterarbeit (Nr. 51413). Bearbeitungszeitraum: 01.12.2019–01.07.2020.

GUTIERREZ FLORES 2020

Gutierrez Flores, Alberto: Simulation des Vibrationswendelförderer-Schikanenverhaltens und Markov-Ketten-basierter Auswahlalgorithmus. Masterarbeit (Nr. 51214). Bearbeitungszeitraum: 01.03.2020–01.09.2020.

HONEIN 2020

Honein, Jawa Emanuelle: Klassifizierung von 3D-CAD-Modellen für Vibrationswendelförderer mit neuronalen Netzen. Master-Semesterarbeit (Nr. 51994). Bearbeitungszeitraum: 01.11.2019–01.05.2020.

MORABITO 2020

Morabito, Valdemar: Prozessanalyse und -optimierung – neue Arbeitswelten in indirekten Bereichen. Bachelorarbeit (Nr. 51154). Bearbeitungszeitraum: 01.11.2019–01.05.2020.

---

HILBCK 2020

Hilbck, Maximilian: Validierung simulationsgestützter Auslegungen von Ordnungsschikanen in Vibrationswendelförderern. Bachelorarbeit (Nr. 53413). Bearbeitungszeitraum: 01.04.2020–01.10.2020.



## Abbildungsverzeichnis

1.1	Wissenschaftssystematik . . . . .	7
1.2	Dieser Arbeit zugrunde gelegte Forschungsmethodik . . . . .	11
2.1	Schematischer Aufbau eines VWF . . . . .	18
2.2	VWF aus der Praxis mit aktiver OS (geschlitzte Bahn) und Schrauben als Fördergut . . . . .	21
2.3	Beispiel: Sechs mögliche, diskrete Orientierungen einer Schraube	24
2.4	Beispiel für eine OS, zusammengesetzt aus mehreren SE . . . . .	24
2.5	Wirkweisen von Schikanenelementen . . . . .	26
2.6	Beispielhafte OS zur Erläuterung des Prinzips der Transitions- matrizen . . . . .	27
2.7	Darstellung von Werkstück-Orientierungen . . . . .	27
2.8	Ablauf der konventionellen Auslegung von OS für VWF . . . . .	31
2.9	Übersicht über die Domänen KI, ML sowie Lernparadigmen und Anwendungsgebiete . . . . .	44
2.10	Elemente eines künstlichen Neurons . . . . .	48
2.11	Aktivierungsfunktion ReLU . . . . .	49
2.12	Schematischer Aufbau eines CNN . . . . .	50
2.13	Schematische Darstellung des Filters im CNN . . . . .	51
2.14	Beispielhafte Darstellung der Faltung durch den Filter . . . . .	52
2.15	Aufbau eines RL-Systems . . . . .	54
3.1	Methode zur automatisierten Generierung von OS für VWF . . . . .	64
4.1	Erster Methodenschritt „Datenaufbereitung“ und erarbeitete Bausteine . . . . .	68
4.2	Beispiel-Bauteile für die zehn Klassen des Datensets TUM10 . . . . .	80
4.3	Architektur des adaptierten CNN . . . . .	81

4.4	Genauigkeit der Klassifizierung im Verlauf von Training und Validierung . . . . .	82
4.5	Im Rahmen dieser Arbeit entwickelte SE . . . . .	85
4.6	Beispiel für ein Fördergut-spezifisches SE . . . . .	86
4.7	Struktur der CAD-Bibliothek als UML-Klassendiagramm (schematisch) . . . . .	89
5.1	Methodenschritt „Bewertung der Schikanenelemente“ und erarbeitete Bausteine . . . . .	92
5.2	Konvergenz der Transitionsmatrizen für fünf repräsentative Simulationsläufe . . . . .	100
6.1	Methodenschritt „Generierung der Ordnungsschikane“ und erarbeitete Bausteine . . . . .	102
6.2	Übersicht über das RL-System und dessen Komponenten . . . . .	106
6.3	Belohnungskonzept für den RL-Agenten . . . . .	108
6.4	Training des RL-Agenten . . . . .	111
6.5	Prädiktion einer OS . . . . .	113
7.1	Für die Implementierung betrachtete Schraube . . . . .	120
7.2	Ergebnisse des CNN-Systems zur Ähnlichkeit-Analyse . . . . .	122
7.3	Simulation einer Schraube auf einer geschlitzten Bahn . . . . .	123
7.4	Verlauf der kumulierten durchschnittlichen Belohnung über den Trainingsfortschritt . . . . .	126
7.5	Ergebnis aus der OS-Prädiktion . . . . .	127
A.1	Klasse 1: Pilzteil . . . . .	194
A.2	Klasse 2: Niet . . . . .	194
A.3	Klasse 3: T-Stück . . . . .	194
A.4	Klasse 4: Griff . . . . .	195
A.5	Klasse 5: Winkel . . . . .	195
A.6	Klasse 6: Mutter . . . . .	195
A.7	Klasse 7: Lager . . . . .	195
A.8	Klasse 8: Zahnrad . . . . .	196
A.9	Klasse 9: Scheibe . . . . .	196
A.10	Klasse 10: Platte . . . . .	196

A.11 Übersicht über die für das Training des RL-Systems verwendeten Fördergut-Teile . . . . .	199
A.12 Schikanenelement 1: Verengte Bahn . . . . .	200
A.13 Schikanenelement 2: Geschlitzte Bahn . . . . .	201
A.14 Schikanenelement 3: Durchgängig geschlitzte Bahn . . . . .	202
A.15 Schikanenelement 4: Durchlass . . . . .	203
A.16 Schikanenelement 5: Aufrichter . . . . .	204
A.17 Schikanenelement 6: Stufe . . . . .	205
A.18 Schikanenelement 7: Ausschnitt . . . . .	205
A.19 Schikanenelement 8: Formausschnitt . . . . .	206
A.20 Schikanenelement 9: Höhenabweiser . . . . .	207
A.21 Schikanenelement 10: Vorsprung und verengte Bahn . . . . .	208
A.22 Schikanenelement 11: Abweiser . . . . .	209
A.23 Rechenzeiten in der Physiksimulation . . . . .	213





## Tabellenverzeichnis

1.1	Merkmale theoretischer und anwendungsorientierter Wissenschaften im Vergleich . . . . .	9
3.1	Zuordnung der spezifischen zu den allgemeinen Anforderungen	62
4.1	Vergleich der betrachteten Fördergut-Parameter in der Literatur und der Anforderungen innerhalb dieser Arbeit . . . . .	74
4.2	Relevante Fördergut-Parameter und Werkzeuge zur Extraktion	75
4.3	Vergleich von Deep-Learning-Ansätzen für die 3D-Modell-Klassifizierung . . . . .	78
7.1	Fördergut-Charakteristika der zu orientierenden Schraube . . .	121
7.2	Wirtschaftlichkeitsbetrachtung mit beispielhaften Kostensätzen	134
7.3	Erfüllungsgrad der an die Methode gestellten Anforderungen .	135
7.4	Bewertung der Zielerreichung . . . . .	137
A.1	Parameter der für das Training des RL-Systems verwendeten Fördergut-Teile . . . . .	197
A.2	Parameter zu Schikanenelement 1: Verengte Bahn . . . . .	200
A.3	Parameter zu Schikanenelement 2: Geschlitzte Bahn . . . . .	201
A.4	Parameter zu Schikanenelement 3: Durchgäng, geschlitzte Bahn	202
A.5	Parameter zu Schikanenelement 4: Durchlass . . . . .	203
A.6	Parameter zu Schikanenelement 5: Aufrichter . . . . .	204
A.7	Parameter zu Schikanenelement 6: Stufe . . . . .	205
A.8	Parameter zu Schikanenelement 7: Ausschnitt . . . . .	205
A.9	Parameter zu Schikanenelement 8: Formausschnitt . . . . .	206
A.10	Parameter zu Schikanenelement 9: Höhenabweiser . . . . .	207
A.11	Parameter zu Schikanenelement 10: Vorsprung u. verengte Bahn	208
A.12	Parameter zu Schikanenelement 11: Abweiser . . . . .	209

## Tabellenverzeichnis

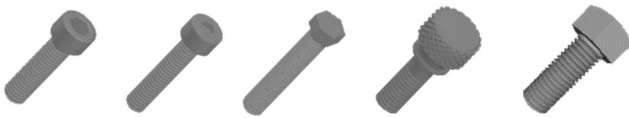
---

A.13 Berechnete Parameterwerte der zur Evaluation erstellten Schikanenelemente . . . . .	210
A.14 Parameter zur Einstellung des SAC-Algorithmus . . . . .	214
A.15 Verwendete Softwarebibliotheken und jeweilige Version . . . . .	215

## **Anhang**

## A.1 Bauteile des Datensets TUM10 (Auswahl)

Die folgenden Übersichten enthalten jeweils eine repräsentative Auswahl der für das Training des CNN verwendeten Fördergut-Teile aus dem Datenset TUM10. Die Abbildungen sind nicht maßstabsgetreu.



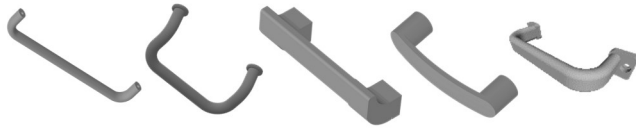
*Abbildung A.1: Klasse 1: Pilzteil*



*Abbildung A.2: Klasse 2: Niet*



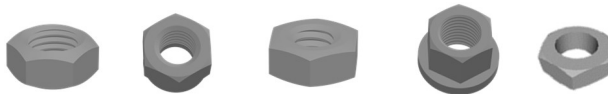
*Abbildung A.3: Klasse 3: T-Stück*



*Abbildung A.4: Klasse 4: Griff*



*Abbildung A.5: Klasse 5: Winkel*



*Abbildung A.6: Klasse 6: Mutter*



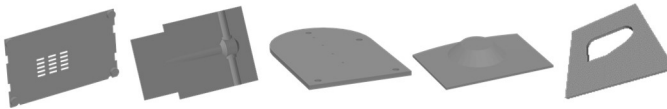
*Abbildung A.7: Klasse 7: Lager*



*Abbildung A.8: Klasse 8: Zahnrad*



*Abbildung A.9: Klasse 9: Scheibe*



*Abbildung A.10: Klasse 10: Platte*

## A.2 Daten für das Training des Reinforcement-Learning-Systems

Tabelle A.1: Parameter der für das Training des RL-Systems verwendeten Fördergut-Teile

Fördergut-Teil (s. Abbildung A.11)	Parameter	Wert
(1)	$l$	28,575 mm
	$d_k$	7,33 mm
	$d_s$	4,756 mm
	$h_k$	3,175 mm
(2)	$l$	28,0 mm
	$d_k$	13,0 mm
	$d_s$	9,5 mm
	$h_k$	11,0 mm
(3)	$l$	38,0 mm
	$d_k$	13,0 mm
	$d_s$	8,0 mm
	$h_k$	8,0 mm
(4)	$l$	24,0 mm
	$d_k$	7,0 mm
	$d_s$	4,0 mm
	$h_k$	4,0 mm
(5)	$l$	34,528 mm
	$d_k$	23,826 mm
	$d_s$	14,283 mm
	$h_k$	9,128 mm
(6)	$l$	20,0 mm
	$d_k$	7,0 mm
	$d_s$	4,0 mm
	$h_k$	4,0 mm
(7)	$l$	25,93 mm
	$d_k$	10,1 mm
	$d_s$	5,86 mm
	$h_k$	5,93 mm
(8)	$l$	20,93 mm
	$d_k$	10,1 mm
	$d_s$	5,86 mm
	$h_k$	5,93 mm

## Anhang

Fördergut-Teil (s. Abbildung A.11)	Parameter	Wert
(9)	$l$	15,39 mm
	$d_k$	10,1 mm
	$d_s$	5,86 mm
	$h_k$	5,93 mm
(10)	$l$	32,9 mm
	$d_k$	13,0 mm
	$d_s$	7,66 mm
	$h_k$	7,9 mm
(11)	$l$	25,0 mm
	$d_k$	8,6 mm
	$d_s$	5,0 mm
	$h_k$	5,0 mm
(12)	$l$	24,0 mm
	$d_k$	7,0 mm
	$d_s$	4,0 mm
	$h_k$	4,0 mm
(13)	$l$	13,9 mm
	$d_k$	6,4 mm
	$d_s$	3,8 mm
	$h_k$	3,9 mm
(14)	$l$	15,0 mm
	$d_k$	8,6 mm
	$d_s$	5,0 mm
	$h_k$	5,0 mm
(15)	$l$	17,0 mm
	$d_k$	8,6 mm
	$d_s$	5,0 mm
	$h_k$	5,0 mm



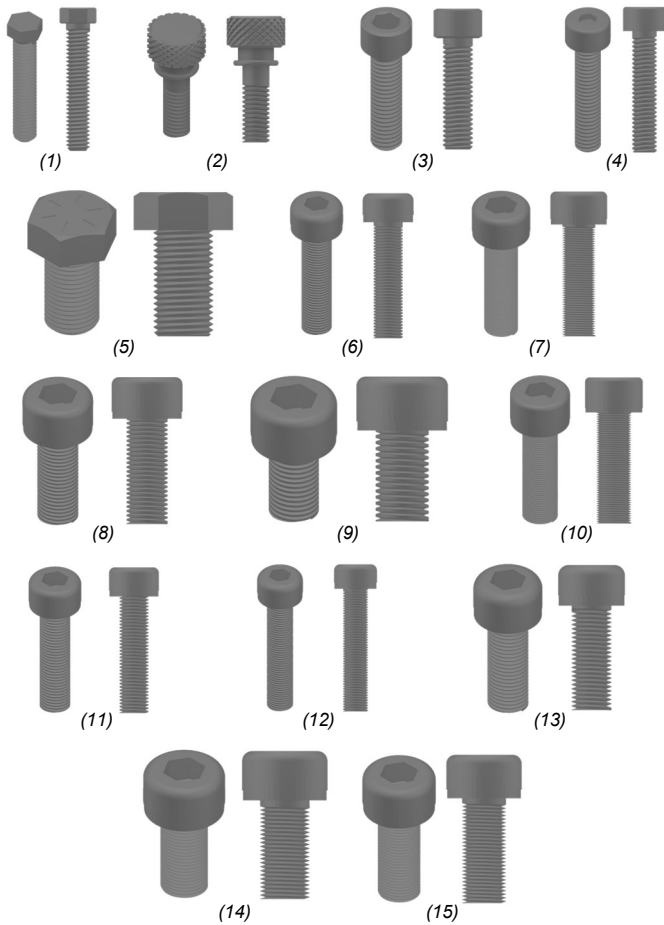


Abbildung A.11: Übersicht über die für das Training des RL-Systems verwendeten Fördergut-Teile (nicht maßstabsgetreu)

### A.3 Generische Schikanenelemente mit Parametern

Die Formelzeichen  $l$ ,  $d_s$ ,  $d_k$  und  $h_k$  beziehen sich auf ein Pilzteil, wie in Abschnitt 7.2 ausführlich beschrieben (s. insb. Abbildung 7.1). Für jeden SE-Parameter wurde über die unten angegebenen Formeln abhängig vom Fördergut-Teil ein Wert berechnet. Zudem wurden zwei weitere Abstufungen mithilfe der Skalierungsfaktoren „Minimum“ und „Maximum“ erzeugt.

#### A.3.1 Schikanenelement 1: Verengte Bahn

Tabella A.2: Parameter zu Schikanenelement 1: Verengte Bahn

Parameter	Formel	Minimum	Maximum
P1: Länge des Ausschnitts	$l_{narrowed} = 1,5 \cdot l$	0,9	1,1
P2: Breite der Bahn	$b_{narrowed} = 0,65 \cdot l$	0,9	1,1

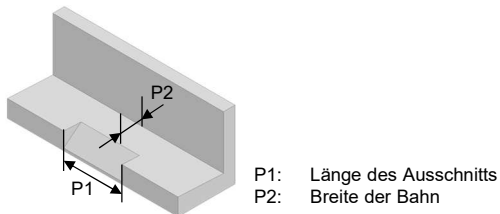


Abbildung A.12: Schikanenelement 1: Verengte Bahn

### A.3.2 Schikanenelement 2: Geschlitzte Bahn

Tabelle A.3: Parameter zu Schikanenelement 2: Geschlitzte Bahn

Parameter	Formel	Minimum	Maximum
P1: Breite des Schlitzes	$b_{grooved} = 1,1 \cdot d_k$	0,8	1,0
P2: Tiefe des Schlitzes	$t_{grooved} = 1,1 \cdot d_k$	0,9	1,1
P3: Abstand zur Wand	$d_{l, grooved} = 0,5 \cdot d_s$	0,9	1,1

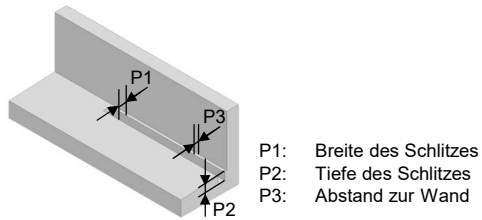


Abbildung A.13: Schikanenelement 2: Geschlitzte Bahn

### A.3.3 Schikanenelement 3: Durchgängig geschlitzte Bahn

Tabelle A.4: Parameter zu Schikanenelement 3: Durchgängig geschlitzte Bahn

Parameter	Formel	Minimum	Maximum
P1: Breite des Schlitzes	$w_{max, slotted} = 0,5 \cdot (d_k + d_s)$	0,9	1,1
P2: Länge des Schlitzes	$l_{slotted} = 2 \cdot l$	0,5	1,5
P3: Abstand zur Wand	$d_{l, slotted} = 0,5 \cdot d_s$	0,9	1,1

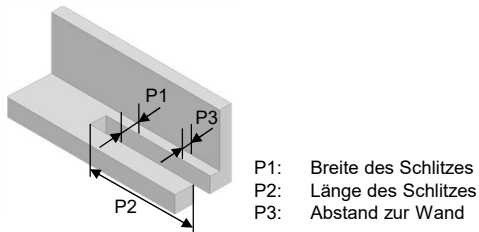
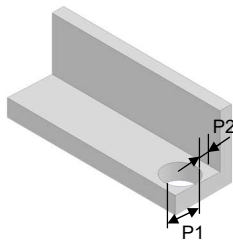


Abbildung A.14: Schikanenelement 3: Durchgängig geschlitzte Bahn

### A.3.4 Schikanenelement 4: Durchlass

Tabelle A.5: Parameter zu Schikanenelement 4: Durchlass

Parameter	Formel	Minimum	Maximum
P1: Durchmesser des Durchlasses	$d_{hole} = 1,1 \cdot (0,5 \cdot l + 0,5 \cdot d_k)$	1,0	1,2
P2: Abstand zur Wand	$d_{l, hole} = 0,5 \cdot d_s$	0,9	1,1



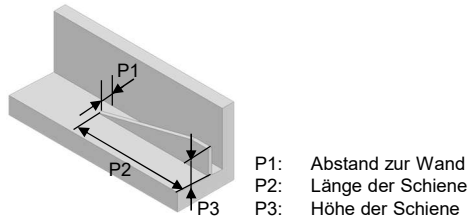
P1: Durchmesser des Durchlasses  
P2: Abstand zur Wand

Abbildung A.15: Schikanenelement 4: Durchlass

**A.3.5 Schikanenelement 5: Aufrichter**

*Tabelle A.6: Parameter zu Schikanenelement 5: Aufrichter*

Parameter	Formel	Minimum	Maximum
P1: Abstand zur Wand	$d_{l, riser} = d_s + 0,5 \cdot (d_k - d_s)$	0,9	1,1
P2: Länge der Schiene	$l_{riser} = 3 \cdot l$	0,5	1,0
P3: Höhe der Schiene	$h_{riser} = 0,8 \cdot (l - h_k)$	0,9	1,1



*Abbildung A.16: Schikanenelement 5: Aufrichter*

### A.3.6 Schikanenelement 6: Stufe

Tabelle A.7: Parameter zu Schikanenelement 6: Stufe

Parameter	Formel	Minimum	Maximum
P1: Höhe der Stufe	$h_{step} = 0,5 \cdot l$	0,8	1,2

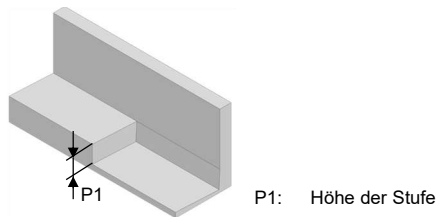


Abbildung A.17: Schikanenelement 6: Stufe

### A.3.7 Schikanenelement 7: Ausschnitt

Tabelle A.8: Parameter zu Schikanenelement 7: Ausschnitt

Parameter	Formel	Minimum	Maximum
P1: Öffnungswinkel	$\alpha_{cutout} = 90 \text{ deg}$	0,67	1,0
P2: Breite der Bahn	$b_{cutout} = 1,5 \cdot d_s$	0,9	1,1

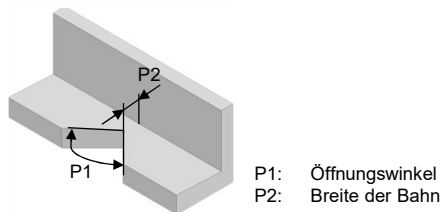
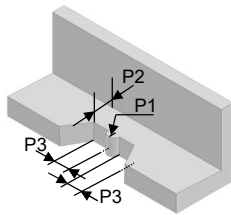


Abbildung A.18: Schikanenelement 7: Ausschnitt

### A.3.8 Schikanenelement 8: Formausschnitt

Tabelle A.9: Parameter zu Schikanenelement 8: Formausschnitt

Parameter	Formel	Minimum	Maximum
P1: Radius des Formausschnitts	$r_{scallop} = 0,25 \cdot l$	0,9	1,1
P2: Breite der Bahn	$b_{scallop} = 1,5 \cdot d_s$	0,9	1,1
P3: Breite der Abstände	$k_{scallop} = 1,2 \cdot d_s$	0,9	1,1



- P1: Radius des Formausschnitts
- P2: Breite der Bahn
- P3: Breite der Abstände

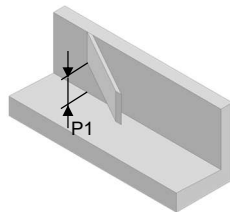
Abbildung A.19: Schikanenelement 8: Formausschnitt



### A.3.9 Schikanenelement 9: Höhenabweiser

Tabelle A.10: Parameter zu Schikanenelement 9: Höhenabweiser

Parameter	Formel	Minimum	Maximum
P1: Durchfahrtshöhe	$z_{blade} = 1,5 \cdot d_k$	1,0	1,2



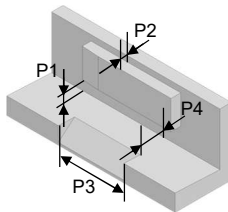
P1: Durchfahrtshöhe

Abbildung A.20: Schikanenelement 9: Höhenabweiser

### A.3.10 Schikanenelement 10: Vorsprung und verengte Bahn

Tabelle A.11: Parameter zu Schikanenelement 10: Vorsprung und verengte Bahn

Parameter	Formel	Minimum	Maximum
P1: Abstand zur Bahn	$z_{projection} = 1,5 \cdot h_k$	1,0	1,2
P2: Tiefe des Vorsprungs	$w_{projection} = 0,9 \cdot (d_k - d_s)$	0,9	1,1
P3: Länge des Ausschnitts	$l_{narrowed} = 1,5 \cdot l$	0,9	1,1
P4: Breite der Bahn	$b_{narrowed} = 0,65 \cdot l$	0,9	1,1



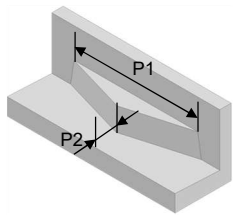
- P1: Abstand zur Bahn
- P2: Tiefe des Vorsprungs
- P3: Länge des Ausschnitts
- P4: Breite der Bahn

Abbildung A.21: Schikanenelement 10: Vorsprung und verengte Bahn

### A.3.11 Schikanenelement 11: Abweiser

Tabelle A.12: Parameter zu Schikanenelement 11: Abweiser

Parameter	Formel	Minimum	Maximum
P1: Länge des Vorsprungs	$l_{break} = 3 \cdot l$	0,5	1,0
P2: Breite der Bahn	$b_{break} = 0.5 \cdot l + 0,25 \cdot d_k$	0,9	1,1



P1: Länge des Vorsprungs  
P2: Breite der Bahn

Abbildung A.22: Schikanenelement 11: Abweiser

**A.4 Berechnete Parameter der erstellten Schikanenelemente**

*Tabelle A.13: Berechnete Parameterwerte der zur Evaluation erstellten Schikanenelemente*

Schikanenelement	Parameter	Bezeichnung	Werte
SE1: Verengte Bahn	P1: Länge des Ausschnitts	$l_{narrowed}$	26,865 mm 29,85 mm 32,835 mm
	P2: Breite der Bahn	$b_{narrowed}$	11,6415 mm 12,935 mm 14,2285 mm
SE2: Geschlitzte Bahn	P1: Breite des Schlitzes	$b_{grooved}$	11,264 mm 12,672 mm 14,08 mm
	P2: Tiefe des Schlitzes	$t_{grooved}$	12,672 mm 14,08 mm 15,488 mm
	P3: Abstand zur Wand	$d_{l, grooved}$	3,465 mm 3,85 mm 4,235 mm
SE3: Durchgängig geschlitzte Bahn	P1: Breite des Schlitzes	$w_{max, slotted}$	9,225 mm 10,25 mm 11,275 mm
	P2: Länge des Schlitzes	$l_{slotted}$	19,9 mm 39,8 mm 59,7 mm
	P3: Abstand zur Wand	$d_{l, slotted}$	3,465 mm 3,85 mm 4,235 mm
SE4: Durchlass	P1: Durchmesser der Durchlasses	$d_{hole}$	17,985 mm 19,7835 mm 21,582 mm
	P2: Abstand zur Wand	$d_{l, hole}$	3,465 mm 3,85 mm 4,235 mm
SE5: Aufrichter	P1: Abstand zur Wand	$d_{l, riser}$	9,225 mm

Schikanenelement	Parameter	Bezeichnung	Werte
			10,25 mm 11,275 mm
	P2: Länge der Schiene	$l_{riser}$	29,85 mm 44,775 mm 59,7 mm
	P3: Höhe der Schiene	$h_{riser}$	8,64 mm 9,6 mm 10,56 mm
SE6: Stufe	P1: Höhe der Stufe	$h_{step}$	7,96 mm 9,95 mm 11,94 mm
SE7: Ausschnitt	P1: Öffnungswinkel	$\alpha_{cutout}$	60,3 deg 74,7 deg 90,0 deg
	P2: Breite der Bahn	$b_{cutout}$	10,395 mm 11,55 mm 12,705 mm
SE8: Formausschnitt	P1: Radius des Formausschnitts	$r_{scallop}$	4,4775 mm 4,975 mm 5,4725 mm
	P2: Breite der Bahn	$b_{scallop}$	10,395 mm 11,55 mm 12,705 mm
	P3: Breite der Abstände	$k_{scallop}$	8,316 mm 9,24 mm 10,164 mm
SE9: Höhenabweiser	P1: Durchfahrtshöhe	$z_{blade}$	19,2 mm 21,12 mm 23,04 mm
SE10: Vorsprung und verengte Bahn	P1: Abstand zur Bahn	$z_{projection}$	11,85 mm 13,035 mm 14,22 mm
	P2: Tiefe des Vorsprungs	$w_{projection}$	4,131 mm 4,59 mm 5,049 mm
	P3: Länge des Ausschnitts	$l_{narrowed}$	26,865 mm

## Anhang

---

Schikanenelement	Parameter	Bezeichnung	Werte
			29,85 mm
			32,835 mm
	P4: Breite der Bahn	$b_{narrowed}$	11,6415 mm
			12,935 mm
			14,2285 mm
SE11: Abweiser	P1: Länge des Vorsprungs	$l_{break}$	29,85 mm
			44,775 mm
			59,7 mm
	P2: Breite der Bahn	$b_{break}$	11,835 mm
			13,15 mm
			14,465 mm

---

## A.5 Rechenzeiten in der Physiksimulation

Zur Erzeugung der für das Training des RL-Systems sowie zur prototypischen Umsetzung der Methode erforderlichen Daten mit der Physiksimulation wurden für jedes Fördergut-Teil jeweils 231 SE mit Fördergut simuliert. Pro SE wurden 250 Simulationsdurchläufe berechnet. Die Höhe der Rechenzeit korreliert mit der Komplexität des simulierten Fördergut-Teils. Diese äußert sich durch die Anzahl der Netzpunkte, welche die Kontur der Computer Aided Design (CAD)-Modelle beschreiben: Je höher die Anzahl, desto länger die Rechenzeit.

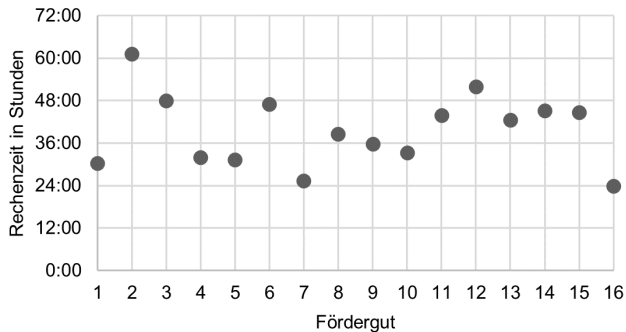


Abbildung A.23: Rechenzeiten in der Physiksimulation; Fördergut 1-15: s. Abbildung A.11, Fördergut 16: zur prototypischen Umsetzung genutzten Schraube: s. Abbildung 7.1

### A.6 Parameter des Reinforcement-Learning-Algorithmus

Für ausführliche Informationen zu den verwendeten Parametern siehe HAARNOJA ET AL. 2018a,b und STABLE BASELINES 2021.

*Tabelle A.14: Parameter zur Einstellung des SAC-Algorithmus für das RL*

Parameter	Wert
Optimierer	Adam (KINGMA & BA 2014)
Strategie	MLP
Trainingsschritte	$10^5$
Lernrate	$3 \cdot 10^{-4}$
Diskontierungsfaktor $\gamma$	0,0
Replay Buffer Size	50.000
Lernbeginn	100
Trainingsfrequenz	1
Anzahl Samples im Mini-Batch	64
Target-Smoothing-Koeffizient $\tau$	0,005
Entropiekoeffizient	automatisch
Target-Update-Intervall	1
Target-Entropie	-3,0
Gradientenschritte	1
Anzahl verdeckter Layer	2
Anzahl verdeckter Units pro Layer	256
Nichtlinearität der Layer	ReLU



## A.7 Verwendete Software und Versionen

Tabelle A.15: Zur prototypischen Umsetzung verwendete Softwarebibliotheken und jeweilige Version

Software-bibliothek	Version	Einsatzzweck	Referenz
Autodesk Inventor Professional 2021	Build 183, Release: 2021	CAD-Konstruktion von SE und Fördergut-Teilen	AUTODESK, INC. 2020
Bullet Physics	2.83	Physiksimulation des Förderprozesses	COUMANS 2019
Python	3.7.9	Programmierung des RL-Systems	PYTHON SOFTWARE FOUNDATION 2021
Stable Baselines	2.10.1	SAC-Agent	A. HILL ET AL. 2018
TensorFlow	1.15.4	NN innerhalb des SAC-Agenten	ABADI ET AL. 2015
TensorBoard	Teil von TensorFlow	Visualisierung und Überprüfung von Lernprozessen	ABADI ET AL. 2015
Optuna	2.5.0	RL-Parameter-optimierung	AKIBA ET AL. 2019
pandas	1.2.2	Werkzeuge zur Datenanalyse	MCKINNEY 2010; REBACK ET AL. 2021
SciPy	1.6.0	Wissenschaftliche Rechenoperationen	VIRTANEN ET AL. 2020
NumPy	1.20.1	Numerische Matrix-Datenverarbeitung	HARRIS ET AL. 2020
Matplotlib	3.3.4	Mathematische Darstellungen wie Graphen und Diagramme	HUNTER 2007
OpenAI Gym	0.18.0	Umgebung im RL-System	BROCKMAN ET AL. 2016