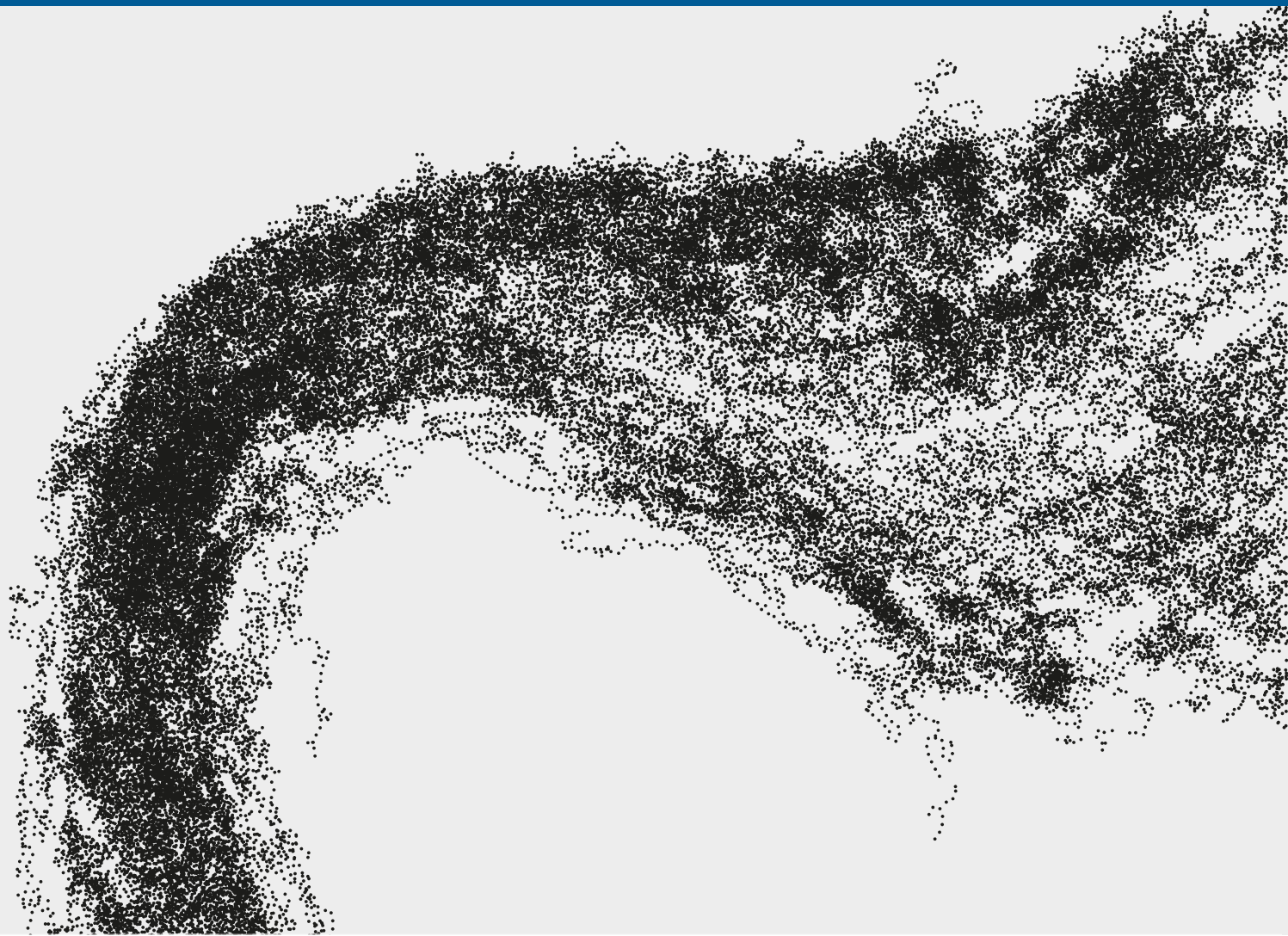


STUTTGARTER BEITRÄGE ZUR PRODUKTIONSFORSCHUNG

PATRICK KÜBLER

Ganzheitliches Verfahren zur Lösung der operativen Planungsprobleme in der Person-zur-Ware-Kommissionierung



Universität Stuttgart



Fraunhofer

IPA

Herausgeber:

Univ.-Prof. Dr.-Ing. Thomas Bauernhansl

Univ.-Prof. Dr.-Ing. Kai Peter Birke

Univ.-Prof. Dr.-Ing. Marco Huber

Univ.-Prof. Dr.-Ing. Oliver Riedel

Univ.-Prof. Dr.-Ing. Dipl.-Kfm. Alexander Sauer

Univ.-Prof. Dr.-Ing. Dr. h.c. mult. Alexander Verl

Univ.-Prof. a.D. Dr.-Ing. Prof. E.h. Dr.-Ing. E.h. Dr. h.c. mult. Engelbert Westkämper

Patrick Kübler

**Ganzheitliches Verfahren zur Lösung
der operativen Planungsprobleme in der
Person-zur-Ware-Kommissionierung**

Kontaktadresse:

Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA, Stuttgart
Nobelstraße 12, 70569 Stuttgart
Telefon 07 11/9 70-11 01
info@ipa.fraunhofer.de; www.ipa.fraunhofer.de

STUTTGARTER BEITRÄGE ZUR PRODUKTIONSFORSCHUNG

Herausgeber:

Univ.-Prof. Dr.-Ing. Thomas Bauernhansl^{1,2}

Univ.-Prof. Dr.-Ing. Kai Peter Birke^{1,4}

Univ.-Prof. Dr.-Ing. Marco Huber^{1,2}

Univ.-Prof. Dr.-Ing. Oliver Riedel³

Univ.-Prof. Dr.-Ing. Dipl.-Kfm. Alexander Sauer^{1,5}

Univ.-Prof. Dr.-Ing. Dr. h.c. mult. Alexander Verl³

Univ.-Prof. a. D. Dr.-Ing. Prof. E.h. Dr.-Ing. E.h. Dr. h.c. mult. Engelbert Westkämper^{1,2}

¹Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA, Stuttgart

²Institut für Industrielle Fertigung und Fabrikbetrieb (IFF) der Universität Stuttgart

³Institut für Steuerungstechnik der Werkzeugmaschinen und Fertigungseinrichtungen (ISW) der Universität Stuttgart

⁴Institut für Photovoltaik (IPV) der Universität Stuttgart

⁵Institut für Energieeffizienz in der Produktion (EEP) der Universität Stuttgart

Titelbild: © AdobeStock

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über www.dnb.de abrufbar.

ISSN: 2195-2892

ISBN (Print): 978-3-8396-1504-1

D 93

Zugl.: Stuttgart, Univ., Diss., 2019

Druck: Mediendienstleistungen des Fraunhofer-Informationszentrum Raum und Bau IRB, Stuttgart
Für den Druck des Buches wurde chlor- und säurefreies Papier verwendet.

© by **FRAUNHOFER VERLAG**, 2019

Fraunhofer-Informationszentrum Raum und Bau IRB

Postfach 80 04 69, 70504 Stuttgart

Nobelstraße 12, 70569 Stuttgart

Telefon 07 11 9 70-25 00

Telefax 07 11 9 70-25 08

E-Mail verlag@fraunhofer.de

URL <http://verlag.fraunhofer.de>

Alle Rechte vorbehalten

Dieses Werk ist einschließlich aller seiner Teile urheberrechtlich geschützt. Jede Verwertung, die über die engen Grenzen des Urheberrechtsgesetzes hinausgeht, ist ohne schriftliche Zustimmung des Verlages unzulässig und strafbar. Dies gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen sowie die Speicherung in elektronischen Systemen.

Die Wiedergabe von Warenbezeichnungen und Handelsnamen in diesem Buch berechtigt nicht zu der Annahme, dass solche Bezeichnungen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und deshalb von jedermann benutzt werden dürften. Soweit in diesem Werk direkt oder indirekt auf Gesetze, Vorschriften oder Richtlinien (z.B. DIN, VDI) Bezug genommen oder aus ihnen zitiert worden ist, kann der Verlag keine Gewähr für Richtigkeit, Vollständigkeit oder Aktualität übernehmen.

Ganzheitliches Verfahren zur Lösung der operativen Planungsprobleme in der Person-zur-Ware-Kommissionierung

Von der Fakultät Konstruktions-, Produktions- und Fahrzeugtechnik
der Universität Stuttgart

zur Erlangung der Würde eines Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Abhandlung

Vorgelegt von

Patrick Kübler (M.Sc.)

aus Schwäbisch-Hall

Hauptberichter: Univ.-Prof. Dr.-Ing. Thomas Bauernhansl

Mitberichter: Univ.-Prof. Dr. rer. pol. Christoph Glock

Tag der mündlichen Prüfung: 04.03.2019

Institut für Industrielle Fertigung und Fabrikbetrieb der Universität Stuttgart

2019

Vorwort

Wo ein Wille ist, ist bekanntermaßen auch ein Weg, doch hat mich die zurückliegende Zeit gelehrt, dass mancher Weg nur gemeinsam mit anderen bis an sein Ziel gegangen werden kann. Am Ende einer herausfordernden und lehrreichen Zeit möchte ich mich deshalb bei all jenen bedanken, die ein Stück des Weges mit mir gegangen sind.

Zuerst möchte ich mich bei meinem Doktorvater Herrn Professor Bauernhansl bedanken, von dem ich persönlich sehr viel gelernt habe und der mir trotz vollem Terminkalender oft mit wertvollem Rat zur Seite gestanden ist. Ebenso möchte ich mich bei Herrn Professor Glock für die fachlich sehr anregenden Diskussionen und die Übernahme des Mitberichts bedanken. Durch seine nicht selbstverständliche Unterstützung hat meine Arbeit stark profitiert.

Für ihre fachliche und emotionale Unterstützung möchte ich mich bei meinen Kollegen vom Fraunhofer-Institut für Produktionstechnik und Automatisierung (IPA) bedanken. Vor allem danke ich meinem Mentor Herrn Eftal Okhan, der mir beigebracht hat, was es heißt, kundenorientiert zu denken und der mir oft mit Rat und Tat zur Seite gestanden ist. Ebenso danke ich Herrn Dr. Markus Weskamp, der mich in schweren Zeiten mit seiner Lebensfreude angesteckt und mir gezeigt hat, dass bei allem Ernst auch die Freude nicht zu kurz kommen darf. Bei Herrn Denis Lock bedanke ich mich für die vielfältige Unterstützung und die gemeinsamen Gespräche und Diskussionen. Herrn Patrik Spalt danke ich dafür, dass er mich auf meinem Weg sehr früh in die richtige Richtung gelenkt hat. Nicht zuletzt möchte ich mich bei Herrn Roman Ungern-Sternberg für die mühevollen und kritischen Durchsichten meiner Arbeit sowie die wertvollen fachlichen Diskussionen bedanken. Ganz herzlich bedanke ich mich bei Frau Dr. Birgit Spaeth für die sprachliche Durchsicht der Arbeit. Meinen Großeltern, Tanten, Onkeln, Cousinen, Cousins und meiner Schwester Hannah Kübler danke ich für den starken Familienzusammenhalt sowie die Unterstützung und menschliche Wärme, die ich durch sie erfahre. Auch wenn ich dies nicht oft zeige, bin ich sehr dankbar, dass es sie gibt.

Meiner Weggefährtin Elena Dauphin danke ich für ihre Liebe und dafür, dass sie immer für mich da ist. Wenige können nachvollziehen, was sie für mich getan hat und wie dankbar ich ihr für alles bin. Ohne sie wäre ich nicht der, der ich bin. Meiner Mutter Susanne Kübler und meinem Vater Günther Kübler bin ich zutiefst dankbar dafür, dass sie meinen Weg nun schon so lange mit mir gehen. Obwohl sie nicht dieselben Möglichkeiten wie ich hatten, haben sie mir alles möglich gemacht. Obgleich sie mir die Freiheit gaben, mein Leben nach meinen Vorstellungen zu gestalten, kann ich mich noch heute auf ihren bedingungslosen Rückhalt und ihre unermüdliche Hilfe verlassen. Ihnen widme ich diese Arbeit.

Patrick Kübler

Kurzzusammenfassung

Kurze und zuverlässige Lieferzeiten werden bei der Kaufentscheidung immer entscheidender. Effizient betriebene Kommissioniersysteme werden damit zu einem wichtigen Faktor. Der Wunsch der Kunden nach individualisierten bzw. personalisierten Produkten führt zu sinkenden Auftragsgrößen, einer zunehmend volatilen Nachfrage und damit zu steigenden Herausforderungen für die Kommissionierung. Um Aufträge möglichst schnell und aufwandsarm kommissionieren zu können, sind drei Planungsprobleme zu lösen: Die Vergabe der Lagerplätze (Lagerplatzvergabe-problem), die Bündelung von Aufträgen in sogenannten Batches (Auftragsbildungsproblem) und die Planung der Kommissioniertouren (Tourenplanungsproblem). Diese Probleme sind stark voneinander abhängig, werden bislang aber separat gelöst. Für die gemeinsame Lösung der Probleme gibt es noch kein geeignetes Verfahren. Die gestiegene Dynamik der Kundennachfrage führt ohne Umlagerungen zu einer sinkenden Effizienz. Häufig schrecken Unternehmen aber davor zurück, Artikel umzulagern, da befürchtet wird, dass den Umlagerungsaufwänden keine entsprechenden Effizienzgewinne gegenüberstehen. In dieser Arbeit wird ein Verfahren vorgestellt, das die drei Planungsprobleme ganzheitlich löst und die zukünftige Dynamik der Kundennachfrage antizipiert. Umlagerungen werden nur durchgeführt, wenn die zu erwartenden Wegstreckenreduktionen den Umlagerungsaufwand übersteigen. Unternehmen profitieren von der ganzheitlichen Lösung der Probleme und der Beherrschung der Dynamik in Form erheblicher Aufwandsreduktionen. Die Leistungsfähigkeit des Verfahrens wird in umfangreichen numerischen Experimenten nachgewiesen. Im Rahmen einer Fallstudie werden die Ergebnisse der numerischen Experimente bestätigt und die Praxistauglichkeit des Verfahrens unter Beweis gestellt.

Short Summary

Short and reliable delivery times are increasingly important for buying decisions. Thus, efficient operated order picking systems turn into a crucial factor. Customers' desire for individualized or personalized products lead to smaller order sizes, an increasingly volatile demand and, therefore, to growing challenges for order picking systems. To be able to pick orders fast and with little effort, three planning problems need to be solved: Assign articles to storage locations (storage location assignment problem), combine orders in batches (order batching problem) and find optimal picking tours (picker routing problem). Even though these problems are strongly interdependent, up to now they are solved separately. There is no method available for solving the three problems jointly. Without relocating items, increased volatile customer demand leads to decreasing efficiency. However, companies hesitate to relocate articles, since there are concerns that the relocation efforts exceed the future efficiency gains. This work presents a method that solves the planning problems jointly and anticipates future dynamics in customer demand. Relocations are only performed if future travel distance reductions exceed the relocation effort. Due to the joint solution of the planning problems and control of dynamics, companies benefit by significant efficiency gains. The performance of the method is proven in comprehensive numerical experiments. The results are confirmed by a case study, where the practicability of the method is demonstrated.

Inhaltsverzeichnis

Inhaltsverzeichnis.....	V
Abbildungsverzeichnis	VIII
Tabellenverzeichnis	XI
Symbolverzeichnis.....	XIII
Abkürzungsverzeichnis	XIX
1 Einleitung	1
1.1 Ausgangssituation	1
1.2 Problemstellung.....	2
1.3 Zielsetzung und Forschungsfrage.....	4
1.4 Wissenschaftlicher Rahmen	5
1.4.1 Wissenschaftstheoretische Positionierung.....	5
1.4.2 Forschungsmethodologie	7
1.4.3 Aufbau der Arbeit.....	9
2 Kommissionierung.....	11
2.1 Lager	11
2.1.1 Definition, Wertschöpfung und Funktion	11
2.1.2 Lagerarten und Lagerhausfunktionen.....	14
2.2 Kommissionierung als Lagerhausfunktion	15
2.2.1 Definition.....	15
2.2.2 Kommissionierziele	16
2.3 Kommissioniersysteme.....	18
2.3.1 Klassifikation.....	18
2.3.2 Manuelle Person-zur-Ware-Kommissioniersysteme	18
2.4 Planungsprobleme der Kommissionierung	22
2.4.1 Strategische und operative Planungsprobleme	22
2.4.2 Lagerplatzvergabe.....	23
2.4.3 Auftragsbildung.....	28
2.4.4 Tourenplanung	31
2.4.5 Interdependenz.....	34

2.5	Zwischenfazit	35
3	Grundlagen der Optimierung und Prognose	37
3.1	Optimierung.....	37
3.2	Optimierungsverfahren.....	39
3.3	Dynamik.....	45
3.4	Prognose.....	47
3.5	Prognoseverfahren	49
4	Reflexion bestehender Ansätze	57
4.1	Ansätze zur gemeinsamen Lösung mehrerer Planungsprobleme	57
4.2	Ansätze zur Lösung des dynamischen Lagerplatzvergabeproblems.....	63
4.3	Zwischenfazit	67
5	Entwurf des Verfahrens	70
5.1	Problembeschreibung	70
5.1.1	Annahmen.....	70
5.1.2	Dynamisches Lagerplatzvergabeproblem	71
5.1.3	Kombiniertes Auftragsbildungs- und Tourenplanungsproblem	74
5.2	Verfahrensaufbau.....	76
5.3	Verfahrensauswahl.....	78
6	Ausgestaltung des Verfahrens	81
6.1	Distanzmatrix	81
6.2	Diskrete evolutionäre Partikelschwarmoptimierung	82
6.2.1	Aufbau	83
6.2.2	Initialisierung	86
6.2.3	Partikelaktualisierung.....	91
6.2.4	Mutation	94
6.2.5	Lokales Suchverfahren	97
6.2.6	Ablauf	98
6.3	Dynamischer Lagerplatzvergabealgorithmus	100
6.3.1	Aufbau	101
6.3.2	Klassifizierung.....	102
6.3.3	Prioritätswertberechnung.....	104
6.3.4	Durchführbarkeit	105
6.3.5	Umlagerungsvorschlag.....	105

6.3.6	Wirtschaftlichkeit	107
6.3.7	Ablauf	110
6.4	Zwischenfazit	112
7	Numerische Experimente	113
7.1	Betrachtungsgegenstand	113
7.2	Probleminstanzen	114
7.2.1	Diskrete evolutionäre Partikelschwarmoptimierung	114
7.2.2	Dynamischer Lagerplatzvergabealgorithmus	116
7.3	Vortest diskrete evolutionäre Partikelschwarmoptimierung	121
7.4	Ergebnisse diskrete evolutionäre Partikelschwarmoptimierung	123
7.5	Ergebnisse dynamischer Lagerplatzvergabealgorithmus	127
7.6	Bewertung der Ergebnisse	131
8	Fallbeispiel	133
8.1	Fallbeschreibung	133
8.2	Ergebnisse	134
8.3	Bewertung der Ergebnisse	139
9	Zusammenfassung	141
	Anhang	145
	Literaturverzeichnis	151

Abbildungsverzeichnis

<i>Abbildung 1: Wissenschaftssystematik</i>	5
<i>Abbildung 2: Forschungsmethodisches Vorgehen</i>	7
<i>Abbildung 3: Heuristischer Bezugsrahmen</i>	9
<i>Abbildung 4: Aufbau der Arbeit</i>	10
<i>Abbildung 5: Wertschöpfungsverständnis</i>	12
<i>Abbildung 6: Lagerhausfunktionen</i>	14
<i>Abbildung 7: Logistikziele</i>	16
<i>Abbildung 8: Bestandteile der Kommissionierzeit</i>	17
<i>Abbildung 9: Klassifikation Kommissioniersysteme</i>	18
<i>Abbildung 10: Layoutvarianten</i>	20
<i>Abbildung 11: Planungsprobleme</i>	22
<i>Abbildung 12: Betriebsstrategien</i>	23
<i>Abbildung 13: Artikelanordnungsmuster</i>	26
<i>Abbildung 14: Routingstrategien</i>	33
<i>Abbildung 15: Auswahl Betriebsstrategien</i>	36
<i>Abbildung 16: Klassifizierung Einzellösungsverfahren</i>	42
<i>Abbildung 17: Zeitreihenmodelle</i>	48
<i>Abbildung 18: Prognoseverfahren</i>	50
<i>Abbildung 19: Prognoseverfahren zur Abschätzung der Modellkoeffizienten</i>	51
<i>Abbildung 20: Bewertungsmatrix bestehender Ansätze</i>	68
<i>Abbildung 21: Verfahrensaufbau</i>	76
<i>Abbildung 22: Block-Layout und dessen Repräsentation durch einen Graphen</i>	81
<i>Abbildung 23: Aufbau DEPSO</i>	83
<i>Abbildung 24: Ablauf DEPSO</i>	85
<i>Abbildung 25: First-Fit-Regel</i>	86
<i>Abbildung 26: Differenz zweier Partikel</i>	92
<i>Abbildung 27: Beispielhafte Partikelbewegung</i>	94
<i>Abbildung 28: Mutationsoperatoren</i>	96

<i>Abbildung 29: Aufbau dynamischer Lagerplatzvergabealgorithmus</i>	101
<i>Abbildung 30: Artikelzeitreihe und Klassengrenzen</i>	103
<i>Abbildung 31: Prioritätswertberechnung</i>	104
<i>Abbildung 32: Umlagerungsfälle</i>	106
<i>Abbildung 33: Schätzverfahren</i>	108
<i>Abbildung 34: Lagerlayout der numerischen Experimente</i>	113
<i>Abbildung 35: Zugriffshäufigkeitsverteilung</i>	115
<i>Abbildung 36: Erzeugung Auftragsdaten DEPSO</i>	115
<i>Abbildung 37: Erzeugung Auftragsdaten dynamischer Lagerplatzvergabealgorithmus</i>	117
<i>Abbildung 38: Verfahrensvergleich numerische Experimente DEPSO</i>	124
<i>Abbildung 39: Vergleich DEPSO-Savings (numerische Experimente)</i>	124
<i>Abbildung 40: Einfluss Anzahl Positionen und Menge je Position auf das Optimierungsergebnis am Beispiel von 200 Aufträgen und SOP (numerische Experimente)</i>	125
<i>Abbildung 41: Rechenzeit DEPSO (numerische Experimente)</i>	126
<i>Abbildung 42: Optimierungsergebnis Szenario 1 dynamische Lagerplatzvergabe</i>	127
<i>Abbildung 43: Umlagerungsaufwand, Ersparnis und Wegstreckenreduktion Szenario 1 dynamische Lagerplatzvergabe</i>	128
<i>Abbildung 44: Anteil umgelagerter Artikel Szenario 1 dynamische Lagerplatzvergabe</i>	129
<i>Abbildung 45: Optimierungsergebnis Szenario 2 dynamische Lagerplatzvergabe</i>	130
<i>Abbildung 46: Umlagerungsaufwand, Ersparnis und Wegstreckenreduktion Szenario 2 dynamische Lagerplatzvergabe</i>	130
<i>Abbildung 47: Anteil umgelagerter Artikel Szenario 2 dynamische Lagerplatzvergabe</i>	131
<i>Abbildung 48: Layout Fallbeispiel</i>	133
<i>Abbildung 49: Kommissionierweglänge mit FCFS und DEPSO (Fallbeispiel)</i>	136

*Abbildung 50: Wegstreckenreduktion durch den Einsatz des DEPSO-Algorithmus
(Fallbeispiel) 136*

Abbildung 51: Rechenzeit DEPSO (Fallbeispiel) 137

Abbildung 52: Kommissionierweglänge nach Umlagerung (Fallbeispiel) 137

*Abbildung 53: Umlagerungsaufwand, Ersparnis und Wegstreckenreduktion
(Fallbeispiel) 138*

Abbildung 54: Anteil umgelagerter Artikel (Fallbeispiel) 139

Tabellenverzeichnis

<i>Tabelle 1: Empfohlene Werte für die Glättungsparameter</i>	56
<i>Tabelle 2: Problemkonstanten und -parameter dynamische Lagerplatzvergabe</i> ..	121
<i>Tabelle 3: Problemklassen Vortest und Parameterwerte</i>	122
<i>Tabelle 4: Ergebnisse Vortest</i>	123
<i>Tabelle 5: Übersicht Wegstreckenreduktionen DEPSO (numerische Experimente)</i>	125
<i>Tabelle 6: Untersuchte Parameterwerte im Vortest des Fallbeispiels</i>	134
<i>Tabelle 7: Optimierungsergebnisse numerische Experimente DEPSO</i>	145
<i>Tabelle 8: Dynamischer Lagerplatzvergabealgorithmus Szenario 1: Kommissionierweglänge DEPSO ohne Umlagerung [in LE]</i>	146
<i>Tabelle 9: Dynamischer Lagerplatzvergabealgorithmus Szenario 1: Umlagerungsaufwand [in LE]</i>	146
<i>Tabelle 10: Dynamischer Lagerplatzvergabealgorithmus Szenario 1: Anzahl umgelagerter Artikel</i>	146
<i>Tabelle 11: Dynamischer Lagerplatzvergabealgorithmus Szenario 1: Rechenzeit [in Sekunden]</i>	147
<i>Tabelle 12: Dynamischer Lagerplatzvergabealgorithmus Szenario 1: Kommissionierweglänge nach Umlagerung ohne Aufwand [in LE]</i>	147
<i>Tabelle 13: Dynamischer Lagerplatzvergabealgorithmus Szenario 2: Kommissionierweglänge DEPSO ohne Umlagerung [in LE]</i>	148
<i>Tabelle 14: Dynamischer Lagerplatzvergabealgorithmus Szenario 2: Umlagerungsaufwand [in LE]</i>	148
<i>Tabelle 15: Dynamischer Lagerplatzvergabealgorithmus Szenario 2: Anzahl umgelagerter Artikel</i>	148
<i>Tabelle 16: Dynamischer Lagerplatzvergabealgorithmus Szenario 2: Rechenzeit [in Sekunden]</i>	149
<i>Tabelle 17: Dynamischer Lagerplatzvergabealgorithmus Szenario 2: Kommissionierweglänge nach Umlagerung ohne Aufwand [in LE]</i>	149

<i>Tabelle 18: Fallstudie: Kommissionierweglänge und Umlagerungsaufwand [in Meter]</i>	150
<i>Tabelle 19: Fallstudie: Anzahl umgelagerter Artikel</i>	150

Symbolverzeichnis

a_t	Niveauwert zum Zeitpunkt t
\hat{a}_T	Schätzwert für den Niveauwert zum Zeitpunkt T
A	Menge aller Artikel, einschließlich des fiktiven, dem Depot zugeordneten Artikel, der durch den Index „0“ repräsentiert wird
A^{Auf}	Anzahl Aufträge
A_t^{Auf}	Anzahl Aufträge zum Zeitpunkt t
A^{gesPos}	Anzahl Positionen aller Artikel
A_t^{gesPos}	Anzahl Positionen aller Artikel zum Zeitpunkt t
A^{maxPos}	Maximale Anzahl Positionen je Auftrag
A^{Partikel}	Anzahl Partikel im Schwarm
A_k^{Pos}	Anzahl Positionen in Auftrag k
$A_{k,t}^{\text{Pos}}$	Anzahl Positionen in Auftrag k zum Zeitpunkt t
A_m^{Pos}	Anzahl Positionen von Artikel m
$A_{m,t}^{\text{Pos}}$	Anzahl Positionen von Artikel m zum Zeitpunkt t
$\hat{A}_{m,t}^{\text{Pos}}$	Schätzwert für die Anzahl Positionen von Artikel m zum Zeitpunkt t
$A_{\text{relKG},t}^{\text{Pos}}$	Anzahl Positionen der relevanten Klassengrenze zum Zeitpunkt t
$A_{m,t}^{\text{Posolr}}$	Anzahl Positionen von Artikel m ohne irreguläre Komponente zum Zeitpunkt t
A^{Unterp}	Anzahl Unterperioden
AL_{kl}^{derzl}	Anzahl derzeit leerer Lagerplätze in Klasse kl
AL_{kl}^{leerw}	Anzahl leer werdender Lagerplätze in Klasse kl
AL_{kl}^{vollw}	Anzahl voll werdender Lagerplätze in Klasse kl
b	Batch
b_t	Trendan- bzw. -abstieg zum Zeitpunkt t
\hat{b}_t	Schätzwert für den Trendan- bzw. -abstieg zum Zeitpunkt t
B	Menge aller Batches
$B_{p,G_{\text{best}}}$	Bewegungswahrscheinlichkeit von Partikel p in Richtung G_{best}

$B_{p,P_{best}}$	Bewegungswahrscheinlichkeit von Partikel p in Richtung P_{best}
BA	Basisartikel
c	Parameter zur Erzeugung einer Artikelnachfrage mit Zugriffshäufigkeit ZH
c_t	Saisonfaktor zum Zeitpunkt t
\hat{c}_t	Schätzwert für den Saisonfaktor zum Zeitpunkt t
C	Kommissioniergerätkapazität
$d_{i,j}$	Entfernung zwischen Lagerplatz i und j
$d_{q,r}$	Entfernung zwischen Knoten q und r
$d_{m,relKG,t}$	Distanz zwischen der Zeitreihe des Artikels m und der relevanten Klassengrenze $relKG$ zum Zeitpunkt t
d^{nTau}	Distanz nach Kantentausch
$d_{m,relKG,t}^{rel}$	Relative Distanz zwischen der Zeitreihe des Artikels m und der relevanten Klassengrenze $relKG$ zum Zeitpunkt t
$\hat{d}_{m,relKG,t}$	Schätzwert für die Distanz zwischen der Zeitreihe des Artikels m und der relevanten Klassengrenze $relKG$ zum Zeitpunkt t
$\hat{d}_{m,relKG,t}^{rel}$	Schätzwert für die relative Distanz zwischen der Zeitreihe des Artikels m und der relevanten Klassengrenze $relKG$ zum Zeitpunkt t
d^{vTau}	Distanz vor Kantentausch
D	Distanzmatrix
Df_{p_1,p_2}	Differenz zwischen Partikel p_1 und p_2
DF_{kl}	Durchführbarkeitsfaktor der Klasse kl
e_t	Zufallsabweichung zum Zeitpunkt t
end	Letzte Position einer Permutation
E	Kanten
$f_{m,n,t}$	Häufigkeit mit der Artikel m und n in Periode t nacheinander kommissioniert werden
G	Graph
G_{best}	Global Best: Bislang beste Permutation des Schwarms
GA	Gegenartikel
i, j	Lagerplätze
i_m^{zuk}	Zukünftiger Lagerplatz von Artikel m

Int_p	Intensität von Partikel p
Int^{\max}	Maximale Intensität aller Partikel
Int^{\min}	Minimale Intensität aller Partikel
$\text{Ir}_{m,t}$	Irreguläre Komponente von Artikel m zum Zeitpunkt t
Irf	Irregulärer Faktor
It^{akt}	Aktuelle Iterationsschleife des DEPSO-Algorithmus
It^{LS}	Aktuelle Iterationsschleife der lokalen Suche
It^{\max}	Maximale Anzahl Iterationen des DEPSO-Algorithmus
$\text{It}^{\max\text{LS}}$	Maximale Anzahl Iterationen der lokalen Suche
k	Auftrag
kl	Klasse
K	Menge aller Aufträge
$\text{KR}_{m,t}$	Kommissionierwegstreckenreduktion in Periode t durch Umlagerung von Artikel m
$\widehat{\text{KR}}_{m,t}$	Schätzwert für die Kommissionierwegstreckenreduktion in Periode t durch Umlagerung von Artikel m
K_t	Kommissioniertour
$K_t(q)$	Knoten an der q . Stelle in der Kommissioniertour K_t
$K_t(1:q)$	Kommissioniersequenz von der 1. bis zur q . Stelle der Kommissioniertour
$K_t(1:(-1):q)$	Kommissioniersequenz von der 1. bis zur q . Stelle der Kommissioniertour in umgekehrter Reihenfolge
K_t^{neu}	Neu gebildete Kommissioniertour
$\text{KW}_{G_{\text{best}}}$	Kommissionierweglänge von G_{best}
KW_{k_1}	Kommissionierweglänge bei separater Kommissionierung von Auftrag k_1
$\text{KW}_{k_1 k_2}$	Kommissionierweglänge bei gemeinsamer Kommissionierung der Aufträge k_1 und k_2
KW_p	Kommissionierweglänge von Partikel p
$\text{KW}^{\text{akttat}}$	Kommissionierweglänge in der aktuellen Periode mit der tatsächlichen Artikelanordnung
KW^{\max}	Längster Kommissionierweg aller Partikel im Schwarm
KW^{tau}	Kommissionierweglänge nach Umlagerung

KW^{ver}	Vergleichskommissionierweglänge
L	Saisonzyklusperiode
m, n	Artikel
M	Maximal zulässige Schwankung
M_p	Mutationswahrscheinlichkeit von Partikel p
$M_{pos,k}$	Menge in Position pos in Auftrag k
$M_{pos,k,t}$	Menge in Position pos in Auftrag k zum Zeitpunkt t
M^{maxPos}	Maximale Menge je Position
N	Menge aller Knoten
$Nä_p$	Nähe des Partikels p
o	Anzahl Perioden, die ein Artikel bereits falsch gelagert und in der Zielklasse hätte sein müssen, um für eine Umlagerungsprüfung in Frage zu kommen
pos	Position
P_p	Permutation von Partikel p
$P_p(j)$	Auftrag an der Stelle j der Permutation von Partikel p
P_t	Prozess zum Zeitpunkt t
p_p^{best}	Personal Best: Bisher beste Permutation von Partikel p
P^{ver}	Vergleichspermutation
PW_m	Prioritätswert des Artikels m
q, r	Knoten
$relKG$	Relevante Klassengrenze
$runden$	Rundungsfunktion
$s_{k,i}$	= 1, falls ein Artikel aus Auftrag k von Lagerplatz i entnommen wird (= 0, sonst)
s_t	Saisonkomponente zum Zeitpunkt t
$sign$	Signumsfunktion
S	Teilmenge der Lagerplätze
$\zeta^{G_{best}}$	Schwellwert für eine Bewegung in Richtung G_{best}
$\zeta^{maxStag}$	Maximale Stagnationsschranke

S_m^{Saison}	Saisonstärke von Artikel m
S^{Stag}	Stagnationsschranke
$S^{\text{Stag}G_{\text{best}}}$	Stagnationsfaktor, gibt an, wie viele Iterationen durchlaufen wurden ohne, dass sich G_{best} verbessert hat
S^{Wahr}	Wahrscheinlichkeitsschranke
$Sav_{k_1 k_2}$	Savingswert der Aufträge k_1 und k_2
Sf	Saisonfaktor
t	Zeitpunkt
T	Gegenwart
T_m^{Trend}	Trendstärke von Artikel m
Tf	Trendfaktor
u	Anzahl Perioden, die ein Artikel in der Zukunft in der Zielklasse sein muss, um für eine Umlagerungsprüfung in Frage zu kommen
U	Untersuchungszeitraum
U_m^{admAuf}	Administrativer Aufwand für die Umlagerung von Artikel m
U_m^{disAuf}	Distanzaufwand für die Umlagerung von Artikel m
U_m^{gesAuf}	Aufwand zur Umlagerung von Artikel m
U_m^{phyAuf}	Physischer Aufwand für die Umlagerung von Artikel m
U_m^{Ziel}	Zukünftiger Zeitraum in dem der Artikel m in der Zielklasse verbleiben wird
$v_{p,k,t}$	Geschwindigkeit des Partikels p an der k . Stelle des Geschwindigkeitsvektors zum Zeitpunkt t
v^{kom}	Gehgeschwindigkeit des Kommissionierers
V	Menge aller Lagerplätze, wobei das Depot durch den Index „0“ repräsentiert wird
$V_{p,t}$	Geschwindigkeitsvektor von Partikel p zum Zeitpunkt t
w_k	Gewicht von Auftrag k
$w_{m,i}$	= 1, falls Artikel m Lagerplatz i zugewiesen werden soll (= 0, sonst)
x	Prognosezeitraum
x_q	x-Koordinate von Knoten q
x_t	Beobachtungswert einer Zeitreihe zum Zeitpunkt t

$\hat{x}_{T,\tau}$	Prognosewert für die Vorhersageperiode ($T + \tau$)
$X_{k,b}$	= 1, falls Auftrag k Batch b zugewiesen wird (= 0, sonst)
y_q	y-Koordinate von Knoten q
y	Prozentualer Anteil der Artikel an der Gesamtmenge aller Artikel
$Y_{i,j,b}$	= 1, falls Lagerplatz i direkt nach Lagerplatz j in Batch b besucht wird (= 0, sonst)
z^{adm}	Zeit für die administrative Umlagerung
z^{phy}	Zeit für die physische Umlagerung
Z	Zielfunktionswert
$Z_{i,b}$	= 1, falls Lagerplatz i in Batch b besucht wird (= 0, sonst)
ZH	Zugriffshäufigkeitsparameter
α	Glättungsparameter
α^{expGI}	Glättungsparameter für die exponentielle Glättung erster Ordnung
α^{HW}	Glättungsparameter für das Holt-Winters-Verfahren
β	Glättungsparameter
β^{HW}	Glättungsparameter für das Holt-Winters-Verfahren
γ	Glättungsparameter
γ^{HW}	Glättungsparameter für das Holt-Winters-Verfahren
Δ	Differenz zwischen Distanz nach Kantentausch und Distanz vor Kantentausch
$\xi_{m,t}$	Verhältnis aus Kommissionierwegstreckenreduktion durch Umlagerung von Artikel m und relativer Distanz zwischen der Zeitreihe des Artikels m und der relevanten Klassengrenze zum Zeitpunkt t
$\hat{\xi}_{m,t}$	Schätzwert für das Verhältnis aus Kommissionierwegstreckenreduktion durch Umlagerung von Artikel m und relativer Distanz zwischen der Zeitreihe des Artikels m und der relevanten Klassengrenze zum Zeitpunkt t
τ	Prognoseschritt

Abkürzungsverzeichnis

DEPSO	Diskrete evolutionäre Partikelschwarmoptimierung
EC	Evolutionary Computation
FCFS	First-Come-First-Served
GE	Gewichtseinheiten
GRASP	Greedy Adaptive Search Procedure
IPA	Institut für Produktionstechnik und Automatisierung
LE	Längeneinheiten
PSO	Partikelschwarmoptimierung
PzW	Person-zur-Ware
SA	Simulated Annealing
SOP	Single Order Picking
TS	Tabu Search
TSP	Traveling-Salesman-Problem

1 Einleitung

1.1 Ausgangssituation

Um zukünftig auf den Märkten der entwickelten Industrienationen erfolgreich zu sein, müssen Unternehmen auf die stark veränderten Konsumbedürfnisse mit zum Teil hochindividualisierten Produkten reagieren. Personalisierte Produktion („Mass Personalization“) wird neben dem nachhaltigen Umgang mit Ressourcen („Mass Sustainability“) zum Schlüssel, um im globalen Wettbewerb bestehen zu können (Bauernhansl 2015; Bauernhansl 2017, S. 29). Unternehmen sehen sich vor die Herausforderung gestellt, eine immer größer werdende Produktvielfalt mit kleiner werdenden Auftragsgrößen und kürzeren Lebenszyklen in ihrem Produktions- bzw. Logistiksystem wirtschaftlich abzubilden. Kurze und zuverlässige Lieferzeiten werden immer entscheidender (Tsamis et al. 2015, S. 272). Mit dem rapiden Wachstum des E-Commerce und dem Vorhandensein mehrerer Anbieter für ein Produkt wird der Lieferservice zum entscheidenden Wettbewerbsfaktor (Albareda-Sambola et al. 2009, S. 655-656). Um diese Herausforderungen erfolgreich beherrschen zu können, müssen Unternehmen über ein Kommissioniersystem verfügen, das in der Lage ist, Aufträge schnell und fehlerfrei zu bearbeiten und damit dem Kunden eine hohe Liefertermintreue zu garantieren (Le-Duc et al. 2007, S. 374-375). Deshalb hat die Kommissionierung in den letzten Jahren wieder eine größere Aufmerksamkeit in Wissenschaft und Praxis erlangt (Manzini et al. 2007, S. 811). Die Kommissionierung als wesentlicher Wertschöpfungsprozess im Lager (Nave 2009, S. 16) ist gleichzeitig die kostenintensivste Tätigkeit (Tompkins et al. 2010, S. 432). Je nach Studie variiert der Anteil der Kommissionierkosten an den gesamten Lagerhauskosten zwischen 50 und 65 % (Frazelle 2002, S. 148; Coyle et al. 2003, S. 329). Die Lagerhauskosten machen durchschnittlich 25 % der Logistikkosten eines Unternehmens aus, wobei dieser Wert je nach Branche zwischen 16 % (Maschinenbau) und 33 % (Einzelhandel) variiert (Mayer et al. 2009, S. 13). Dennoch existieren in vielen Unternehmen im

Bereich der Kommissionierung noch signifikante Optimierungspotentiale (Koch 2014, S. 8). Da Effizienzsteigerungspotentiale in der Fertigung durch Optimierungsmaßnahmen häufig weitestgehend ausgeschöpft sind, rücken angrenzende Bereiche wie die logistischen Prozesse in den Fokus (Glock et al. 2011, S. 65). Die Kommissionierung ist der arbeitsintensivste Bereich im Lager, wenn manuell kommissioniert wird oder aber kapitalintensiv, wenn dies automatisiert erfolgt. Daher sehen viele Experten hier die höchste Priorität für produktivitätssteigernde Maßnahmen (de Koster et al. 2007, S. 481). Eine gut organisierte Kommissionierung, die Artikel schnell kommissionieren und damit kurze Durchlaufzeiten realisieren kann, ermöglicht eine hohe Reaktionsfähigkeit auf immer kurzfristiger eingehende Bestellungen und eine schnelle Belieferung der Kunden (Bottani et al. 2012, S. 127). Maßnahmen, die die Effizienz von Kommissioniersystemen erhöhen, sind nicht nur ein Mittel, die hohen Kommissionierkosten zu senken, sondern ermöglichen es Unternehmen auch, den Kunden einen höheren Lieferservice zu garantieren (Chiang et al. 2011, S. 219-220). Eine schlecht organisierte Kommissionierung gefährdet die Wettbewerbsposition, weil sie Kunden durch den schlechten Lieferservice abschreckt und die Logistikkosten unnötig ansteigen lässt (Wäscher 2004, S. 324). Die Bedeutung der Kommissionierung wird mit dem Voranschreiten der personalisierten Produktion zunehmen, da das Logistiksystem in der Lage sein muss, auch kurzfristige Änderungen umzusetzen und mit einer größer werdenden Artikelvielfalt umgehen zu können. Mit der Vielfalt wird das Problem der Dynamik zunehmen, da die Kundennachfrage volatiler wird. Unternehmen müssen Wege finden, mit dieser steigenden Dynamik umzugehen (Choy et al. 2014, S. 111-112).

1.2 Problemstellung

Bei der Planung von Kommissioniersystemen werden zwei Ebenen unterschieden: Die strategische und die operative Planungsebene (de Koster et al. 2007, S. 485). Im Rahmen der strategischen Planung geht es unter anderem darum, das Lagerlayout und die eingesetzten Betriebsmittel festzulegen (Gu et al. 2007, S. 3). Die operative Planung umfasst die Lagerplatzvergabe, die Tourenplanung und die Auftragsbildung

(Caron et al. 1998, S. 713). In einer Vielzahl von Arbeiten wird auf die starke *Interdependenz* der operativen Planungsprobleme hingewiesen (unter anderem van Gils et al. 2017, S. 5; de Koster et al. 2007, S. 486; Rao et al. 2013a, S. 1309; Sooksaksun et al. 2012, S. 220; Rouwenhorst et al. 2000, S. 522; Chan et al. 2011, S. 2688). Trotz der vielseitigen Abhängigkeiten werden die Planungsprobleme bisher separat gelöst (de Koster et al. 2007, S. 486). Es wurden Lösungsansätze für die Einzelprobleme entwickelt, jedoch existiert kein Verfahren, das die drei Probleme ganzheitlich löst und die Interdependenz berücksichtigt (van Gils et al. 2017, S. 26). Bislang gibt es nur wenige Arbeiten, die sich mit der Kombination mehrerer Probleme beschäftigen (de Koster et al. 2007, S. 497-498; Matusiak et al. 2014, S. 968). Problematisch ist, dass dies zu lokalen Optima führt, die weit von einer optimalen Lösung entfernt sein können (Hsieh et al. 2011, S. 618). Die wenigen Ansätze, die zwei Planungsprobleme gemeinsam lösen, zeigen, dass dadurch signifikante Verbesserungen erzielt werden können (siehe etwa Kulak et al. 2012, S. 71). Ein Problem dieser Verfahren ist deren mangelnde Praxistauglichkeit, da sie zu komplex sind und nur für sehr stark vereinfachte Planungssituationen entwickelt wurden (van Gils et al. 2017, S. 26). Mehrfach wird die Notwendigkeit einer ganzheitlichen Methode zur Lösung der drei Planungsprobleme hervorgehoben (siehe unter anderem van Gils et al. 2017, S. 26; Pan et al. 2015, S. 13; de Koster et al. 2007, S. 497-498; Wäscher 2004, S. 344; Henn et al. 2011, S. 26; Koch et al. 2016, S. 151; Tsai et al. 2008, S. 6554). So auch von Koch (2014, S. 178): „*Von besonderem Interesse wären darüber hinaus Untersuchungen über eine mögliche Integration der drei operativen Planungsprobleme der Kommissionierung. Derzeit werden die Teilprobleme der Kommissionierung üblicherweise separat gelöst, obwohl sie starke Interdependenzen aufweisen. Ansätze für eine integrierte Betrachtung der Kommissionierung wären hier erstrebenswert.*“

Der effiziente Betrieb eines Kommissioniersystems muss auch dann sichergestellt sein, wenn die Nachfrage stark schwankt, die Produkte sinkende Lebenszyklen haben und die Variantenvielfalt zunimmt. Die *Dynamik* der Kundennachfrage hat die

größte Auswirkung auf das Lagerplatzvergabeproblem, da sie dazu führt, dass Artikel nicht optimal gelagert werden und das System nicht mehr effizient betrieben werden kann (ten Hompel et al. 2010, S. 57-58). Bisherige Verfahren zur Lagerplatzvergabe sind aber meist statisch (Kofler et al. 2015, S. 416). Die wenigen Verfahren zur dynamischen Lagerplatzvergabe geben oft keinen Anhaltspunkt, ob die zu erwartende Aufwandsreduktion den notwendigen Umlagerungsaufwand übersteigt (Kofler et al. 2015, S. 416). Für die integrierte Lösung der drei Planungsprobleme gibt es demnach noch kein geeignetes Verfahren, das den Einfluss der zunehmenden Dynamik der Kundennachfrage auf die Lagerplatzvergabeentscheidung berücksichtigt.

1.3 Zielsetzung und Forschungsfrage

Ziel der Arbeit ist es, ein Verfahren zu entwickeln, das die drei operativen Planungsprobleme der Kommissionierung ganzheitlich löst und deren Interdependenz berücksichtigt. Das Verfahren soll in einem dynamischen Umfeld, in dem die Nachfrage stark schwankt und sich über die Zeit verändert, einsetzbar sein. Es soll zur Lösung von praxisrelevanten Problemgrößen eingesetzt werden können und sicherstellen, dass im Falle von Umlagerungen die zukünftig zu erwartende Aufwandsreduktion nicht durch den Umlagerungsaufwand aufgehoben wird. Es wird von einem manuellen Person-zur-Ware-Kommissioniersystem ausgegangen, da nach einer Expertenschätzung 80 % der Systeme abgedeckt und so ein möglichst großer Anwenderkreis erschlossen werden kann (de Koster et al. 2007, S. 485). Um den Kommissionieraufwand zu reduzieren, soll die Kommissionierweglänge minimiert werden. Daraus leiten sich folgende Anforderungen an das Verfahren ab:

- Wegstreckenreduktion
- Interdependenz
- Dynamik
- Umlagerungsaufwand
- Praxistauglichkeit

Aus der Zielformulierung resultiert folgende handlungsleitende Forschungsfrage:
 „Wie können die operativen Planungsprobleme in manuellen Person-zur-Ware-Kommissioniersystemen unter Berücksichtigung ihrer Interdependenz mit dem Ziel möglichst kurzer Kommissionierwege gemeinsam so gelöst werden, dass das System trotz zunehmender Dynamik effizient betrieben werden kann?“

1.4 Wissenschaftlicher Rahmen

1.4.1 Wissenschaftstheoretische Positionierung

Ausgangspunkt der wissenschaftstheoretischen Positionierung der Arbeit bildet deren Zuordnung zu einer Wissenschaftskategorie. Grundlegend können Wissenschaften in Formal- und Realwissenschaften unterteilt werden (Abbildung 1).

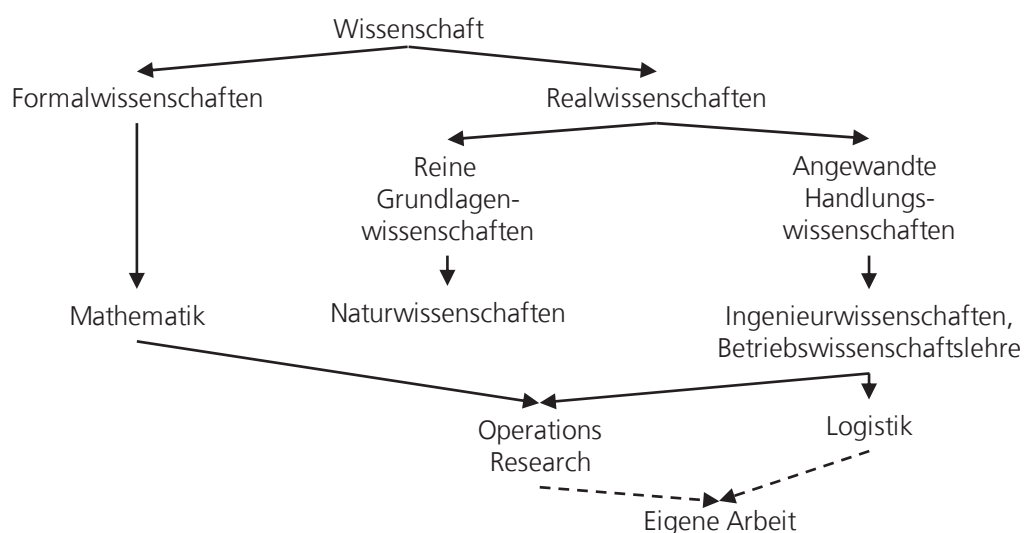


Abbildung 1: Wissenschaftssystematik (in Anlehnung an Ulrich et al. 1976, S. 305)

Unterscheidungsmerkmal ist die syntaktische Form der verwendeten Sätze (Schanz 1987, S. 2039). Die Sätze der *Formalwissenschaften* besitzen allein aufgrund ihrer Form Gültigkeit (Carnap 1931, S. 432-433). Aufgabe ist die logische Analyse der Sprache (Carnap 1931, S. 435). Formalwissenschaften sind nur auf ihre logische Wahrheit hin überprüfbar. Zu den Formalwissenschaften werden die Philosophie, die Logik und die Mathematik gezählt (Ulrich et al. 1976, S. 305). Die Sätze der

Realwissenschaften sind synthetisch (Schanz 1987, S. 2039). Sinnlich wahrnehmbare Wirklichkeitsausschnitte werden beschrieben, erklärt oder gestaltet (Ulrich et al. 1976, S. 305). Sie umfassen *Grundlagen-* und *Handlungswissenschaften*. Erstere versuchen Erklärungsmodelle für Wirklichkeitsausschnitte zu bilden (Ulrich et al. 1976, S. 305-306). Dagegen analysieren Handlungswissenschaften menschliche Handlungsalternativen mit dem Ziel, soziale und technische Systeme zu gestalten. Um dieses Ziel zu erreichen, werden Entscheidungsmodelle bzw. -prozesse entwickelt. Handlungswissenschaften umfassen neben den Sozialwissenschaften auch die Ingenieurwissenschaften (Ulrich et al. 1976, S. 305). Grundlagenwissenschaften verfolgen eine kognitive Zielsetzung: Man strebt nach Erkennen um des Erkennens willen (Schanz 1987, S. 2041). Dagegen ist die Handlungswissenschaft an praktisch verwertbarem Wissen interessiert (Ulrich 1984, S. 172).

Operations Research beschäftigt sich mit der Analyse von praxisnahen Problemen durch die Anwendung mathematischer Methoden (Domschke et al. 2015, S. 1). Ein wichtiges Merkmal ist die Interdisziplinarität, da Methoden und Anwendungen aus den Bereichen Mathematik, Wirtschaftswissenschaften, Informatik und Ingenieurwissenschaften von Bedeutung sind (Werners 2013, S. 1). Die *Logistik* betrachtet die Subsysteme Auftragsabwicklung, Lagerhaltung, Verpackung, Transport und Lagerhaus (Pfohl 2010, S. 68). Das Kommissionierlager ist ein Lagerhausbereich und damit Betrachtungsgegenstand der Logistik (Pfohl 2010, S. 118). Die Problemstellung der vorliegenden Arbeit ist den angewandten Handlungswissenschaften zuzuordnen. Die zur Lösung eingesetzten Methoden entstammen vorwiegend den Formalwissenschaften. Dem Verständnis Ulrichs folgend, wonach Realwissenschaften ohne Formalwissenschaften nicht auskommen (Ulrich et al. 1976, S. 306) und basierend auf dem erklärten Ziel der Arbeit, Entscheidungen zur Optimierung von Kommissioniersystemen zu unterstützen, ist die vorliegende Arbeit in erster Linie den Handlungswissenschaften zuzuordnen.

1.4.2 Forschungsmethodologie

Der *Forschungsprozess*, der dieser Arbeit zugrunde liegt, basiert auf dem explorativen Forschungszyklus von Tomczak (Abbildung 2), der als eigenständige Forschungsmethodologie in den Handlungswissenschaften verstanden wird (Kubicek 1977, S. 13; Ulrich 1984, S. 192; Tomczak 1992, S. 83).

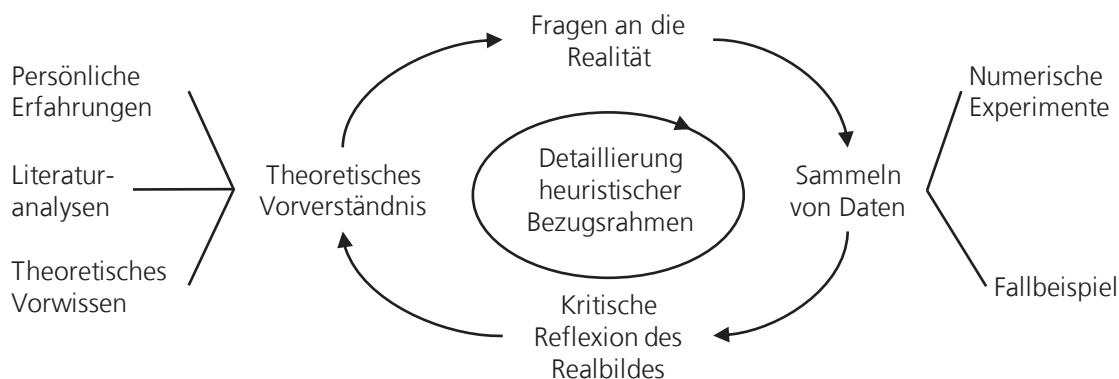


Abbildung 2: Forschungsmethodisches Vorgehen (in Anlehnung an Kubicek 1977, S. 14)

Forschung ist ein iterativer Lernprozess (Kubicek 1977, S. 13). Ausgangspunkt ist ein als unzureichend beherrscht angesehenes Phänomen der Praxis (Zohm 2004, S. 11), wie es in der Problemstellung in Kapitel 1.2 beschrieben wurde. Das theoretische Vorverständnis des Forschers formt dieses Problem und hilft ihm, es zu strukturieren (Tomczak 1992, S. 84). Verschiedene Theorien und erste Strukturierungen werden untersucht. Die Praxisrelevanz wird hinterfragt (Walter-Busch 1984, S. 227). Die theoretische Perspektive wird präzisiert. Eine genauere Definition des Problems wird so möglich. Passende Erklärungen und Anhaltspunkte für mögliche Lösungen werden sichtbar (Tomczak 1992, S. 84). Durch das wiederholte Stellen von Fragen an die Realität, dem Sammeln von Daten und der kritischen Reflexion des so gewonnenen Realitätsbildes wird Erfahrungswissen aufgebaut. Dies führt zur Formulierung weiterer Fragen (Kubicek 1977, S. 28). Den Kern des Forschungsprozesses bildet ein *heuristischer Bezugsrahmen*. Er expliziert das Vorverständnis des Forschers, steuert den Forschungsprozess und bietet eine Orientierungshilfe zur Lösung des Problems. Er umfasst die relevanten Analyseeinheiten sowie deren Zusammenhänge (Kubicek 1977, S. 16 und 18).

Theoretisches Vorwissen in den Bereichen Operations Research und Logistik eignete sich der Autor in seinem Studium an der Technischen Universität Darmstadt an. Dort lernte er die Arbeiten der Professoren Domschke (Operations Research) und Pfohl (Logistik) kennen. Beide gehören auf ihrem Gebiet zu den prägenden Persönlichkeiten (Nyhuis 2008, S. 4; Stadtler 2007). Bei Lehrveranstaltungen von Herrn Professor Glock kam der Autor zum ersten Mal in Kontakt mit dem Einsatz von Methoden des Operations Research zur Optimierung von Kommissioniersystemen. Wichtige persönliche Erfahrungen konnte der Autor am Institut für Industrielle Fertigung und Fabrikbetrieb sammeln. Hier bearbeitete er in Kooperation mit dem Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA zahlreiche industrienaher Forschungsprojekte. Den Schwerpunkt bildeten Projekte im Bereich Logistik. Für die vorliegende Arbeit wurden umfangreiche Analysen der relevanten Literatur durchgeführt. Die Sammlung von Daten und die anschließende Validierung, erfolgten im Rahmen numerischer Experimente und eines Fallbeispiels. *Numerische Experimente* eignen sich für einen Vergleich verschiedener Verfahren (Reschke 2013, S. 4). Einfache Verfahren können wertvolle Referenzen liefern, auch wenn diese nicht den „State-of-the-Art“ repräsentieren (Gibson et al. 1992, S. 64-65; Barr et al. 1995, S. 13). Sie ermöglichen es, verallgemeinerbare Aussagen über die Leistungsfähigkeit der Verfahren zu treffen. So kann eine fundierte Auswahl des geeigneten Verfahrens für einen konkreten Anwendungsfall erfolgen (Reschke 2013, S. 4). Weiterer Vorteil numerischer Experimente sind deren Nachvollziehbarkeit und Wiederholbarkeit (Reschke 2013, S. 108). Die Anwendung des entwickelten Verfahrens in einem realen Anwendungsfall erfolgte in Form eines *Fallbeispiels*. Mithilfe eines Fallbeispiels kann die praktische Relevanz eines Untersuchungsgegenstandes analysiert werden (Roll 2003, S. 315). Abbildung 3 zeigt den heuristischen Bezugsrahmen der Dissertation. Bestandteile sind die Person-zur-Ware-Kommissionierung und die operativen Planungsprobleme. Es gilt zu untersuchen, welche Auswirkung Interdependenz und Dynamik auf diese Objekte haben.

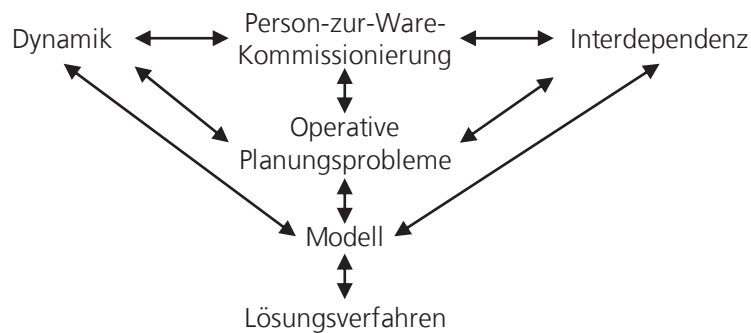


Abbildung 3: Heuristischer Bezugsrahmen

Das im vorausgegangenen Kapitel beschriebene Problem soll modelliert und mithilfe eines zu entwickelnden Verfahrens gelöst und damit die in Kapitel 1.3 formulierte Forschungsfrage beantwortet werden.

1.4.3 Aufbau der Arbeit

Der Aufbau der Arbeit (Abbildung 4) orientiert sich an den Phasen der angewandten Forschung nach Ulrich (2001, S. 195). In Kapitel 1 wurden die Problemstellung, die Zielsetzung der Arbeit und der wissenschaftliche Rahmen beschrieben. In Kapitel 2 werden der Betrachtungsgegenstand Kommissionierung sowie die operativen Planungsprobleme und deren Interdependenz untersucht. In Kapitel 3 werden die Grundlagen der Optimierung, der Prognose und der Dynamik beschrieben. In Kapitel 4 werden bestehende Ansätze analysiert, die mehr als eines der Planungsprobleme oder das dynamische Lagerplatzvergabeproblem lösen. In Kapitel 5 wird ein erster Entwurf des Lösungsverfahrens vorgestellt und jeweils ein Modell für das dynamische Lagerplatzvergabeproblem und das kombinierte Auftragsbildungs- und Tourenplanungsproblem formuliert. Die diskrete evolutionäre Partikelschwarmoptimierung sowie der dynamische Lagerplatzvergabealgorithmus werden in Kapitel 6 dargestellt. Die Validierung der Verfahren wird im Zuge umfangreicher numerischer Experimente in Kapitel 7 durchgeführt. Die praktische Relevanz wird in Kapitel 8 in Form eines Fallbeispiels nachgewiesen.

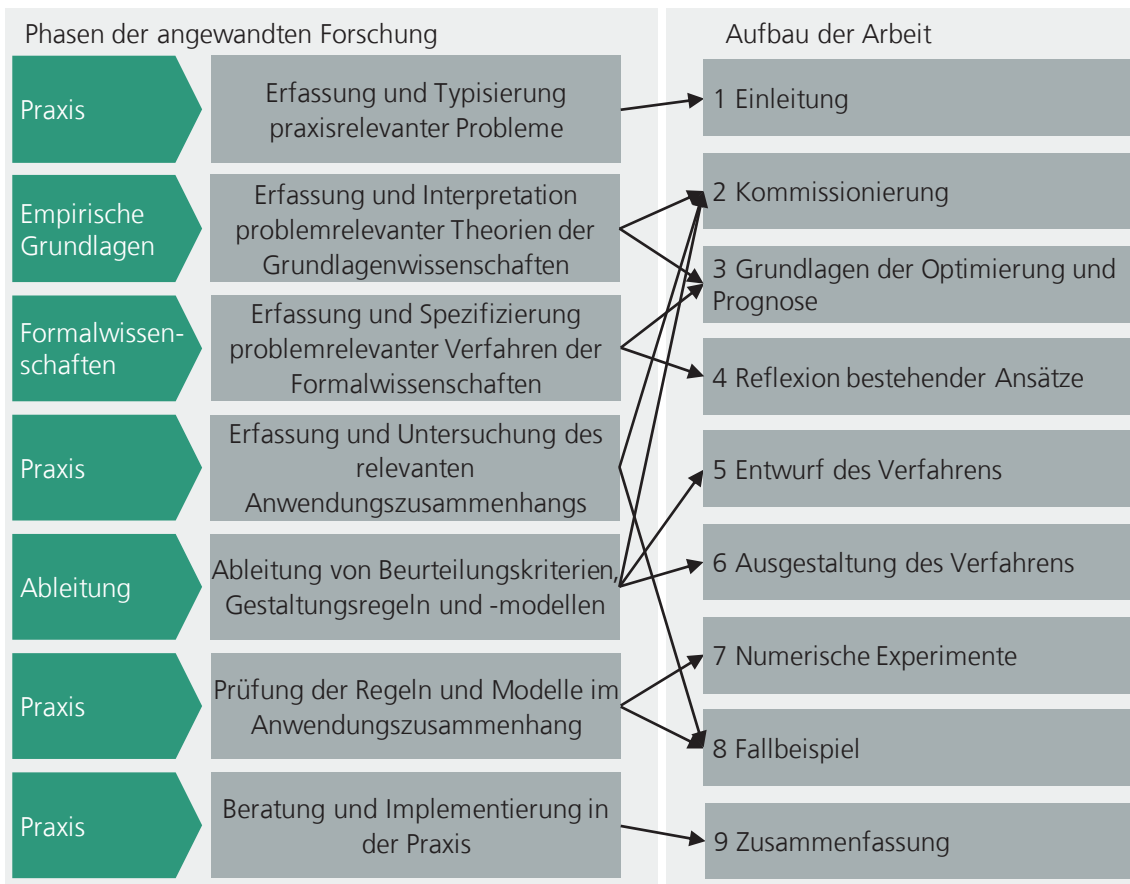


Abbildung 4: Aufbau der Arbeit (in Anlehnung an Ulrich 2001, S. 195)

Die Arbeit schließt mit einer Zusammenfassung der wesentlichen Erkenntnisse, deren Anwendungsmöglichkeiten in der Praxis und denkbaren Weiterentwicklungen in Kapitel 9.

2 Kommissionierung

Das Kapitel zur Kommissionierung untersucht die Grundlagen der Arbeit. Begonnen wird mit einer Diskussion zur Funktion des Lagers. Anschließend wird der Kommissioniervorgang beschrieben und unterschiedliche Kommissioniersysteme untersucht. Die Auseinandersetzung mit den grundlegenden Begriffen führt zur Eingrenzung des Anwendungszusammenhangs. Nach einer Analyse der operativen Planungsprobleme und deren Interdependenz werden die Auswirkungen auf die zu entwickelnden Verfahren untersucht. Ziel ist es, geeignete Betriebsstrategien und damit Gestaltungsregeln für das zu entwickelnde Verfahren zu identifizieren.

2.1 Lager

2.1.1 Definition, Wertschöpfung und Funktion

Definition

Lagern, also das „Aufbewahren und Bereithalten der Bestände einer Anzahl von Artikeln“ (Gudehus 2012, S. 613), geschieht in einem *Lager* bzw. *Lagerhaus* und stellt eine Zeitüberbrückung ankommender und abgehender Güter dar. Lagern ist eine geplante Unterbrechung des Materialflusses (Martin 2014, S. 336). Der Lagerprozess setzt sich aus den Teilprozessen Einlagern, Aufbewahren und Auslagern zusammen (Gudehus 2012, S. 613). Obwohl mit dem Lagern von Gütern häufig negative Effekte verbunden werden und der wertschöpfungslose Charakter der Zeitüberbrückung hervorgehoben wird, ist die Lagerung von Gütern in der Praxis nahezu unumgänglich (ten Hompel et al. 2010, S. 3).

Wertschöpfung

Dieses Verständnis von logistischen Aktivitäten resultiert aus der verbreiteten Ansicht, dass lediglich Produktionsprozesse wertschöpfend sind (Adolph et al. 2016, S. 563). Ausgehend von der Annahme, dass jeder Wirtschaftsprozess der Befriedi-

gung von Bedürfnissen von Menschen dient, lässt sich auch ein anderes *Wertschöpfungsverständnis* rechtfertigen: Bedürfnisse werden nicht allein durch die wirtschaftliche Tätigkeit, sondern erst durch die von ihr ausgelöste Nutzenstiftung befriedigt. Gemäß Abbildung 5 können fünf Nutzenarten unterschieden werden, die durch wirtschaftliche Tätigkeiten entstehen können: Gestalt-, Informations-, Ort-, Zeitnutzen und der Nutzen aus dem Recht am Gut (Pfohl 2010, S. 20-21). Die letzten vier Nutzenarten können unter dem Begriff „Servicewert“ subsumiert und der Gestaltnutzen als „Produktwert“ bezeichnet werden. Der Produktwert wird durch die Produktionsprozesse beeinflusst (Adolph et al. 2016, S. 563-564). Der Ort- und Zeitnutzen sowie teilweise auch der Informationsnutzen entsteht im Logistiksystem. Da der Zweck von Unternehmen in der Wertschöpfung besteht, also der Erzeugung von Gütern mit Werteigenschaften, die die Bedürfnisse ihrer Kunden befriedigen, können logistische Aktivitäten nicht per se als Verschwendung deklariert werden.

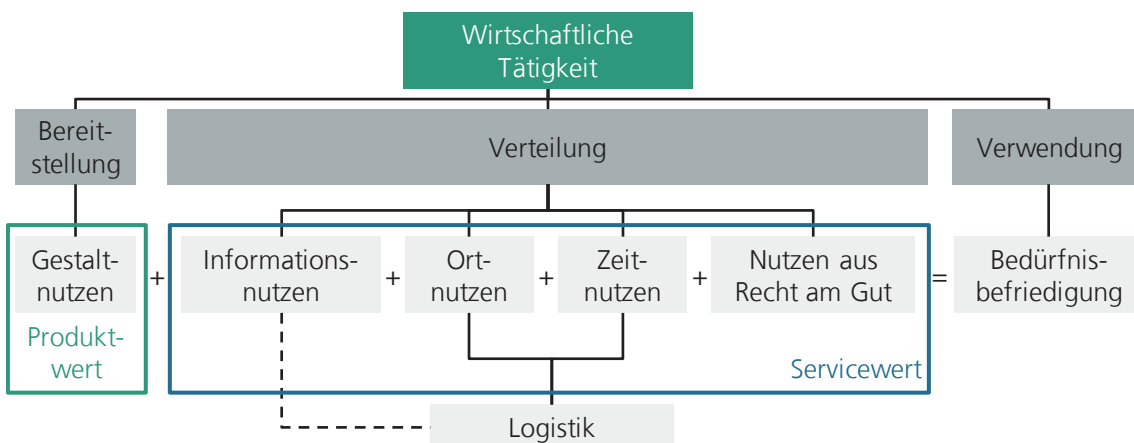


Abbildung 5: Wertschöpfungsverständnis (in Anlehnung an Pfohl 2010, S. 21 und Adolph et al. 2016, S. 564)

Die Kundennachfrage gilt erst als erfüllt, wenn alle fünf Nutzenarten in ausreichendem Maß befriedigt werden konnten. Eine in Stuttgart hergestellte Maschine (Gestaltnutzen) kann in München erst dann ein Bedürfnis befriedigen, wenn der potentielle Kunde weiß, dass es die Maschine gibt (Informationsnutzen), er sie in München (Ortsnutzen) zu dem von ihm gewünschten Zeitpunkt nutzen kann (Zeitnutzen) und er die notwendigen Rechte zur Nutzung der Maschine besitzt (Nutzen aus dem

Recht am Gut). Logistik ist demnach nicht einfach nur eine notwendige Verschwendung, sondern kann wertschöpfend sein (Pfohl 2010, S. 20-21). Diesem Wertschöpfungsverständnis folgend, lässt sich auch das Vorhandensein von Lagern rechtfertigen (Adolph et al. 2016, S. 563-564).

Funktion

Das Lager nimmt eine Schlüsselstellung innerhalb der Logistikkette ein (Pfohl 2010, S. 113), da es die Diskontinuität des Materialflusses glättet (Manzini 2012, S. vii) und in seiner Funktion als Auslieferungslager den Güterfluss vom Lieferanten zum Kunden aufspaltet (Pfohl 2010, S. 113). Bestände kommen in Form von Roh-, Hilfs- und Betriebsstoffen sowie Zwischen-, Halb- und Fertigprodukten entlang der gesamten Wertschöpfungskette vor. Sie bilden einen Puffer zwischen den Input- und Outputflüssen eines Unternehmens, wenn diese zeitlich und/oder quantitativ voneinander abweichen. Demnach kann nur auf Bestandshaltung verzichtet werden, wenn alle Input- und Outputströme vollständig miteinander synchronisiert werden können (Koch 2012, S. 34). Lagerbestände erleichtern die Spezialisierung der Produktion in verschiedenen Werken bzw. Unternehmen entlang der Wertschöpfungskette (Koch 2012, S. 35). Bestände können außerdem durch Rüstanforderungen, die eine Bildung von Produktionslosen erforderlich machen, entstehen (Erlach 2010, S. 18). Auch eröffnen Bestände Potentiale für kurzfristige Reaktionen und damit eine schnelle Belieferung der Kunden (Wiendahl 2011, S. 147). Notwendig wird die Bildung von Beständen häufig, wenn Güter transportiert werden müssen, da Lade- und Raumkapazitäten möglichst effizient ausgelastet werden sollen (ten Hompel et al. 2010, S. 3-4).

2.1.2 Lagerarten und Lagerhausfunktionen

Lagerarten

Nach ihrer Funktion im Wertschöpfungsprozess können Beschaffungs-, Produktions- und Distributionslager unterschieden werden. *Beschaffungslager* versorgen die Produktion mit Material und *Distributionslager* die Kunden mit den Erzeugnissen des Unternehmens. *Produktionslager* nehmen Zwischenerzeugnisse vorausgehender Fertigungsschritte auf und versorgen nachfolgende Schritte (Martin 2014, S. 338). Kommissioniert wird entweder für einen Kundenauftrag (Distributionslager) oder für Fertigungs- bzw. Montageaufträge (Beschaffungs- und Produktionslager) (Martin 2014, S. 396). Das Verfahren dieser Arbeit kann in allen drei Lagerarten eingesetzt werden.

Lagerhausfunktionen

Die Lager- und Bewegungsprozesse werden durch eine Reihe von Lagerhausfunktionen realisiert (siehe Abbildung 6).

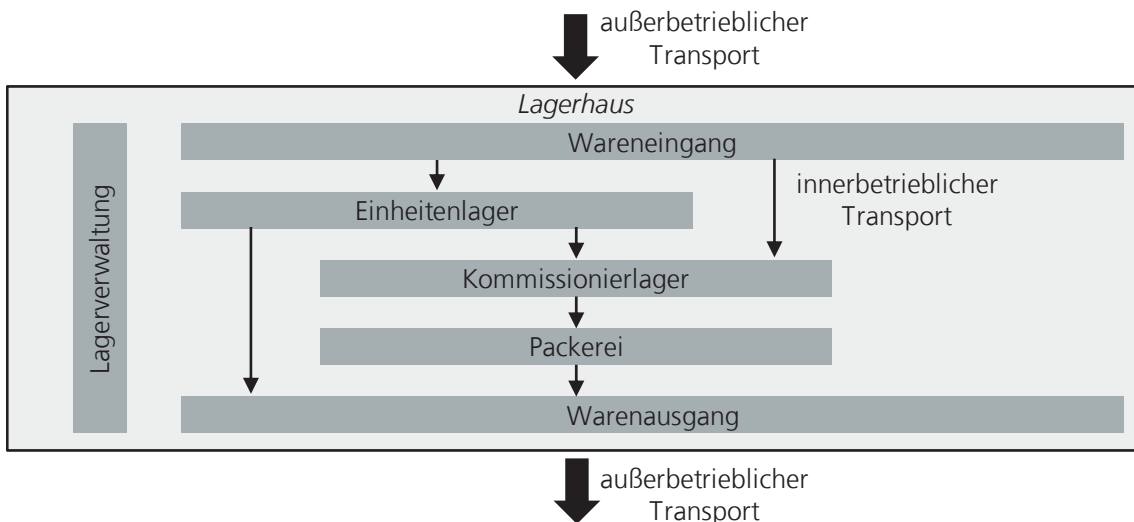


Abbildung 6: Lagerhausfunktionen (Pfohl 2010, S. 118)

Im *Wareneingang* wird das Transportmittel entladen, die Produkte entgegengenommen, das Bestandsverzeichnis aktualisiert und die Wareneingangsprüfung durchgeführt (de Koster et al. 2007, S. 483). Über den innerbetrieblichen Transport werden

die Produkte in das *Einheitenlager* gebracht. Dort werden die Güter zur Zeitüberbrückung eingelagert und in derselben logistischen Einheit ausgelagert. Aufgrund der Gleichförmigkeit der Tätigkeiten sind diese Lager in hohem Maße automatisierbar. Gehen die Einheiten nicht direkt zum Warenausgang, sondern in ein Kommissionierlager, wird das Einheitenlager auch als Reservelager bezeichnet. Während die Produkte dort in großen Mengen über einen relativ langen Zeitraum gelagert werden, werden die Güter im *Kommissionierlager* nur über eine kurze Dauer in kleinen Mengen und Einheiten gelagert (Pfohl 2010, S. 118-119). Das Kommissionieren erfolgt in den meisten Fällen manuell (Behnisch et al. 2017, S. 64-72; de Koster et al. 2007, S. 485). Nach dem Verlassen des Kommissionierlagers müssen die Artikel eines Auftrags entsprechend der Kundenspezifikationen in der *Packerei* verpackt und auf geeignete Ladungsträger (wie z.B. Paletten) gestapelt und zum *Warenausgang* gebracht werden (de Koster et al. 2007, S. 483). Die *Lagerverwaltung* steuert und koordiniert die Lager- und Bewegungsprozesse innerhalb des Lagerhauses und bildet damit die Schnittstelle zur Auftragsabwicklung (Pfohl 2010, S. 119).

2.2 Kommissionierung als Lagerhausfunktion

2.2.1 Definition

Die Kommissionierung ist die wichtigste Lagerhausfunktion und hat laut VDI-Richtlinie 3590 (VDI 1994, S. 2) „[...] das Ziel, aus einer Gesamtmenge von Gütern (Sortiment) Teilmengen auf Grund von Anforderungen (Aufträge) zusammenzustellen“. Die Notwendigkeit hierfür entsteht, wenn Artikel sortenrein in großer Zahl im Wareneingang von den Zulieferern oder der eigenen Produktion angeliefert werden, die Kunden bzw. nachfolgende Fertigungsstufen aber kleine Mengen unterschiedlicher Artikel benötigen (Wäscher 2004, S. 324). Das Kommissionieren ist die komplexeste Aufgabe der innerbetrieblichen Logistik. Kommissioniersysteme können mithilfe einer Vielzahl von Verfahren, Techniken und Kombinationen verschiedener Strategien aufgebaut und organisiert werden (Gudehus 2012, S. 707).

2.2.2 Kommissionierziele

Lieferservice und Logistikkosten sind die zwei Ziele der Logistik und damit auch der Kommissionierung (siehe Abbildung 7).

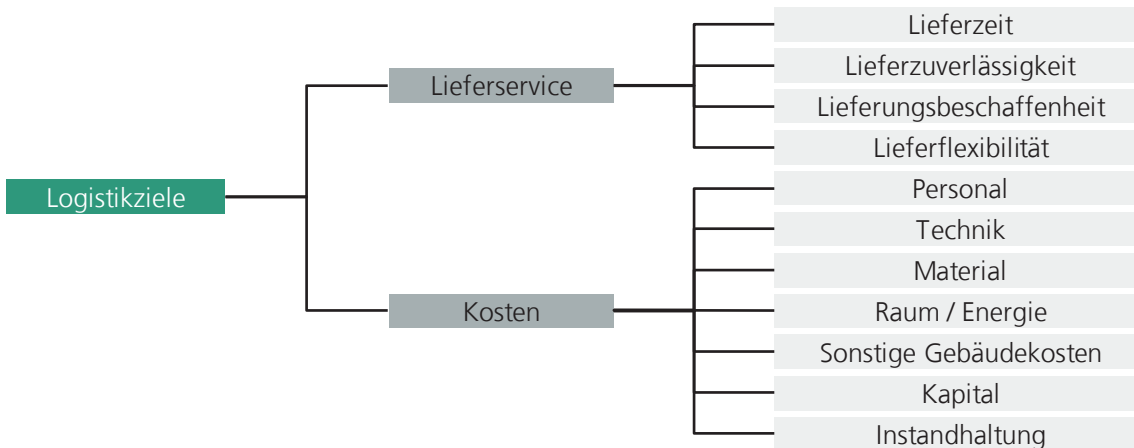


Abbildung 7: Logistikziele (in Anlehnung an Pfohl 2010, S. 35-37 und ten Hompel et al. 2011, S. 213)

Der *Lieferservice* wird durch die Faktoren Lieferzeit, Lieferzuverlässigkeit, Lieferungsbeschaffenheit und Lieferflexibilität bestimmt (Pfohl 2010, S. 37). Ziel der Kommissionierung ist es, den Lieferservice bei gleichzeitiger Minimierung der Kommissionierkosten zu maximieren. Beide Ziele lassen sich durch eine Reduzierung der Auftragsdurchlaufzeiten realisieren. So ermöglichen kürzere Durchlaufzeiten kürzere Lieferzeiten (de Koster et al. 2007, S. 486) und erhöhen die Flexibilität für die Annahme kurzfristig eintreffender Aufträge bzw. Auftragsänderungen (Wiendahl 2011, S. 3). Gleichzeitig sinkt der Umlaufbestand (Wiendahl 2011, S. 15), was einen positiven Effekt auf die Kapitalkosten hat. Lange Durchlaufzeiten, die aufgrund schlecht geplanter oder ausgeführter Prozesse entstehen, führen zu erhöhtem Personalaufwand. Eine Optimierung der Kommissionierprozesse hat damit eine positive Wirkung auf die Personalkosten (Koch 2014, S. 13). Zur Reduktion der Auftragsdurchlaufzeit gibt eine Analyse der Arbeitszeit eines Kommissionierers erste Hinweise. Gemäß einer Analyse (Abbildung 8) von Tompkins et al. (2010, S. 434) verbringt ein Kommissionierer 10 % seiner Zeit damit, die Kommissionierung vorzubereiten, indem er Kommissionierlisten druckt, die Listen sortiert oder Etiketten vorbereitet.

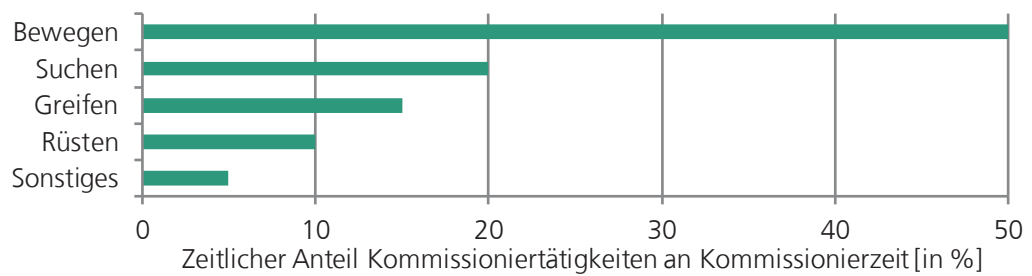


Abbildung 8: Bestandteile der Kommissionierzeit (Tompkins et al. 2010, S. 434)

In Analogie zu der Vorbereitung von Maschinen wird von *Rüstzeit* gesprochen (Yu et al. 2008, S. 1055). Nach diesen vorbereitenden Tätigkeiten bewegt sich der Kommissionierer vom Depot zu den Lagerplätzen, die zu kommissionierende Artikel enthalten (Grosse et al. 2013, S. 655). Die *Bewegungszeit* ist mit 50 % der größte Zeitbestandteil. Am Regal angekommen, muss der Kommissionierer das gesuchte Teil identifizieren (Grosse et al. 2013, S. 655). Die *Suchzeit* kann bis zu 20 % der Zeit in Anspruch nehmen. Das *Greifen* umfasst nur 15 % der Kommissionierzeit. Zeitbestandteile wie die Greif- und Suchzeit sind nicht oder nur mit hohem Aufwand beeinflussbar oder aber, wie im Fall der Rüstzeit und der sonstigen Zeiten vernachlässigbar. Damit kommt der Bewegungszeit die zentrale Bedeutung bei der Reduzierung der Kommissionierzeit und damit der Auftragsdurchlaufzeit in der Kommissionierung zu (Henn et al. 2010, S. 85). Da angenommen werden kann, dass die Bewegungszeit bei konstanter Bewegungsgeschwindigkeit eine lineare Funktion der Weglänge ist, entspricht das Ziel der Bewegungszeitreduktion dem der Minimierung der Weglänge einer Kommissioniertour (Jarvis et al. 1991, S. 94). Folglich ist die Minimierung der Kommissioniertourlänge die zentrale Optimierungsgröße (siehe unter anderem Petersen et al. 1999, S. 483; Rao et al. 2013b, S. 4979; Caron et al. 1998, S. 714; Meneghetti et al. 2013, S. 304; Le-Duc et al. 2005, S. 3562; de Koster et al. 2007, S. 486). Das Optimierungsziel der zu entwickelnden Verfahren besteht demnach darin, die gesamte Länge aller Kommissioniertouren (Gesamttourlänge), bei gegebenem Bestand an Aufträgen zu minimieren (Koch 2014, S. 15).

2.3 Kommissioniersysteme

2.3.1 Klassifikation

Es werden manuelle und automatische Kommissioniersysteme unterschieden (siehe Abbildung 9).

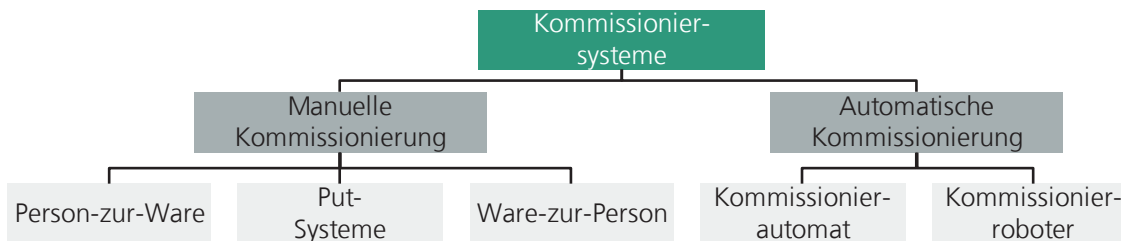


Abbildung 9: Klassifikation Kommissioniersysteme (in Anlehnung an de Koster et al. 2007, S. 484)

Automatische Systeme kommen als Automaten (sogenannte Dispenser) oder Roboter vor. Vollständig automatisierte Systeme werden selten eingesetzt (de Koster et al. 2007, S. 485). Manuelle Kommissioniersysteme sind entweder nach dem Prinzip Person-zur-Ware oder Ware-zur-Person organisiert bzw. nehmen eine Zwischenform in Form eines Put-Systems ein. Bei dem Prinzip Ware-zur-Person wird der von einem Kommissionierer angeforderte Artikel auf einer Palette oder in einem Behälter an die Kommissionierstation befördert und dort durch den Kommissionierer entnommen. Der Ein- und Auslagerungsprozess wird z. B. von einem automatischen Regalbediengerät oder einem Paternostersystem durchgeführt (de Koster et al. 2007, S. 484). Put-Systeme sind eine Mischform der Person-zur-Ware- und der Ware-zur-Person-Systeme. Zu kommissionierende Artikel werden entweder über ein Person-zur-Ware- oder ein Ware-zur-Person-System kommissioniert und anschließend auf verschiedene Behälter verteilt, die jeweils einem Auftrag zugeordnet sind (de Koster et al. 2007, S. 483-484). Das Prinzip Person-zur-Ware wird im nächsten Unterkapitel aufgrund seiner Bedeutung für diese Arbeit eingehender untersucht.

2.3.2 Manuelle Person-zur-Ware-Kommissioniersysteme

Bei den Person-zur-Ware-Systemen geht oder fährt ein Kommissionierer Gänge entlang und entnimmt Artikel aus Lagerplätzen, meist aus Regalen. Die Artikel werden

entweder in Greifhöhe des Kommissionierers (sog. *Low-Level-Kommissionierung*) oder in Hochregalen (sog. *High-Level-Kommissionierung*) gelagert. Im zweiten Fall sind zusätzliche Betriebsmittel wie Gabelstapler oder Man-on-Board-Systeme notwendig. Laut einer Expertenschätzung sind 80 % der Kommissioniersysteme in westeuropäischen Lagerhäusern nach dem Prinzip der manuellen Person-zur-Ware Low-Level-Kommissionierung organisiert (de Koster et al. 2007, S. 485). Der Grund hierfür ist, dass Menschen aufgrund ihrer kognitiven Fähigkeiten flexibel und in Echtzeit auf Änderungen reagieren können (Grosse et al. 2013, S. 653). Diese Flexibilität ist wichtig, da sich die Artikel häufig stark in Form und Größe unterscheiden und die Nachfrage Schwankungen unterliegt (Petersen et al. 2004a, S. 11). Es ist davon auszugehen, dass aufgrund der zunehmenden Dynamik der Märkte und der Vielfalt zu kommissionierender Artikel auch in Zukunft hauptsächlich Menschen zur Kommissionierung eingesetzt werden (Möller 2011, S. 177). Person-zur-Ware-Systeme haben einen minimalen technischen Aufwand und kurze Auftragsdurchlaufzeiten, sie bieten die Möglichkeit, verschiedene Auftragsarten gleichzeitig zu bearbeiten und sind hochgradig flexibel gegenüber schwankenden Durchsatzanforderungen und Sortimentsveränderungen (Gudehus 2012, S. 717).

Layout

Es gibt unterschiedliche Möglichkeiten, die Regale im Kommissionierlager anzuordnen. Das in Abbildung 10 oben links dargestellte *Ein-Block-Layout* besteht aus einem rechteckigen Kommissionierbereich mit mehreren zueinander parallelen Regalreihen, zwischen denen jeweils ein Kommissioniergang verläuft.

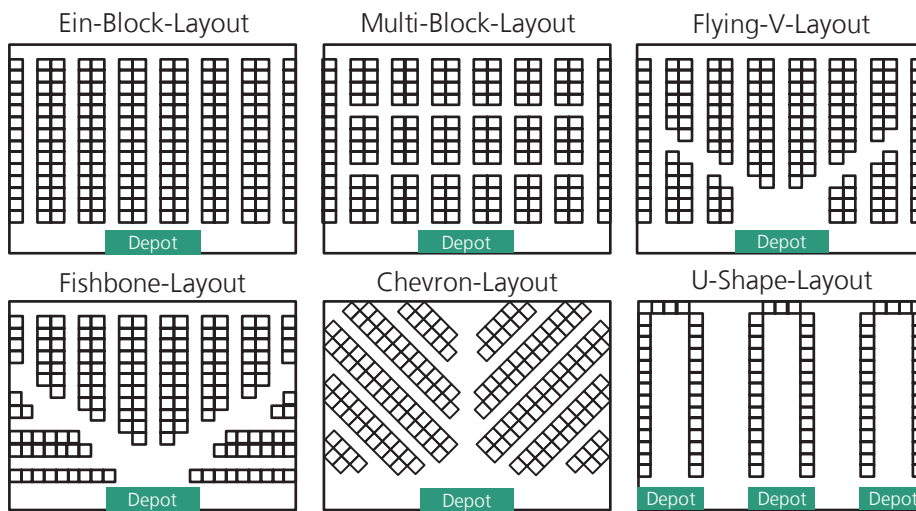


Abbildung 10: Layoutvarianten

Zu beiden Enden der Kommissioniergänge verläuft ein Quergang (Wäscher 2004, S. 325). Oft sind alle Regalreihen und Kommissioniergänge gleich groß. Über das Depot wird das Lager betreten und es werden kommissionierte Artikel abgelegt. Kommissioniergänge können über den vorderen oder den hinteren Quergang gewechselt werden (Henn et al. 2011, S. 2-3). Die Gänge sind breit genug für zwei Kommissionierwagen, so dass in beide Richtungen kommissioniert bzw. überholt werden kann, ohne dass es zu einer Blockade kommt. Kommissioniert werden kann auf beiden Seiten des Gangs, ohne die Position wesentlich zu verändern. Die Gänge sind schmal genug, um die horizontale Bewegung des Kommissionierers vernachlässigen zu können (Wäscher 2004, S. 326). Dieses Layout wird in der Literatur als Basislayout bezeichnet (Roodbergen et al. 2001b, S. 32). Durch das Einfügen weiterer Quergänge wird aus dem Ein-Block- ein *Multi-Block-Layout* (Abbildung 10 Mitte oben). Diese sollen den Kommissionierern den Wechsel zwischen den Kommissioniergängen erleichtern, um so die Kommissioniereffizienz zu erhöhen (Koch 2014, S. 11). Neben den klassischen Blocklayouts werden in der Literatur weitere Layoutvarianten wie beispielsweise das Flying-V- (Abbildung 10 rechts oben), das Fishbone- (Abbildung 10 links unten), das Chevron- (Abbildung 10 Mitte unten) (Meller et al. 2009, S. 2-4) oder das U-Shape-Layout (Abbildung 10 rechts unten) (Glock et al. 2011, S. 65-66) beschrieben.

In manuellen Kommissioniersystemen werden häufig Fachbodenregale eingesetzt. Da dieser Regaltyp einen durchgehenden Regalboden besitzt, können Artikel mit unterschiedlichen Abmessungen eingelagert werden. Höhe und Breite der Fächer können durch unterschiedliche Formen der Fachteilung flexibel aufgebaut werden und eignen sich damit besonders gut für die Kommissionierung verschiedenartiger Güter. Die Regalhöhe beträgt in der Regel zwei Meter, um ohne Hilfsmittel wie etwa Leitern kommissionieren zu können (ten Hompel et al. 2010, S. 76). Kommissioniersysteme, bei denen die vertikale Bewegung des Kommissionierers im Vergleich zur horizontalen Bewegung vernachlässigt werden kann, dominieren in der Praxis (Rao et al. 2013a, S. 1309). Aufgrund des breiten potentiellen Anwenderkreises stehen diese Systeme im Fokus der vorliegenden Dissertation. Das zu entwickelnde Verfahren soll auch für Multi-Block-Layouts einsetzbar sein, da dort kürzere Kommissionierwege realisiert werden können und der potentielle Anwenderkreis des Verfahrens weiter ausgebaut werden kann. Daher sind manuelle Person-zur-Ware Low-Level-Kommissioniersysteme mit Blocklayout der Anwendungszusammenhang dieser Arbeit.

Kommissionierauftrag

Artikel, die gemeinsam kommissioniert werden sollen, werden in einer *Kommissionierliste* bzw. einem *Kommissionierauftrag* zusammengefasst. Sie bestehen meist aus mehreren Auftragszeilen bzw. *Positionen*, die jeweils den benötigten Artikel mit der zugehörigen *Menge* enthalten. Ein Kommissionierauftrag enthält entweder nur Artikel von einem Kunden- oder Fertigungsauftrag (Pick-by-Order) oder eine Kombination aus Artikeln verschiedener Aufträge in einem *Batch* (Pick-by-Batch). Der Kommissionierauftrag gibt vor, aus welchem Lagerplatz die zu kommissionierenden Artikel zu entnehmen sind (Chen et al. 2015, S. 158). Die Reihenfolge der Zeilen entspricht der Reihenfolge, in der die Lagerorte besucht werden sollen. Der Kommissionierer wird mithilfe des Kommissionierauftrags durch das Lager geführt (Scholz et al. 2015, S. 5).

2.4 Planungsprobleme der Kommissionierung

2.4.1 Strategische und operative Planungsprobleme

Damit ein Kommissioniersystem eine hohe Leistung zu niedrigen Kosten erbringen kann, sind einige wichtige Entscheidungen zu treffen. Die entsprechenden Planungsprobleme haben, wie in Abbildung 11 dargestellt, entweder einen strategischen oder einen operativen Charakter.

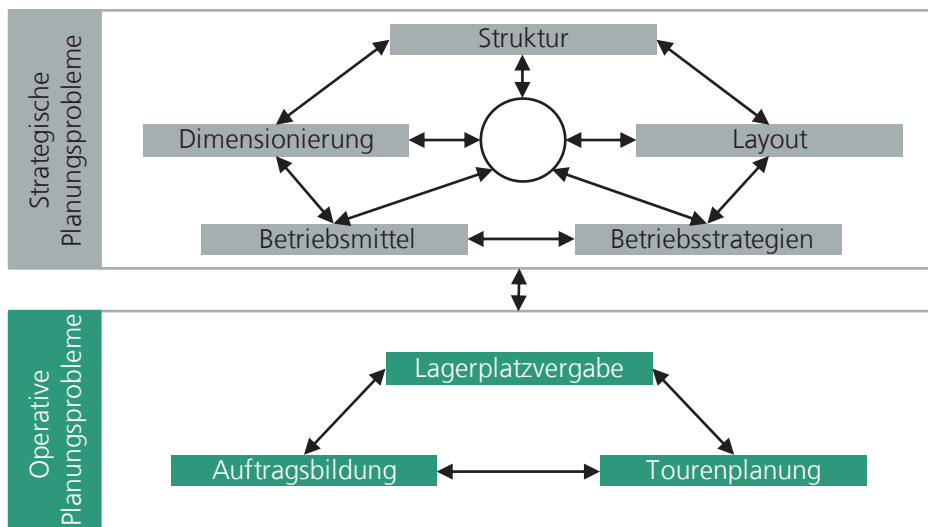


Abbildung 11: Planungsprobleme (in Anlehnung an Gu et al. 2007, S. 2 und de Koster et al. 2007, S. 485)

Im Zuge der *strategischen Planungsprobleme* werden die *Lagerstruktur* und damit der Materialfluss festgelegt, die Funktionsbereiche definiert sowie die relative Lage der Funktionsbereiche zueinander bestimmt (Gu et al. 2007, S. 3). Ein geeigneter Kommissioniersystemtyp muss festgelegt werden. Je nach Anforderungen an das Kommissioniersystem kommen dabei entweder Person-zur-Ware-, Ware-zur-Person- oder vollautomatische Systeme in Frage. Bei der *Dimensionierung* wird die notwendige Gesamtkapazität des Kommissionierlagers bestimmt und der ermittelte Kapazitäts- in einen Flächenbedarf umgerechnet (Gu et al. 2010, S. 540-541). Die Festlegung des *Layouts* umfasst die Einplanung der Funktionsbereiche innerhalb des Lagergebäudes und das Layout des Kommissionierlagers (de Koster et al. 2007,

S. 487). *Betriebsmittel* sind etwa in Form von Lagermitteln und Materialbediengeräten auszuwählen. Die *Betriebsstrategien* (Abbildung 12) geben den Rahmen für die operativen Planungsprobleme vor.

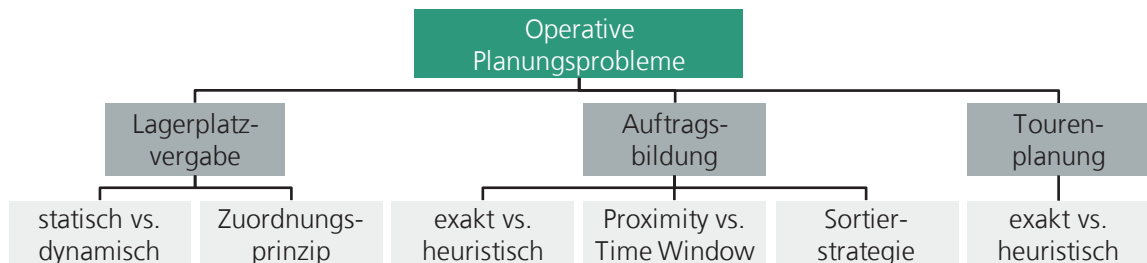


Abbildung 12: Betriebsstrategien

In der strategischen Planung wird entschieden, welche Betriebsstrategie eingesetzt werden soll. Im Zuge der operativen Planung erfolgt deren Ausgestaltung (Gu et al. 2007, S. 3-4). Auf operativer Ebene werden die Lagerplatzvergabe, die Auftragsbildung und die Tourenplanung gelöst. Die Probleme und Betriebsstrategien werden im Folgenden beschrieben und Lösungsverfahren aufgezeigt.

2.4.2 Lagerplatzvergabe

Das *Lagerplatzvergabeproblem* wird wie folgt definiert: „Wie sollen die Artikel (bei gegebenem Lagerlayout, gegebenen Lagerplätzen und gegebenen Aufträgen) den Lagerplätzen zugeordnet werden, so dass die für die Bearbeitung aller Aufträge notwendige Gesamttourlänge minimiert werden kann?“ (Wäscher 2004, S. 330). Das zu lösende Lagerplatzvergabeproblem kann als *quadratisches Zuordnungsproblem* modelliert werden (Wäscher 2004, S. 330-331). Entscheidungen zur Lagerplatzvergabe beeinflussen nahezu alle wesentlichen Lagerhausziele, wie die Kommissionierzeit und die Kommissionierkosten oder den Flächennutzungsgrad (Muppani et al. 2008, S. 492). Erfolgt die Lagerplatzvergabe ohne einen systematischen Ansatz, besteht die Gefahr, dass die Effizienz des Kommissionierlagers und der Lagernutzungsgrad sinken und die Kosten steigen (Choy et al. 2014, S. 111). Einer Schätzung zufolge haben die meisten Lagerhäuser Mehrkosten in Höhe von bis zu 30 %, weil weniger als 15 % der Artikel effizient gelagert werden (Petersen et al. 2004b,

S. 997). Das Lagerplatzvergabeproblem ist in der Regel NP-schwer, weshalb für praxisrelevante Probleminstanzen keine optimalen Lösungsverfahren in Frage kommen (Pan et al. 2009, S. 261). Daher werden heuristische Verfahren eingesetzt (Wäscher 2004, S. 331). Bei steigender Lösungsqualität der Verfahren, steigt auch die notwendige Rechenzeit. Allerdings spielt die Rechenzeit eine untergeordnete Rolle, da das Lagerplatzvergabeproblem in größeren Zeitintervallen gelöst werden muss (Wäscher 2004, S. 336). Es existieren verschiedene *Zuordnungsprinzipien* (Koch 2014, S. 16). Diese beinhalten eine Reihe von Regeln, die festlegen, wie Artikel den vorhandenen Lagerplätzen zugeordnet werden sollen (Bindi et al. 2007, S. 2). Man unterscheidet die feste, die zufällige und die klassenbasierte Lagerplatzvergabe sowie das Family Grouping (de Koster et al. 2007, S. 488-491).

Zufällige Lagerplatzvergabe

Bei der *zufälligen bzw. chaotischen* Lagerplatzvergabe wird jeder ankommende Artikel einem freien Lagerplatz zugeordnet, der zufällig aus der Menge aller freien Lagerplätze ausgewählt wird (Roodbergen 2001, S. 14). Die zufällige Lagerplatzvergabe ist einfach umzusetzen und führt zu einem höheren Lagernutzungsgrad als andere Zuordnungsprinzipien (Petersen et al. 2004a, S. 12). Sie erzeugt aber die längsten Kommissionierwege (Rao et al. 2013a, S. 1309), weil auch häufig zu kommissionierende Artikel am entferntesten Punkt zum Depot gelagert werden können (Garfinkel 2005, S. 1-2). Die zufällige Lagerplatzvergabe wird häufig in der Praxis eingesetzt und in der Literatur als Benchmark für die Bewertung der Leistungsfähigkeit anderer Zuordnungsprinzipien eingesetzt (Kofler et al. 2011, S. 77).

Feste Lagerplatzvergabe

Bei der *festen Lagerplatzvergabe* wird jedem Artikel ein fester Lagerplatz zugewiesen (de Koster et al. 2007, S. 489). Meist erfolgt die Vergabe der Lagerplätze nach der Nachfrageintensität der Artikel unter Berücksichtigung der Entfernung des Lagerorts zum Depot. Artikel mit einer hohen Nachfrageintensität werden nahe zum Depot gelagert (Mantel et al. 2007, S. 302-303). Diese Strategie führt im Vergleich

zu den anderen Zuordnungsprinzipien zu den kürzesten Kommissionierwegen (Petersen 1999, S. 1053-1054). Eine feste Lagerplatzvergabe kann zu einem niedrigen Lagernutzungsgrad führen, weil Lagerplätze auch dann für Artikel reserviert werden, wenn diese nicht auf Lager liegen (de Koster et al. 2007, S. 489). Schwierigkeiten entstehen bei einer festen Lagerplatzvergabe, wenn sich das Artikelsortiment und die Nachfrageintensität der Artikel stark ändern, da die Artikel häufig umgelagert werden müssen (Le-Duc et al. 2005, S. 3563).

Klassenbasierte Lagerplatzvergabe

Bei der *klassenbasierten Lagerplatzvergabe* wird das Lager in verschiedene Zonen eingeteilt, denen die Artikel fest zugewiesen werden. Innerhalb der Zonen werden die Artikel meist zufällig gelagert (Mantel et al. 2007, S. 303). Die Artikelklassen werden aufgrund gemeinsamer Produktcharakteristika wie der Kommissionierhäufigkeit gebildet. Die Wegzeitvorteile der festen Lagerplatzvergabe können so mit dem höheren Lagernutzungsgrad der zufälligen Lagerplatzvergabe kombiniert werden (Chiang et al. 2011, S. 221). Die zufällige und die feste Lagerplatzvergabe sind Extremfälle der klassenbasierten Lagerplatzvergabe: Bei der zufälligen Lagerplatzvergabe besteht das Lager aus nur einer Klasse, wohingegen bei der festen Lagerplatzvergabe jeder Artikel seine eigene Klasse bildet (van den Berg 1999, S. 754). In einer Ausprägung der klassenbasierten Lagerplatzvergabe wird das Lager in drei Klassen eingeteilt und die Artikel der Klasse A, B oder C zugewiesen. Die Schnelldreher werden der A-Klasse und die Langsamdreher der C-Klasse zugeteilt. Artikel der Klasse A werden nah zum Depot gelagert (de Koster et al. 2007, S. 489). Die klassenbasierte Lagerplatzvergabe wird häufig in der Praxis eingesetzt, weil sie relativ einfach umsetzbar ist. Häufige Sortiments- bzw. Nachfrageänderungen bereiten weit weniger Probleme als bei der festen Lagerplatzvergabe (Le-Duc et al. 2005, S. 3562). Es gibt verschiedene Möglichkeiten, die Artikelklassen im Lager anzuordnen. Abbildung 13 zeigt die bekanntesten Artikelanordnungsmuster. Bei der *Radialanordnung* entscheidet die Entfernung eines Lagerplatzes zum Depot über die Zu-

ordnung zu einer Klasse. Bei der *Längsanordnung* ist die Nähe der Kommissioniergänge zum Depot und bei der *Queranordnung* die Nähe der Quergänge zum Depot entscheidend.

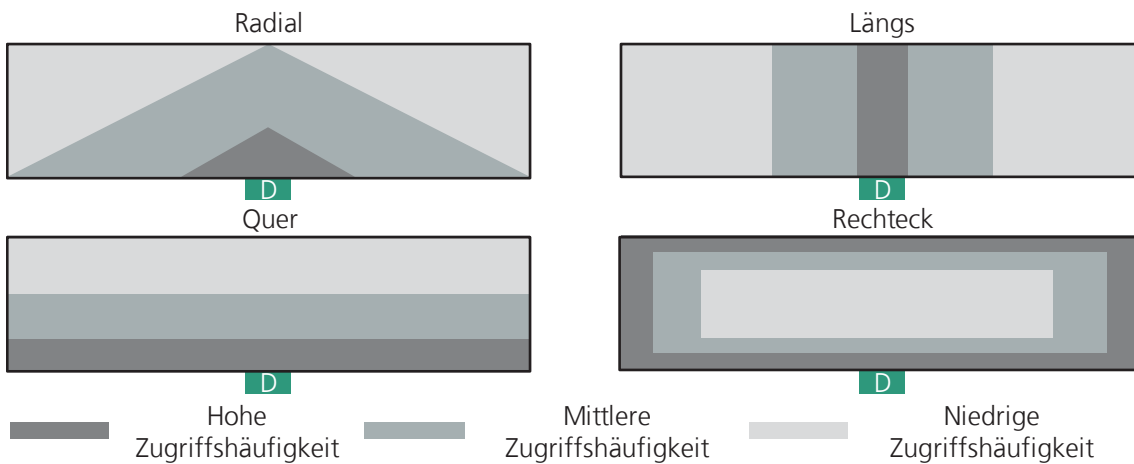


Abbildung 13: Artikelanordnungsmuster (in Anlehnung an Petersen et al. 1999, S. 489)

Die Lagerplätze einer Klasse liegen bei der *Rechteckanordnung* in einem Rechteck um das Zentrum des Lagers (Petersen et al. 1999, S. 488-490).

Family Grouping

Die bislang beschriebenen Zuordnungsprinzipien berücksichtigen keine Abhängigkeiten zwischen Artikeln. Es kann aber vorkommen, dass Artikel häufig gemeinsam in Aufträgen auftreten. Dann kann es sinnvoll sein, diese Artikel nah beieinander zu lagern, um so den notwendigen Kommissionieraufwand zu senken. In diesem Fall spricht man von einer *Lagerplatzvergabe unter Auswertung von Verbundbeziehungen* (Reschke 2013, S. 52). Durch die Berücksichtigung von Verbundbeziehungen ist es häufig möglich, Einsparungen der Wegstrecke zu realisieren, da in einer Tour mehrere Artikel auf einmal kommissioniert werden können (Glock et al. 2011, S. 67). Allerdings können die meisten Verfahren nur die Korrelation zwischen zwei Artikeln berücksichtigen, auch wenn Verbundbeziehungen zwischen mehreren Artikeln bestehen (Battini et al. 2015, S. 1078). Eine weitere Möglichkeit ist die Lagerplatzvergabe nach *Auswertung der Kontaktintensitäten*. Artikel, die häufig innerhalb einer Kommissioniertour aufeinander folgen, werden nah beieinander gelagert

(Reschke 2013, S. 82). Die Kontaktintensitäten stehen für die Häufigkeit, mit der ein Artikel in allen Kommissioniertouren direkt nach einem anderen Artikel kommissioniert wird. Die Kontaktintensitäten lassen sich aus den Kommissionieraufträgen ableiten (Reschke 2013, S. 40). Erfolgt die Lagerplatzvergabe unter Berücksichtigung von Verbundbeziehungen oder Kontaktintensitäten, spricht man von *Family Grouping* (de Koster et al. 2007, S. 491). Die Lagerplatzvergabe unter Berücksichtigung der Kontaktintensitäten sollte nicht eingesetzt werden, wenn Aufträge gebündelt kommissioniert werden. Der Grund dafür ist, dass sich durch die Bündelung von Aufträgen sehr lange Kommissioniersequenzen ergeben. Diese Sequenzen treten aber sehr selten auf, wodurch die Verfahren ihren Vorteil verlieren (Mantel et al. 2007, S. 304). Die Verfahren sind anfällig gegenüber Volatilität, da die Artikelkorrelationen, mehr noch als die Nachfrageintensitäten der Einzelartikel, Änderungen unterworfen sind. Eine einmal gefundene Artikelanordnung läuft so Gefahr, schnell zu veralten. Daher sollte das Nachfrageprofil stabil sein (Mantel et al. 2007, S. 304).

Dynamische Lagerplatzvergabe

Die meisten Verfahren wurden für das statische Lagerplatzvergabeproblem entwickelt (Kofler et al. 2015, S. 416). Die Literatur, die sich mit der dynamischen Variante des Problems beschäftigt, ist stark limitiert (Kofler et al. 2010, S. 6). In der statischen Version wird von einem unveränderten Materialfluss ausgegangen. Es bleibt unberücksichtigt, dass sich die Zusammensetzung des Materialflusses im Zeitverlauf ändern kann (Carlo et al. 2010, S. 40). Notwendig wird eine Umlagerung, wenn die ursprüngliche Artikelanordnung mit der Zeit ineffizient wird (Quintanilla et al. 2015, S. 1406), weil sich etwa das Zugriffsverhalten auf einzelne Artikel oder die Sortimentszusammensetzung ändert. Auch an- bzw. auslaufende Vertriebsaktionen (ten Hompel et al. 2010, S. 57-58) oder eine saisonal schwankende Nachfrage können eine Umlagerung notwendig machen (Liu 2004, S. 37). Es stellt sich somit in vielen Fällen nicht die Frage, ob umgelagert werden soll, sondern wie häufig und in welchem Umfang (Kofler et al. 2011, S. 81). In der Literatur werden zwei grundlegende Strategien unterschieden. Bei der sogenannten *Rewarehousing-Strategie* wird eine

neue optimale Artikelanordnung ermittelt und über eine große Anzahl an Umlagerungen realisiert. Aufgrund des damit verbundenen hohen Aufwands erfolgt dies unregelmäßig und in größeren Zeitabständen (Kofler 2015, S. 72). Umfangreiches Rewarehousing kann den laufenden Betrieb des Kommissioniersystems stark negativ beeinflussen, da Personal und Betriebsmittel gebunden werden. Es besteht die Gefahr, dass die neue Artikelanordnung schnell veraltet (Kofler et al. 2011, S. 81). Bei der zweiten Strategie, der sogenannten *Healing-Strategie*, wird regelmäßig eine kleine Anzahl Umlagerungen vorgenommen. Diese können entweder während des operativen Betriebs oder während der Stillstandszeiten des Systems durchgeführt werden (Kofler 2015, S. 72).

Umlagerungen sollten nur durchgeführt werden, wenn die erzielbaren Effizienzgewinne die notwendigen Aufwände übersteigen (Kofler et al. 2015, S. 422). Grosse et al. (2013) haben untersucht, wann Artikel umgelagert werden sollten und welchen Einfluss Lerneffekte auf diese Entscheidung haben. Häufig erfolgt die Umlagerung von Artikeln in der Praxis jedoch rein erfahrungsbasiert, ohne methodische Unterstützung (Hou et al. 2010, S. 369). Bestehende Verfahren zur dynamischen Lagerplatzvergabe geben oft keinen Anhaltspunkt, ob die zu erwartenden Effizienzgewinne den notwendigen Umlagerungsaufwand übersteigen (Kofler et al. 2015, S. 416). Aufgrund der zentralen Bedeutung für die vorliegende Arbeit werden Verfahren zur dynamischen Lagerplatzvergabe in Kapitel 4.2 detailliert untersucht.

2.4.3 Auftragsbildung

Das zweite operative Planungsproblem ist das *Auftragsbildungsproblem*: „Ausgehend von bekannten Lagerorten der Artikel, festgelegter Kapazität des Kommissioniergeräts und einer gegebenen Menge an Aufträgen, sollen diese derart zu Kommissionieraufträgen zusammengefasst werden, dass die benötigte Gesamttourlänge minimiert wird“ (Wäscher 2004, S. 337). Dadurch kann die Kapazität des Kommissionierers bzw. des Kommissioniergeräts besser ausgelastet werden (Grosse et al. 2013, S. 654). Die Kapazität kann z.B. durch die maximale Anzahl an Artikeln (Koch 2014, S. 26) oder das maximale Gewicht der Aufträge (Kulak et al. 2012, S. 57)

vorgegeben sein. Das Auftragsbildungsproblem ist ein *Bin-Packing-Problem*. Es wird ein Weg gesucht, eine vorgegebene Menge Aufträge so auf Batches zu verteilen, dass die Kapazität eines Batches nicht überschritten und die Anzahl notwendiger Batches minimiert wird (Won et al. 2005, S. 1430). Für Batches mit mehr als zwei Aufträgen ist das Problem NP-schwer (Gademann et al. 2005, S. 65). Nach der zeitlichen Informationsverfügbarkeit werden zwei Situationen unterschieden. Ist lediglich eine Teilmenge der Aufträge zum Start des Planungslaufs bekannt, kommt das *Time Window Batching* zum Einsatz (van Nieuwenhuyse et al. 2009, S. 655). Sind alle Aufträge zu Beginn bekannt, wird das *Proximity Batching* eingesetzt. Die Bildung der Kommissionieraufträge erfolgt auf Basis der Artikelanordnung im Lager. Aufträge mit vielen identischen Artikeln oder mit Artikeln, die nah beieinander liegen, werden zusammengefasst (Koch 2014, S. 25). In der vorliegenden Arbeit wird davon ausgegangen, dass alle notwendigen Informationen zu Beginn des Planungslaufes verfügbar sind.

Wenn mehrere Aufträge gemeinsam kommissioniert werden, muss eine Vereinzelung der Artikel und Zuordnung der Artikel zu den Aufträgen erfolgen. Hierfür gibt es die beiden Sortierstrategien *Sort-while-Pick* (die Sortierung findet während dem Kommissionierprozess statt) und *Sort-after-Pick* (das Sortieren erfolgt nach Abschluss der Kommissionierung im Rahmen eines zusätzlichen Sortierprozesses) (Gu et al. 2007, S. 16). Beim *Sort-while-Pick* kann der Aufwand für den zusätzlichen Sortierprozess nach der Kommissionierung vermieden werden. Das Kommissioniergerät hat unterschiedliche Behälter, die jeweils einem Auftrag zugeordnet sind. So können die Artikel den Aufträgen bereits während dem Kommissionierprozess zugeordnet werden. Das bedingt allerdings, dass Aufträge nicht auf verschiedene Batches aufgeteilt werden, da sonst nach dem Kommissionierprozess ein zusätzlicher Sortieraufwand entsteht (de Koster et al. 1999, S. 1480).

Zur Lösung des Auftragsbildungsproblems werden Prioritätsregeln, Seed-Algorithmen, Savings-Algorithmen, Metaheuristiken oder exakte Verfahren eingesetzt. *Prioritätsregeln* ordnen jedem Auftrag einen bestimmten Prioritätswert zu. Anschlie-

ßend werden die Aufträge entsprechend ihres Prioritätswerts den Batches so zugeordnet, dass die Kapazitätsgrenze des Kommissioniergeräts nicht überschritten wird (Wäscher 2004, S. 337). Die bekannteste und in der Praxis am häufigsten eingesetzte Prioritätsregel ist die *First-Come-First-Served-Regel*: Aufträge werden in der Reihenfolge ihres Eintreffens solange dem offenen Batch zugewiesen, bis die Kapazitätsgrenze erreicht ist und ein neuer Batch eröffnet wird (de Koster et al. 1999, S. 1483). *Seed-Algorithmen* bilden Batches in den Schritten Seed-Auswahl und Auftragsergänzung sequentiell. In der Seed-Auswahl wird ein Seed-Auftrag (Startauftrag) ausgewählt, der dem bis dato leeren Batch zugeordnet wird. Im zweiten Schritt werden dem Batch so lange Aufträge zugeteilt, bis die Kapazitätsgrenze erreicht wird. Regeln für die Auswahl des Seed-Auftrags werden als *Seed-Selection-Rules* und für die Auftragsergänzung als *Order-Congruency-Rules* bezeichnet (Henn et al. 2011, S. 10). *Savings-Algorithmen* basieren auf dem von Clarke et al. (1964) entwickelten Algorithmus für das Tourenplanungsproblem (Wäscher 2004, S. 338-339). Bei der ursprünglichen Version von Elsayed et al. (1989, S. 1110-1112) werden für alle möglichen Auftragskombinationen Savings-Werte berechnet. Zunächst wird die Wegstrecke für jeden Auftrag unter der Annahme ermittelt, dass er separat in einer Tour kommissioniert wird. Anschließend wird die Wegstrecke berechnet, die notwendig ist, wenn beide Aufträge gemeinsam in einer Tour kommissioniert werden. Der Savings-Wert beider Aufträge ergibt sich aus der Differenz der Gesamtlänge zweier separater Kommissioniertouren und der Wegstrecke der gemeinsamen Kommissionierung. Die Auftragspaare werden nach abnehmenden Savings-Werten sortiert und den Batches zugeordnet. Erweiterungen dieser Basisvariante finden sich unter anderem in de Koster et al. (1999) und Elsayed et al. (1989). Eine *Metaheuristik* ist ein Algorithmus, der aus einer globalen Steuereinheit und lokalen Suchoperatoren besteht (Schwede et al. 2011, S. 160). Es handelt sich um allgemeine Prinzipien zur Steuerung von heuristischen Verfahren (Suhl et al. 2013, S. 13). Wie Heuristiken sind Metaheuristiken keine exakten Verfahren (Suhl et al. 2013, S. 20) und gehören der Klasse der Verbesserungsverfahren an (Nickel et al. 2011, S. 226). Sie sind nicht auf einen speziellen Problemtyp beschränkt (Suhl et al. 2013, S. 137) und

haben sich als äußerst leistungsfähig erwiesen (Zimmermann 2008, S. 298). Die erste Metaheuristik für das Auftragsbildungsproblem wurde von Gademann et al. (2005) mit dem *Iterated Descent* entwickelt (Koch 2014, S. 40). Eine Erweiterung des Verfahrens von Gademann et al. (2005) wurde von Henn et al. (2010) entwickelt. Das *Iterated Local Search* unterscheidet sich im Wesentlichen in den berücksichtigten Nachbarschaften und der etwas umfangreicher gestalteten Störungsphase (Koch 2014, S. 44). Weitere Metaheuristiken wurden von Albareda-Sambola et al. (2009) (*Variable Neighborhood Search*), Henn et al. (2012) (*Tabu Search*), Henn et al. (2010) (*Ameisenalgorithmus*), Hsu et al. (2005) (*Genetischer Algorithmus*) und Cheng et al. (2015) (*Partikelschwarmoptimierung*) entwickelt. Zur Lösung des Auftragsbildungsproblems wurden bislang kaum *exakte Verfahren* entwickelt. Grund ist die hohe Komplexität des Problems. Gademann et al. (2005) haben ein Spalten-generierungsverfahren entwickelt, das nur sehr kleine Problemgrößen lösen kann. Für praxisrelevante Problemgrößen lassen sich auf diesem Weg keine Lösungen finden (Chen et al. 2005, S. 333).

2.4.4 Tourenplanung

Das *Tourenplanungsproblem* in der Kommissionierung wird wie folgt definiert: „In welcher Reihenfolge sollen die in einem Batch enthaltenen Artikel (mit gegebener Artikelanordnung im Lager) kommissioniert werden, so dass die Gesamttourlänge minimiert wird?“ (Wäscher 2004, S. 340). Das Tourenplanungsproblem ist ein Spezialfall des *Traveling-Salesman-Problems (TSP)*: Der Kommissionierer beginnt seine Kommissioniertour am Depot, muss in der Folge alle im Kommissionierauftrag enthaltenen Lagerorte besuchen und wieder zum Ausgangspunkt zurückkehren. Im Unterschied zum klassischen TSP ergibt sich die Besonderheit, dass einige Knoten (die Schnittpunkte von Kommissioniergängen und Quergängen) nicht besucht werden müssen, andere aber mehrfach besucht werden können. In Abgrenzung zum klassischen Problem wird dieser Fall als *Steiner TSP* bezeichnet. Im Allgemeinen ist sowohl das klassische als auch das Steiner TSP nicht mit polynomialem Aufwand

lösbar (de Koster et al. 2007, S. 494). Für das rechteckige Ein-Block-Layout, entwickelten Ratliff et al. (1983) einen Algorithmus, der das Problem optimal löst. Dieser Algorithmus wurde nachfolgend auf weitere Layoutvarianten ausgeweitet, siehe hierzu Cornuéjols et al. (1985), de Koster et al. (1998) sowie Roodbergen et al. (2001a, 2001b). Cambazard et al. (2018) haben einen optimalen Algorithmus für Block-Layouts mit bis zu acht Quergängen entwickelt. Pansart et al. (2018) haben schließlich einen optimalen Algorithmus für Block-Layouts mit einer beliebigen Anzahl an Quergängen entwickelt. *Optimale Algorithmen* werden in der Praxis allerdings kaum eingesetzt. Gründe hierfür sind die hohe Komplexität der Verfahren (de Koster et al. 1998, S. 470) und die hohen Rechenzeiten. Einige Autoren weisen darauf hin, dass optimale Routen für Kommissionierer nicht intuitiv verständlich sind und diese daher von den geplanten Routen abweichen würden (Gademann et al. 2005, S. 73). Demgegenüber steht, dass auch einfache Heuristiken für komplexere Lagerlayouts, wie etwa das Multi-Block-Layout, verzweigte, nicht-intuitive Routen erzeugen (Scholz et al. 2017, S. 493). Die fortschreitenden technologischen Möglichkeiten im Bereich der Assistenzsysteme unterstützen Kommissionierer beim Folgen der Routenvorschläge. Dies spricht zukünftig für den vermehrten Einsatz von effizienteren Verfahren für die Tourenplanung (Chen et al. 2013, S. 77). Derzeit werden in der Praxis allerdings meist Heuristiken zur Lösung des Tourenplanungsproblems eingesetzt (Won et al. 2005, S. 1431). Aufgrund ihrer praktischen Bedeutung sind einige der bekanntesten Routingstrategien in Abbildung 14 dargestellt.

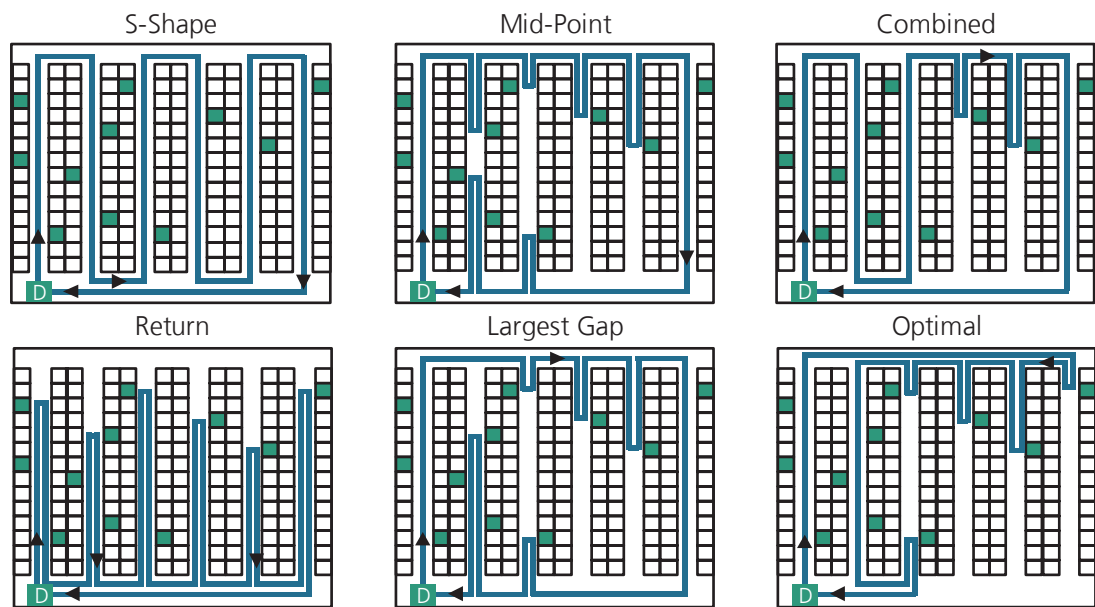


Abbildung 14: Routingstrategien (in Anlehnung an Roodbergen 2001, S. 36)

Die *S-Shape-Strategie* (Abbildung 14 links oben) ist die am häufigsten in der Praxis eingesetzte Routingstrategie: Der Kommissionierer betritt nur die Gänge, die zu kommissionierende Artikel enthalten, durchquert diese bis zum anderen Ende und geht von dort zum nächsten Gang (Roodbergen 2001, S. 33). Diese Strategie dient in der Literatur meist als Benchmark für andere Heuristiken (Chen et al. 2013, S. 78). Bei der *Return-Strategie* (Abbildung 14 links unten) werden die Gänge an der Seite mit dem Depot betreten und bis zum letzten Artikel durchschritten. Dort angekommen, wird wieder zur selben Seite zurückgekehrt und der nächste Gang betreten. Für die *Mid-Point-Strategie* (Abbildung 14 Mitte oben) wird das Kommissionierlager gedanklich in zwei Hälften geteilt. Nur der erste und letzte Gang mit Artikeln werden ganz durchschritten. Bei den restlichen Gängen werden Artikel der unteren Hälfte vom unteren Quergang und die oberen vom oberen Quergang aus kommissioniert. Bei der *Largest-Gap-Strategie* (Abbildung 14 Mitte unten) werden der erste und letzte Gang ganz und alle weiteren Gänge nur bis zur größten Lücke zwischen zwei Artikeln, einem Artikel und dem Eingang oder einem Artikel und dem Ausgang durchschritten (Roodbergen 2001, S. 33-34). Die *Combined-Strategie* (Abbildung

14 rechts oben) ist eine Kombination aus S-Shape- und Return-Strategie. Die Entscheidung wird mittels eines dynamischen Programmierungsansatzes getroffen (Roodbergen et al. 2001a, S. 1871-1872). Außerdem zeigt Abbildung 14 das Ergebnis des von Ratliff et al. (1983) entwickelten optimalen Algorithmus.

2.4.5 Interdependenz

Die Interdependenz wurde in Kapitel 1.4.2 als wesentliches Objekt des heuristischen Bezugsrahmens identifiziert. *Interdependenz* bezeichnet in der Entscheidungstheorie die Wechselbeziehung bzw. gegenseitige Abhängigkeit verschiedener Entscheidungsfelder in einem Unternehmen (o.V. 2013, S. 159). In einer Vielzahl von Arbeiten wird auf die starke Interdependenz der operativen Planungsprobleme hingewiesen (unter anderem van Gils et al. 2017, S. 5; de Koster et al. 2007, S. 486; Rao et al. 2013a, S. 1309; Sooksaksun et al. 2012, S. 220; Rouwenhorst et al. 2000, S. 522; Chan et al. 2011, S. 2688). Dies wird von einer Reihe statistischer Untersuchungen gestützt, die die starke Interdependenz der Planungsprobleme nachweisen (van Gils et al. 2017, S. 18-19). So weisen Petersen et al. (2004a, S. 14-16), Ho et al. (2006, S. 3402-3414) und Ho et al. (2008, S. 334-343) eine statistisch signifikante Abhängigkeit zwischen dem Lagerplatzvergabe- und dem Auftragsbildungsproblem nach. Somit sollten Auftragsbildungsmethoden Lagerplatzinformationen berücksichtigen (van Gils et al. 2017, S. 18). Die Interdependenz zwischen der Lagerplatzvergabe und der Tourenplanung wird unter anderem von Petersen et al. (1999, S. 494-496), Manzini et al. (2007, S. 819-820), Theys et al. (2010, S. 758-762) und Shqair et al. (2014, S. 128-134) nachgewiesen. Besonders stark ist die Abhängigkeit zwischen dem Auftragsbildungs- und dem Tourenplanungsproblem: Um das Auftragsbildungsproblem lösen zu können, muss die Reihenfolge bekannt sein, in der die Lagerplätze nacheinander besucht werden. Um allerdings das Tourenplanungsproblem lösen zu können, müssen die Aufträge bereits den Batches zugewiesen worden sein (Scholz et al. 2015, S. 2). Unter anderem zeigen Kulak et al. (2012, S. 71), dass durch die kombinierte Betrachtung der beiden Probleme signifikante Verbesserungen erzielt werden können.

2.5 Zwischenfazit

In diesem Kapitel wurde die Grundlagenliteratur des Objektbereichs Kommissionierung analysiert. Die strategischen Planungsprobleme bilden den Rahmen für die Ausgestaltung der operativen Planungsprobleme. Das im Zuge der Dissertation entwickelte Verfahren betrachtet die strategischen Probleme nicht. Die Betriebsstrategien werden im Rahmen der strategischen Planung ausgewählt. Bevor das Verfahren entwickelt werden kann, sind daher zunächst geeignete Betriebsstrategien festzulegen. In Kapitel 1.3 wurden die Wegstreckenreduktion, die Interdependenz, die Dynamik, der Umlagerungsaufwand und die Praxistauglichkeit als Anforderungen festgelegt. Die Auswahl der jeweiligen Betriebsstrategie erfolgt anhand der Ergebnisse der Literaturanalyse und der definierten Anforderungen.

Lagerplatzvergabe

Eine statische Lagerplatzvergabe ist in einem dynamischen Umfeld ungeeignet. Deshalb erfolgt sie im Verfahren *dynamisch*. Aufgrund der höheren Akzeptanz in der Praxis und dem formulierten Ziel, das System in einem dynamischen Umfeld dauerhaft effizient betreiben und Umlagerungsaufwände berücksichtigen zu wollen, kommt nur die *Healing-Strategie* in Frage. Die zufällige Lagerplatzvergabe erzeugt die längsten Kommissionierwege und ist deshalb genauso ungeeignet wie die feste Lagerplatzvergabe, die das Risiko birgt, schnell zu veralten und sensibel auf Nachfrageänderungen zu reagieren. Die Verfahren des Family Groupings sind meist nicht kompatibel mit der Bündelung von Aufträgen. Auch reagieren sie sehr sensibel auf Nachfrageänderungen. Die *klassenbasierte Lagerplatzvergabe* erfüllt alle Kriterien und wird daher ausgewählt.

Auftragsbildung

Exakte Auftragsbildungsverfahren liefern für praxisrelevante Problemgrößen keine Lösung. Daher kommen nur *Heuristiken* in Frage. Als Sortierstrategie wird das Prin-

zip *Sort-while-Pick* unterstellt, da so vermieden werden kann, dass Wegstreckenreduktionen, die durch die Bündelung von Aufträgen entstehen, durch zusätzliche Sortieraufwände aufgehoben werden. Aus demselben Grund sind *Aufträge nicht teilbar*. In dieser Arbeit wird davon ausgegangen, dass alle Aufträge zu Beginn des Planungslaufs bekannt sind und damit das *Proximity Batching* zum Einsatz kommt.

Tourenplanung

Aufgrund ihrer hohen Rechenzeiten sind exakte Verfahren für ein kombiniertes Verfahren nicht geeignet. Daher werden *heuristische Verfahren* eingesetzt.

Abbildung 15 zeigt die für das zu entwickelnde Verfahren einzusetzenden Betriebsstrategien.

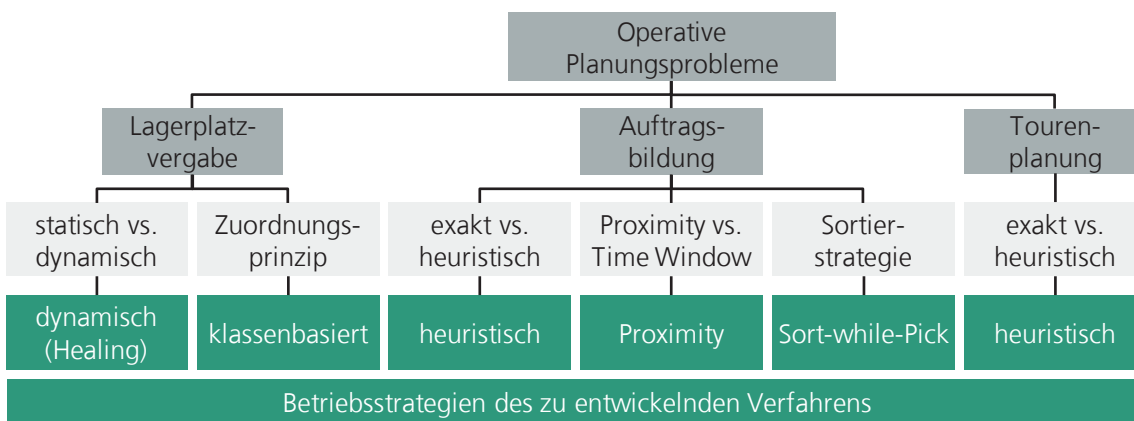


Abbildung 15: Auswahl Betriebsstrategien

3 Grundlagen der Optimierung und Prognose

Nachdem im vorangegangenen Kapitel die Grundlagen der Kommissionierung untersucht wurden, werden in den nachfolgenden Kapiteln die Grundlagen der Optimierung und Prognose beschrieben. Dies soll die Auswahl geeigneter Lösungsverfahren für das formulierte Problem ermöglichen. Die Dynamik wurde als wesentlicher Bestandteil des heuristischen Bezugsrahmens identifiziert und wird daher ebenfalls analysiert.

3.1 Optimierung

Unter einem *Optimum* werden Parameter oder Eigenschaften verstanden, für die das beste Resultat eines angestrebten Zielwerts erreicht wird. Im Zuge einer *Optimierung* wird nach dem Optimum unter gegebenen Voraus- und Zielsetzungen gesucht (März et al. 2011, S. 11). Mathematisch beschreibt der Begriff Optimierung „die Bestimmung des Maximums oder Minimums einer Funktion [...], die auf einem (beschränkten) Bereich [...] oder Zustandsraum definiert ist“ (Kallrath 2013, S. 1). Modelle spielen eine zentrale Rolle: „Ein Modell ist ein vereinfachtes (isomorphes oder homomorphes) Abbild eines realen Systems oder Problems“ (Domschke et al. 2015, S. 3). Bei einem *Optimierungsmodell* handelt es sich um „formale Darstellungen eines Entscheidungs- und Planungsproblems, das in seiner einfachsten Form mindestens eine Alternativmenge und eine diese bewertende Zielfunktion enthält. Es wird entwickelt, um mit geeigneten Verfahren optimale oder suboptimale Lösungsvorschläge ermitteln zu können“ (Domschke et al. 2015, S. 4). Bezogen auf ihre Komplexität können Optimierungsprobleme in zwei Klassen unterteilt werden: Probleme, die auch im ungünstigsten Fall mit polynomialem Rechenaufwand lösbar sind, werden der *Klasse P* zugeordnet. Konnte bislang für ein Problem noch kein Algorithmus entwickelt werden, der auch die am schwersten zu lösende Probleminstanz mit polynomialem Aufwand lösen kann, gehört dieses Problem zur Klasse der *NP-schweren Probleme* (Domschke et al. 2015, S. 133). Mit polynomialem Auf-

wand lösbar ist ein Modell, wenn der Rechenaufwand mit der Problemgröße proportional steigt. Für Probleme der Klasse NP gilt dies nicht. Hier steigt der Lösungsaufwand exponentiell. Es wird davon ausgegangen, dass Probleme der Klasse P effizient bzw. gut lösbar sind. Dagegen gelten Probleme der Klasse NP als schwer lösbar (Zimmermann 2008, S. 272).

Optimierungsmodelle werden nach dem Typ ihrer Zielfunktion und den Nebenbedingungen unterschieden in (Domschke et al. 2015, S. 7):

- Lineare Optimierungsmodelle
- Ganzzahlige und binäre Optimierungsmodelle
- Dynamische Optimierungsmodelle
- Nichtlineare Optimierungsmodelle

Wie noch gezeigt wird, kann für das dynamische Lagerplatzvergabeprobem und für das kombinierte Auftragsbildungs- und Tourenplanungsproblem jeweils ein ganzzahliges Optimierungsmodell formuliert werden (siehe Kapitel 5.1.2 und 5.1.3). *Ganzzahlige Optimierungsmodelle* lassen die Modellierung von Null-Eins-Entscheidungen, wie etwa die Zuordnung von Arbeitsschritten zu Mitarbeitern, zu (Kallrath 2013, S. 1-2). Die ganzzahlige Optimierung unterscheidet sich von der kontinuierlichen linearen Optimierung durch Variablen, die einer Ganzzahligkeitsbedingung unterliegen. Im Unterschied zu linearen Optimierungsmodellen sind diese Probleme in der Regel sehr schwer zu lösen (Suhl et al. 2013, S. 9).

Die drei Planungsprobleme dieser Arbeit lassen sich jeweils mithilfe eines ganzzahligen Optimierungsmodells beschreiben. Das Lagerplatzvergabeprobem ist ein *quadratisches Zuordnungsproblem* (Reschke 2013, S. 42-44). Neben der Lagerplatzvergabe wird es unter anderem auch im Bereich der innerbetrieblichen Standortplanung eingesetzt, wenn eine vorgegebene Anzahl gleich großer Maschinen auf gleich große Plätze so angeordnet werden sollen, dass die Summe der Transportkosten zwischen den Maschinen minimal ist (Domschke et al. 2015, S. 131). Das Auftragsbildungsproblem ist ein Bin-Packing-Problem (Domschke et al. 2015, S. 129). Beim *Bin-Packing-Problem* soll eine gegebene Menge an Gegenständen unterschiedlicher

Größe in eine minimale Anzahl gleich großer Behälter gepackt werden. Das Gewicht aller Gegenstände in einem Behälter darf dessen Kapazität nicht überschreiten. Kein Artikel darf schwerer als die Kapazität eines Behälters sein, da die Gegenstände unteilbar sind (Scholl et al. 1997, S. 627). Das Tourenplanungsproblem kann als ganzzahliges Optimierungsproblem in Form eines TSP modelliert werden (Domschke et al. 2015, S. 129). Dem *TSP* liegt die Vorstellung eines Handlungsreisenden zugrunde, der auf seiner Reise eine vorgegebene Anzahl Kunden besucht und am Ende seiner Reise wieder zum Ausgangsort zurückkehren muss. Er muss während seiner Reise alle Kunden einmal besuchen und zwar so, dass die Länge seiner Rundreise minimiert wird. Es wird die optimale Reihenfolge gesucht, in der der Handlungsreisende die Kundenorte besuchen soll (Zimmermann 2008, S. 309).

3.2 Optimierungsverfahren

Nach dem Aufstellen eines Optimierungsmodells wird es mithilfe eines *Optimierungsverfahrens* unter Verwendung von Daten gelöst (Domschke et al. 2015, S. 2). Unter einem *Algorithmus* versteht man „eine genau definierte Rechen-, Handlungs- und/oder Verarbeitungsvorschrift zur Lösung eines Problems/Problemtyps“ (Suhl et al. 2013, S. 132). Während *exakte Verfahren* die Ermittlung einer optimalen Lösung erlauben, bieten *heuristische Verfahren* keine Garantie dafür, dass ein Optimum gefunden wird (Kallrath 2013, S. 83).

Heuristiken

Heuristiken bestehen aus Vorgehensregeln, die für die Lösung der jeweiligen Problemstellung geeignet sind (Domschke et al. 2015, S. 135). Da die meisten ganzzahligen und kombinatorischen Probleme zur Klasse NP gehören, bietet sich hier der Einsatz von Heuristiken an (Zimmermann 2008, S. 272). Kennzeichnend für alle Heuristiken ist, dass sie nicht den gesamten Lösungsraum untersuchen. Die Suche des Restlösungsraums erfolgt nicht willkürlich, sondern folgt im Voraus definierten Regeln. Es lassen sich Eröffnungsverfahren, die zur Ermittlung einer ersten zulässigen

Lösung eingesetzt werden, von Verbesserungsverfahren unterscheiden (Zimmermann 2008, S. 273-275). *Eröffnungsverfahren* können bei jedem Verfahrensschritt eine größtmögliche Verbesserung des Zielfunktionswertes anstreben (sogenannte Greedy-Verfahren) oder versuchen, eine zulässige Lösung ohne Berücksichtigung der Zielfunktion zu ermitteln (sogenannte uniformierte Verfahren). *Lokale Such- bzw. Verbesserungsverfahren* starten mit einer zulässigen Lösung und versuchen, diese im weiteren Verlauf zu verbessern. In jeder Iteration wird von der aktuellen Lösung zu einer Lösung in der Nachbarschaft fortgeschritten. Die Nachbarschaft enthält alle Lösungen, die aus der aktuellen Lösung mithilfe einer Transformationsvorschrift erzeugt werden können. Eine Transformationsvorschrift für das TSP ist der Tausch von Kanten gegen Kanten, die noch nicht in der Rundreise enthalten sind. Die Nachbarschaft enthält alle Lösungen, die so aus der aktuellen Lösung entwickelt werden können (Domschke et al. 2015, S. 136). Ein *2opt-Algorithmus* prüft alle Vertauschungsmöglichkeiten von jeweils zwei Kanten einer gegebenen Rundreise gegen zwei andere. Kann die Rundreise durch den Tausch verkürzt werden, wird er durchgeführt. Die Überprüfung wird so lange fortgeführt, bis bei der letzten Überprüfung aller paarweisen Tauschmöglichkeiten keine Verbesserung mehr erzielt werden konnte (Domschke et al. 2015, S. 154). Der 2opt-Algorithmus liefert sehr gute Lösungen für das TSP. Er benötigt sehr kurze Rechenzeiten. Das Ergebnis liegt oft nur wenige Prozentpunkte vom globalen Optimum entfernt (Englert et al. 2007, S. 1296).

Metaheuristiken

Ein wesentlicher Nachteil lokaler Suchverfahren ist, dass sie häufig zu lokalen Optima konvergieren (Talbi 2009, S. 125). Um dieses Problem zu beheben, wurden sogenannte *Metastrategien* (Domschke et al. 2015, S. 136) bzw. *Metaheuristiken* entwickelt (Suhl et al. 2013, S. 13). Metaheuristiken werden zur Lösung von Optimierungsproblemen eingesetzt, für die es keine exakten Verfahren gibt, die in der Lage sind, optimale Lösungen innerhalb akzeptabler Rechenzeiten zu finden oder für die

es keine problemspezifischen Algorithmen gibt (Boussaïd et al. 2013, S. 82). Metaheuristiken sind nicht für ein bestimmtes Problem entwickelt worden und damit problemunspezifisch (Blum et al. 2011, S. 4135). Es handelt sich um übergeordnete Frameworks, die versuchen, durch die Kombination einzelner untergeordneter Heuristiken, den Lösungsraum eines Optimierungsproblems zu untersuchen (Blum et al. 2003, S. 270). Die untergeordneten Heuristiken bedienen sich unterschiedlicher Lösungskonzepte und werden von der Metaheuristik so gesteuert, dass der Lösungsraum einerseits ganzheitlich erforscht, andererseits aber auch detailliert ausgewertet werden kann (Osman et al. 1996, S. 513-514). Während *Exploration* für die Suche nach Regionen im Lösungsraum mit einer hohen Lösungsqualität steht (Birattari et al. 2001, S. 3), bedeutet *Exploitation* die besonders intensive Suche innerhalb dieser vielversprechenden Regionen. Metaheuristiken unterscheiden sich in der Art und Weise, in der sie versuchen, eine Balance zwischen diesen beiden Eigenschaften zu erreichen (Birattari et al. 2001, S. 3). Eine häufig in der Literatur verwendete Klassifizierung grenzt Metaheuristiken voneinander ab, die bei jeder Iterationsstufe eine (*Einzellösungsverfahren*) oder gleichzeitig mehrere Lösungen untersuchen (*populationsbasierte Metaheuristiken*) (Boussaïd et al. 2013, S. 83).

Einzellösungsverfahren

Einzellösungsverfahren starten mit einer Anfangslösung und bewegen sich im Suchverlauf unter Beschreibung einer Bahnkurve von dieser weg. Daher werden diese Verfahren auch als Trajektorie-Verfahren bezeichnet (Boussaïd et al. 2013, S. 83). Abbildung 16 klassifiziert diese Verfahren nach der Art und Weise, wie sie dem Problem der Konvergenz in lokalen Optima begegnen.

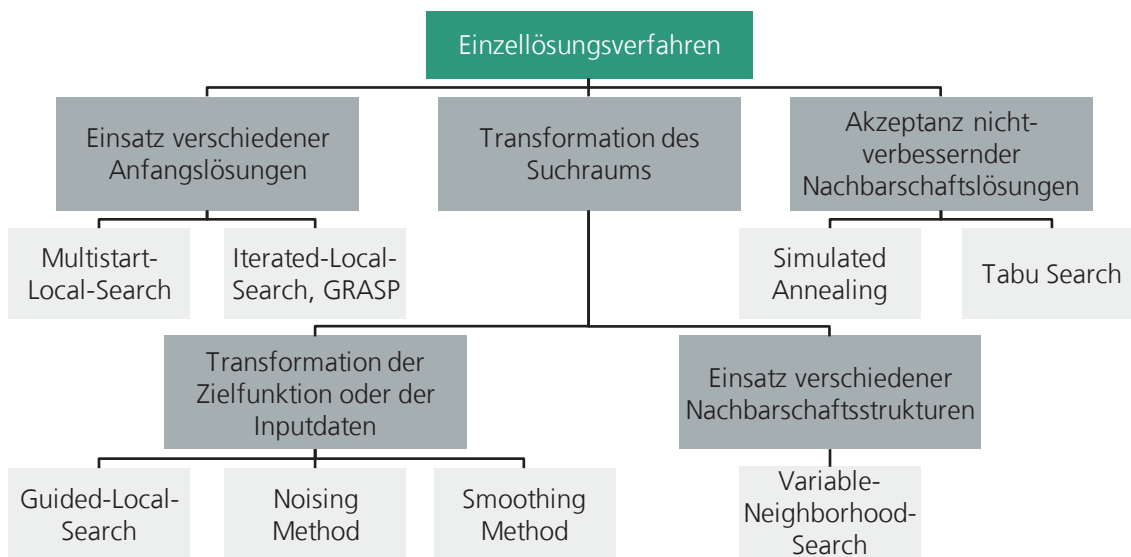


Abbildung 16: Klassifizierung Einzellösungsverfahren (in Anlehnung an Talbi 2009, S. 125)

Da die Lösungsqualität einfacher lokaler Suchverfahren stark von der gewählten Startlösung abhängt, setzen Verfahren wie das Multistart-Local-Search, das Iterated-Local-Search (Stützle 1998) oder das Greedy-Adaptive-Search-Procedure (GRASP) (Feo et al. 1989) verschiedene Startlösungen ein (Talbi 2009, S. 125). Andere Verfahren wiederum versuchen, durch die temporäre Akzeptanz nicht-verbessernder Nachbarschaftslösungen lokalen Optima zu entgehen (Domschke et al. 2015, S. 137). Ein Beispiel für ein solches Verfahren wurde von Kirkpatrick et al. (1983) entwickelt und wird aufgrund seiner Analogie zum Abkühlverhalten von Metallen als *Simulated Annealing (SA)* bezeichnet. Mit dem *Tabu Search (TS)* wurde von Glover (1986) ein weiteres Einzellösungsverfahren entwickelt. Im Unterschied zu SA-Algorithmen nutzen TS-Algorithmen Wissen aus dem bisherigen Suchverlauf durch den Einsatz einer Art Gedächtnis. Dieses Gedächtnis soll dafür sorgen, dass Lösungen, die kürzlich untersucht wurden, nicht erneut berechnet werden (Glover et al. 1993, S. 6-8). Andere Verfahren wie das *Guided-Local-Search* (Voudouris 1997), die *Noising Method* (Charon et al. 1993) oder die *Smoothing Method* (Glover et al. 1986) transformieren die Zielfunktion oder die Inputdaten, um lokale Optima zu verlassen (Talbi 2009, S. 125). Beim *Variable-Neighborhood-Search* (Mladenović et

al. 1997) werden entweder zufallsbasiert oder systematisch unterschiedliche Nachbarschaftsstrukturen getestet, um verschiedene lokale Optima besuchen zu können und so die Wahrscheinlichkeit zu erhöhen, das globale Optimum zu finden (Talbi 2009, S. 150).

Populationsbasierte Metaheuristiken

Populationsbasierte Metaheuristiken berechnen in jeder Iteration gleichzeitig mehrere Lösungen (Blum et al. 2003, S. 284). Sie können entweder dem Evolutionary Computation oder der Schwarmintelligenz zugeordnet werden. Unter dem Begriff *Evolutionary Computation* werden Algorithmen zusammengefasst, die durch die Evolutionstheorie von Darwin inspiriert wurden, nach der sich eine Population aus Individuen durch Rekombination, Mutation und Selektion an eine sich verändernde Umwelt anpasst. Diese Methoden werden auch als *Evolutionäre Algorithmen* bezeichnet (Boussaïd et al. 2013, S. 90). Die Individuen einer Population stehen für Punkte im Lösungsraum des Optimierungsproblems. Zu Beginn der Optimierung wird die Population zufällig initialisiert und im weiteren Verlauf in bessere Regionen geführt. Die Qualität einer Lösung drückt sich im Fitnesswert eines Individuums aus (Bäck et al. 1993, S. 1). Jede Iteration besteht aus einer neuen Generation von Individuen. Diese Individuen können sich fortpflanzen, unterliegen allerdings einer Selektion. Individuen mit einem höheren Fitnesswert können sich eher fortpflanzen, als solche mit einem niedrigeren Wert. Neue Individuen entstehen entweder durch die Kombination zweier Individuen (Rekombination) oder durch Mutation, was zum Entstehen neuer Merkmale führt (Boussaïd et al. 2013, S. 90). Die größte Verbreitung haben in diesem Bereich *Genetische Algorithmen*, deren Prinzipien von Holland (1975) entwickelt wurden (Boussaïd et al. 2013, S. 90-91). Genetische Algorithmen eignen sich im Besonderen für kombinatorische Optimierungsprobleme (Gerdes et al. 2004, S. 2). Als primärer Suchoperator wird die Rekombination genutzt. Die Mutation ist meist nur ein Hintergrundoperator, der mit einer geringen Wahrscheinlichkeit angewendet wird, allerdings dafür sorgt, dass alle Punkte im Suchraum erreicht

werden können (Weicker 2015, S. 128). Der Selektionsdruck wird durch eine fitnessproportionale Selektion der Eltern erzeugt (Weicker 2015, S. 86). Neben den Genetischen Algorithmen zählen Boussaïd et al. (2013, S. 91-98) noch folgende Verfahren zum Evolutionary Computation:

- *Evolutionäre Programmierung* (Fogel et al. 1966)
- *Evolutionsstrategien* (Rechenberg 1973)
- *Koevolutionäre Algorithmen* (Hillis 1990)
- *Genetische Programmierung* (Koza 1992)
- *Cultural Algorithms* (Reynolds 1994)
- *Estimation of Distribution Algorithms* (Mühlenbein et al. 1996)
- *Differential Evolution Algorithms* (Storn et al. 1997)
- *Scatter Search* (Glover 1998)

Die zweite Gruppe der populationsbasierten Metaheuristiken ist durch das in der Natur beobachtbare kollektive Verhalten von sozialen Insekten und anderen Tiergruppen inspiriert worden (Boussaïd et al. 2013, S. 98). Individuen zeigen für sich genommen sehr einfache Verhaltensweisen, allerdings ist ihr kollektives und soziales Verhalten außerordentlich komplex. Für diesen Zusammenhang hat sich der Begriff *Schwarmintelligenz* etabliert. Bezogen auf die Lösung von Optimierungsproblemen bestehen Schwarmintelligenzsysteme aus einer Grundgesamtheit einfacher Agenten. Diese sind in der Lage, bestimmte Operationen durchzuführen und lokal untereinander und mit ihrer Umgebung zu interagieren (Das et al. 2008, S. 119). Obgleich normalerweise keine zentrale Kontrollinstanz vorhanden ist, führen diese lokalen Interaktionen zu globalen Verhaltensweisen des Schwarms (Boussaïd et al. 2013, S. 98). Mit der ersten Veröffentlichung zur PSO lösten Kennedy et al. (1995) einen wissenschaftlichen Trend aus. Seither steigen die Publikationen zu diesem Thema mit jedem Jahr exponentiell an. Eine Vielzahl Autoren nutzen diese schnelle Art der Optimierung (Bogon 2013, S. 39). Die Grundversion der PSO nutzt ausschließlich einfache mathematische Operatoren und ist konzeptionell relativ simpel (Das et al. 2008, S. 126). Zu Beginn der Optimierung wird stochastisch eine bestimmte Anzahl

an autonomen Einheiten (Partikel) im Suchraum erzeugt. Die Partikel sind Lösungen des Optimierungsproblems (Boussaïd et al. 2013, S. 99). Ein Partikel wird durch seine aktuelle Position im Suchraum und seine Geschwindigkeit beschrieben, mit der er sich durch den mehrdimensionalen Suchraum bewegt. Die Gesamtheit aller Partikel wird als Schwarm bezeichnet. Jeder Partikel hat einen Speicher, der seine bisher beste Position und die beste Position aus der Nachbarschaft festhält. Die PSO kombiniert lokale und globale Sucheigenschaften miteinander. Anfangs tendiert sie zur globalen Suche, fokussiert sich aber im Suchverlauf zunehmend auf die lokale Suche. Sie ist hochgradig effizient (Önüt et al. 2008, S. 788). Önüt et al. (2008, S. 797) zeigen, dass sie in der Lage ist, sehr gute Lösungen innerhalb relativ kurzer Rechenzeiten zu erzeugen. Allerdings tendiert sie manchmal dazu, zu früh eine Lösung zu fokussieren und somit in einem lokalen Optimum zu verharren (Sedighzadeh et al. 2009, S. 487). Um diesen Nachteil zu beheben, kann die PSO um evolutionäre Bestandteile erweitert werden. Dadurch kann verhindert werden, dass zu schnell ein lokales Optimum fokussiert wird (Wei et al. 2002, S. 1278). Zusammenfassend bietet die PSO viele Möglichkeiten, um flexibel gestaltet und genutzt zu werden. Durch mannigfaltige Erweiterungs- und Kombinationsmöglichkeiten kann jegliche Art von Problem effizient gelöst werden (Bogon 2013, S. 58).

Gemäß Boussaïd et al. (2013, S. 98-110) zählen noch folgende Verfahren zu dieser Kategorie:

- *Künstliche Immunsysteme* (unter anderem Farmer et al. 1986)
- *Ameisenalgorithmen* (Dorigo 1996)
- *Bacterial Foraging Optimization Algorithmen* (Passino 2002)
- *Bienenalgorithmen* (unter anderem Karaboga 2005)
- *Biogeographieoptimierungsalgorithmen* (Simon 2008)

3.3 Dynamik

Neben der Interdependenz wurde im heuristischen Bezugsrahmen in Kapitel 1.4.2 die Dynamik als zentrale Herausforderung dieser Arbeit genannt. Die Dynamik ist

neben der Vielzahl, der Vielfalt und der Intransparenz eine der vier Dimensionen von komplexen Systemen (Jäger et al. 2013, S. 342). Sie bezeichnet Veränderungen eines Systemzustandes im Laufe der Zeit. Die verfügbare Reaktionszeit sinkt bei zunehmender Dynamik, wobei eigentlich eine immer höhere Reaktionszeit notwendig wäre. Eine besondere Herausforderung stellt heute, wie in Kapitel 1.1 beschrieben, die Dynamik der Kundenbedürfnisse dar (Thommen 2008, S. 181). Unternehmen agieren heute mehr denn je in einem turbulenten Umfeld. Als mögliche Ursachen für die Intensivierung der Dynamik werden häufig die Globalisierung, die Entwicklung neuer Wirtschaftsräume und -mächte, der technologische Wandel, die zunehmende Innovationsgeschwindigkeit sowie verkürzte Produktlebenszyklen genannt (Tyssen 2012, S. 1-2). Unternehmen sehen sich hoher Unsicherheit gegenüber. Trotzdem basieren bestehende Prozesse und Methoden oft auf einer statischen Betrachtungsweise. Im dynamischen Umfeld wird es für Unternehmen entscheidend, frühzeitig zukünftige Entwicklungen zu erkennen und notwendige Veränderungen einzuleiten (Tyssen 2012, S. 5). Somit ist es für Unternehmen unabdingbar, *Prognosefähigkeiten* aufzubauen, um den Entwicklungen besser begegnen zu können (Zobel 2005, S. 4). Vorhersagefähigkeit trägt entscheidend zur langfristigen Prosperität von Unternehmen bei. Die Fähigkeit zur Prognose möglicher Entwicklungen wird zum kritischen Erfolgsfaktor im dynamischen Wettbewerb. Unternehmen können erwarteten Veränderungen wesentlich besser begegnen als unerwarteten (Zobel 2005, S. 234). Die Fähigkeiten eines Unternehmens, Kompetenzen zum erfolgreichen Umgang mit sich drastisch verändernden Umweltsituationen aufzubauen, werden als dynamische Fähigkeiten bezeichnet. Sie ermöglichen neue und innovative Formen von Wettbewerbsvorteilen (Teece et al. 1997, S. 516). Damit können dynamische Fähigkeiten als eigentliche Quelle für langfristige Wettbewerbsvorteile verstanden werden (Teece 1998, S. 72). Sie erleichtern eine frühzeitige Adaption, die durch Volatilitäten notwendig wird. Dynamische Fähigkeiten differenzieren wissensbasierte, lernende Unternehmen der Zukunft nachhaltig voneinander (Bessant et al. 2002, S. 495).

3.4 Prognose

Eine *Prognose* ist die Vorhersage eines zukünftigen Zustandes oder der zukünftigen Entwicklung eines Systems. Sie basiert auf dem gegenwärtigen Zustand und der vergangenen Entwicklung des Systems (Vogel 2015, S. 11). Empirische Zusammenhänge zwischen betrachteten Variablen sollen aufgedeckt werden (Bamberg et al. 2008, S. 4-5). Der Unterschied zu anderen Vorhersagemethoden, wie etwa der Prognose, ist die Wissenschaftlichkeit (Vogel 2015, S. 9). Prognosen werden eingesetzt, um Unsicherheiten zu reduzieren und das Risiko von Fehlentscheidungen zu verringern (Frerichs et al. 1980, S. 1). Sie haben eine Entscheidungsunterstützungsfunktion (Brockhoff 2005, S. 766). Voraussetzung für die Durchführung von Prognosen ist die Gültigkeit der Zeitstabilitätshypothese, wonach ein in der Vergangenheit beobachtetes Erklärungsmuster auch in der Zukunft gilt (Makridakis et al. 1980, S. 9). Eine Prognose kann immer falsch sein (Vogel 2015, S. 15). Auch wenn ein Prognosesystem nur vergleichbare Prognosesicherheiten bietet, wie qualitative Vorhersagen von Menschen, entsteht mit der Automatisierbarkeit der Prognose dennoch ein beachtlicher Nutzeneffekt. Gerade in der Lagerlogistik ist das von entscheidender Bedeutung, da für eine große Anzahl lagerhaltiger Artikel Prognosen zu erstellen sind (Mertens et al. 2012, S. 4). Im Fokus dieser Arbeit steht die Fortsetzung zeitlich erhobener Datenreihen in die Zukunft, um so den zukünftigen Kommissionieraufwand prognostizieren zu können. Diesem Zusammenhang widmet sich die Zeitreihenanalyse (Vogel 2015, S. 9). Das mathematische Modell einer Zeitreihe ist der *stochastische Prozess*. Eine Zeitreihe ist ein endlicher Ausschnitt aus der Realisierung eines stochastischen Prozesses. Bei der Prognose von Zeitreihen wird versucht, das Modell eines stochastischen Prozesses an die beobachteten Zeitreihenwerte anzupassen. Das Verteilungsgesetz des Prozesses soll möglichst treffend beschrieben und so die Schätzung von künftigen Prozesswerten ermöglicht werden (Vogel 2015, S. 19). Gedanklich werden die beobachteten Werte als das Ergebnis eines stochastischen Prozesses zuzüglich einer nicht vorhersagbaren Zufallsabweichung aufgefasst. Für die Beobachtungswerte x_t gilt somit (Schröder 2012, S. 16):

$$x_t = P_t + e_t \quad (P_t = \text{Prozess}; e_t = \text{Zufallsabweichung}) \quad (3.1)$$

Für die Zufallsabweichungen wird angenommen, dass der Erwartungswert null, die Varianz konstant und die Verteilung annähernd normal ist. Der zugrunde liegende Prozess ist unbekannt und wird als deterministische Funktion der Zeit betrachtet (Schröder 2012, S. 16). Eine Zeitreihe beinhaltet eine Menge von Beobachtungswerten, die zu unterschiedlichen Zeitpunkten erhoben werden. Die Abstände zwischen den Zeitpunkten sind in der Regel konstant und damit äquidistant (Bormann 1975, S. 6). Wenn nach jeder Beobachtung die Vorhersage wiederholt wird, entspricht das Prognoseintervall dem Zeitabstand zwischen zwei Beobachtungen. Der Vorhersagezeitraum ist die Länge der Periode, auf die sich ein Vorhersagewert bezieht. Sie setzt sich aus einem oder mehreren Prognoseintervallen zusammen (Schröder 2012, S. 15). Der Verlauf einer Zeitreihe kann über die Zeit konstant (Abbildung 17 oben) oder trendförmig (Abbildung 17 unten) sein (Kühne et al. 2011, S. 324).

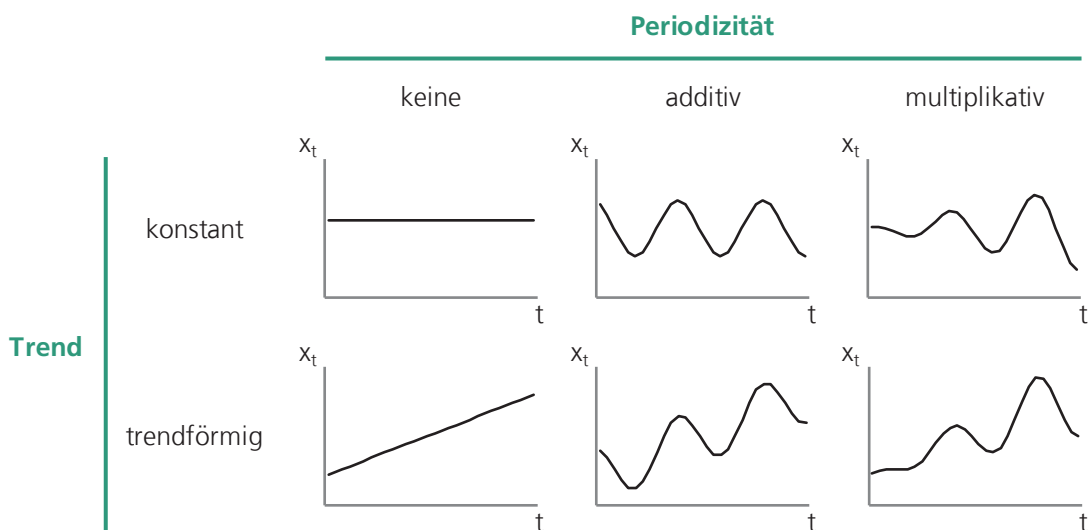


Abbildung 17: Zeitreihenmodelle

Weitere Modellgruppen entstehen, wenn diesen Modellen periodische Schwankungen überlagert werden. Wenn die Amplituden der Schwankungen unabhängig von der Höhe des Grundwertes sind, sind die Schwankungen additiv überlagert (Abbildung 17 Mitte). Multiplikativ überlagerte Schwankungen (Abbildung 17 rechts) ha-

ben zeitabhängige Amplituden (Schröder 2012, S. 16-17). Am Anfang der Zeitreihenanalyse steht somit die Auswahl eines Modells, das das Verhalten einer Zeitreihe möglichst genau abbildet (Schröder 2012, S. 20). Unter Verwendung von x_t (Beobachtungswert), a_t (Niveauwert), b_t (Trendanstieg/-abstieg), s_t (Saisonkomponente) und e_t (Zufallsabweichung) können die in Abbildung 17 dargestellten Zeitreihenmodelle wie folgt mathematisch beschrieben werden (Rößler et al. 2016, S. 236; Schröder 2012, S. 19):

- Konstantes Modell ohne periodische Schwankungen:

$$x_t = a_t + e_t \quad (3.2)$$

- Konstantes Modell mit additiv überlagerten periodischen Schwankungen:

$$x_t = a_t + s_t + e_t \quad (3.3)$$

- Konstantes Modell mit multiplikativ überlagerten periodischen Schwankungen:

$$x_t = a_t \cdot s_t + e_t \quad (3.4)$$

- Trendförmiges Modell ohne periodische Schwankungen:

$$x_t = a_t + b_t \cdot t + e_t \quad (3.5)$$

- Trendförmiges Modell mit additiv überlagerten periodischen Schwankungen:

$$x_t = a_t + b_t \cdot t + s_t + e_t \quad (3.6)$$

- Trendförmiges Modell mit multiplikativ überlagerten periodischen Schwankungen:

$$x_t = (a_t + b_t \cdot t) \cdot s_t + e_t \quad (3.7)$$

3.5 Prognoseverfahren

Prognoseverfahren können in qualitative und quantitative Verfahren unterteilt werden (siehe Abbildung 18). *Qualitative Prognosen* sind subjektive Beurteilungen. Sie sind fast immer verbal und selten mit Zahlen verbunden. Die wichtigsten qualitativen Techniken sind Befragungen, experimentelle Feldstudien und historische Analogien (Vogel 2015, S. 11-12). Aufgrund der mangelnden Rationalisierbarkeit dieser Methoden spielen sie für diese Arbeit keine Rolle und werden nicht weiter betrachtet. *Quantitative Verfahren* sind zumeist Rechenverfahren und werten empirische Daten

aus. Mit ihrer Hilfe können entweder zukünftige Zustände (Zeitreihenanalysen) oder Wirkungen von Einflussfaktoren (Wirkungsprognosen) beschrieben werden.

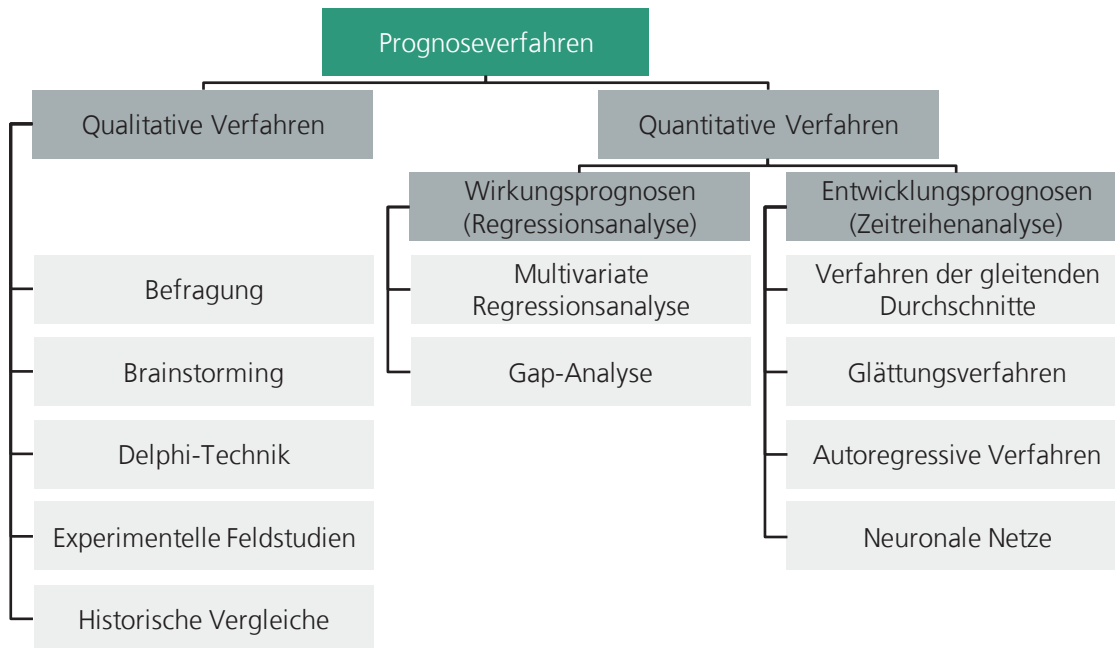


Abbildung 18: Prognoseverfahren (in Anlehnung an Vogel 2015, S. 11 und S. 13 und Schröder 2012, S. 20-21)

Wirkungsprognosen basieren auf Analysen kausaler Effekte von einstellbaren Variablen. So lässt sich beispielsweise die Wirkung der beiden Faktoren Preis und Werbeausgaben auf den Absatz untersuchen. Dagegen erlauben *Zeitreihenanalysen* die Vorhersage zukünftiger Zustände (Vogel 2015, S. 13). Da im Rahmen des zu entwickelnden Verfahrens Informationen über zukünftige Systemzustände in die Lagerplatzvergabeentscheidung einfließen sollen, liegen *Zeitreihenanalysen* im Fokus dieser Arbeit. Bei den *Verfahren der gleitenden Durchschnitte* wird aus einer bestimmten Anzahl Beobachtungswerte ein Mittelwert gebildet. Dabei wird jeweils die jüngste Beobachtung statt der ältesten in die Berechnung einbezogen (Schröder 2012, S. 22). Dagegen fließen die Beobachtungswerte bei den *Glättungsverfahren* unterschiedlich stark gewichtet in die Prognose ein (Schröder 2012, S. 26). Bei den *autoregressiven Verfahren* werden Zeitreihen durch stochastische Prozesse modelliert (Mertens et al. 2012, S. 6). *Neuronale Netze* bilden die Intelligenz biologischer

Systeme nach und erlernen selbstständig den funktionalen Zusammenhang des zugrunde liegenden Prozesses (Mertens et al. 2012, S. 5). Während sich die beiden erstgenannten Verfahrenstypen relativ einfacher mathematischer Methoden bedienen, sind die beiden letztgenannten Verfahrenstypen vergleichsweise kompliziert und mathematisch weniger geschulten Mitarbeitern aus der Praxis nur schwer verständlich zu machen (Mertens et al. 2012, S. 4-5). Da die Praxistauglichkeit des zu entwickelnden Verfahrens in Kapitel 1.3 als wesentliches Anforderungskriterium abgeleitet wurde, wird im Folgenden nicht weiter auf die autoregressiven Verfahren und die Neuronalen Netze eingegangen.

Um zukünftige Zustände vorhersagen zu können, ist ein Verfahren (Abbildung 19) auszuwählen, das die unbekanntenen Koeffizienten des formulierten Modells schätzt.

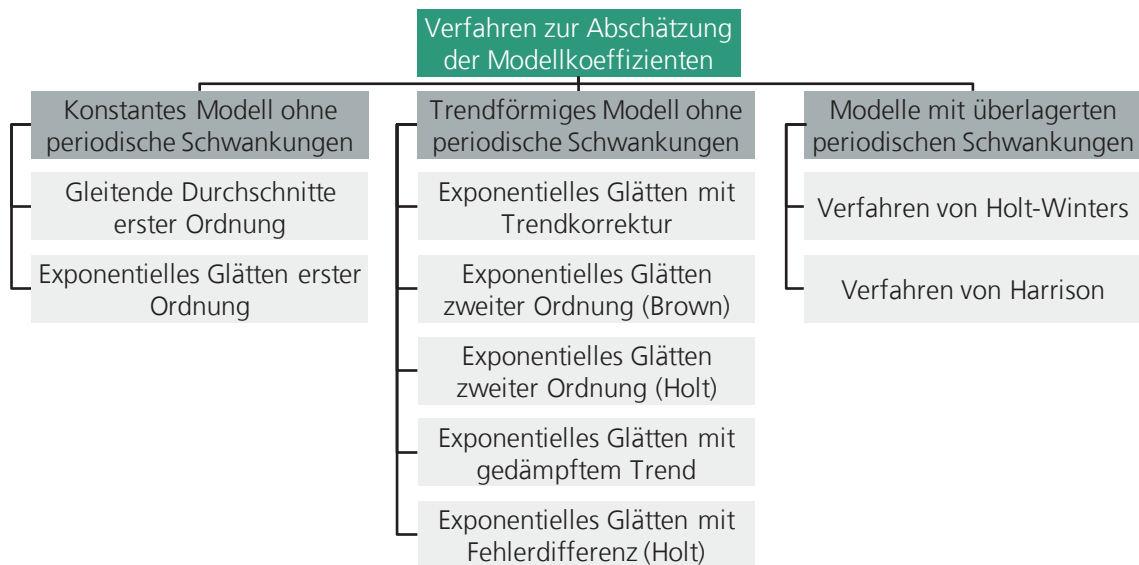


Abbildung 19: Prognoseverfahren zur Abschätzung der Modellkoeffizienten (in Anlehnung an Schröder 2012, S. 20 und Schuhr 2012, S. 48)

Die Eignung eines Verfahrens richtet sich nach dem verwendeten Modell und damit nach dem zugrunde liegenden Prozess (Schröder 2012, S. 20). Daher werden im Folgenden, ausgehend von dem eingesetzten Modell, geeignete Prognoseverfahren vorgestellt. Der Index T bezeichnet nachfolgend die Gegenwart und $(T + \tau)$ die Vorhersageperiode. Das Symbol $\hat{}$ kennzeichnet einen Schätzwert.

Konstantes Modell ohne periodische Schwankungen

Zur Abschätzung von a_t im *konstanten Modell ohne periodische Schwankungen* (Abbildung 17 links oben) kann das Verfahren der gleitenden Durchschnitte erster Ordnung oder das exponentielle Glätten erster Ordnung eingesetzt werden. Der Schätzwert \hat{a}_T wird als Prognosewert $\hat{x}_{T,\tau} = \hat{a}_T$ verwendet. Beim *Verfahren der gleitenden Durchschnitte erster Ordnung* wird aus einer bestimmten Anzahl von Beobachtungswerten x_t ein Mittelwert gebildet und als Prognosewert für die Periode $(T + \tau)$ verwendet. In jeder Periode wird jeweils der jüngste anstatt des ältesten Beobachtungswertes in die Berechnung einbezogen. Hauptnachteil dieser Methode ist, dass alle Werte mit demselben Gewicht in die Prognose einfließen. Dagegen ist es plausibel anzunehmen, dass Daten aus der jüngeren Vergangenheit die Entwicklung der Zeitreihe besser repräsentieren als ältere Daten. Eine unterschiedliche Gewichtung kann so zur Verbesserung der Prognose beitragen (Schröder 2012, S. 20-22).

Viele Vorhersagemethoden basieren auf dem Prinzip des *exponentiellen Glättens* (Schröder 2012, S. 24). Es unterstellt, dass das Gewicht der Zeitreihenwerte für die Prognose mit zunehmendem Alter abnimmt. Gleichzeitig wird der Prognosefehler der Gegenwart in die Berechnung miteinbezogen (Kühne et al. 2011, S. 324). Durch die Einführung eines Glättungsparameters α ergeben sich zum Zeitpunkt t für das konstante Modell folgende Gleichungen (Kühne et al. 2011, S. 324):

$$\hat{a}_t = \hat{a}_{t-1} + \alpha \cdot (x_t - \hat{a}_{t-1}) \quad (3.8)$$

bzw. nach Umformulierung:

$$\hat{a}_t = \alpha \cdot x_t + (1 - \alpha) \cdot \hat{a}_{t-1} \quad (3.9)$$

mit $0 < \alpha \leq 1$.

Die Wahl des Glättungsparameters legt fest, welcher Anteil des Prognosefehlers in die Prognose eingeht (Hüttner 1986, S. 56). Für große α gehen die jüngsten Beobachtungswerte stark in die Prognose ein. Umgekehrt gewichten kleine α Beobachtungswerte, die weiter in der Vergangenheit liegen, stärker. Ändert sich die Zeitreihe grundlegend, ist die Wahl eines hohen α sinnvoll. Handelt es sich bei dem

letzten Zeitreihenwert um eine starke einmalige Abweichung, sollte ein kleines α zu besseren Ergebnissen führen (Hansmann 1983, S. 31). In praktischen Anwendungsfällen wird meist ein Wert zwischen 0,1 und 0,3 gewählt (Glaser 1986, S 7).

Trendförmiges Modell ohne periodische Schwankungen

Für die Schätzung der beiden Koeffizienten im *trendförmigen Modell ohne periodische Schwankungen* (Abbildung 17 links unten) können das exponentielle Glätten zweiter Ordnung oder das exponentielle Glätten erster Ordnung mit Trendkorrektur eingesetzt werden (Schröder 2012, S. 20). Das exponentielle Glätten zweiter Ordnung (nach Brown) unterscheidet sich von dem exponentiellen Glätten mit Trendkorrektur lediglich in der Organisation des Rechenvorgangs, nicht aber in den Ergebnissen (Schröder 2012, S. 41). Beide Verfahren sind aufgrund ihres geringen Parametrierungsaufwands, der aus dem Einsatz eines einzelnen Glättungsparameters resultiert, sehr verbreitet (Kühne et al. 2011, S. 325; Schröder 2012, S. 36). Neben diesen beiden Verfahren existieren noch das Zwei-Parameter-Modell nach Holt, das Zwei-Parameter-Modell mit gedämpftem Trend sowie das Drei-Parameter-Modell mit Fehlerdifferenz Ausdruck (Schröder 2012, S. 21). Da diese Verfahren aufgrund des Einsatzes mehrerer Glättungsparameter einen deutlich höheren Parametrierungsaufwand haben (Schröder 2012, S. 41), werden Sie im Folgenden nicht weiter betrachtet. Zur Durchführung der *exponentiellen Glättung mit Trendkorrektur* sind folgende Gleichungen notwendig (Röblier et al. 2016, S. 236):

$$\hat{a}_t = \alpha \cdot x_t + (1 - \alpha) \cdot \hat{a}_{t-1} \quad (3.10)$$

$$\hat{b}_t = \alpha \cdot (\hat{a}_t - \hat{a}_{t-1}) + (1 - \alpha) \cdot \hat{b}_{t-1} \quad (3.11)$$

$$\hat{x}_{T,\tau} = \hat{a}_t + \hat{b}_t \cdot \tau + \frac{(1 - \alpha)}{\alpha} \cdot \hat{b}_t \quad (3.12)$$

Zu Beginn des Verfahrens sind Startwerte für die Glättungswerte erforderlich. Für deren Festlegung gibt es in der Literatur keine einheitliche Empfehlung (Weber 1990, S. 200). Allerdings wird der Einfluss der Startwerte im Laufe des Verfahrens durch die Gewichtung zunehmend geringer. Daher sollte ein grober Schätzwert aus-

reichend sein (Tempelmeier 2006, S. 49). So bietet es sich an, für \hat{a}_2 den Beobachtungswert aus der zweiten Periode und für \hat{b}_2 die mit dem Glättungsparameter α multiplizierte Differenz aus dem zweiten und dem ersten Beobachtungswert zu wählen (Rößler et al. 2016, S. 236).

Modelle mit überlagerten periodischen Schwankungen

Ist der Erhebungsabstand der Beobachtungswerte ein Jahresbruchteil, sind häufig Saisonalitäten zu beobachten (Rinne et al. 2002, S. 313). Es treten offensichtliche Unterschiede zwischen den Saisons auf (Franses 1998, S. 15). Diese Unterschiede wiederholen sich relativ gleichförmig jedes Jahr (Schuhr 2012, S. 47). Weist eine Zeitreihe überlagerte periodische Schwankungen auf, ist neben dem Niveauwert und ggf. dem Trendanstieg bzw. -abstieg noch die Saisonkomponente zu schätzen. Für *Zeitreihen mit überlagerten periodischen Schwankungen* (Abbildung 17 Mitte und rechts) wird oft das *Verfahren von Holt-Winters* eingesetzt, da es sich in der Praxis gegenüber dem Verfahren von Harrison als überlegen erwiesen hat (Kühne et al. 2011, S. 327). Das ebenfalls für diese Art von Zeitreihen entwickelte *Verfahren von Harrison* (1965) wird daher nicht weiter beschrieben. Im Holt-Winters-Verfahren wird die Saisonzyklusperiode mit L gekennzeichnet. Im Falle einer Reihe mit Monatsabstand der Erhebungszeitpunkte gilt $L = 12$. Werden die Beobachtungswerte quartalsweise erhoben, nimmt L den Wert 4 an. Im Gegensatz zu den bereits vorgestellten Verfahren kommen zur Schätzung der drei Koeffizienten des Modells (3.6 und 3.7) beim Holt-Winters-Verfahren drei Glättungsparameter (α , β und γ) zum Einsatz (Schuhr 2012, S. 49-50). Im Fall additiv überlagerter Schwankungen (Abbildung 17 Mitte unten) werden folgende Gleichungen zur Berechnung der Prognosekoeffizienten und der Prognosewerte verwendet (Schuhr 2012, S. 53):

$$\hat{a}_t = \alpha \cdot (x_t - \hat{c}_{t-L}) + (1 - \alpha) \cdot (\hat{a}_{t-1} + \hat{b}_{t-1}) \quad (3.13)$$

$$\hat{b}_t = \beta \cdot (\hat{a}_t - \hat{a}_{t-1}) + (1 - \beta) \cdot \hat{b}_{t-1} \quad (3.14)$$

$$\hat{c}_t = \gamma \cdot (x_t - \hat{a}_t) + (1 - \gamma) \cdot \hat{c}_{t-L} \quad (3.15)$$

$$\hat{x}_{T,\tau} = \hat{a}_t + \hat{b}_t \cdot \tau + \hat{c}_{T+\tau-\lfloor \frac{\tau}{L} \rfloor L} \quad (3.16)$$

In (3.16) symbolisiert $\left\lceil \frac{\tau}{L} \right\rceil$ die kleinste natürliche Zahl, die nicht kleiner als der Quotient $\frac{\tau}{L}$ ist (Schuhr 2012, S. 50). Im multiplikativen Fall (Abbildung 17 rechts unten) ergeben sich die Gleichungen (Schuhr 2012, S. 53):

$$\hat{a}_t = \alpha \cdot \frac{x_t}{\hat{c}_{t-L}} + (1 - \alpha) \cdot (\hat{a}_{t-1} + \hat{b}_{t-1}) \quad (3.17)$$

$$\hat{b}_t = \beta \cdot (\hat{a}_t - \hat{a}_{t-1}) + (1 - \beta) \cdot \hat{b}_{t-1} \quad (3.18)$$

$$\hat{c}_t = \gamma \cdot \frac{x_t}{\hat{a}_t} + (1 - \gamma) \cdot \hat{c}_{t-L} \quad (3.19)$$

$$\hat{x}_{T,\tau} = (\hat{a}_t + \hat{b}_t \cdot \tau) \cdot \hat{c}_{T+\tau-\left\lceil \frac{\tau}{L} \right\rceil L} \quad (3.20)$$

Im Fall des konstanten Modells mit periodischen Schwankungen (Abbildung 17 Mitte und rechts oben) entfallen jeweils die Trendkomponenten. Auch beim Holt-Winters-Verfahren müssen für die Prognosekoeffizienten zu Beginn geeignete Startwerte festgelegt und Glättungsparameter bestimmt werden (Schuhr 2012, S. 52). Ein praxistauglicher Ansatz für den additiven Fall findet sich in Rößler et al. (2016, S. 236):

$$\hat{a}_{L+1} = x_{L+1} \quad (3.21)$$

$$\hat{b}_{L+1} = \frac{x_{L+1} - x_1}{L} \quad (3.22)$$

$$\hat{c}_t = x_t - (\hat{b}_{L+1} \cdot t + x_1 - \hat{b}_{L+1}) \quad (3.23)$$

Im multiplikativen Fall ändert sich Gleichung (3.23) zu (Rößler et al. 2016, S. 236):

$$\hat{c}_t = \frac{x_t}{\hat{b}_{L+1} \cdot t + x_1 - \hat{b}_{L+1}} \quad (3.24)$$

Neben einfachen heuristischen Verfahren können auch komplexere Optimierungsverfahren zur Bestimmung der Startwerte eingesetzt werden (Schuhr 2012, S. 55). Die Berechnung der Prognosekoeffizienten wird durch die im Intervall]0,1[wählbaren Glättungsparameter α , β und γ gesteuert. Bei sehr kleinen Werten nahe null beeinflussen die Beobachtungswerte am historischen Rand die Prognosekoeffizienten annähernd so stark wie die Werte am aktuellen Zeitreihenrand. Das Gewicht der jungen Beobachtungswerte kann somit durch die Wahl größerer Glättungsparameter erhöht werden (Schuhr 2012, S. 56). Bei einem vermuteten stabilen Verlauf können alle drei Werte auf 0,1 gesetzt werden. Liegen stärkere zufällige Störungen vor,

sollte zumindest α erhöht werden (Götze 2000, S. 110). Grundsätzlich sollten bei starken Zufallsschwankungen kleine Glättungsparameter gewählt werden. Weist die Zeitreihe einen strukturellen Wandel der Trendentwicklung und/oder instabile Saisonschwankungen auf, sollten große Parameterwerte gewählt werden (Schuhr 2012, S. 56). Neben dem Einsatz von Optimierungsverfahren, wie etwa dem Hill-Climbing-Verfahren (Chatfield et al. 1988, S. 134), kann auch auf die Empfehlungen von Silver et al. (2017, S. 104) zurückgegriffen werden (siehe Tabelle 1). Sie schlagen für unterschiedliche Werte des Glättungsparameters, der in der exponentiellen Glättung erster Ordnung eingesetzt werden würde, entsprechende Parameterwerte für das Holt-Winters-Verfahren vor.

α^{expGl}	α^{HW}	β^{HW}	γ^{HW}
0,3	0,51	0,176	0,5
0,1	0,19	0,053	0,1
0,01	0,02	0,005	0,05

Tabelle 1: Empfohlene Werte für die Glättungsparameter (in Anlehnung an Silver et al. 2017, S. 104)

In umfangreichen Studien erwies sich das Holt-Winters-Verfahren als robust und weitgehend automatisierbar. Die Prognoseergebnisse sind im Allgemeinen nicht weniger zuverlässig als die Prognosen komplexerer und rechenaufwändigerer Alternativenverfahren. Speziell im Bereich der Lagerhaltungs- und Produktionsplanung, in der routinemäßig Prognosen für zukünftige Absatzzahlen von mehreren tausend Artikeln erstellt werden müssen, ist das Verfahren von Holt-Winters eine attraktive Prognosemethode (Schuhr 2012, S. 70).

4 Reflexion bestehender Ansätze

Nachdem in den vorangegangenen Kapiteln die grundlegenden Begriffe beschrieben wurden, werden in diesem Kapitel relevante Lösungsansätze für die in Kapitel 1.2 formulierte Problemstellung analysiert. Die Analyse erfolgt zweigeteilt. Zunächst werden Ansätze untersucht, die mehr als ein operatives Planungsproblem lösen. Anschließend werden Lösungsansätze für das dynamische Lagerplatzvergabeproblem analysiert und abschließend die Defizite der Lösungsansätze zusammengefasst. Die Anforderungen an das Lösungsverfahren wurden in Kapitel 1.3 definiert:

- *Wegstreckenreduktion*
- *Interdependenz*
- *Dynamik*
- *Umlagerungsaufwand*
- *Praxistauglichkeit*

Der Betrachtungsgegenstand sind *manuelle Person-zur-Ware (PzW) Low-Level-Kommissioniersysteme*. Das Verfahren soll, wie in Kapitel 2.3.2 beschrieben, auch für *Multi-Block-Layouts* einsetzbar sein.

4.1 Ansätze zur gemeinsamen Lösung mehrerer Planungsprobleme

Won et al. (2005) entwickeln eine mathematische Formulierung für das kombinierte Auftragsbildungs- und Tourenplanungsproblem. Zur Lösung des Problems beschreiben sie zwei heuristische Algorithmen. Der erste Algorithmus löst beide Planungsprobleme sequentiell, der zweite löst diese kombiniert. Die Autoren gehen von der Annahme aus, dass jeder Kommissionierer einem Kommissioniergang zugeordnet ist und jeder Gang am Anfangspunkt ein eigenes Depot hat. Es findet keine gangübergreifende Kommissionierung statt. Entnahmen bzw. Bewegungen sind daher nur innerhalb eines Ganges möglich. Somit liegen eine andere Form des Tourenplanungsproblems und ein anderer Betrachtungsgegenstand vor. Die Optimierung erfolgt nach den beiden Zielgrößen Auftragsliegezeit und Kommissionierzeit und nicht nach der Wegstrecke. Das Lagerplatzvergabeproblem wird nicht betrachtet.

Tsai et al. (2008) formulieren ein Optimierungsmodell, in welchem neben den Kosten für die zurückgelegte Strecke auch Strafkosten für zu früh bzw. zu spät fertigestellte Aufträge berücksichtigt werden. Zur Lösung entwickeln sie einen zweistufigen Genetischen Algorithmus. Der erste Genetische Algorithmus soll die optimale Anzahl Batches und die Zuordnung der Aufträge zu den Batches finden. Der zweite Algorithmus ist ein Unterverfahren des ersten Algorithmus und sucht den wegstreckenoptimalen Routenplan für jeden Batch. Das Aufteilen eines Auftrags auf mehrere Batches wird zugelassen. Damit liegt jedoch eine andere Form des Auftragsbildungsproblems vor. Sollte das Batchvolumen die Kapazität des Kommissioniergeräts übersteigen, wird dies mit Strafkosten belegt. Der Ansatz löst das Auftragsbildungs- und Tourenplanungsproblem in einem gemeinsamen Verfahren. Allerdings wird nicht ausschließlich das Ziel der Wegstreckenreduktion verfolgt. Die Abhängigkeit der beiden Planungsprobleme mit dem Lagerplatzvergabeproblem wird nicht berücksichtigt. Die Validierung der Verfahren erfolgt ausschließlich mithilfe eines Ein-Block-Layouts. In den numerischen Experimenten werden zwar praxisnahe Problemgrößen von bis zu 250 Aufträgen untersucht, zur Berechnung der Lösung sind allerdings bis zu 130 Minuten notwendig.

Ene et al. (2012) berücksichtigen alle drei Planungsprobleme. In dem betrachteten Lager mit Zwei-Block-Layout werden Artikel für eine Montagelinie kommissioniert. Das Verfahren optimiert die Probleme in zwei Stufen mit dem Ziel, die Bewegungszeit zu minimieren: Im ersten Schritt werden eingehende Artikel mithilfe einer klassenbasierten Lagerplatzvergaberegeln den Lagerplätzen zugeordnet. Das Auftragsbildungs- und Tourenplanungsproblem wird mithilfe eines Genetischen Algorithmus gelöst. Der Ansatz berücksichtigt zwar alle drei Planungsprobleme, allerdings ist das betrachtete Lagerplatzvergabeproblem stark vereinfacht. Es werden lediglich neue Artikel, die bislang nicht gelagert wurden, eingeplant. Belegte Lagerplätze werden vor dem Optimierungslauf gesperrt. Diese Art der Lagerplatzvergabe ist nicht in der Lage, mit der zunehmend volatilen Kundennachfrage umzugehen, da einmal eingelagerte Artikel nicht mehr umgelagert werden und sich somit eine historisch ge-

wachsene Artikelanordnung verfestigt. Die Lösung des Auftragsbildungs- und Tourenplanungsproblems findet keinen Eingang in die Lösung des Lagerplatzvergabe-problems. Damit wird die Interdependenz der drei Probleme nicht berücksichtigt. *Kulak et al. (2012)* formulieren ein ganzzahliges Optimierungsmodell zur gemeinsamen Lösung des Auftragsbildungs- und Tourenplanungsproblems. Die Kapazitätsrestriktion wird in Form des maximalen Batchgewichts angegeben. Aufträge dürfen nicht auf mehrere Batches aufgeteilt werden. Zur Lösung entwickeln sie zwei Algorithmen, die sich jeweils aus einer Tabu Search-Heuristik und einem Seed-Algorithmus zusammensetzen. Die Verfahren wurden für Multi-Block-Layouts entwickelt. Zielgröße ist in diesem Ansatz die Minimierung der Kommissionierweglänge. Das Lagerplatzvergabeproblem wird von den Autoren nicht berücksichtigt. Mit dem Einsatz des Zopt-Algorithmus konnten innerhalb kurzer Rechenzeiten sehr gute Optimierungsergebnisse erzielt werden.

Azadnia et al. (2013) entwickeln eine Methode zur gemeinsamen Optimierung des Auftragsbildungs- und Tourenplanungsproblems. Es wird die Verspätung von Aufträgen minimiert. Betrachtet wird ein manuelles Person-zur-Ware-Kommissioniersystem mit Ein-Block-Layout. Aufträge dürfen die Kapazität des Kommissioniergeräts nicht überschreiten und nicht auf mehrere Batches aufgeteilt werden. Die Methode besteht aus vier Phasen: Im ersten Schritt wird die Korrelation zwischen den Aufträgen mithilfe einer Data-Mining-Methode ermittelt. In die Korrelationsberechnung fließen neben gleichen Artikeln auch gleiche Liefertermine ein. Im nächsten Schritt werden die Aufträge Batches zugeordnet. Die Zuordnung erfolgt so, dass die Korrelation zwischen den Aufträgen maximiert wird. Anschließend wird mithilfe eines Genetischen Algorithmus eine wegstreckenminimale Kommissioniertour je Batch ermittelt. Abschließend ermittelt ein weiterer Genetischer Algorithmus die Batchsequenz, also die Reihenfolge, in der die Kommissionieraufträge freigegeben werden sollen. Der letzte Schritt erfolgt mit dem Ziel, die Verspätung von Aufträgen zu vermeiden. Der Optimierungsansatz berücksichtigt das Lagerplatzvergabeproblem nicht. Der Ansatz nutzt verschiedene Optimierungsverfahren und Methoden und ist entsprechend komplex. Die Verfahren wurden nicht für den Einsatz in Multi-Block-

Layouts entwickelt und verfolgen ein anderes Optimierungsziel. Rechenzeiten werden von den Autoren nicht angegeben. Durch die Kombination mehrerer Metaheuristiken und einer Data-Mining-Methode ist allerdings von hohen Rechenaufwänden auszugehen. Nicht zuletzt, da der Tourenplanungsalgorithmus durch die Integration in den Auftragsbildungsalgorithmus mehrfach durchlaufen werden muss.

Grosse et al. (2014) entwickeln ein Verfahren für das kombinierte Auftragsbildungs- und Tourenplanungsproblem, mit dem Anspruch einer möglichst einfachen Implementierung, um so die Akzeptanz in der Praxis zu erhöhen. Das untersuchte Auftragsbildungsproblem unterscheidet sich von dem dieser Arbeit, da ein Auftrag auf mehrere Batches aufgeteilt werden kann. Hierzu entwickeln die Autoren einen SA-Algorithmus für das Auftragsbildungsproblem und kombinieren ihn mit vier verschiedenen Routingstrategien. Die Kapazität wird in dem maximal zu transportierenden Gewicht angegeben. Minimiert wird die Kommissionierweglänge in einem Ein-Block-Layout. Die Autoren wählen ein mehrstufiges Vorgehen: Zunächst wird mithilfe verschiedener Konstruktionsheuristiken eine zulässige Startlösung für das Auftragsbildungsproblem ermittelt. Der SA-Algorithmus optimiert diese Lösung, indem Artikel zwischen den Batches getauscht werden. Anschließend wird für jeden Batch eine Kommissioniertour mithilfe einer Routingstrategie bestimmt. Das untersuchte Auftragsbildungsproblem entspricht nicht dem in dieser Arbeit betrachteten Problem. Für das Lagerplatzvergabeprobem wird kein Lösungsverfahren entwickelt. Es wird lediglich beschrieben, wie die Artikel im Rahmen der numerischen Experimente angeordnet werden.

Dem Optimierungsansatz von *Matusiak et al. (2014)* liegt eine spezielle Planungssituation zugrunde: Jeder Auftrag muss in einer fest vorgegebenen Reihenfolge kommissioniert werden, da die zu beliefernden Verkaufsläden Artikel mit dem Family Grouping anordnen. Das Verfahren wurde für ein Zwei-Block-Layout entwickelt, das mehrere Abgabepunkte für die kommissionierten Artikel vorsieht. Der entwickelte Ansatz besteht aus zwei Algorithmen: Einem A*-Algorithmus für das Tourenplanungsproblem und einem SA-Algorithmus für das Auftragsbildungsproblem. Das

Verfahren wurde für eine Planungssituation entwickelt, in der die Kommissionierreihenfolge für jeden Auftrag zu Beginn der Optimierung bereits fest vorgegeben ist. Das Lagerplatzvergabeproblem findet keine Berücksichtigung.

Chen et al. (2015) formulieren ein nichtlineares gemischt-ganzzahliges Optimierungsmodell zur Lösung des Auftragsbildungs- und Tourenplanungsproblems. Ziel ist die Minimierung der Verspätung von Aufträgen. Zur Lösung des Problems entwickeln sie einen Genetischen Algorithmus, der die Auftragsbildung optimiert und einen Ameisenalgorithmus zur Lösung des Tourenplanungsproblems, der in den Genetischen Algorithmus eingebettet ist. Das Optimierungsverfahren nutzt ein komplexes nichtlineares Optimierungsmodell und einen aufwändigen Optimierungsalgorithmus. Dies zeigt sich auch in der Validierung: Die größte untersuchte Probleminstanz umfasst acht Aufträge mit zehn Artikeln und einer Kommissioniergerätkapazität von fünf Artikeln. Das zugrundeliegende Lager mit Ein-Block-Layout hat zwölf Lagerplätze. Die Autoren kommen zu dem Schluss, dass die Rechenzeit problematisch ist. Das Lagerplatzvergabeproblem wird nicht untersucht.

Cheng et al. (2015) nutzen das von *Kulak et al. (2012)* formulierte Optimierungsmodell und entwickeln einen Algorithmus, der aus einem PSO-Algorithmus und einem Ameisenalgorithmus besteht. Der PSO-Algorithmus bestimmt die Anzahl notwendiger Batches und die Zuordnung von Aufträgen zu Batches. Je Batch wird mithilfe des Ameisenalgorithmus die kürzeste Kommissioniertour berechnet. Die Methode wird mit dem von *Tsai et al. (2008)* entwickelten Genetischen Algorithmus verglichen. Im Rahmen einer Simulationsstudie kommen die Autoren zu dem Ergebnis, dass das Verfahren dem Genetischen Algorithmus sowohl in Bezug auf das Optimierungsergebnis, als auch in Bezug auf die Rechenzeit überlegen ist. Allerdings wird das Verfahren mithilfe eines Lagers validiert, das nur zwölf Lagerplätze beinhaltet, die in einem Ein-Block-Layout angeordnet sind. Die Lagerplatzvergabe findet keine Berücksichtigung. Die Autoren machen keine Angabe bezüglich der Rechenzeiten. Allerdings ist durch die Integration des Ameisenalgorithmus in den PSO-Algorithmus von erhöhten Rechenaufwänden auszugehen. Der Einsatz des Verfahrens in einem Multi-Block-Layout ist nicht vorgesehen.

Lin et al. (2016) entwickeln einen verbesserten PSO-Algorithmus für das kombinierte Auftragsbildungs- und Tourenplanungsproblem. Sie setzen eine alternative Lösungscodierungsform ein. Das Lager wird als Koordinatennetz dargestellt. Jeder Auftrag wird symbolisch mittels seines Auftragsmittelpunkts in das Netz eingetragen. Der Mittelpunkt basiert auf den Lagerplätzen der enthaltenen Artikel. Die Batches werden mit ihrem Batchmittelpunkt in das Netz eingetragen. Die Entfernung zwischen zwei Mittelpunkten dient zur Bewertung der Ähnlichkeit von Aufträgen und zur Zuteilung der Aufträge zu Batches. Optimierungskriterium ist die Kommissionierweglänge. Der PSO-Algorithmus speichert schlechte Positionen und berücksichtigt diese in der Partikelaktualisierung. Das Verfahren kann in Multi-Block-Layouts eingesetzt werden. Bei der Validierung in Form von numerischen Studien wird ein Lager mit 16 Lagerplätzen untersucht. In der Fallstudie wird eine Lösung für 47 Aufträge berechnet. Das Lagerplatzvergabeprobem wird nicht betrachtet. Die Komplexität des Verfahrens ist nicht zuletzt aufgrund der Codierungsform als hoch zu bewerten.

Menéndez et al. (2017) entwickeln einen Variable-Neighborhood-Search-Algorithmus für das kombinierte Auftragsbildungs- und Tourenplanungsproblem. Als Routingstrategie kommt die Combined-Heuristik zum Einsatz. Minimiert wird die Verzugszeit der Aufträge. Das Verfahren wurde für ein Ein-Block-Layout entwickelt. Auch dieser Ansatz löst das Lagerplatzvergabeprobem nicht. Das Verfahren wird mithilfe relativ kleiner Problemgrößen von 20-80 Aufträgen validiert. Es sind bis zu 300 Sekunden notwendig, um eine Problem Instanz mit 80 Aufträgen in einem Ein-Block-Layout zu lösen.

Das Verfahren von *Scholz et al. (2017a)* basiert auf einem Iterated-Local-Search-Algorithmus. Das Verfahren wurde für Multi-Block-Layouts entwickelt. Mit dem Algorithmus können das Auftragsbildungs- und das Tourenplanungsproblem gelöst werden. Hierzu wird zunächst mithilfe der First-Come-First-Served-Heuristik eine zulässige Anfangslösung für das Auftragsbildungsproblem bestimmt. Diese wird in der Local-Search-Phase bis zum Erreichen eines lokalen Optimums verbessert. In einer sich anschließenden Störungsphase wird das lokale Optimum verlassen und erneut

in der Local-Search-Phase optimiert. In den Iterated-Local-Search-Algorithmus wurden verschiedene Routingalgorithmen integriert. Die Autoren geben zu bedenken, dass durch die Integration des exakten Routingalgorithmus in den Auftragsbildungsalgorithmus die Rechenzeiten stark ansteigen, da der Algorithmus mehrfach durchlaufen werden muss. Auch dieser Ansatz berücksichtigt das Lagerplatzvergabeproblem nicht. Das Verfahren wird mithilfe relativ kleiner Problemgrößen von 20-80 Aufträgen validiert.

Scholz et al. (2017b) formulieren ein Optimierungsmodell für das Auftragsbildungs- und Tourenplanungsproblem. Minimiert wird die Verzugszeit der Aufträge. Zur Lösung des Problems wird ein Variable-Neighborhood-Descent-Algorithmus eingesetzt. Das Tourenplanungsproblem wird mithilfe der Combined-Heuristik gelöst. Das Verfahren kann auch in Multi-Block-Layouts eingesetzt werden. Es wird im Zuge von numerischen Experimenten mit bis zu 200 Aufträgen validiert. Hierfür sind Rechenzeiten von bis zu 60 Minuten erforderlich.

Valle et al. (2017) entwickeln ein ganzzahliges Optimierungsmodell für das kombinierte Auftragsbildungs- und Tourenplanungsproblem. Minimiert werden soll die Kommissionierweglänge. Die Autoren entwickeln Branch-and-Cut-Algorithmen für kleine Instanzen, um das formulierte Problem optimal zu lösen. Das von den Autoren entwickelte optimale Verfahren kann nur bis zu einer Problemgröße von 20 Aufträgen eingesetzt werden, ist äußerst komplex, rechenzeitintensiv und damit nicht für praxisrelevante Problemgrößen einsetzbar.

4.2 Ansätze zur Lösung des dynamischen Lagerplatzvergabeproblems

Nachdem im vorangegangenen Kapitel Ansätze vorgestellt wurden, die mehrere Planungsprobleme lösen, werden in diesem Kapitel Methoden untersucht, die zur Lösung des dynamischen Lagerplatzvergabeproblems eingesetzt werden können. Es werden nur Methoden vorgestellt, die für manuelle Person-zur-Ware-Systeme entwickelt wurden. Verfahren, die für automatische Kommissioniersysteme oder für Reservelager entwickelt wurden, sind für diese Arbeit nicht relevant und werden nicht

weiter untersucht. Verfahren für automatische Kommissioniersysteme bzw. für Reservelager finden sich z.B. in Christofides et al. (1973), Jaikumar et al. (1990), Muralidharan et al. (1995), Moon et al. (2001), Carlo et al. (2010), Chen et al. (2011), Ang et al. (2012), Pazour et al. (2015) und Quintanilla et al. (2015).

Sadiq et al. (1996) entwickeln ein dynamisches Verfahren, das Artikelkorrelationen berücksichtigt. Artikel, die häufig gemeinsam kommissioniert werden, werden im Zuge des Family Groupings möglichst nah beieinander gelagert. Das Verfahren gliedert sich in zwei Phasen. In der globalen Zuordnungsphase wird entschieden, welche Artikel im Kommissionierlager eingelagert und welche aufgrund von Platzmangel nur im Reservelager vorgehalten werden. Es werden sowohl historische Daten als auch Prognosedaten genutzt. Die Prognose ist nicht Teil des Verfahrens. In der lokalen Zuordnungsphase werden paarweise Artikelkorrelationen bestimmt und in einer Korrelationsmatrix gespeichert. Basierend auf dieser Matrix werden mithilfe zweier Cluster-Algorithmen die Artikelcluster gebildet. Die Cluster werden festen Lagerplätzen zugewiesen. Die paarweisen Wechsel werden zwischen gleich großen oder benachbarten Clustern durchgeführt. Berücksichtigt werden sowohl der Umlagerungsaufwand, als auch die Kommissionierweglänge. Das Verfahren ist allerdings nur für ein einreihiges Kommissioniersystem und nicht für das in dieser Arbeit betrachtete System mit mehreren Kommissioniergängen einsetzbar. Das kombinierte Auftragsbildungs- und Tourenplanungsproblem wird nicht betrachtet.

Pierre et al. (2003) entwickeln einen dynamischen klassenbasierten Lagerplatzvergabealgorithmus für manuelle Person-zur-Ware-Kommissioniersysteme. Artikel werden, basierend auf der Anzahl Positionen je Periode, einer der drei Klassen zugeteilt. Zunächst wird die Klassifizierung der Artikel überprüft und bestimmt, welche Artikel in eine andere Klasse eingeteilt werden müssen. Die Analyse basiert auf der historischen Nachfrageintensität der Artikel. Prognosedaten werden nicht verwendet. Die falsch gelagerten Artikel erhalten einen Prioritätswert, der sich aus drei Faktoren berechnet. Die Prioritätswerte werden in einer Prioritätswertmatrix gespeichert. Im nächsten Schritt wird sichergestellt, dass die so gefundene Neuklassifizierung durch-

führbar ist. Ist eine Klasse überbelegt, müssen die Artikel mit den niedrigsten Prioritätswerten aus der Matrix entfernt werden. Anschließend werden die Artikel innerhalb ihrer Klasse basierend auf ihrer historischen Nachfrageintensität den Lagerplätzen so zugewiesen, dass die am häufigsten zu kommissionierenden Artikel möglichst nah zum Depot gelagert werden. Es kann nicht sichergestellt werden, dass die Umlagerungen tatsächlich vorteilhaft sind, da kein Vergleich zwischen Umlagerungsaufwand und potentieller Wegeinsparung vorgenommen wird. Die Autoren lagern so lange Artikel um, bis eine maximale Umlagerungszeit erreicht wird. Das Verfahren ist für den Einsatz in einem Multi-Block-Layout geeignet. Das kombinierte Auftragsbildungs- und Tourenplanungsproblem wird nicht betrachtet.

Neuhäuser (2013) stellt eine simulationsbasierte Methode vor, die die Ermittlung des optimalen Zeitpunktes für eine Lagerreorganisation und damit die Festlegung von Reorganisationszyklen ermöglicht. Es wird nicht nur die Lagerplatzvergabe, sondern auch die Nachschubversorgung vom Reservelager an das Kommissionierlager betrachtet. Beobachtungsgegenstand sind manuelle Low-Level-Kommissioniersysteme. Im Gegensatz zur vorliegenden Arbeit werden Palettenstellplätze untersucht. Die Methode wurde für den Lebensmitteleinzelhandel entwickelt. Die Lagerplatzvergabe erfolgt klassenbasiert. Die Entscheidung basiert nicht auf der Zugriffshäufigkeit, sondern auf qualitativen und quantitativen Stabilitätskriterien, um empfindliche Ware wie Eier vor schweren Gegenständen wie Konserven zu schützen. Die Bewertung einer Umlagerung erfolgt nicht ausschließlich nach der Wegeinsparung. Es wird ein umfangreiches kennzahlenbasiertes Bewertungssystem erarbeitet. Es besteht aus Kennzahlen zur Beschreibung der Kommissionierleistung, zur Beschreibung der Nachschubleistung und zur Beschreibung des Systemzustands. Es werden funktionale Abhängigkeiten zwischen den Kennzahlen approximiert und Algorithmen zur Zuordnung von Artikeln zu Lagerplätzen vorgestellt. Die Berechnung des optimalen Reorganisationszeitpunktes erfolgt unter Verwendung von Simulations- und Echtdateen und nicht mithilfe von Prognosedaten. Voraussetzung für die Anwendung der Methode ist eine detaillierte Kenntnis der Kosten- und Leistungsfunktionen eines Kommissioniersystems. Die spezifischen Abhängigkeiten des Systems

müssen untersucht und in die Methode integriert werden. Das Auftragsbildungs- und Tourenplanungsproblem ist nicht Gegenstand der Arbeit.

Kofler et al. (2015) erweitern den Ansatz von *Pierre et al. (2003)*, so dass neben einer klassenbasierten Lagerplatzvergabe auch andere Zuordnungsprinzipien genutzt werden können. Dies wird über die Einführung von virtuellen Zonen erreicht, die nicht mit den realen im Lager vorzufindenden Zonen übereinstimmen müssen. Mithilfe der Prioritätswertmatrix von *Pierre et al.* werden zu favorisierende Umlagerungsartikel identifiziert. Die Autoren lassen den Algorithmus jede Periode pauschal 50 Umlagerungen durchführen. Eine Prüfung, ob eine Umlagerung vorteilhaft ist, erfolgt nicht. Die Anzahl Umlagerungen wird nicht systematisch bestimmt. Das kombinierte Auftragsbildungs- und Tourenplanungsproblem ist nicht Gegenstand der Untersuchung.

Manzini et al. (2015) formulieren zwei kostenbasierte gemischt-ganzzahlige Optimierungsmodelle, um Entscheidungen zur Technologieauswahl und zur Zuteilung von Artikeln zu Lagerbereichen zu unterstützen. Dabei wird berücksichtigt, dass Artikel Lebenszyklen unterliegen. Das erste Optimierungsmodell soll die optimale Lagerkapazität einer Klasse festlegen und die Artikel den Klassen zuweisen. Das zweite Modell unterstützt bei strategischen Entscheidungen, wie der Auswahl eines Kommissioniersystems oder der Bestimmung der Zonengrößen und des Layouts. Die Autoren führen einen rollierenden Faktor zur Bestimmung der Popularität eines Artikels ein. Dieser wird periodenweise berechnet. Optimierungsmodell eins sieht Umlagerungsvorgänge zwischen den Lagerbereichen vor. Wie die Optimierungsprobleme in realen Anwendungsfällen gelöst werden können, ist nicht Gegenstand der Veröffentlichung. Das Auftragsbildungs- und Tourenplanungsproblem wird nicht betrachtet.

Li et al. (2016) modellieren einen Spezialfall des dynamischen Lagerplatzvergabeproblems als quadratisches Zuordnungsproblem. Artikel, die stark korreliert sind, sollen möglichst im selben Kommissioniergang bzw. in nah zueinander liegenden Gängen gelagert werden. Zusätzlich werden die Kommissioniergänge in einem Ein-

Block-Layout in drei Klassen eingeteilt. Die Artikel mit der höchsten Zugriffshäufigkeit sollen möglichst in den Gang eingelagert werden, der am nächsten zum Wareneingang liegt. Um die Artikelkorrelationen zu bestimmen, wird eine Data-Mining-Technik und zur Lösung des Zuordnungsproblems ein Genetischer Algorithmus eingesetzt. Das von den Autoren betrachtete Problem unterscheidet sich jedoch von dem in dieser Dissertation untersuchten Problem, da lediglich Lagerplätze für neu eintreffende Artikel gesucht werden. Umlagerungen werden nicht durchgeführt.

4.3 Zwischenfazit

Abbildung 20 fasst die Ergebnisse der Analyse zusammen. Es existiert noch kein Verfahren, welches die drei operativen Planungsprobleme der Kommissionierung ganzheitlich löst. Es konnte lediglich ein Ansatz (Ene et al. 2012) identifiziert werden, der alle drei Planungsprobleme betrachtet. Allerdings ist das dort untersuchte Lagerplatzvergabeproblem stark vereinfacht, denn es werden lediglich neuen Artikeln Lagerplätze zugewiesen. Die bestehende Artikelanordnung wird nicht optimiert und die Interdependenz mit den anderen Verfahren nicht berücksichtigt. Die meisten Verfahren, die für das kombinierte Auftragsbildungs- und Tourenplanungsproblem entwickelt wurden, sind sehr kompliziert und benötigen hohe Rechenaufwände. In einigen Verfahren werden andere Kommissioniersysteme untersucht. Andere können nicht in einem Multi-Block-Layout eingesetzt werden. Kulak et al. (2012) erzielen durch den Einsatz des 2opt-Algorithmus für das Tourenplanungsproblem gute Ergebnisse. Lin et al. (2016) und Cheng et al. (2015) ermitteln mithilfe der Partikelschwarmoptimierung gute Ergebnisse für das Auftragsbildungsproblem, allerdings nur für sehr kleine Probleminstanzen.

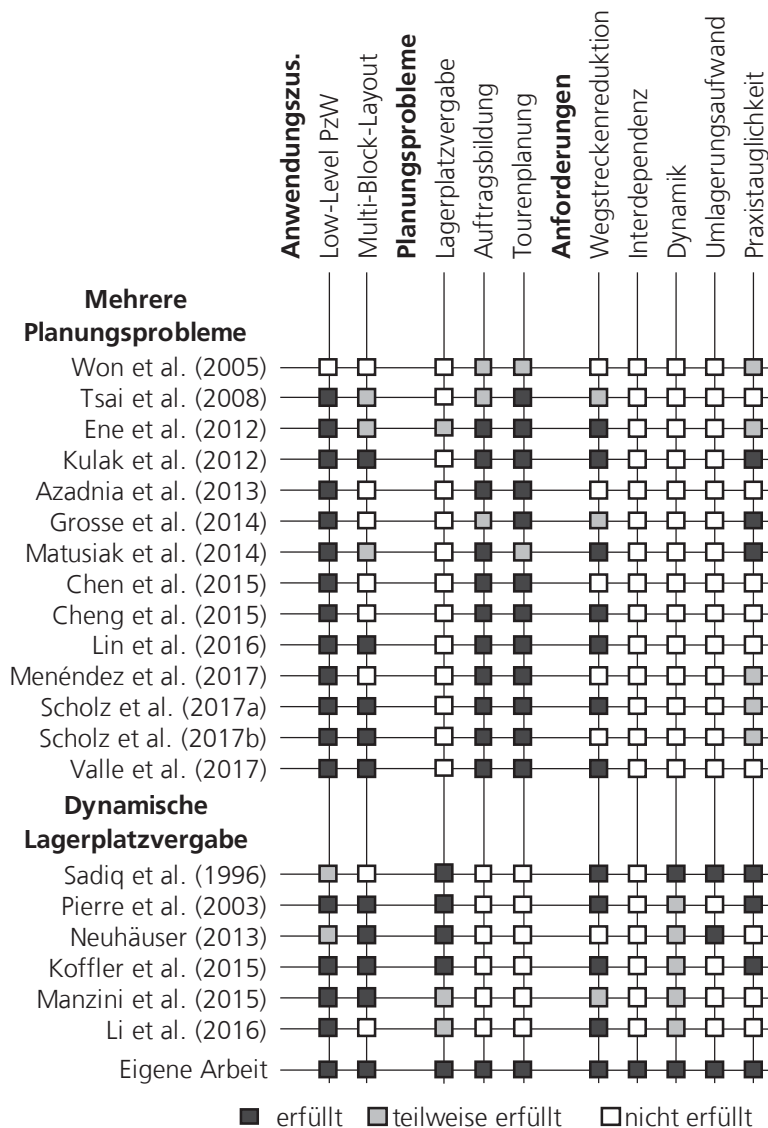


Abbildung 20: Bewertungsmatrix bestehender Ansätze

Es gibt nur wenige Arbeiten, die sich mit der dynamischen Variante des Lagerplatzvergabeproblems befassen. In Kapitel 3.3 wurden Prognosefähigkeiten als kritische Komponente zur Beherrschung der Dynamik identifiziert. Allerdings nutzen lediglich Sadiq et al. (1996) Prognosedaten. Den Umlagerungsaufwand berücksichtigen ausschließlich Sadiq et al. (1996) und Neuhäuser (2013). Das Verfahren von Sadiq et al. (1996) ist aber nur in einem einreihigen Kommissioniersystem einsetzbar. Auch der Ansatz von Neuhäuser (2013) wurde für ein anderes Kommissioniersystem konzipiert. Es setzt die detaillierte Kenntnis der Kosten- und Leistungsfunktionen eines Kommissioniersystems voraus, basiert auf einem komplizierten Kennzahlensystem

und einem umfangreichen Simulationsmodell. Pierre et al. (2003) entwickeln ein praxistaugliches Verfahren, das allerdings weder Prognosedaten noch Umlagerungsaufwände berücksichtigt. Damit kann der in Kapitel 1.2 postulierte Forschungsbedarf bestätigt werden.

Für die formulierte Problemstellung existiert noch kein geeignetes ganzheitliches Lösungsverfahren. Dessen Entwicklung und Validierung ist Gegenstand der nachfolgenden Kapitel.

5 Entwurf des Verfahrens

In diesem Kapitel werden das dynamische Lagerplatzvergabeproblem und das Auftragsbildungs- und Tourenplanungsproblem formal beschrieben und die dem Verfahren zugrunde liegenden Annahmen genannt. Anschließend wird der Aufbau des Verfahrens dargestellt und eine Verfahrensauswahl für die einzelnen Bausteine des Gesamtverfahrens vorgenommen.

5.1 Problembeschreibung

5.1.1 Annahmen

In Kapitel 2 wurden der Anwendungszusammenhang eingegrenzt und Gestaltungsregeln für das zu entwickelnde Verfahren abgeleitet. Daraus ergeben sich einige Annahmen für die Verfahren. Diese werden in der Literatur häufig getroffen, wie etwa die Arbeiten von Kulak et al. (2012, S. 57-59), Azadnia et al. (2013, S. 4), Koch (2014, S. 120-121) und Cheng et al. (2015, S. 807) zeigen. Wie in Kapitel 2 gezeigt worden ist, werden diese Systeme häufig in der Praxis eingesetzt. Den Verfahren liegen folgende Annahmen zugrunde:

1. Um den Sortieraufwand zu begrenzen, dürfen Aufträge nicht auf mehrere Batches aufgeteilt werden (siehe Kapitel 2.5).
2. Ein Auftrag übersteigt nicht die maximale Kapazität des Kommissioniergeräts.
3. Die Kapazität des Kommissioniergeräts wird mit dem maximalen Gewicht angegeben, das das Gerät transportieren kann.
4. Alle Aufträge sind zum Start des Auftragsbildungs- und Tourenplanungsproblems bekannt.
5. Es wird ein manuelles Person-zur-Ware Low-Level-Kommissioniersystem unterstellt.
6. Es liegt ein Block-Layout mit beliebig vielen Blocks vor.
7. Artikel können ohne wesentliche Positionsveränderung sowohl von der rechten als auch der linken Seite des Kommissioniergangs entnommen werden.

8. Kommissioniergänge können in beiden Richtungen begangen werden.
9. Kommissionierer können sich in einem Kommissioniergang überholen.
10. Vertikale Bewegungen können vernachlässigt werden.
11. Der Lagerplatz eines Artikels ist zu jedem Zeitpunkt bekannt.
12. Ein Lagerplatz enthält maximal eine Artikelnummer.
13. Eine Artikelnummer liegt in maximal einem Lagerplatz.
14. Jeder Artikel kann in jedem Lagerplatz gelagert werden.
15. Es wird eine klassenbasierte Lagerplatzvergabe unterstellt.
16. Ziel ist die Minimierung der Kommissionierweglänge die notwendig ist, um alle Aufträge zu kommissionieren.

5.1.2 Dynamisches Lagerplatzvergabeproblem

Am Ende jeder Periode sollen die Artikel identifiziert werden, die falsch gelagert werden. Ein Artikel ist falsch gelagert, wenn sein aktueller Lagerplatz vom optimalen Lagerplatz abweicht. Hierzu sind den falsch gelagerten Artikeln (zunächst hypothetisch) neue Lagerplätze zuzuweisen. Aus der Menge falsch gelagerter Artikel, sollen jene Artikel identifiziert werden, für die sich eine Umlagerung lohnt. Eine Umlagerung lohnt sich, wenn die zu erwartende zukünftige Wegstreckenreduktion im Untersuchungszeitraum den notwendigen Umlagerungsaufwand übersteigt. Ist dies der Fall, soll die Umlagerung durchgeführt werden. Es wird angenommen, dass sich der Umlagerungsaufwand aus drei Komponenten zusammensetzt: Dem physischen Aufwand zur Entnahme des Artikels aus dem Lagerplatz, dem administrativen Aufwand zur Verbuchung des neuen Lagerplatzes und dem Distanzaufwand zwischen neuem und altem Lagerplatz. Aus Vereinfachungsgründen werden der physische und der administrative Aufwand über die Gehgeschwindigkeit des Kommissionierers in die Einheit Kommissionierstrecke umgerechnet. Die Artikel sollen so im Lager angeordnet werden, dass die Weglänge, die zur Kommissionierung aller Aufträge im untersuchten Zeitraum zurückgelegt wird, unter Berücksichtigung des Umlagerungsaufwands minimiert wird. Das statische Lagerplatzvergabeproblem kann als quadratisches Zuordnungsproblem modelliert werden. Eine Formulierung für das

statische Problem findet sich in Reschke (2013, S. 39-42). Eine formale Darstellung des untersuchten dynamischen Problems kann durch die Erweiterung des statischen Problems um eine Mehrperiodenbetrachtung und den Umlagerungsaufwand gefunden werden.

Indizes und Mengen

- $m, n \in A$ Artikel, einschließlich des fiktiven, dem Depot zugeordneten Artikels, der durch den Index „0“ repräsentiert wird
- $i, j \in V$ Lagerplätze, wobei das Depot durch den Index „0“ repräsentiert wird
- $t \in U$ Untersuchte Perioden in der Zukunft ohne aktuelle Periode „0“

Parameter

- $d_{i,j}$ Entfernung zwischen Lagerplatz i und j
- $f_{m,n,t}$ Häufigkeit, mit der Artikel m und n in Periode t nacheinander kommissioniert werden
- v^{kom} Gehgeschwindigkeit des Kommissionierers
- z^{adm} Zeit für die administrative Umlagerung
- z^{phy} Zeit für die physische Umlagerung
- i_m^{akt} Aktueller Lagerplatz von Artikel $m \in V$

Entscheidungsvariable

- $W_{m,i}$ = 1, falls Artikel m Lagerplatz i zugewiesen werden soll (= 0, sonst)

Zielfunktion

$$\min Z = \sum_{t \in U} \sum_{m \in A} \sum_{\substack{n \in A \\ n \neq m}} \sum_{i \in V} \sum_{\substack{j \in V \\ j \neq i}} f_{m,n,t} \cdot d_{i,j} \cdot W_{m,i} \cdot W_{n,j} + \sum_{m \in A} \sum_{i \in V} W_{m,i} \cdot \text{sign}(|i_m^{akt} - i|) \cdot \left(d_{i_m^{akt},i} + v^{kom} \cdot (z^{adm} + z^{phy}) \right) \quad (5.1)$$

Die Zielfunktion minimiert die zukünftige Kommissionierweglänge unter Berücksichtigung des dazu notwendigen Umlagerungsaufwands. Hierzu wird die Häufigkeit,

mit der die Strecke zwischen dem Lagerplatz von Artikel m und dem Lagerplatz von Artikel n begangen wird, mit der Länge dieser Strecke multipliziert. Wenn ein Artikel nicht umgelagert werden soll und damit $i = i_m^{\text{akt}}$, wird die Signumsfunktion (sign) und damit der Umlagerungsaufwand zu null. Wenn Artikel m auf Lagerplatz i umgelagert werden soll ($i \neq i_m^{\text{akt}}$ und $W_{m,i} = 1$), nimmt die Signumsfunktion den Wert 1 an. Zu beachten ist, dass die Entscheidungsvariable $W_{m,i}$ nicht zeitabhängig ist. Es wird ein Lagerplatz für den gesamten Untersuchungszeitraum gesucht.

Nebenbedingungen

$$\sum_{m \in A} W_{m,i} \leq 1 \quad \forall i \in V \quad (5.2)$$

Stellt sicher, dass jedem Lagerplatz maximal ein Artikel zugewiesen wird.

$$\sum_{i \in V} W_{m,i} = 1 \quad \forall m \in A \quad (5.3)$$

Stellt sicher, dass jeder Artikel genau einem Lagerplatz zugewiesen wird.

$$W_{m,i} \in 0, 1 \quad \forall i \in V, m \in A \quad (5.4)$$

Definiert den Wertebereich der Variable.

Das so formulierte Problem ist am Ende jeder Periode zu lösen. Die Häufigkeit, mit der Artikel m und Artikel n nacheinander kommissioniert werden, ist zum Start des Lagerplatzvergabeproblems nicht bekannt und kann erst nach Lösung des zugehörigen Auftragsbildungs- und Tourenplanungsproblems aus Kapitel 5.1.3 bestimmt werden. Hier zeigt sich auch formal die starke Interdependenz der drei Planungsprobleme. Da alle drei Probleme NP-schwer sind, ist ein simultaner Lösungsansatz für praxisrelevante Problemgrößen nicht realistisch (Reschke 2013, S. 42-46; Wäscher 2004, S. 330-331). Ein weiteres Problem ist, dass die Auftragsdaten des Untersuchungszeitraums in der Regel vorab nicht bekannt sind (Reschke 2013, S. 40). Wie in Kapitel 2.5 beschrieben, soll mit dem dynamischen Lagerplatzvergabealgorithmus eine Heuristik zur Lösung des Optimierungsproblems eingesetzt werden, für die definitionsgemäß keine Garantie für das Finden einer optimalen Lösung besteht. Wie an späterer Stelle noch beschrieben wird, wird daher auch kein optimaler Lagerplatz, sondern ein möglichst guter Lagerplatz innerhalb einer Klasse ermittelt.

5.1.3 Kombiniertes Auftragsbildungs- und Tourenplanungsproblem

Bei dem untersuchten Auftragsbildungsproblem sollen Aufträge Batches so zugeteilt werden, dass die notwendige Kommissionierweglänge minimiert und die Kapazität des Kommissioniergeräts nicht überschritten wird. Für jeden Batch soll eine wegstreckenminimale Kommissioniertour gefunden werden, die alle zu kommissionierenden Artikel bzw. deren Lagerplätze enthält. Es muss sichergestellt werden, dass jeder Auftrag genau einem Batch zugewiesen wird. Eine Kommissioniertour beginnt und endet im Depot. Die Lagerplätze eines Batches müssen in einer zusammenhängenden Tour kommissioniert werden (Verbot sogenannter Kurzzyklen). Eine formale Darstellung des Problems findet sich in Kulak et al. (2012, S. 57-58):

Indizes und Mengen

$b \in B$	Batches
$k \in K$	Aufträge
$i, j \in V$	Lagerplätze
$S \subset V$	Teilmenge der Lagerplätze

Parameter

C	Kommissioniergerätkapazität
w_k	Gewicht von Auftrag k
$d_{i,j}$	Entfernung zwischen Lagerplatz i und j
$s_{k,i}$	= 1, falls ein Artikel aus Auftrag k von Lagerplatz i entnommen wird (= 0, sonst)

Entscheidungsvariablen

$X_{k,b}$	= 1, falls Auftrag k Batch b zugewiesen wird (= 0, sonst)
$Y_{i,j,b}$	= 1, falls Lagerplatz i direkt nach Lagerplatz j in Batch b besucht wird (= 0, sonst)
$Z_{i,b}$	= 1, falls Lagerplatz i in Batch b besucht wird (= 0, sonst)

Zielfunktion

$$\min Z = \sum_{b \in B} \sum_{\substack{i \in V \\ i \neq j}} d_{i,j} \cdot Y_{i,j,b} \quad (5.5)$$

Die Zielfunktion minimiert die Wegstrecke aller Kommissioniertouren.

Nebenbedingungen

$$\sum_{\substack{j \in V \\ j \neq i}} Y_{i,j,b} = Z_{i,b} \quad \forall b \in B, i \in V \quad (5.6)$$

$$\sum_{\substack{i \in V \\ i \neq j}} Y_{i,j,b} = Z_{j,b} \quad \forall b \in B, j \in V \quad (5.7)$$

Die Nebenbedingungen (5.6) und (5.7) stellen sicher, dass jeder besuchte Lagerplatz genau einen Vorgänger und einen Nachfolger hat. Außerdem wird sichergestellt, dass der betreffende Lagerplatz in der Tour besucht wird.

$$\sum_{\substack{i \in S \\ j \in V \setminus S}} Y_{i,j,b} \geq Z_{i,b} \quad \forall b \in B, S \subset V \quad (5.8)$$

Die Subtour-Eliminationsbedingungen stellen sicher, dass die Lösung aus einer zusammenhängenden Tour besteht, die alle Lagerplätze eines Batches enthält.

$$Z_{i,b} \geq s_{k,i} \cdot X_{k,b} \quad \forall b \in B, i \in V, k \in K \quad (5.9)$$

Abhängig von den Aufträgen, die einem Batch zugewiesen werden, werden Lagerplätze festgelegt, die in diesem Batch besucht werden müssen.

$$\sum_{b \in B} X_{k,b} = 1 \quad \forall k \in K \quad (5.10)$$

Stellt sicher, dass jeder Auftrag genau einem Batch zugewiesen wird.

$$\sum_{k \in K} w_k \cdot X_{k,b} \leq C \quad \forall b \in B \quad (5.11)$$

Die Aufträge, die einem Batch zugewiesen werden, dürfen die Kapazität des Kommissioniergeräts nicht übersteigen.

$$X_{k,b}, Y_{i,j,b}, Z_{i,b} \in 0, 1 \quad \forall i, j \in V, b \in B, k \in K \quad (5.12)$$

Definiert den Wertebereich der Variablen.

5.2 Verfahrensaufbau

Die im vorausgegangenen Kapitel formulierten Probleme sind aufgrund ihrer Komplexität für praxisrelevante Problemgrößen nicht mithilfe von exakten Verfahren lösbar. Daher wurde ein heuristisches Verfahren entwickelt. Das Tourenplanungs- und Auftragsbildungsproblem muss täglich neu gelöst werden und kann daher integriert berechnet werden. Das Lagerplatzvergabeproblem hat einen längeren Planungshorizont und wird üblicherweise nicht täglich neu gelöst. Getroffene Entscheidungen haben damit in der Regel eine längere Gültigkeit. Daher wurde ein iteratives Lösungsverfahren entwickelt. Abbildung 21 zeigt den Aufbau des Verfahrens.

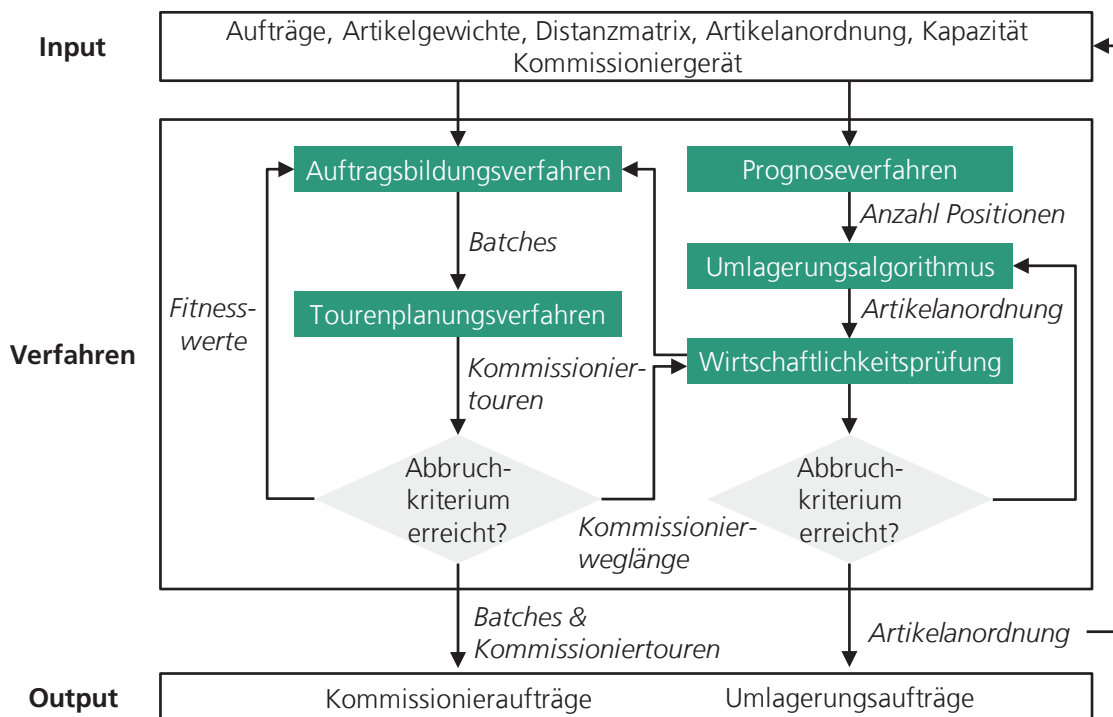


Abbildung 21: Verfahrensaufbau

Input sind Aufträge, die Artikelgewichte, die Distanzmatrix, die Artikelanordnung sowie die Kapazität des Kommissioniergeräts. Aufträge können sowohl Kunden- als auch Produktionsaufträge, etwa für die Versorgung von Montagelinien, sein. Sie können aus beliebig vielen Positionen bestehen. Jede Position kann aus einer beliebigen Artikelmenge bestehen, solange ein Auftrag nicht die Kapazität des Kommis-

sioniergeräts übersteigt. Die Entfernungen zwischen den Lagerorten im Layout werden in der Distanzmatrix gespeichert. Die Artikelanordnung enthält die Information, in welchem Lagerort die Artikel gelagert werden. Die Kapazität des Kommissioniergeräts ist das maximale Gewicht, das das Gerät bzw. der Kommissionierer transportieren kann.

Das *Verfahren* selbst unterteilt sich in zwei Bereiche. Das Verfahren zur Lösung des kombinierten Auftragsbildungs- und Tourenplanungsproblems ist auf der linken Seite von Abbildung 21 dargestellt und kann unabhängig von dem Verfahren zur Lagerplatzvergabe eingesetzt werden. Mithilfe des Auftragsbildungsverfahrens werden die Aufträge Batches zugeordnet. Diese werden an das Tourenplanungsverfahren weitergegeben. Dort werden für die einzelnen Batches Kommissioniertouren und deren Länge bestimmt. Die Kommissionierweglängen werden zur Bewertung der aktuellen Auftragsbildungslösung genutzt und in Form von Fitnesswerten an das Auftragsbildungsverfahren zurückgegeben. Beide Verfahren werden so lange durchlaufen, bis ein bestimmtes Abbruchkriterium (wie z.B. eine maximale Rechenzeit oder Anzahl Iterationen) erreicht wird. Wenn das Verfahren terminiert ist, werden die gefundene Zuordnung der Aufträge zu Batches und die Kommissioniertouren in Form von Kommissionieraufträgen ausgegeben.

Das Lagerplatzvergabeverfahren ist auf der rechten Seite von Abbildung 21 dargestellt. Mithilfe eines Prognoseverfahrens wird die zukünftige Anzahl Positionen je Artikelnummer prognostiziert. Im Umlagerungsalgorithmus wird bestimmt, welche Artikel umgelagert werden sollten und es werden potentielle Lagerplätze für diese Artikel ermittelt. Anschließend erfolgt eine Wirtschaftlichkeitsprüfung der Umlagerung einzelner Artikel. Für den umzulagernden Artikel wird zunächst der notwendige Umlagerungsaufwand bestimmt. Anschließend wird mithilfe des Verfahrens zur Lösung des kombinierten Auftragsbildungs- und Tourenplanungsproblems und den Aufträgen der aktuellen Periode, die Kommissionierweglänge berechnet, die sich durch die neue Artikelanordnung ergeben würde. Die Wegstreckenreduktion wird in die Zukunft projiziert. Wenn die zukünftig zu erwartende Wegstreckenre-

duktion den notwendigen Umlagerungsaufwand übersteigt, wird ein Umlagerungsauftrag für den untersuchten Artikel generiert. Dieses Verfahren wird solange wiederholt, bis ein Abbruchkriterium (wie z.B. eine maximale Rechenzeit oder Anzahl Umlagerungsprüfungen) erreicht wird. Die bis dahin generierten Umlagerungsaufträge werden ausgegeben.

5.3 Verfahrensauswahl

Nachdem der Aufbau des Gesamtverfahrens beschrieben wurde, werden Verfahren für die einzelnen Bestandteile ausgewählt. Zur Lösung des Auftragsbildungsproblems wird eine modifizierte Version der PSO eingesetzt. Önüt et al. (2008, S. 797) zeigen, dass die PSO in der Lage ist, sehr gute Lösungen innerhalb relativ kurzer Rechenzeiten zu erzeugen. Auch Hembeker et al. (2007, S. 363-364) zeigen, dass die PSO sehr gut zur Lösung kombinatorischer Optimierungsprobleme geeignet ist. PSO-Algorithmen sind zur Lösung großer Probleminstanzen einsetzbar und zeichnen sich im Allgemeinen durch eine hohe Rechenzeiteffizienz aus. In seiner ursprünglichen Version ist der PSO-Algorithmus für den kontinuierlichen Suchraum entwickelt worden und somit nicht für diskrete Anwendungsfälle einsetzbar (Kashan et al. 2013, S. 83). Er muss zunächst angepasst werden, damit er auch im diskreten Fall eingesetzt werden kann.

Cheng et al. (2015) entwickeln einen PSO-Algorithmus für das Auftragsbildungsproblem und zeigen, dass dieser – bezogen auf das Optimierungsergebnis – dem Genetischen Algorithmus von Tsai et al. (2008) überlegen ist. Auch Lin et al. (2016) setzen einen PSO-Algorithmus für das Auftragsbildungsproblem ein und erzielen damit sehr gute Optimierungsergebnisse. Beide Ansätze sind allerdings nur für sehr kleine Probleminstanzen einsetzbar. Die Herausforderung besteht darin, die PSO geeignet zu modifizieren und mit einem Tourenplanungsverfahren zu kombinieren, so dass auch praxisrelevante Problemgrößen gelöst werden können. Ein Problem der PSO ist, dass sie in manchen Fällen dazu neigt, frühzeitig eine Lösung zu fokussieren und so möglicherweise in einem lokalen Optimum zu verharren (Sedighizadeh et al. 2009, S. 487). Dieses Verhalten konnte während der Entwicklung des Verfahrens

beobachtet werden. Um diesen Nachteil zu beheben, wird die PSO um evolutionäre Bestandteile erweitert. Dadurch kann verhindert werden, dass der Algorithmus zu schnell ein lokales Optimum fokussiert (Wei et al. 2002, S. 1278). Die so entwickelte PSO basiert auf dem von Tian et al. (2013) für das Two-Stage-Assembly-Scheduling-Problem entwickelten Verfahren und erweitert dieses. Um die Rechenzeit zu verbessern und den Schwarm möglichst schnell in vielversprechende Suchregionen zu bewegen, kann bei der Schwarminitialisierung eine problembezogene Heuristik eingesetzt werden und die Lösung in Form eines Partikels in den Schwarm „injiziert“ werden (vgl. hierzu Lam et al. 2007). Aus diesem Grund wird der Savings-Algorithmus von Clarke und Wright in das Verfahren integriert. Kulak et al. (2012) zeigen, dass mithilfe des *2opt-Algorithmus* für das Tourenplanungsproblem gute Optimierungsergebnisse innerhalb sehr kurzer Rechenzeiten erzielt werden können. Dies ist vor dem Hintergrund relevant, dass der Tourenplanungsalgorithmus im Zuge des Auftragsbildungsalgorithmus vielfach durchlaufen werden muss. Die Integration einer komplexeren Metaheuristik würde die Rechenzeit stark ansteigen lassen. Dieses Problem ist bei dem Lagerplatzvergabeverfahren noch gravierender, da hier das Auftragsbildungsverfahren selbst vielfach durchlaufen werden muss. Daher wird zur Lösung des Tourenplanungsproblems eine Startlösung mithilfe des Nearest-Neighbor-Algorithmus berechnet und mit dem *2opt-Algorithmus* verbessert. Das im Weiteren als „diskrete evolutionäre Partikelschwarmoptimierung“ (DEPSO) bezeichnete Verfahren zur Lösung des kombinierten Auftragsbildungs- und Tourenplanungsproblems wird in Kapitel 6.2 vorgestellt.

Für die Entwicklung des *dynamischen Lagerplatzvergabealgorithmus* dient das Verfahren von Pierre et al. (2003) als Vorbild. Es wird allerdings stark erweitert bzw. an den notwendigen Stellen angepasst. So wird in das Verfahren ein Prognoseverfahren zur Bestimmung der zukünftigen Positionen je Artikelnummer integriert. Die Auswahl eines allgemeingültigen Prognoseverfahrens ist nicht möglich und hängt von der vorliegenden Zeitreihe ab. Eine Hilfestellung für die Auswahl eines geeigneten Verfahrens bieten die Ausführungen in Kapitel 3.5. Eine weitergehende methodische Unterstützung findet sich unter anderem bei Schröder (2012), Schuhr (2012)

und Vogel (2015). In den numerischen Experimenten wurde das Verfahren von Holt-Winters gewählt, da hier Zeitreihen mit saisonalen Schwankungen und Trend vorlagen. Dagegen wurde im Zuge des Fallbeispiels die exponentielle Glättung erster Ordnung eingesetzt, da hier zwar ein Trend, aber keine Saisonalitäten, sondern starke unsystematische Schwankungen erkennbar waren. Der Umlagerungsvorschlag wird mithilfe des prioritätswertbasierten Umlagerungsalgorithmus generiert. Hierfür wurde eine Umlagerungslogik entwickelt. Zur Integration von Zukunftsdaten wird ein Prioritätswert berechnet. Im Zuge der Wirtschaftlichkeitsprüfung wird zunächst der Umlagerungsaufwand ermittelt. Der vom Umlagerungsalgorithmus generierte Umlagerungsvorschlag wird vorläufig umgesetzt. Mithilfe des DEPSO-Algorithmus und den Aufträgen der aktuellen Periode wird die Kommissionierweglänge berechnet, die sich mit der neuen Artikelanordnung ergeben würde. Ein im Zuge dieser Arbeit entwickeltes Schätzverfahren bestimmt, basierend auf den Prognosedaten und der berechneten Wegstreckenreduktion der aktuellen Periode, die zukünftig zu erwartende Wegstreckenreduktion. Der Umlagerungsvorschlag wird übernommen, wenn der zukünftige Nutzen den Aufwand übersteigt. Der dynamische Lagerplatzvergabealgorithmus ist so konzipiert, dass auch andere Verfahren zur Lösung des kombinierten Auftragsbildungs- und Tourenplanungsproblems integriert werden können.

6 Ausgestaltung des Verfahrens

In diesem Kapitel werden der DEPSO-Algorithmus und der dynamische Lagerplatzvergabealgorithmus beschrieben. Da sich die Verfahren vor allem an Anwender richten, wird auf die Darstellung der Algorithmen in Pseudocode verzichtet. Die übersichtliche verbale Darstellung solcher Verfahren ist in der Literatur weit verbreitet und findet sich unter anderem auch bei Kulak et al. (2012). Wo sinnvoll, unterstützen graphische Darstellungen die Verständlichkeit. Aufgrund ihrer Bedeutung für die zu lösenden Probleme wird zunächst auf die Distanzmatrix eingegangen.

6.1 Distanzmatrix

Das Tourenplanungsproblem in einem Multi-Block-Layout ist ein Steiner TSP. Das Problem kann mithilfe eines Graphen $G = (N,E)$ mit N Knoten und E Kanten dargestellt werden (Abbildung 22).

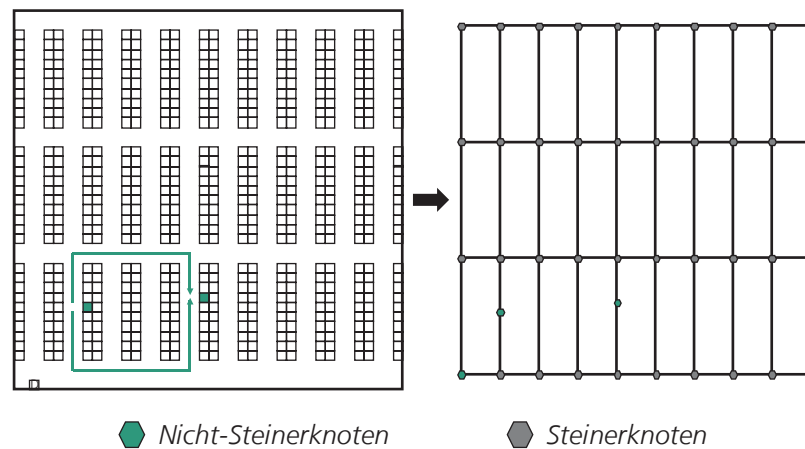


Abbildung 22: Block-Layout und dessen Repräsentation durch einen Graphen

Die N Knoten setzen sich aus Steinerknoten und Nicht-Steinerknoten zusammen. Die Steinerknoten befinden sich an den Kreuzungen der Gänge. Die Nicht-Steinerknoten liegen in der Mitte der Kommissioniergänge an Lagerorten mit zu kommissionierenden Artikeln. Das Ziel besteht darin, die kürzeste Steiner-Tour zu finden, die alle Nicht-Steinerknoten mindestens einmal besucht. Die Steinerknoten müssen

nicht, können aber besucht werden. Da das Steiner TSP ein Spezialfall des klassischen TSP ist, kann es in dieses überführt werden, indem der kürzeste Weg zwischen den Nicht-Steinerknoten (inklusive des Depots) mithilfe der Manhattan-Metrik berechnet wird. Die Manhattan-Distanz zwischen zwei Knoten mit den Koordinaten (x_1, y_1) und (x_2, y_2) kann mithilfe von (6.1) berechnet werden:

$$d_{1,2} = |x_1 - x_2| + |y_1 - y_2| \quad (6.1)$$

Die Entfernung zweier Knoten, die in unterschiedlichen Kommissioniergängen desselben Blocks liegen (siehe Abbildung 22) kann nicht direkt mithilfe der Manhattan-Metrik berechnet werden, da der kürzeste Weg durch die Lagerregale verlaufen würde. In diesem Fall muss die Länge der Tour, die über den oberen bzw. den unteren Steinerknoten verläuft, mithilfe der Manhattan-Metrik berechnet werden. Die kürzere Tour wird behalten und die längere verworfen. Für ein Lager werden einmalig die Entfernungen zwischen allen Knoten berechnet und in einer Distanzmatrix D (ohne die Steinerknoten) gespeichert. Mithilfe dieser symmetrischen Distanzmatrix können alle verfügbaren Heuristiken, die für das klassische TSP entwickelt wurden, zur Lösung des Tourenplanungsproblems in einem Multi-Block-Layout eingesetzt werden (Theys et al. 2010, S. 756-758). Damit kann die Forderung aus Kapitel 2.3.2, wonach das Verfahren auch für Multi-Block-Layouts einsetzbar sein soll, erfüllt werden.

6.2 Diskrete evolutionäre Partikelschwarmoptimierung

Zur Lösung des in Kapitel 5.1.3 beschriebenen kombinierten Auftragsbildungs- und Tourenplanungsproblems wurde ein Verfahren entwickelt, das Elemente der Evolutionary Computation und der Schwarmintelligenz vereint. Solche Kombinationen finden sich u.a. in Lee (2007), Sadri et al. (2006), Wei et al. (2002), Yin (2006) und Zhiming et al. (2008). Lian et al. (2008) und Liu et al. (2008) erweitern den PSO-Algorithmus um Mutationsoperatoren und setzen ihn zur Lösung des Flow-Shop-Scheduling-Problems ein. Tian et al. (2013) erweitern diesen Ansatz um ein lokales Suchverfahren zur Lösung des Two-Stage-Assembly-Scheduling-Problems. Das hier

vorgestellte Verfahren basiert auf diesem Algorithmus, wurde aber erweitert und zur Lösung des kombinierten Auftragsbildungs- und Tourenplanungsproblems angepasst. Weder das Auftragsbildungsproblem noch das kombinierte Problem wurde bislang mithilfe eines solchen Verfahrens gelöst. Um die Rechenzeit zu verbessern und den Schwarm möglichst schnell in vielversprechende Suchregionen zu ziehen, wird in den Schwarm zu Beginn eine Lösung in Form eines Partikels injiziert. Diese wird mithilfe des Savings-Algorithmus von Clarke und Wright berechnet. Ebenfalls neu ist die Integration des 2opt-Algorithmus zur Berechnung der kürzesten Tour in diese Art von Algorithmus. Der Optimierungszeitraum ist frei wählbar. Unternehmen können eine optimierte Lösung für den Auftragsbestand eines ganzen Tages oder für einen beliebigen anderen Zeitraum ermitteln.

6.2.1 Aufbau

Der Aufbau des DEPSO-Algorithmus ist in Abbildung 23 dargestellt. In der *Initialisierung* des Verfahrens werden, unter Nutzung eines geeigneten Codierungsschemas, zufallsbasiert Startlösungen erzeugt. Eine Startlösung wird durch einen Partikel repräsentiert. Eine der Startlösungen wird mithilfe des Savings-Algorithmus berechnet. Neben der Position der Partikel im Suchraum ist zusätzlich deren Geschwindigkeitsvektor zufallsbasiert zu erzeugen.

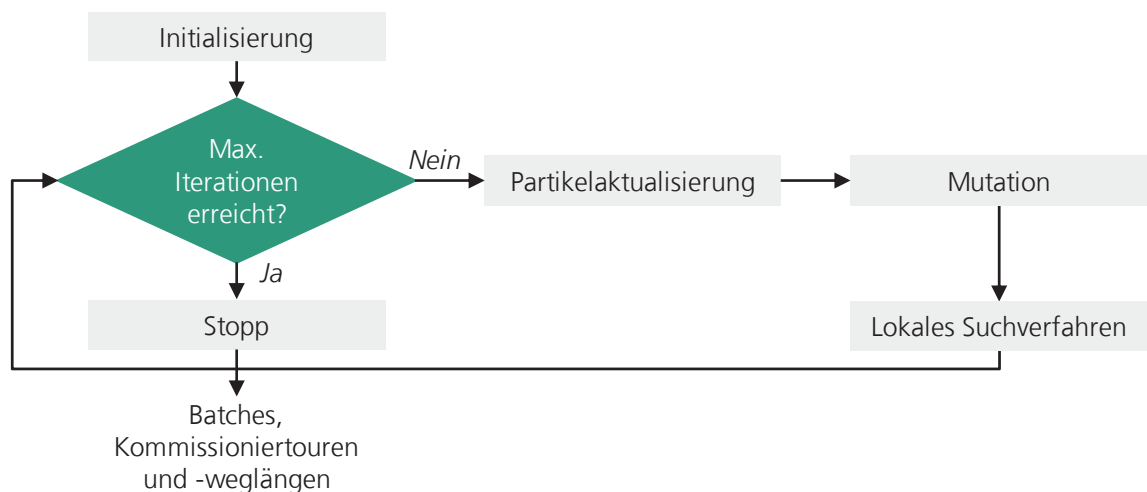


Abbildung 23: Aufbau DEPSO

Je nach Codierungsschema ist eine Regel festzulegen, mit der die Aufträge auf die Batches verteilt werden. Mithilfe des Nearest-Neighbor-Algorithmus und des 2opt-Algorithmus wird für jeden Partikel eine Kommissioniertour ermittelt und der Fitnesswert des Partikels in Form der Kommissionierweglänge ausgegeben. Anschließend wird die Initialisierungslösung der Partikel als deren bisher beste Lösung (P_p^{best} = Personal Best Partikel p) gespeichert. Unter den P_p^{best} wird derjenige mit dem kürzesten Kommissionierweg als bisher beste Schwarmlösung (G_{best} = Global Best) gespeichert. In der sich anschließenden *Partikelaktualisierung* bewegen sich die Partikel zu einem neuen Punkt im Suchraum. Die Bewegung der Partikel wird von deren Bewegungswahrscheinlichkeit und dem Geschwindigkeitsvektor gesteuert. Neben der Position des Partikels wird auch dessen Geschwindigkeitsvektor aktualisiert. Für die neuen Positionen der Partikel im Suchraum werden mithilfe des Tourenplanungsverfahrens neue Fitnesswerte berechnet. Anschließend wird geprüft, ob sich P_p^{best} und G_{best} verbessert haben. Wenn eine Verbesserung eingetreten ist, wird der entsprechende Wert aktualisiert. Befinden sich Partikel in der Nähe des G_{best} und P_p^{best} wird ihre Position im Zuge der *Mutation* verändert, damit ein Partikel nicht in dieser Position verharret. Der Mutationsvorgang wird mithilfe einer Mutationswahrscheinlichkeit gesteuert. Die Stärke des Eingriffs richtet sich nach der geschätzten Nähe des Partikels zum globalen Optimum. Wenn seit mehreren Iterationen keine verbesserte Schwarmlösung gefunden werden konnte, besteht die Gefahr, dass das Verfahren stagniert. Mithilfe eines *lokalen Suchverfahrens* wird versucht, die Lösung des G_{best} zu verbessern. Da dies zusätzlichen Rechenaufwand bedeutet, wird die lokale Suche nicht in jeder Iterationsschleife durchgeführt. Das Verfahren wird erst gestartet, wenn der G_{best} stagniert und somit keine Verbesserung mehr zu erwarten ist. Beim Überschreiten einer adaptiven Schranke schaltet sich das Verfahren automatisch ein. Die Qualität der mithilfe des lokalen Suchverfahrens gefundenen Lösung wird durch die vom Tourenplanungsverfahren zu berechnende Kommissionierweglänge bewertet. Der G_{best} wird aktualisiert, wenn sich die Lösung durch das lokale Suchverfahren verbessert hat. Sowohl die Lösungsqualität als auch

die Rechenzeit können verbessert werden, indem im Gegensatz zu dem Vorgehen von Tian et al. (2013, S. 487) ein Partikel auf den neu gefundenen G_{best} gesetzt wird. Dieser Partikel wird mit hoher Wahrscheinlichkeit in der nächsten Mutationsphase mutiert und so kann die Region um den neuen G_{best} schneller und intensiver untersucht werden. Partikelaktualisierung, Mutation und lokale Suche werden so lange iterativ durchlaufen, bis die maximale Anzahl Iterationen erreicht wurde. Das Verfahren terminiert und gibt die bis dahin gefundene beste Zuteilung der Aufträge zu den Batches sowie die Kommissioniertouren und die Kommissionierweglängen aus. Abbildung 24 stellt den Ablauf des DEPSO-Algorithmus graphisch dar.

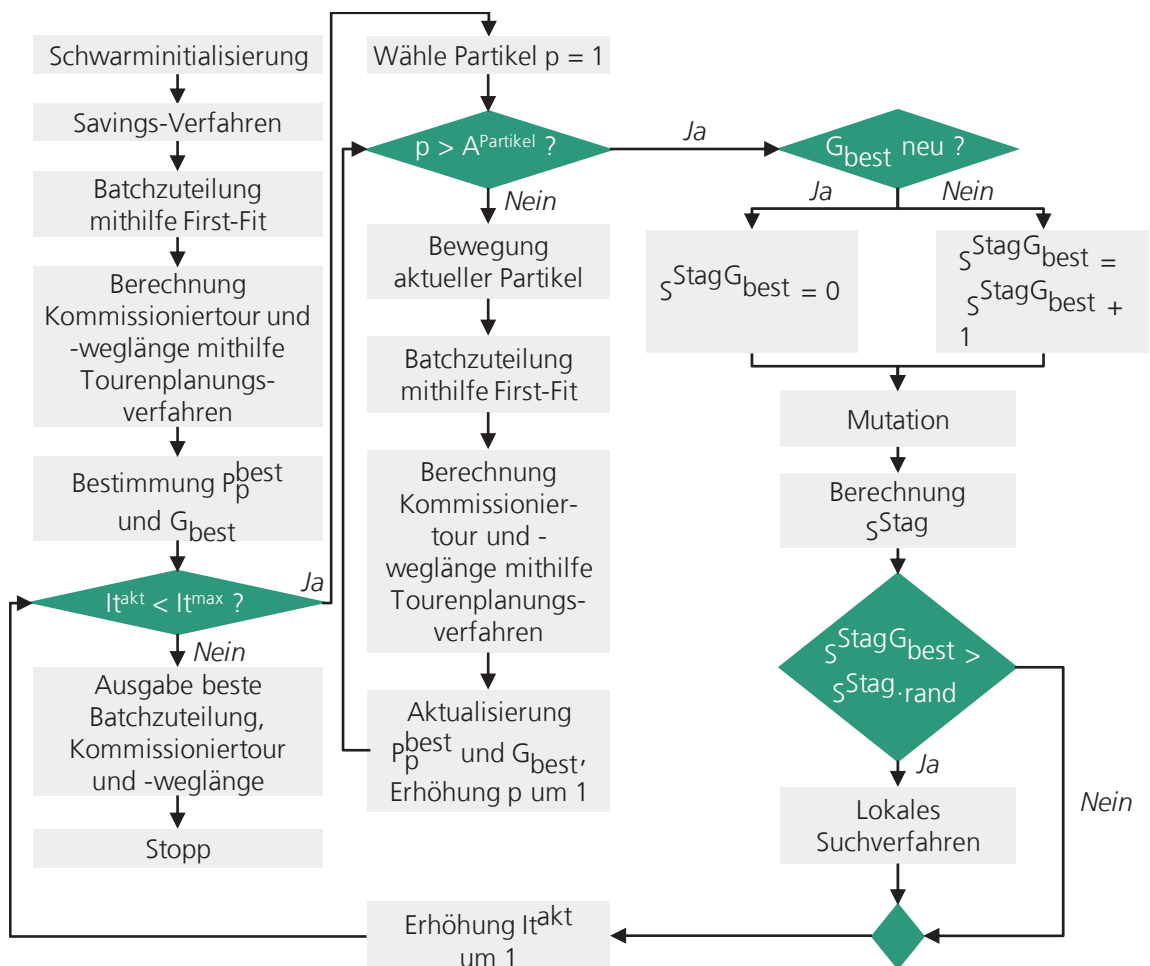


Abbildung 24: Ablauf DEPSO

In den nachfolgenden Kapiteln wird auf die einzelnen Bestandteile des Verfahrens detailliert eingegangen.

6.2.2 Initialisierung

Codierung

Damit der DEPSO-Algorithmus zur Lösung des vorliegenden Problems eingesetzt werden kann, muss die Lösung des Optimierungsproblems in eine für den Algorithmus verarbeitbare Form gebracht werden. Dieser Vorgang wird als *Codierung* bezeichnet. Es wurde folgendes Codierungsschema gewählt: Ein Partikel ist die Permutation aller K einzuplanender Aufträge. Jeder Auftrag kommt genau einmal in der Permutation vor. Die Permutation gibt die Reihenfolge an, in der die Aufträge eingeplant werden sollen. Zum Beispiel bedeutet $P_p = \{123, 99, 174, 244, 32, 40, 22, 175, 34, 152, \dots\}$, dass in Partikel p Auftrag 174 an 3. Stelle eingeplant wird. Im Zuge der Schwarminitialisierung wird die vor Beginn des Verfahrens festgelegte Anzahl Partikel A^{Partikel} zufallsbasiert erzeugt.

Zuordnungsregel

Da bei dieser Art der Codierung die Zuteilung zu Batches nicht direkt erfolgt, müssen die Aufträge entsprechend einer Zuordnungsregel den Batches zugewiesen werden. Im vorliegenden Fall wird die *First-Fit-Regel* gewählt, mit der Batches simultan erzeugt werden (siehe Abbildung 25).

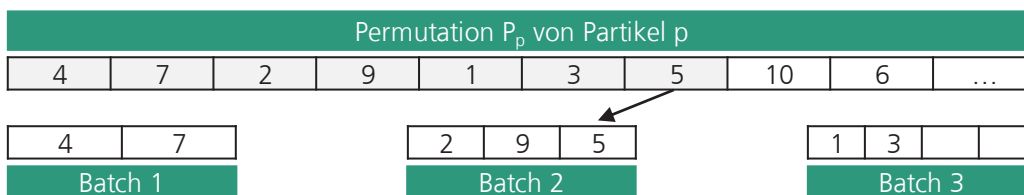


Abbildung 25: First-Fit-Regel

Es sind mehrere Batches gleichzeitig geöffnet. Die Nummerierung der Batches entspricht der Reihenfolge, in der sie eröffnet und abgearbeitet werden. Die Aufträge der Permutation werden der Reihe nach eingeplant. Der jeweils zuzuordnende Auftrag wird dem Batch mit der niedrigsten Nummer zugewiesen, in dem noch Platz für den Auftrag ist (Koch 2014, S. 34). Durch die Anwendung der First-Fit-Regel

werden die Aufträge kompakter und damit auf weniger Batches verteilt, als etwa bei der sequentiell verteilenden Next-Fit-Regel (Koch 2014, S. 107).

Geschwindigkeitsvektor

Alle Partikel haben zu jedem Zeitpunkt im Optimierungsverlauf eine spezifische Geschwindigkeit. Der Geschwindigkeitsvektor $V_{p,t} = \{v_{p,1,t}, v_{p,2,t}, \dots, v_{p,k,t}\}$ beschreibt die Geschwindigkeit des Partikels p zum Zeitpunkt t . $v_{p,k,t} \in \{-1, 0, 1\}$ ist dem k . Auftrag in der Permutation zugeordnet. Die Geschwindigkeitsausprägungen haben folgende Bedeutung:

- Wenn $v_{p,k,t} = 1$, dann wird der Partikel an der Stelle k von G_{best} angezogen und bewegt sich in dessen Richtung.
- Wenn $v_{p,k,t} = -1$, dann wird der Partikel an der Stelle k von P_p^{best} angezogen und bewegt sich in dessen Richtung.
- Wenn $v_{p,k,t} = 0$, dann bewegt sich der Partikel nicht.

Tourenplanungsverfahren

Die Bestimmung der Kommissioniertouren und der Kommissionierweglängen erfolgt innerhalb des DEPSO-Algorithmus mithilfe der Nearest-Neighbor-Heuristik und einem 2opt-Verfahren. Die *Nearest-Neighbor-Heuristik* ist ein heuristisches Eröffnungsverfahren. Mit ihr wird eine zulässige Anfangslösung bestimmt. Hierzu wird der Kommissioniertour, ausgehend vom Depot, immer der am nächsten zum aktuellen Knoten liegende Knoten hinzugefügt. Dies wird solange wiederholt, bis alle Knoten besucht wurden. Die Tour endet im Depot. Das Verfahren kann wie folgt zusammengefasst werden:

Ablauf Nearest-Neighbor-Heuristik (I.)

Schritt 1

Wähle das Depot als Ausgangspunkt der Kommissioniertour K_t und speichere es als aktuellen Knoten.

Schritt 2

Berechne mithilfe der Distanzmatrix die Entfernung vom aktuellen Knoten zu allen anderen bislang noch unbesuchten Knoten mit zu kommissionierenden Artikeln. Wähle den Knoten mit der geringsten Entfernung, speichere ihn als aktuellen Knoten und füge ihn am Ende der Kommissioniertour K_t ein. Wiederhole Schritt 2, bis alle Knoten besucht wurden, die zu kommissionierende Artikel enthalten.

Schritt 3

Wähle das Depot als Endpunkt der Tour und gebe K_t aus.

Zur Verbesserung dieser Lösung wird das *2opt-Verfahren* eingesetzt. Es prüft systematisch alle Tauschmöglichkeiten zweier Kanten gegen zwei bislang noch nicht in der Tour enthaltene Kanten. Der Tausch wird angenommen, wenn die Tour dadurch verkürzt werden kann. Sobald ein Tausch angenommen wurde, wird wieder zum Start des Verfahrens gewechselt und mit den Vertauschungsprüfungen begonnen. Wird ein Tausch verworfen, wird der Tausch der nächsten Kanten untersucht. Das Verfahren wird abgebrochen, wenn bei der letzten Überprüfung aller paarweisen Tauschmöglichkeiten keine Verbesserung mehr erzielt werden konnte (Domschke et al. 2015, S. 154). Das 2opt-Verfahren läuft wie folgt ab:

Ablauf 2opt-Verfahren

(II.)

Schritt 1

Speichere die Kommissioniertour aus dem Nearest-Neighbor-Verfahren als aktuelle Kommissioniertour K_t . Berechne die Länge der aktuellen Kommissioniertour mithilfe der Distanzmatrix und speichere sie als aktuelle Kommissionierweglänge KW .

Schritt 2

Wähle Stelle 1 als Stelle q der aktuellen Kommissioniertour. Wähle Stelle 3 als Stelle r der aktuellen Kommissioniertour. Der Knoten an der Stelle q (der Stelle r) in der aktuellen Kommissioniertour ist $K_t(q)$ ($K_t(r)$).

Schritt 3

Berechne die Distanzen d^{nTau} (Distanz nach Tausch) und d^{vTau} (Distanz vor Tausch):

$$d^{nTau} = d_{K_t(q),K_t(r)} + d_{K_t(q+1),K_t(r+1)} \quad (6.2)$$

$$d^{v\text{Tau}} = d_{Kt(q),Kt(q+1)} + d_{Kt(r),Kt(r+1)} \quad (6.3)$$

$$\Delta = d^{n\text{Tau}} - d^{v\text{Tau}} \quad (6.4)$$

Dabei steht $d_{Kt(q),Kt(r)}$ für die Distanz zwischen den Knoten $Kt(q)$ und $Kt(r)$.

Schritt 4

Falls $\Delta \geq 0$, dann verwerfe den Tausch und gehe wie folgt vor:

- a) Wenn r nicht die vorletzte Stelle in der Tour ist, behalte zunächst q . Erhöhe r um 1. Gehe zu Schritt 3.
- b) Wenn r die vorletzte Stelle in der Tour ist und q nicht die viertletzte Stelle, erhöhe q um 1 und setze $r = q + 2$. Gehe zu Schritt 3.
- c) Wenn q die viertletzte Stelle in der Tour und r die vorletzte ist, breche das Verfahren ab und gebe die Kommissioniertour Kt und die Kommissionierweglänge KW aus.

Falls $\Delta < 0$, akzeptiere den Tausch und setze:

$$Kt^{neu} = (Kt(1 : q), Kt(r), Kt((r - 1) : (-1) : (q + 1)), Kt((r + 1) : end)) \quad (6.5)$$

Dabei bezeichnet $Kt(1:q)$ die Kommissioniersequenz von der 1. bis zur q . Stelle der aktuellen Kommissioniertour und $Kt((r-1):(-1):(q+1))$ die umgekehrte Kommissioniersequenz von der $(r-1)$. bis zur $(q+1)$. Stelle in der aktuellen Kommissioniertour. Das Symbol end steht für die letzte Stelle in der aktuellen Kommissioniertour.

Schritt 5

Setze $Kt = Kt^{neu}$. Berechne die aktuelle Kommissionierweglänge KW mithilfe der Distanzmatrix. Fahre mit Schritt 2 fort.

Savings-Verfahren

Um den Schwarm schnell auf eine vielversprechende Region im Suchraum zu lenken, können bestimmte Partikel zuerst mithilfe einer Heuristik berechnet und der Schwarm schon frühzeitig beeinflusst werden. Um Rechenzeit und Optimierungsergebnis zu verbessern, wird ein Partikel im Schwarm mithilfe des Savings-Verfahrens berechnet. Da dieses Verfahren nur einmal bei der Initialisierung durchgeführt wird,

hat es keinen nennenswerten Einfluss auf die Rechenzeit. Das hier eingesetzte Verfahren wurde von Clarke et al. (1964) entwickelt und erstmals von Elsayed et al. (1989) zur Lösung des Auftragsbildungsproblems eingesetzt. Der Savings-Wert $Sav_{k_1k_2}$ der Aufträge k_1 und k_2 ergibt sich aus der Differenz aus den Weglängen zweier separater Kommissioniertouren (KW_{k_1} , KW_{k_2}) und der Weglänge der kombinierten Kommissioniertour ($KW_{k_1k_2}$):

$$Sav_{k_1k_2} = KW_{k_1} + KW_{k_2} - KW_{k_1k_2} \quad (6.6)$$

Ein positiver Wert drückt eine Wegeinsparung aus, die durch die gemeinsame Kommissionierung zweier Aufträge erreicht werden kann. Die Weglängen werden mithilfe des oben beschriebenen Tourenplanungsverfahrens ermittelt. Das hier eingesetzte Verfahren läuft in den folgenden Schritten ab:

Ablauf Savings-Verfahren (III.)

Schritt 1

Berechne die Savings-Werte aller möglichen Auftragskombinationen mithilfe von (6.6) und den Tourenplanungsverfahren (I.) und (II.). Berechne nur Savings-Werte für Auftragskombinationen, die die Kapazitätsrestriktion des Kommissioniergeräts nicht übersteigen. Speichere die Werte in einer Savings-Liste.

Schritt 2

Sortiere die Savings-Liste in absteigender Reihenfolge.

Schritt 3

Wähle die erste bislang ungeprüfte Auftragskombination in der Savings-Liste und teile sie einem Batch zu. Dabei können folgende Fälle auftreten:

- (1) Das Auftragspaar wurde noch keinem bestehenden Batch zugeordnet. Eröffne einen neuen Batch und weise diesem das Paar zu.
- (2) Einer der beiden Aufträge wurde bereits einem Batch zugewiesen. Falls der zweite Auftrag noch Platz hat, dann füge ihn dort ein. Falls nicht, gehe zur nächsten Auftragskombination in der Liste.
- (3) Beide Batches wurden bereits zugeordnet. Dann gehe zur nächsten Kombination in der Liste.

Wiederhole Schritt 3 solange bis alle Savings-Werte geprüft wurden.

Schritt 4

Sollten noch nicht zugeordnete Aufträge übrig sein, weise diese Aufträge jeweils neu zu eröffnenden Batches zu. Wenn alle Aufträge zugewiesen wurden, dann beende das Verfahren und gebe die Zuteilung der Aufträge zu Batches aus.

Bestimmung von P_p^{best} und G_{best}

Die Kommissionierweglänge, die mithilfe des Tourenplanungsverfahrens berechnet wurde, ist der Fitnesswert eines Partikels. Nachdem dieser für alle Partikel bestimmt wurde, wird die Initialisierungslösung der Partikel als ihre bisher beste Permutation P_p^{best} gespeichert. Die Permutation des Partikels mit dem kürzesten Kommissionierweg wird als G_{best} gespeichert. Der Stagnationsfaktor $S^{\text{stag}G_{\text{best}}}$ gibt an, wie viele Iterationen durchlaufen wurden ohne dass sich G_{best} verbessert hat. Zu Beginn wird $S^{\text{stag}G_{\text{best}}}$ zu Null gesetzt.

6.2.3 Partikelaktualisierung

In dieser Phase bewegen sich die Partikel im Suchraum. Die Bewegung wird von der Bewegungswahrscheinlichkeit und dem Geschwindigkeitsvektor der Partikel gesteuert.

Bewegungswahrscheinlichkeit

Jeder Partikel hat zwei Bewegungswahrscheinlichkeiten. Eine in Richtung P_p^{best} und eine in Richtung G_{best} . Die Bewegungswahrscheinlichkeit berechnet sich aus der Differenz Df_{p_1, p_2} der Permutationen P_{p_1} und P_{p_2} der Partikel p_1 und p_2 mit der Länge K mithilfe von (6.7):

$$Df_{p_1, p_2} = \frac{\sum_{j=1}^K \text{sign}(|P_{p_1}(j) - P_{p_2}(j)|)}{K} \quad (6.7)$$

Die Signumsfunktion (sign) in (6.7) nimmt den Wert 0 an, wenn $P_{p_1}(j) = P_{p_2}(j)$. Ansonsten wird die Funktion zu 1. Abbildung 26 zeigt die Berechnung der Differenz anhand eines Beispiels.

Partikel unterscheiden sich an jeder Stelle: $Df_{p_1,p_2} = \frac{10}{10} = 1$										
Partikel 1	4	8	2	7	9	1	3	5	6	10
Partikel 2	5	6	1	3	10	8	4	2	9	7
Gleich?	1	1	1	1	1	1	1	1	1	1
Partikel unterscheiden sich an keiner Stelle: $Df_{p_1,p_2} = \frac{0}{10} = 0$										
Partikel 1	4	8	2	7	9	1	3	5	6	10
Partikel 2	4	8	2	7	9	1	3	5	6	10
Gleich?	0	0	0	0	0	0	0	0	0	0

Abbildung 26: Differenz zweier Partikel

Im Gegensatz zum Vorgehen von Tian et al. (2013, S. 484) wird hier die Differenz nicht mit einer Zufallszahl zwischen 0 und 1 multipliziert. Die Autoren hindern die Partikel durch die Multiplikation mit der Zufallszahl daran, sich zu schnell zu bewegen. Hier wird darauf verzichtet, damit sich der Schwarm schneller in vielversprechende Regionen bewegt und die Rechenzeit sinkt. Damit gilt für die Bewegungswahrscheinlichkeit in Richtung P_p^{best} eines Partikels p:

$$B_{p,P_{best}} = Df_{p,P_{best}} \tag{6.8}$$

und für die Bewegungswahrscheinlichkeit in Richtung G_{best} :

$$B_{p,G_{best}} = Df_{p,G_{best}} \tag{6.9}$$

Partikelbewegung

Eine Bewegung von Partikel p an der Stelle h in Richtung G_{best} wird geprüft, wenn:

$$v_{p,h,t} = 1 \text{ und } B_{p,G_{best}} > S^{G_{best}} \cdot \text{rand} \tag{6.10}$$

Dabei steht rand für eine gleichverteilte Zufallszahl zwischen null und eins. Eine Bewegung von Partikel p an der Stelle h in Richtung P_p^{best} wird geprüft, wenn:

$$v_{p,h,t} = -1 \text{ und } B_{p,P_{best}} > \text{rand} \tag{6.11}$$

Ist $v_{p,h,t} = 0$ wird keine Bewegung durchgeführt und die nächste Stelle im Partikel geprüft. Der Schwellwert $S^{G_{best}}$ wurde entgegen dem Verfahren von Tian et al.

(2013, S. 484) eingeführt, um dem Verfahren zu ermöglichen, vorzugweise dem G_{best} oder dem P_p^{best} zu folgen. Der Schwellwert ist vor Beginn des Verfahrens mithilfe von Vortests festzulegen. Die Vorgehensweise zur Durchführung von Vortests findet sich in den Kapiteln 7.3 und 8.2. Die Bewegung eines Partikels p läuft in folgenden Schritten ab:

Ablauf Partikelbewegung

(IV.)

Schritt 1

Starte das Verfahren an der Stelle $h = 1$ der Permutation P_p des Partikels p .

Schritt 2

Wenn (6.10) erfüllt ist, dann wähle G_{best} als Vergleichspermutation P^{ver} . Wenn (6.11) erfüllt ist, dann wähle P_p^{best} als Vergleichspermutation P^{ver} . Ist keine der beiden Bedingungen erfüllt, fahre mit Schritt 5 fort.

Schritt 3

Bestimme die Stelle r in der Permutation P_p , die den Auftrag $P^{\text{ver}}(h)$ enthält.

Schritt 4

Wenn Bedingung (6.12) erfüllt ist, tausche den Auftrag an der Stelle h mit dem Auftrag an Stelle r in der Permutation P_p .

$$v_{p,r,t} \neq 0 \text{ oder } v_{p,r,t} = 0 \text{ und } \text{rand} < 0,5 \quad (6.12)$$

Aktualisiere die Bewegungswahrscheinlichkeiten.

Schritt 5

Aktualisiere die Geschwindigkeit $v_{p,h,t}$ an der Stelle h . Wähle hierzu zufällig aus $\{-1, 0, 1\}$. Wenn das Ende des Partikels noch nicht erreicht wurde, erhöhe h um 1 und fahre mit Schritt 2 fort. Ansonsten breche das Verfahren ab.

Abbildung 27 zeigt eine beispielhafte Partikelbewegung an der Stelle $h = 3$ eines Partikels p . Es sei angenommen, dass Bedingung (6.10) erfüllt sei. Der zu tauschende Auftrag wird an der Stelle h in G_{best} mit Auftrag 2 identifiziert. Dieser befindet sich aktuell in Partikel p an der Stelle $r = 8$. Da die Geschwindigkeit $v_{p,8,t} \neq 0$, werden beide Aufträge getauscht und $v_{p,3,t}$ zufallsbasiert aktualisiert.

Position	1	2	3	4	5	6	7	8	9	10
G_{best}	4	8	2	7	9	1	3	5	6	10
P_p	5	6	1	3	10	8	4	2	9	7
$V_{p,t}$	1	-1	1	0	0	1	1	-1	-1	1

↓

Position	1	2	3	4	5	6	7	8	9	10
G_{best}	4	8	2	7	9	1	3	5	6	10
P_p	5	6	2	3	10	8	4	1	9	7
$V_{p,t+1}$	1	-1	-1	0	0	1	1	-1	-1	1

Abbildung 27: Beispielhafte Partikelbewegung

Das Ergebnis ist in der unteren Hälfte von Abbildung 27 dargestellt.

Aktualisierung

Nachdem die Bewegung des Partikels durchgeführt wurde, werden die Aufträge mithilfe der First-Fit-Regel Batches zugeteilt und die Kommissioniertour sowie die Kommissionierweglänge mit dem Tourenplanungsverfahren bestimmt. Anschließend werden gegebenenfalls P_p^{best} und G_{best} aktualisiert und die Aktualisierung des nächsten Partikels durchgeführt. Nachdem alle Partikel aktualisiert wurden, wird geprüft, ob sich G_{best} verbessert hat. Falls ja, wird der Stagnationsfaktor $S^{stagG_{best}}$ zu null gesetzt, ansonsten um eins erhöht.

6.2.4 Mutation

Wenn ein Partikel sich nahe am G_{best} bzw. seinem P_p^{best} befindet, besteht die Gefahr, dass er dort verharrt und sich im Laufe des Optimierungsverfahrens nicht mehr bewegt. Das Konzept der Mutation wird in das Verfahren integriert, um diese Partikel gezielt verändern zu können.

Mutationswahrscheinlichkeit

Die Mutation der Partikel wird mithilfe einer Mutationswahrscheinlichkeit gesteuert. Zur Berechnung der Mutationswahrscheinlichkeit muss zunächst die Intensität Int_p eines Partikels p berechnet werden:

$$\text{Int}_p = \frac{1}{3} \cdot (\text{Df}_{p,p^{\text{best}}} + \text{Df}_{p,G_{\text{best}}} + \text{Df}_{p^{\text{best}},G_{\text{best}}}) \quad (6.13)$$

Die Intensität eines Partikels gibt an, wie sehr sich der Partikel p von seinem p^{best} und von dem G_{best} unterscheidet und wie nahe p^{best} und G_{best} zueinander liegen. Je näher die drei Punkte im Lösungsraum beieinander liegen, desto höher ist die Intensität des Partikels. Mithilfe der maximalen Intensität aller Partikel Int^{max} und der minimalen Intensität aller Partikel Int^{min} kann die Mutationswahrscheinlichkeit M_p eines Partikels wie folgt berechnet werden:

$$M_p = \frac{\text{Int}^{\text{max}} - \text{Int}_p}{\text{Int}^{\text{max}} - \text{Int}^{\text{min}}}, \quad \text{falls } \text{Int}^{\text{max}} \neq \text{Int}^{\text{min}} (= 1, \text{sonst}) \quad (6.14)$$

Je größer die Differenz ist, desto größer ist die Intensität und umso kleiner ist die Mutationswahrscheinlichkeit.

Relative Nähe zum bislang besten Wert

Es wird unterstellt, dass ein Partikel näher am globalen Optimum ist als ein anderer Partikel, wenn sein Fitnesswert höher ist. Dann sollte nur ein kleiner Eingriff vorgenommen werden, um den Partikel nicht zu weit vom vermuteten globalen Optimum zu entfernen. Sonst sollte ein größerer Eingriff erfolgen. Die Stärke des Eingriffs wird durch die relative Nähe Nä_p des Partikels p zum bislang besten Wert festgelegt. Diese kann mithilfe der Kommissionierweglänge des Partikels p (KW_p), des G_{best} ($\text{KW}_{G_{\text{best}}}$) und die des aktuell schlechtesten Partikels (KW^{max}) berechnet werden.

$$\text{Nä}_p = \frac{\text{KW}_p - \text{KW}_{G_{\text{best}}}}{\text{KW}^{\text{max}} - \text{KW}_{G_{\text{best}}}} \quad (6.15)$$

Mutationsoperatoren

Es werden drei Mutationsoperatoren eingesetzt:

- Swap-Operator: Zwei Positionen werden zufällig ausgewählt und die Aufträge getauscht.
- Shift-Operator: Der Auftrag einer zufällig bestimmten Position wird auf eine andere zufällig bestimmte Position verschoben. Die nachfolgenden Aufträge werden nach hinten verschoben.

- Inverse-Operator: Zwei Positionen werden zufällig ausgewählt und die Aufträge getauscht. Die Reihenfolge der dazwischen liegenden Aufträge wird umgekehrt.

Der Swap-Operator führt zu dem schwächsten, der Inverse-Operator zum stärksten Eingriff. Abbildung 28 zeigt die Operatoren anhand eines Beispiels.

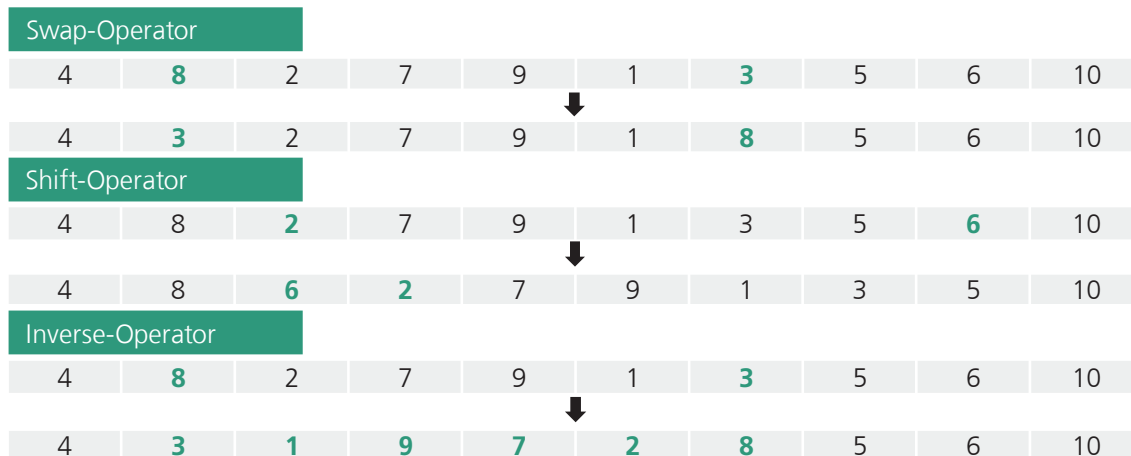


Abbildung 28: Mutationsoperatoren

Der Ablauf der Mutation kann wie folgt beschrieben werden:

Ablauf Mutation (V.)

Schritt 1

Berechne die Intensität der Partikel mithilfe von (6.13).

Schritt 2

Wähle Partikel 1 als aktuellen Partikel p .

Schritt 3

Berechne die Mutationswahrscheinlichkeit M_p des Partikels mithilfe von (6.14).

Schritt 4

Falls $M_p > \text{rand}$, berechne $N\ddot{a}_p$ mithilfe von (6.15) und fahre mit Schritt 5 fort. Sonst erhöhe p um 1 und führe Schritt 3 durch. Beende das Verfahren, wenn der aktuelle Partikel der letzte noch nicht untersuchte Partikel war.

Schritt 5

Falls $N\ddot{a}_p < 0,5$, wende den Swap-Operator an. Ist $0,5 \leq N\ddot{a}_p < 0,8$ wende den Shift-Operator an. Sonst wende den Inverse-Operator an.

Schritt 6

Wähle zufallsbasiert zwei Positionen im Geschwindigkeitsvektor $V_{p,t}$ des Partikels aus. Tausche die beiden Geschwindigkeiten. Erhöhe p um 1 und führe Schritt 3 durch. Beende das Verfahren, wenn der aktuelle Partikel der letzte noch nicht untersuchte Partikel war.

6.2.5 Lokales Suchverfahren

Wenn sich G_{best} seit einigen Iterationsschleifen nicht mehr verbessert hat, droht das Verfahren zu stagnieren und möglicherweise in einem lokalen Optimum zu verharren. Um dies zu verhindern, wird ein lokales Suchverfahren eingesetzt. Dadurch kann aus einem lokalen Optimum ausgebrochen werden. Da die lokale Suche Rechenzeit in Anspruch nimmt, wird sie nicht in jeder Iterationsschleife durchgeführt. Die lokale Suche wird nur durchgeführt, wenn sich G_{best} einige Zeit nicht mehr verbessert hat. Um den Zeitanteil der lokalen Suche anfangs niedrig zu halten und im Zeitverlauf steigen zu lassen, wird eine adaptive Stagnationsschranke S^{Stag} , die im Zeitverlauf sinkt eingesetzt:

$$S^{Stag} = \text{runden} \left(1 + S^{\maxStag} \cdot \frac{It^{\max} - It^{\text{akt}}}{It^{\max}} \right) \quad (6.16)$$

In (6.16) steht S^{\maxStag} für die vor dem Beginn des Verfahrens festgelegte maximale Stagnationsschranke, It^{\max} für die maximale Anzahl Iterationen des DEPSO-Algorithmus und It^{akt} für die aktuelle Iterationsschleife des DEPSO-Algorithmus. Das Symbol runden steht für die Rundungsfunktion; sie ordnet einer reellen Zahl die nächstliegende nicht größere bzw. nicht kleinere ganze Zahl zu. Ist die Ziffer an der ersten wegfallenden Dezimalstelle eine 0, 1, 2, 3 oder 4, dann wird abgerundet. Ist die Ziffer an der ersten wegfallenden Dezimalstelle eine 5, 6, 7, 8 oder 9, dann wird aufgerundet. Ist die Anzahl der Iterationen, in denen sich G_{best} nicht mehr verbessert

hat ($S^{\text{Stag}G_{\text{best}}}$), größer als die Stagnationsschranke multipliziert mit einer Zufallszahl, schaltet sich die lokale Suche ein. Die lokale Suche läuft wie folgt ab:

Ablauf lokales Suchverfahren (VI.)

Schritt 1

Prüfe, ob $S^{\text{Stag}G_{\text{best}}} > S^{\text{Stag}} \cdot \text{rand}$. Falls ja, setze den Iterationszähler der lokalen Suche It^{LS} auf 0 und fahre mit Schritt 2 fort, ansonsten breche das Verfahren ab.

Schritt 2

Führe den Swap-Operator durch.

Schritt 3

Führe die Batchzuteilung mithilfe der First-Fit-Regel durch.

Schritt 4

Berechne die Kommissioniertour und die Kommissionierweglänge mithilfe des Tourenplanungsverfahrens in (I.) und (II.). Falls sich der Kommissionierweg verkürzt hat, setze $S^{\text{Stag}G_{\text{best}}}$ auf null, aktualisiere G_{best} , wähle zufallsbasiert einen Partikel und setze ihn gleich dem G_{best} und beende das Verfahren. Falls nicht, erhöhe It^{LS} um 1 und fahre mit Schritt 2 fort. Beende das Verfahren, wenn die vorgegebene maximale Iterationsanzahl It^{maxLS} erreicht ist.

Das Ergebnis der lokalen Suche wird also nur akzeptiert, wenn eine bessere Lösung gefunden wurde. Ansonsten wird der ursprüngliche G_{best} beibehalten.

6.2.6 Ablauf

Vor Beginn der DEPSO müssen folgende Parameter festgelegt werden:

- Anzahl Partikel: A^{Partikel}
- Anzahl Iterationen: It^{max}
- Schwellwert G_{best} : $S^{\text{G}_{\text{best}}}$
- Maximale Anzahl Iterationen des lokalen Suchverfahrens: It^{maxLS}
- Maximale Stagnationsschranke: S^{maxStag}

Die Parameter haben einen wesentlichen Einfluss auf die Leistungsfähigkeit des Verfahrens. Die Einstellung der Parameter erfolgt im Rahmen von Vortests, bei denen

unterschiedliche Parameterkombinationen getestet werden. Die Parameterkombination, die die besten Resultate liefert, wird für das Verfahren genutzt. Die Durchführung von Vortests wird in den Kapiteln 7.3 und 8.2 beschrieben.

Damit sind alle Bestandteile des DEPSO-Algorithmus bekannt und dessen Ablauf kann wie folgt beschrieben werden:

Ablauf DEPSO

(VII.)

Schritt 1

Erzeuge mithilfe der Codierungsregel aus Kapitel 6.2.2 zufallsbasiert für die vorgegebene Anzahl Partikel jeweils eine Permutation mit den einzuplanenden Aufträgen. Erzeuge für jeden Partikel, wie in Kapitel 6.2.2 beschrieben, zufallsbasiert einen Geschwindigkeitsvektor.

Schritt 2

Ermittle mithilfe des Savings-Verfahrens aus (III.) eine Lösung und ersetze einen Partikel durch die gefundene Permutation.

Schritt 3

Führe die Batchzuteilung für alle Partikel mithilfe der First-Fit-Regel durch.

Schritt 4

Ermittle die Kommissioniertour und die Kommissionierweglänge der Partikel mithilfe des Tourenplanungsverfahrens aus (I.) und (II.).

Schritt 5

Setze die aktuelle Permutation der Partikel als ihren P_p^{best} und wählen den Partikel mit dem kürzesten Kommissionierweg als G_{best} .

Schritt 6

Wähle den ersten Partikel als $p = 1$.

Schritt 7

Führe mithilfe von (IV.) die Bewegung des Partikels p durch.

Schritt 8

Führe die Batchzuteilung für Partikel p mithilfe der First-Fit-Regel durch.

Schritt 9

Ermittle die Kommissioniertour und die Kommissionierweglänge von Partikel p mithilfe des Tourenplanungsverfahrens aus (I.) und (II.).

Schritt 10

Aktualisiere gegebenenfalls P_p^{best} und G_{best} . Wenn noch nicht alle Partikel aktualisiert wurden, erhöhe p um 1 und fahre mit Schritt 7 fort. Sonst fahre mit Schritt 11 fort.

Schritt 11

Prüfe, ob G_{best} aktualisiert wurde, falls ja, setze $S^{\text{Stag}G_{\text{best}}} = 0$. Sonst erhöhe $S^{\text{Stag}G_{\text{best}}}$ um 1.

Schritt 12

Führe die Mutation mithilfe von (V.) durch.

Schritt 13

Führe das lokale Suchverfahren mithilfe von (VI.) durch.

Schritt 14

Prüfe, ob $It^{\text{akt}} < It^{\text{max}}$. Falls ja, fahre mit Schritt 6 fort und erhöhe It^{akt} um 1. Sonst breche das Verfahren ab und gebe die beste bislang gefundene Batchzuteilung, die Kommissioniertouren und die Kommissionierweglängen aus.

6.3 Dynamischer Lagerplatzvergabealgorithmus

Das in Kapitel 5.1.2 beschriebene dynamische Lagerplatzvergabeproblem setzt voraus, dass die zukünftigen Aufträge und die Häufigkeiten, mit denen zwei Artikel nacheinander kommissioniert werden, bekannt sind. Beides trifft zum Planungsstart der Lagerplatzvergabe nicht zu. Zur Auflösung dieses Konflikts wurde ein dynamischer Lagerplatzvergabealgorithmus entwickelt, der im Folgenden vorgestellt wird. Die Lagerplatzvergabe erfolgt, wie in Kapitel 2.5 beschrieben, klassenbasiert. Das Verfahren bietet zwei wesentliche Neuheiten im Vergleich zu bestehenden Verfahren: Zum einen wird die Interdependenz der drei Planungsprobleme erstmals berücksichtigt. Artikel werden so angeordnet, dass sie möglichst effizient mithilfe des

Auftragsbildungs- und Tourenplanungsverfahren kommissioniert werden können. Außerdem werden Umlagerungen nur getätigt, wenn die zukünftige Kommissionierweinsparung den Aufwand übersteigt. Die zukünftige Artikelnachfrage kann so zum Planungszeitpunkt in der Artikelanordnung berücksichtigt werden. Das Verfahren hat das von Pierre et al. (2003) entwickelte Prioritätswertverfahren lose als Vorbild. Allerdings wurde es aufgrund seiner in Kapitel 4.2 identifizierten Unzulänglichkeiten für das zu lösende Problem stark erweitert. So wurden ein Prognoseverfahren und das Verfahren zur Lösung des kombinierten Auftragsbildungs- und Tourenplanungsproblems integriert, eine eigene Umlagerungslogik, ein neuartiges Schätzverfahren und eine Wirtschaftlichkeitsprüfung entwickelt. Im Folgenden wird der Aufbau des Verfahrens beschrieben.

6.3.1 Aufbau

Abbildung 29 zeigt den Aufbau des dynamischen Lagerplatzvergabealgorithmus.

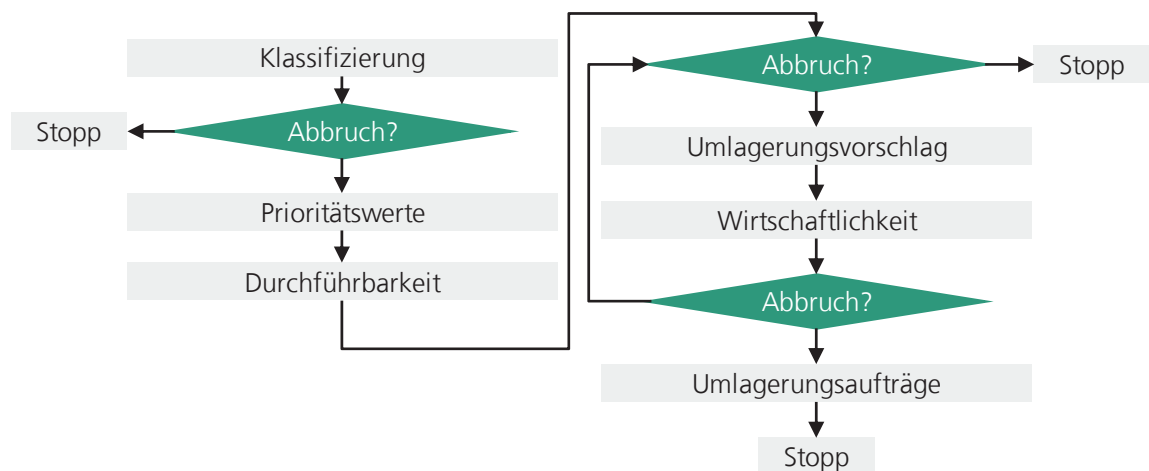


Abbildung 29: Aufbau dynamischer Lagerplatzvergabealgorithmus

Das Verfahren wird am Ende jeder Periode durchgeführt. Es ist unternehmensspezifisch festzulegen, ob eine Periode einen Tag, eine Woche, einen Monat oder ein Quartal umfassen soll. Das Verfahren beginnt mit der Überprüfung der *Klassifizierung* der Artikel. Dazu werden zunächst die Positionen je Artikel der aktuellen Periode sowie die Soll- und Ist-Klassifizierung der vergangenen Perioden bestimmt. An-

schließlich werden die zukünftig zu erwartenden Positionen je Artikelnummer prognostiziert und die Soll-Klassifizierung der Artikel für den Prognosezeitraum bestimmt. Basierend darauf werden Artikel identifiziert, die aktuell in der falschen Klasse gelagert werden. Für diese Artikel werden *Prioritätswerte* berechnet und die *Durchführbarkeit* der neuen Klassifizierung sichergestellt. Für den Artikel mit dem höchsten Prioritätswert, der in eine höhere Klasse aufsteigen soll, wird mithilfe der Umlagerungslogik ein *Umlagerungsvorschlag* erstellt. Der Artikel wird hypothetisch umgelagert und die *Wirtschaftlichkeit* der Umlagerung geprüft. Hierzu wird die Kommissionierweglänge mithilfe des Auftragsbestands der aktuellen Periode und der hypothetischen Artikelanordnung mithilfe des DEPSO-Algorithmus berechnet. Die Wegstreckenreduktion der aktuellen Periode ergibt sich aus der Differenz der Kommissionierweglänge mit Umlagerung und der ohne Umlagerung. Die Wegstreckenreduktion wird mithilfe eines Schätzverfahrens in die Zukunft projiziert. Wenn diese den Umlagerungsaufwand übersteigt, wird die Umlagerung angenommen, ansonsten verworfen. Es werden so lange Umlagerungsprüfungen durchgeführt, bis ein Abbruchkriterium erreicht ist und die *Umlagerungsaufträge* ausgegeben werden. Im Folgenden werden die einzelnen Verfahrensbestandteile beschrieben.

6.3.2 Klassifizierung

Das Verfahren teilt Artikel mithilfe einer ABC-Analyse entsprechend ihrer Nachfrage in die drei Klassen A, B und C ein. Es ist unternehmensspezifisch festzulegen, wie viel Prozent der Artikel in Klasse A, B bzw. C gelagert werden sollen. Die Klassifizierung basiert auf Zeitreihen, die die Anzahl Positionen je Artikelnummer und Periode abbilden. Am Ende jeder Periode werden die Anzahl Positionen je Artikelnummer ermittelt und gespeichert. Mithilfe eines Prognoseverfahrens, wie etwa dem in Kapitel 3.5 beschriebenen Holt-Winters-Verfahren, wird die Anzahl Positionen je Artikelnummer für die Zukunft bestimmt. Ein Beispiel für eine Artikelzeitreihe ist in Abbildung 30 dargestellt. Neben der Artikelzeitreihe sind auch die Verläufe der beiden Klassengrenzen dargestellt. Die Anzahl Positionen der jeweiligen Klassengrenze entspricht der Anzahl Positionen des letzten noch in der Klasse enthaltenen Artikels.

Die Klassengrenzen variieren über die Zeit, da sich die Gesamtanzahl Positionen aller Artikel verändern kann. Von der (theoretisch richtigen) Klassifizierung ist die tatsächliche physische Lagerung des Artikels zu unterscheiden. Da ein Lager in den meisten Fällen nicht voll belegt ist, gibt es mehr Lagerplätze als zu lagernde Artikel. Allerdings können manche Lagerbereiche voll belegt sein, wohingegen in anderen Bereichen noch freie Lagerplätze verfügbar sein können.

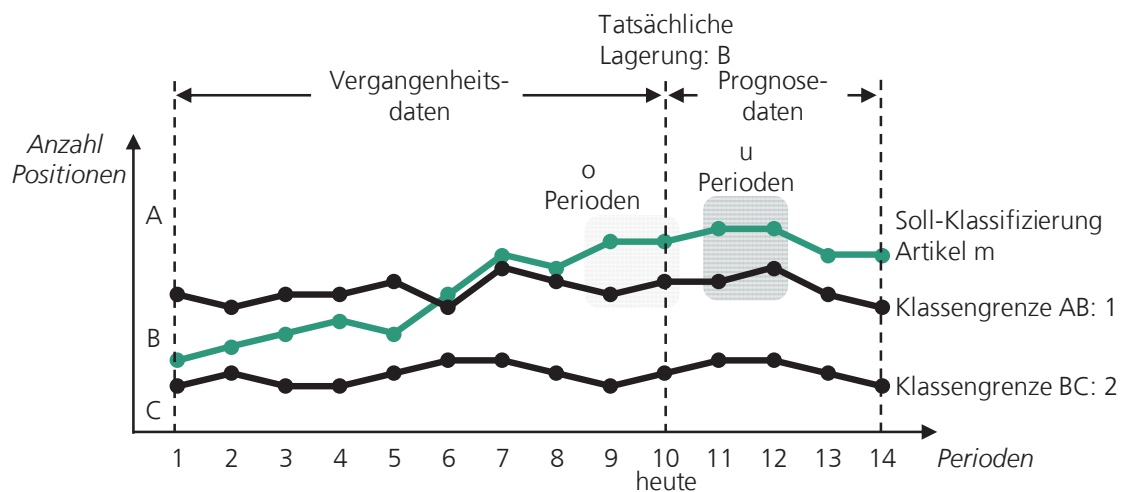


Abbildung 30: Artikelzeitreihe und Klassengrenzen

In dem in Abbildung 30 dargestellten Beispiel wird Artikel m in der aktuellen Periode 10 (physisch) in Klasse B gelagert. Von Periode 1 bis 5 war der Artikel richtig gelagert. Ab Periode 6 hätte der Artikel (theoretisch) in Klasse A gelagert werden müssen. Gemäß Prognose wird der Artikel die nächsten vier Perioden in der Zielklasse A verbleiben. Ein Artikel sollte umgelagert werden, wenn die Zeitreihe mindestens o Vergangenheitsperioden (inklusive der aktuellen Periode) in der Zielklasse lag und mindestens u Zukunftsperioden in der Zielklasse liegen wird. Die beiden Schranken wurden eingeführt, um Umlagerungsvorschläge nicht prüfen zu müssen, die sich mit großer Wahrscheinlichkeit ohnehin nicht lohnen würden. So kann die begrenzte Rechenzeit gezielter eingesetzt werden. Alle Artikel, die diese Kriterien erfüllen, gelten als falsch gelagert und werden in einer Liste gespeichert. Sollte die Liste leer sein, bricht das Verfahren ab.

6.3.3 Prioritätswertberechnung

Für die falsch gelagerten Artikel werden Prioritätswerte berechnet. Die Vergabe von Prioritätswerten erfolgt, da aufgrund von Rechenzeitbeschränkungen in vielen Fällen nicht alle Artikelumlagerungen geprüft werden können. Durch die Einführung der Schwellwerte o und u werden manche Artikel von der Umlagerungsprüfung ausgeschlossen. Deshalb kann eine Situation entstehen, in der nicht alle verbleibenden Artikel umgelagert werden können. Um eine zulässige Lösung zu erzeugen, müssen einige Artikel von der Umlagerung ausgeschlossen werden. Die Entscheidung hierfür basiert auf den Prioritätswerten. Die Berechnung des Prioritätswerts eines Artikels basiert auf dessen prognostizierten Zeitreihenwerte $\hat{A}_{m,t}^{\text{Pos}}$ (Abbildung 31).

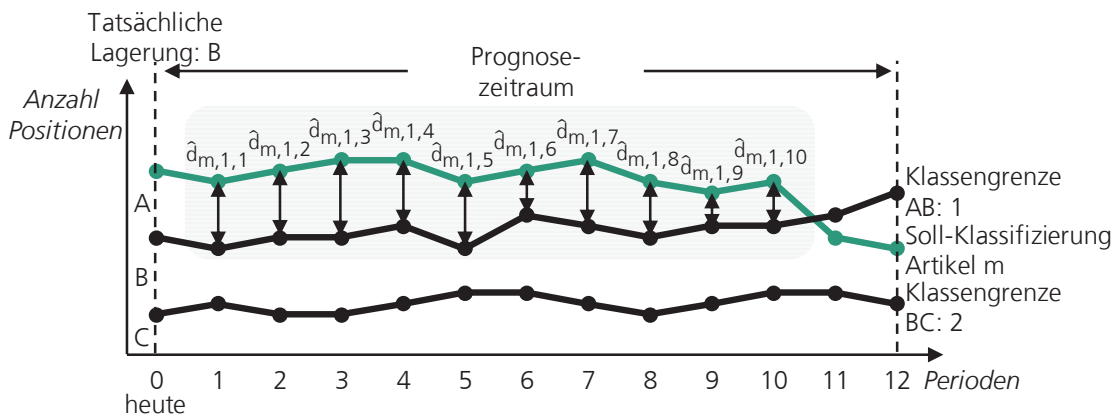


Abbildung 31: Prioritätswertberechnung

Betrachtet werden die Perioden im Zeitraum U_m^{Ziel} , in dem der Artikel zukünftig in der Zielklasse sein wird. Die Grenze zwischen Klasse A und B erhält den Wert 1 und die Grenze zwischen B und C den Wert 2. Aus der Gesamtbetrachtung aller Artikelnummern lassen sich die Anzahl Positionen $\hat{A}_{\text{relKG},t}^{\text{Pos}}$ der relevanten Klassengrenze relKG (= Grenze zur Zielklasse) bestimmen. Somit lassen sich die Distanzen $\hat{d}_{m,\text{relKG},t}$ zwischen der Zeitreihe eines Artikels m und der relevanten Klassengrenze für die Perioden bestimmen:

$$\hat{d}_{m,\text{relKG},t} = |\hat{A}_{m,t}^{\text{Pos}} - \hat{A}_{\text{relKG},t}^{\text{Pos}}| \quad \forall t \in U_m^{\text{Ziel}} \quad (6.17)$$

Die Distanzen werden über die Perioden addiert. Der Prioritätswert PW_m des Artikels m berechnet sich damit wie folgt:

$$PW_m = \sum_{t \in U_m^{\text{Ziel}}} \hat{d}_{m, \text{relKG}, t} \quad (6.18)$$

Die Prioritätswerte der Artikel werden in einer Prioritätswertliste gespeichert und nach absteigenden Werten sortiert.

6.3.4 Durchführbarkeit

Es muss sichergestellt werden, dass die Artikel in der Prioritätswertliste umgelagert werden können, es also ausreichend freie Lagerplätze in den Klassen kl (A, B oder C) gibt. Hierzu ist die Anzahl derzeit leerer Lagerplätze in Klasse kl AL_{kl}^{derzl} , die Anzahl leer werdender Lagerplätze in Klasse kl AL_{kl}^{leerw} und die Anzahl voll werdender Lagerplätze in Klasse kl AL_{kl}^{vollw} zu ermitteln. Damit kann der Durchführbarkeitsfaktor DF_{kl} der Klasse kl wie folgt berechnet werden:

$$DF_{kl} = AL_{kl}^{\text{derzl}} + AL_{kl}^{\text{leerw}} - AL_{kl}^{\text{vollw}} \quad (6.19)$$

Wenn alle DF_{kl} größer oder gleich null sind, können alle Umlagerungen durchgeführt werden. Gibt es negative Durchführbarkeitsfaktoren, fehlen in der betreffenden Klasse freie Lagerplätze. In diesem Fall ist die Klasse mit dem niedrigsten Durchführbarkeitsfaktor auszuwählen und $|DF_{kl}|$ Artikel, die in die Klasse wechseln sollen, aus der Prioritätswertliste zu entfernen. Gewählt werden die Artikel mit den niedrigsten Prioritätswerten. Anschließend werden alle Durchführbarkeitsfaktoren aktualisiert und das Vorgehen so lange wiederholt, bis alle Durchführbarkeitsfaktoren nicht negativ sind.

6.3.5 Umlagerungsvorschlag

Zur Erzeugung eines Umlagerungsvorschlags wird die Prioritätswertliste in zwei Listen aufgeteilt. Die Aufsteigerliste enthält alle Artikel, die in eine höhere Klasse wechseln sollen und die Absteigerliste alle Artikel, die in eine niedrigere Klasse wechseln sollen. Dabei ist A die höchste und C die niedrigste Klasse. Die Listen sind wiederum nach absteigenden Prioritätswerten sortiert. Aus der Aufsteigerliste ist der oberste Artikel zu wählen und eine Umlagerung zu prüfen. Haben mehrere Artikel denselben Prioritätswert, wird der Artikel mit dem Lagerplatz, der am entferntesten zum Depot liegt, gewählt. Sollten dann immer noch mehrere Artikel zur Wahl stehen,

wird ein Artikel zufallsbasiert ausgewählt. Dabei sind die in Abbildung 32 dargestellten Umlagerungsfälle möglich.

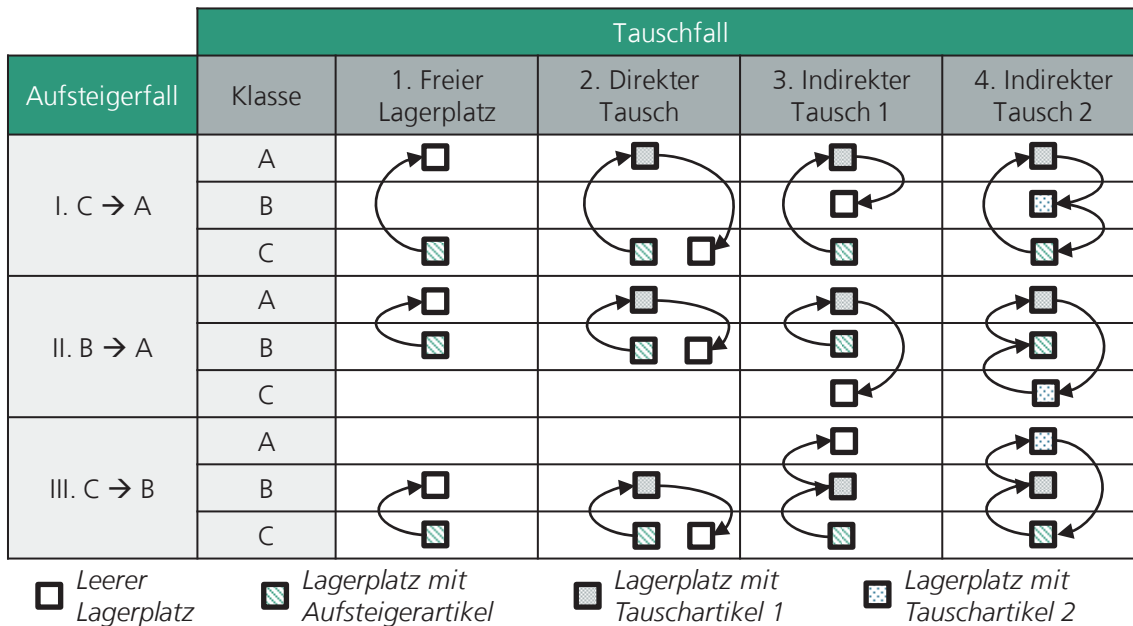


Abbildung 32: Umlagerungsfälle

Für jeden der drei Aufsteigerfälle gibt es vier mögliche Tauschfälle. Im Tauschfall 1 wird der Aufsteigerartikel auf einen freien Lagerplatz in der Zielklasse gelagert. Tauschfall 1 hat den niedrigsten Umlagerungsaufwand. In den anderen drei Fällen wird ein Artikel in der Zielklasse aus einem Lagerplatz entnommen und der Aufsteigerartikel in diesen eingelagert. Im Tauschfall 2 wird der zu tauschende Artikel in die Ist-Klasse des Aufsteigerartikels umgelagert. Dort wird ein Lagerplatz zufallsbasiert ausgewählt. Dies kann theoretisch auch der aktuelle Lagerplatz des Aufsteigerartikels sein. Im Tauschfall 3 wird der zu tauschende Artikel auf einen zufällig ausgewählten freien Lagerplatz in der bislang unbeteiligten Klasse gelagert. Im Tauschfall 4 ist ein weiterer Artikel involviert. Die drei Artikel tauschen ihren Lagerplatz. Tauschfall 4 hat den höchsten Umlagerungsaufwand. Zur Bestimmung des Absteigerartikels wurde wie folgt vorgegangen: Ausgewählt wird der Artikel mit dem höchsten Prioritätswert, der aus der Zielklasse des Aufsteigerartikels absteigt. Auf diese Weise wird der Artikel umgelagert, der zukünftig am wenigsten kommissioniert wird. Dadurch ist die höchste Einsparung zu erwarten. Sollten mehrere Artikel

den gleichen Prioritätswert haben, wird der Artikel mit dem Lagerplatz gewählt, der am nächsten zum Depot liegt, da so der zukünftige Kommissionierweg des Aufsteigerartikels gesenkt werden kann. Sollte es weiterhin mehrere zur Auswahl stehende Artikel geben, wird aufgrund des Umlagerungsaufwands Tauschfall 1 vor 2, 3 und 4 sowie 2 und 3 vor 4 priorisiert. Zwischen 2 und 3 wird zufallsbasiert ein Artikel ausgewählt.

6.3.6 Wirtschaftlichkeit

Im Zuge der Wirtschaftlichkeitsprüfung werden der Umlagerungsaufwand und die zukünftig zu erwartende Kommissionierwegeinsparung bestimmt.

Umlagerungsaufwand

Der Umlagerungsaufwand setzt sich aus drei Komponenten zusammen. Der *Distanzaufwand* U_m^{disAuf} entspricht der Strecke zwischen dem aktuellen und dem zukünftigen Lagerort eines Artikels m . Der *physische Umlagerungsaufwand* wird bei der Entnahme eines Artikels aus dem aktuellen Lagerplatz und der Einlagerung des Artikels in den neuen Lagerplatz notwendig. Hierfür ist unternehmensspezifisch eine pauschale Zeit z^{phy} für einen physischen Umlagerungsvorgang zu bestimmen. Diese Zeit wird mithilfe der durchschnittlichen Gehgeschwindigkeit des Kommissionierers v^{kom} in ein Kommissionierwegäquivalent umgerechnet:

$$U_m^{\text{phyAuf}} = z^{\text{phy}} \cdot v^{\text{kom}} \quad (6.20)$$

Der administrative Aufwand für die Verbuchung der Umlagerung im System wird mithilfe der Zeit für die administrative Umlagerung z^{adm} berechnet:

$$U_m^{\text{admAuf}} = z^{\text{adm}} \cdot v^{\text{kom}} \quad (6.21)$$

Der gesamte Umlagerungsaufwand eines Artikels m berechnet sich wie folgt:

$$U_m^{\text{gesAuf}} = U_m^{\text{disAuf}} + U_m^{\text{phyAuf}} + U_m^{\text{admAuf}} \quad (6.22)$$

Im Zuge einer Umlagerungsprüfung wird der Umlagerungsaufwand aller beteiligten Artikel berechnet und bei der Wirtschaftlichkeitsprüfung berücksichtigt.

Kommissionierwegeinsparung

Zunächst wird die Kommissionierweglänge KW^{aktat} bestimmt, die in der aktuellen Periode mit der tatsächlichen Artikelanordnung zurückgelegt wurde. Diese kann entweder gemessen oder mithilfe des DEPSO-Algorithmus bestimmt werden. Sollte in der aktuellen Periode noch kein Umlagerungsvorschlag angenommen worden sein, wird KW^{aktat} als Vergleichskommissionierweglänge KW^{ver} genutzt. Anschließend wird die Kommissionierweglänge KW^{tau} mithilfe des DEPSO-Algorithmus und dem Auftragsbestand der aktuellen Periode ermittelt, der resultiert wäre, wenn die Umlagerung am Ende der letzten Periode durchgeführt worden wäre. Die Kommissionierwegstreckenreduktion $KR_{m,0}$ der aktuellen Periode ist damit:

$$KR_{m,0} = KW^{ver} - KW^{tau} \tag{6.23}$$

Ist $KR_{m,0}$ kleiner oder gleich null, wird der Umlagerungsvorschlag verworfen und ein neuer Vorschlag erstellt. Sonst wird die zukünftige Kommissionierwegeinsparung mit dem in Abbildung 33 dargestellten Verfahren, basierend auf dem Aufsteigerartikel, geschätzt.

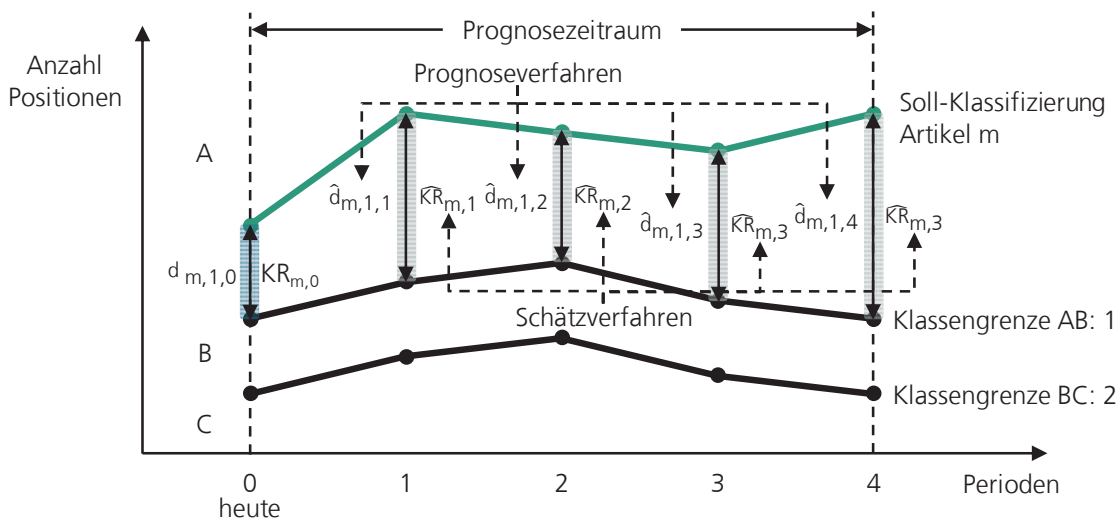


Abbildung 33: Schätzverfahren

Die Anzahl Positionen des Artikels und der relevanten Klassengrenze lassen sich aus den Aufträgen der aktuellen Periode bestimmen und so die Distanz $d_{m,relKG,0}$ berechnen. Mithilfe dieser Informationen lässt sich der relative Abstand $d_{m,relKG,0}^{rel}$ des Artikels m zur relevanten Klassengrenze relKG zum Zeitpunkt 0 bestimmen:

$$d_{m,\text{relKG},0}^{\text{rel}} = \frac{d_{m,\text{relKG},0}^{\text{Pos}}}{A_{\text{relKG},t}} \cdot 100\% \quad (6.24)$$

Hiermit lässt sich ein Faktor $\xi_{m,0}$ berechnen, der die Wegstreckenreduktion $KR_{m,0}$ ins Verhältnis zum relativen Abstand setzt:

$$\xi_{m,0} = \frac{KR_{m,0}}{d_{m,\text{relKG},0}^{\text{rel}}} \quad (6.25)$$

Dieser Faktor gibt an, mit wie viel Wegstreckenreduktion bei einem Prozent Abweichung von der relevanten Klassengrenze zu rechnen ist. Dieser Faktor kann als Schätzer für die zukünftigen Perioden genutzt werden:

$$\hat{\xi}_{m,t} = \xi_{m,0} \quad (6.26)$$

Mithilfe der prognostizierten Zeitreihenwerte des Artikels m und der relevanten Klassengrenze lassen sich mit (6.27) für die zukünftigen Perioden, in denen der Artikel in der Zielklasse verbleiben wird, die zu erwartenden Wegstreckenreduktionen schätzen:

$$\widehat{KR}_{m,t} = \hat{\xi}_{m,t} \cdot \hat{d}_{m,\text{relKG},t}^{\text{rel}} \quad \forall t \in U_m^{\text{Ziel}} \quad (6.27)$$

Der Umlagerungsvorschlag wird angenommen, wenn gilt:

$$\sum_{t \in U_m^{\text{Ziel}}} \widehat{KR}_{m,t} > U_m^{\text{gesAuf}} \quad (6.28)$$

Sonst wird der Umlagerungsvorschlag verworfen. In beiden Fällen werden die beteiligten Artikel aus der Prioritätswertliste entfernt. Wenn das Abbruchkriterium noch nicht erreicht wurde, wird der nächste Umlagerungsvorschlag geprüft. Wenn der Umlagerungsvorschlag angenommen wird, wird die berechnete Kommissionierweglänge nach dem Tausch KW^{tau} zur aktuellen Vergleichskommissionierweglänge KW^{ver} :

$$KW^{\text{ver}} = KW^{\text{tau}} \quad (6.29)$$

Der nächste Umlagerungsvorschlag wird gegen die aktualisierte Vergleichskommissionierweglänge getestet.

6.3.7 Ablauf

Zu Beginn des Verfahrens werden eine maximale Anzahl an Umlagerungsprüfungen und die Schwellwerte o und u festgelegt. Damit lässt sich der Ablauf des dynamischen Lagerplatzvergabeverfahrens wie folgt beschreiben:

Ablauf dynamischer Lagerplatzvergabealgorithmus (VIII.)

Schritt 1

Bestimme die Anzahl Positionen aller Artikel und die Klassengrenzen für die aktuelle und für die vergangenen ($o-1$) Perioden.

Schritt 2

Ermittle mithilfe eines geeigneten Prognoseverfahrens die Anzahl Positionen aller Artikel für den Prognosezeitraum.

Schritt 3

Bestimme die Artikelklassifizierung aller Artikel für die aktuelle Periode, die ($o-1$) vergangenen Perioden sowie die prognostizierten Perioden.

Schritt 4

Identifiziere alle Artikel, die seit mindestens o Perioden falsch gelagert wurden und mindestens u Perioden in der Zielklasse verbleiben werden. Gibt es keine in diesem Sinne falsch gelagerten Artikel, breche das Verfahren ab. Sonst fahre mit Schritt 6 fort.

Schritt 6

Berechne mithilfe von (6.18) die Prioritätswerte der Artikel.

Schritt 7

Prüfe mithilfe von (6.19) die Durchführbarkeit der neuen Klassifizierung und stelle sie gegebenenfalls mithilfe des in Kapitel 6.3.4 beschriebenen Vorgehens sicher.

Schritt 8

Prüfe, ob Artikel übrig sind. Falls nicht, breche das Verfahren ab. Sonst fahre mit Schritt 9 fort.

Schritt 9

Erstelle mithilfe der in Kapitel 6.3.5 beschriebenen Logik einen Umlagerungsvorschlag.

Schritt 10

Berechne mithilfe von (6.22) den Umlagerungsaufwand, der zur Umsetzung der geplanten Umlagerung notwendig ist.

Schritt 11

Wurde bislang noch kein Umlagerungsvorschlag geprüft, bestimme die Kommissionierweglänge der aktuellen Periode mit der tatsächlichen Artikelanordnung mithilfe des DEPSO-Algorithmus aus (VII.). Wurde bislang noch kein Umlagerungsvorschlag angenommen, speichere diese als Vergleichskommissionierweglänge.

Schritt 12

Führe die Umlagerung hypothetisch durch und berechne mithilfe des DEPSO-Algorithmus aus (VII.) die daraus resultierende Kommissionierweglänge für die aktuelle Periode.

Ist diese größer als die Vergleichskommissionierweglänge, dann verwerfe den Umlagerungsvorschlag. Ist die maximale Anzahl an Umlagerungsprüfungen erreicht, breche das Verfahren ab und gebe die Umlagerungsaufträge aus. Ansonsten fahre mit Schritt 8 fort.

Ist die berechnete Kommissionierweglänge kleiner als die Vergleichskommissionierweglänge, dann speichere sie als Tauschkommissionierweglänge und fahre mit Schritt 13 fort.

Schritt 13

Bestimme mithilfe von (6.27) die zukünftig zu erwartende Wegstreckenreduktion.

Schritt 14

Prüfe, ob die zukünftigen Wegstreckenreduktionen den Umlagerungsaufwand übersteigen. Falls ja, führe die Umlagerung durch und aktualisiere die Artikelanordnung. Speichere die Tauschkommissionierweglänge als Vergleichskommissionierweglänge. Sonst verwerfe die Umlagerung.

Schritt 15

Entferne die am Umlagerungsvorschlag beteiligten Artikel aus der Prioritätswertliste.

Schritt 16

Prüfe, ob die maximale Anzahl an Umlagerungsprüfungen erreicht ist. Falls nicht, fahre mit Schritt 8 fort. Sonst breche das Verfahren ab und gebe die Umlagerungsaufträge aus.

6.4 Zwischenfazit

Der DEPSO-Algorithmus löst das Auftragsbildungs- und das Tourenplanungsproblem gemeinsam mit dem Ziel, die *Kommissionierweglänge* zu minimieren. Auch der dynamische Lagerplatzvergabealgorithmus verfolgt dieses Ziel. Er berücksichtigt die *Interdependenz* der drei Planungsprobleme, indem er eine Umlagerung unter Einsatz des Auftragsbildungs- und Tourenplanungsverfahrens bewertet. Die Artikelanordnung passt sich so an die tatsächlichen Kommissioniertouren an. Auftragsbildung, Tourenplanung und Lagerplatzvergabe werden damit ganzheitlich in einem iterativen Verfahren gelöst und deren Interdependenz wird berücksichtigt. Durch die Verwendung von Prognosedaten kann bei einer Umlagerungsprüfung die zukünftig zu erwartende Dynamik antizipiert werden. Umlagerungsvorschläge werden, nicht nur bezogen auf ihre zukünftig zu erwartende Wegstreckenreduktion, sondern auch hinsichtlich des notwendigen *Umlagerungsaufwands*, bewertet. Je Periode wird nur eine kleine Anzahl an Artikeln umgelagert und keine vollständige Lagerreorganisation durchgeführt.

7 Numerische Experimente

Im Rahmen der numerischen Experimente soll die Leistungsfähigkeit des DEPSO-Algorithmus und des dynamischen Lagerplatzvergabealgorithmus untersucht werden. Die Leistungsfähigkeit drückt sich in der erzielten Wegstreckenreduktion aus. Im Gegensatz zum dynamischen Lagerplatzvergabealgorithmus spielt beim DEPSO-Algorithmus auch die Rechenzeit eine wesentliche Rolle. Entscheidungen müssen hier sehr viel kurzzyklischer getroffen werden. Da der DEPSO auch unabhängig vom dynamischen Lagerplatzvergabealgorithmus eingesetzt werden kann, wird zunächst nur der DEPSO untersucht. In einem zweiten Teil des Kapitels wird dann die Leistungsfähigkeit des dynamischen Lagerplatzvergabealgorithmus untersucht. Zunächst wird in Kapitel 7.1 das der Untersuchung zugrunde liegende Kommissioniersystem vorgestellt. In Kapitel 7.2 werden die zu lösenden Problem instanzen für die Experimente mit dem DEPSO und dem dynamischen Lagerplatzvergabealgorithmus sowie deren Erzeugung beschrieben. Die optimalen Parametereinstellungen werden im Rahmen von umfangreichen Vortests in Kapitel 7.3 bestimmt. In den Kapiteln 7.4 (DEPSO) und 7.5 (dynamische Lagerplatzvergabe) werden die Ergebnisse der numerischen Experimente vorgestellt.

7.1 Betrachtungsgegenstand

Für die numerischen Experimente gelten die in Kapitel 5.1.1 getroffenen Annahmen. Untersucht wird das in Abbildung 34 dargestellte Kommissionierlager.

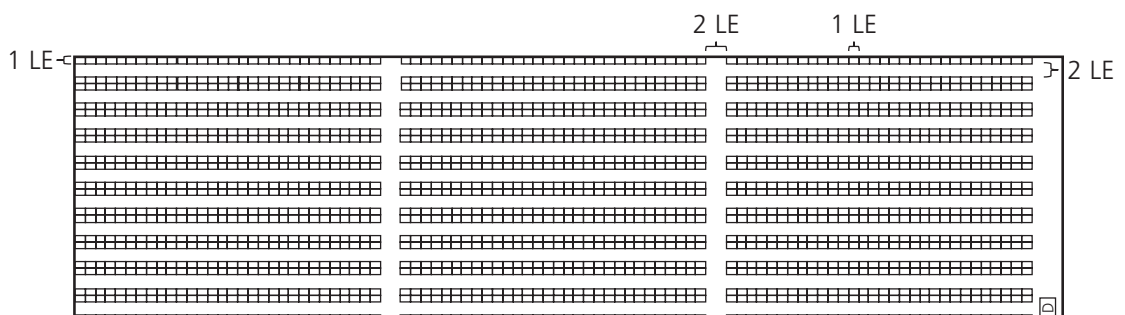


Abbildung 34: Lagerlayout der numerischen Experimente

Es besteht aus zehn Kommissioniergängen. Auf jeder Seite eines Kommissioniergangs sind 90 Lagerorte angeordnet. An einem Lagerort sind vier Lagerplätze vertikal in Regalböden übereinander angeordnet. Die Unterscheidung in Lagerorte und -plätze wird getroffen, da ein Kommissionierer seinen Standpunkt nicht verändern muss, wenn er Artikel aus den Lagerplätzen eines Lagerortes entnimmt. Das System verfügt insgesamt über 7.200 Lagerplätze. Das Lager hat vier Quergänge und drei Blocks. Ein Lagerort ist eine Längeneinheit (LE) breit und eine LE lang. Kommissionier- und Quergänge sind zwei LE breit. Das Depot befindet sich auf der rechten Seite des untersten Kommissioniergangs im rechten Quergang. Die Lagerplatzvergabe erfolgt klassenbasiert. Die Artikel haben ein zufällig erzeugtes Gewicht zwischen 0,1 und 1,0 Gewichtseinheiten (GE). Das Artikelsortiment umfasst 6.000 Artikel. Die Kapazität des Kommissioniergeräts sind 100 GE. Die Klasse A umfasst 5 %, B 15 % und C 80 % der Artikel des Sortiments. Die Lagerplätze werden den Klassen basierend auf ihrer Nähe zum Depot (Radialanordnung) zugewiesen.

7.2 Probleminstanzen

Für die Tests der Verfahren wurden umfangreiche Daten genutzt. In den nächsten beiden Kapiteln wird beschrieben, wie diese Daten erzeugt wurden.

7.2.1 Diskrete evolutionäre Partikelschwarmoptimierung

Für die Tests des DEPSO-Algorithmus wurden unterschiedliche Problemklassen definiert. Eine Klasse wird durch eine Ausprägung der Merkmale Anzahl Aufträge (A^{Auf}), maximale Anzahl Positionen pro Auftrag (A^{maxPos}) und maximale Menge je Position (M^{maxPos}) charakterisiert. Bei der Erzeugung der Daten wurde berücksichtigt, dass die Häufigkeit der Zugriffe auf einzelne Artikel stark voneinander abweichen kann. Dieser Zusammenhang lässt sich durch eine Zugriffsfunktion abbilden und mithilfe des Zugriffshäufigkeitsparameters ZH steuern. Dieser gibt den Anteil der 20 % zugriffstärksten Artikel an den Zugriffen auf alle Artikel an. Für die zugehörige Verteilungsfunktion (siehe Abbildung 35) gilt (Reschke 2013, S. 119-121):

$$Z = y^c \tag{7.1}$$

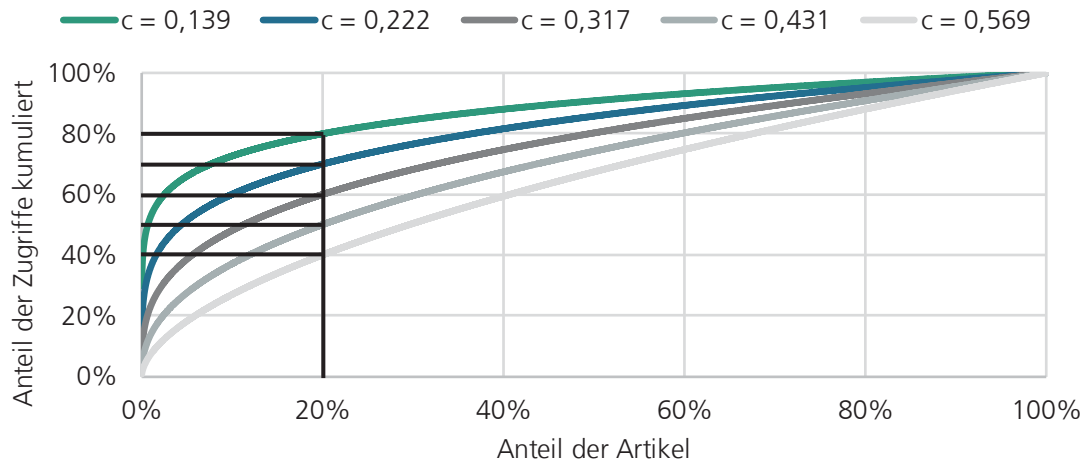


Abbildung 35: Zugriffshäufigkeitsverteilung (in Anlehnung an Reschke 2013, S. 120)

c kann in Abhängigkeit von ZH wie folgt bestimmt werden:

$$c = \frac{\log_{10} ZH}{\log_{10} 0.2} \quad (7.2)$$

Die Vorgehensweise zur Erzeugung der Testdaten zeigt Abbildung 36. Zufallszahlen werden im Folgenden immer per Gleichverteilung gezogen.

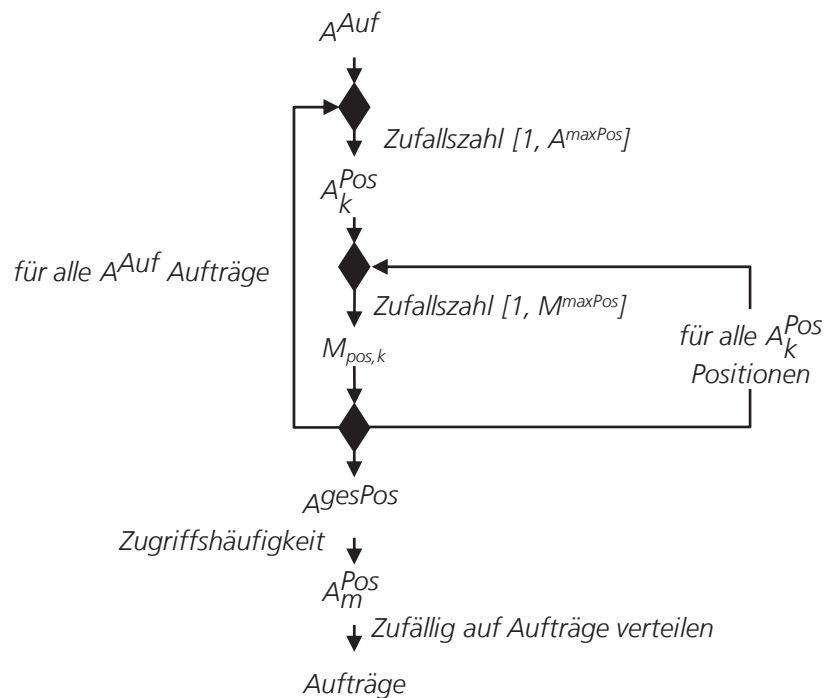


Abbildung 36: Erzeugung Auftragsdaten DEPSO

Zunächst werden A^{Auf} Aufträge erzeugt. Für jeden Auftrag werden über eine Zufallszahl aus $[1, A^{\text{maxPos}}]$ die Anzahl Positionen A_k^{Pos} festgelegt. Für jede Auftragszeile des Auftrags wird über eine Zufallszahl aus $[1, M^{\text{maxPos}}]$ die Menge je Position bestimmt. Nachdem die Anzahl Positionen und die Menge je Position für jeden der A^{Auf} Aufträge festgelegt wurden, kann die Gesamtanzahl Positionen (A^{gesPos}) über alle Aufträge (und damit Artikel) bestimmt werden. Über die Zugriffsverteilung werden die Anzahl Positionen je Artikelnummer (A_m^{Pos}) ermittelt. Diese werden, unter Beachtung der Anzahl Positionen der Aufträge, zufallsbasiert auf die Aufträge verteilt. Wenn alle Positionen eines Auftrags mit Artikelnummern gefüllt sind, wird der Auftrag geschlossen. Eine Artikelnummer kann nur einmal in einem Auftrag enthalten sein. Falls Artikel übrig bleiben, die nicht auf offene Aufträge verteilt werden können, weil sie bereits in den Aufträgen enthalten sind, dann wird ein zufällig bestimmter Artikel des offenen Auftrags mit einem anderen Artikel aus einem zufällig bestimmten, bereits geschlossenen Auftrag (der diesen Artikel noch nicht enthält) getauscht und der offene Artikel dem offenen Auftrag zugeteilt.

In den Experimenten wurden folgende Ausprägungen der Merkmale untersucht:

- Anzahl Aufträge (A^{Auf}): {50, 100, 150, 200}
- Maximale Anzahl Positionen pro Auftrag (A^{maxPos}): {2, 6, 10}
- Maximale Menge je Position (M^{maxPos}): {2, 6, 10}

Für den Zugriffshäufigkeitsparameter ZH wurde 0,6 gewählt. Das bedeutet, dass 20 % der Artikel für 60 % der Zugriffe verantwortlich sind. Zur Beschreibung der Problemklassen wird folgende Nomenklatur verwendet: $A^{\text{Auf}}_{A^{\text{maxPos}}}_{M^{\text{maxPos}}}$. Es wurden zufallsbasiert jeweils 40 Instanzen für 35 Problemklassen erzeugt. Die Problemklasse 50_2_2 wurde nicht untersucht, da sie mit hoher Wahrscheinlichkeit Aufträge erzeugt, die alle in einen Batch passen.

7.2.2 Dynamischer Lagerplatzvergabealgorithmus

Die Vorgehensweise zur Erzeugung der Testdaten für den dynamischen Lagerplatzvergabealgorithmus ist in Abbildung 37 dargestellt. Die Aufträge der ersten Periode

(z.B. Monat) werden wie bei der Erzeugung der Testdaten für den DEPSO-Algorithmus erstellt. Sie werden gleichmäßig auf die Unterperioden (z.B. Tag) verteilt. Sollten Aufträge übrig bleiben, werden diese zufallsbasiert auf die Unterperioden verteilt. Anschließend werden die Anzahl Positionen je Artikel mithilfe der Artikelnachfrage für die Perioden erzeugt. Daraus ergibt sich die Anzahl Positionen über alle Artikel je Periode (A_t^{gesPos}). Anschließend werden Aufträge eröffnet und wie in Periode 1, die Anzahl Positionen der Aufträge und die Mengen je Position zufallsbasiert bestimmt. Es werden so lange Aufträge eröffnet, bis A_t^{gesPos} Positionen erzeugt wurden.

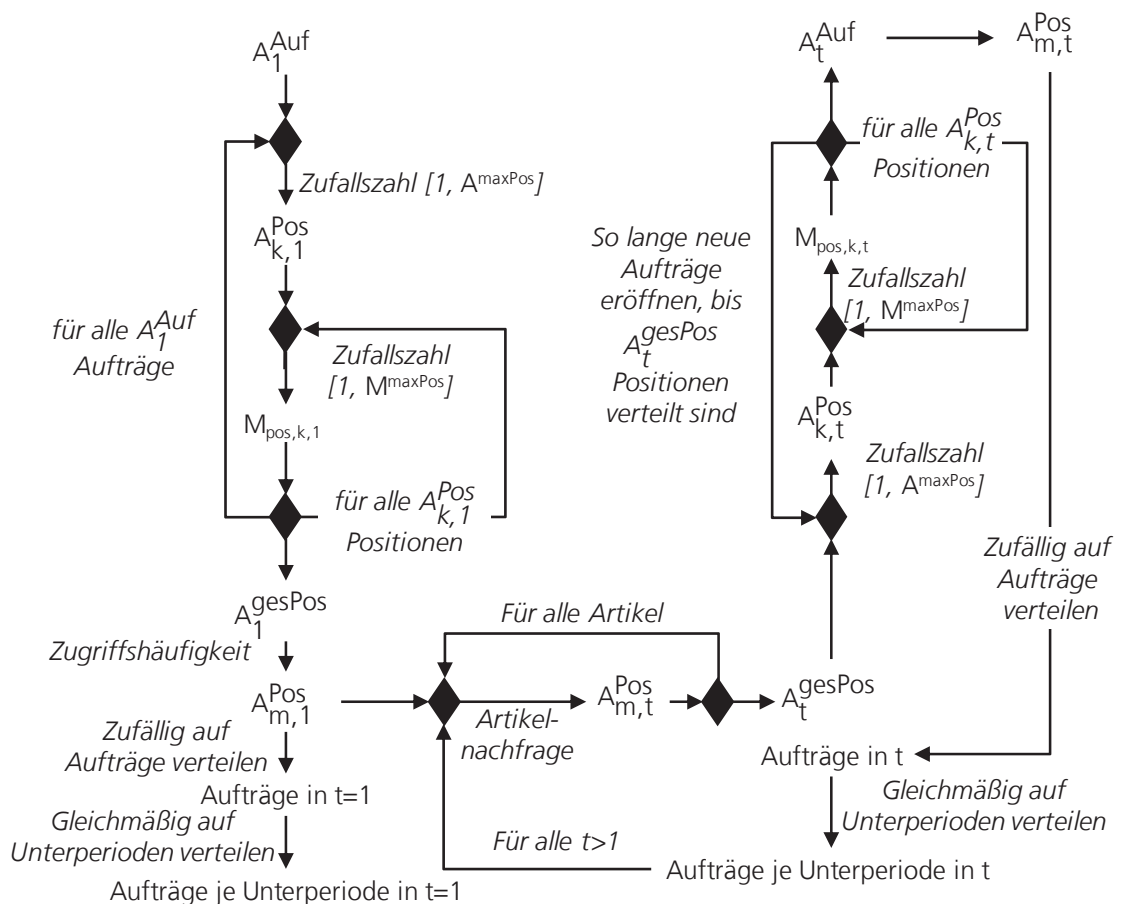


Abbildung 37: Erzeugung Auftragsdaten dynamischer Lagerplatzvergabealgorithmus

Anschließend werden die Artikel wie in Periode 1 auf die Aufträge verteilt.

Artikelnachfrage

Die Anzahl Positionen eines Artikels in der Periode t wurde mithilfe der Artikelnachfrage über den gesamten Untersuchungszeitraum mittels der folgenden Funktion modelliert:

$$A_{m,t}^{\text{Pos}} = (\text{Runden}) \left(A_{m,1}^{\text{Pos}} + T_m^{\text{Trend}} \cdot (t - 1) + S_m^{\text{Saison}} \cdot \cos\left(\frac{2 \cdot \pi \cdot (t - 1)}{L}\right) + Ir_{m,t} \right) \quad \forall t \in U \quad (7.3)$$

Es wurde eine maximal zulässige Schwankung (M) definiert, die dafür sorgt, dass die Gesamtanzahl Positionen nicht unrealistisch hohen Schwankungen unterliegt. Die maximale Anzahl Positionen je Periode A_t^{gesPos} darf maximal M -mal größer als die minimale Anzahl Positionen je Periode sein. Es muss daher gelten:

$$\max \{A_t^{\text{gesPos}}\} \leq M \cdot \min \{A_t^{\text{gesPos}}\} \quad \forall t \in U \quad (7.4)$$

Zur Bestimmung von T_m^{Trend} , S_m^{Saison} und $Ir_{m,t}$ wurde wie folgt vorgegangen: Zu Beginn sind (artikelübergreifend) Trend- (Tf), Saison- (Sf) und ein irregulärer Faktor (Irf) festzulegen. Für einen Artikel m gilt:

$$T_m^{\text{Trend}} = Tf \cdot A_{m,1}^{\text{Pos}} \cdot \text{rand} \quad (7.5)$$

$$S_m^{\text{Saison}} = Sf \cdot A_{m,1}^{\text{Pos}} \cdot \text{rand} \quad (7.6)$$

Die Anzahl Positionen pro Periode ohne irreguläre Komponente ($A_{m,t}^{\text{Posolr}}$) wird wie folgt berechnet:

$$A_{m,t}^{\text{Posolr}} = A_{m,1}^{\text{Pos}} + T_m^{\text{Trend}} \cdot (t - 1) + S_m^{\text{Saison}} \cdot \cos\left(\frac{2 \cdot \pi \cdot (t - 1)}{L}\right) \quad \forall t \in U \quad (7.7)$$

Die irreguläre Komponente ergibt sich zu:

$$|Ir_{m,t}| = Irf \cdot A_{m,t}^{\text{Posolr}} \cdot \text{rand} \quad \forall t \in U \quad (7.8)$$

Damit die irreguläre Komponente auch negative Werte annehmen kann, wird wie folgt vorgegangen:

$$Ir_{m,t} = |Ir_{m,t}| \cdot (-1), \text{ falls } \text{rand} \leq 0,5 (= |Ir_{m,t}|, \text{ sonst}) \quad \forall t \in U \quad (7.9)$$

Um sicherzustellen, dass A-Artikel aus Periode 1 auch zu C-Artikeln werden können und umgekehrt, also eine Situation mit hoher Dynamik entsteht, wurde folgendes Vorgehen gewählt:

Ablauf Festlegung Artikelnachfrage (IX.)Schritt 1

Erstelle eine Liste mit Anzahl Positionen je Artikelnummer aus Periode 1 mit allen Artikeln. Sortiere diese absteigend.

Schritt 2

Wähle den obersten Artikel als Basisartikel BA.

Schritt 3

Setze die Wahrscheinlichkeitsschranke S^{wahr} auf 1.

Schritt 4

Berechne T_{BA}^{Trend} , S_{BA}^{Saison} und $lr_{BA,t}$ mithilfe von (7.5) – (7.9).

Schritt 5

Falls $\text{rand} < S^{\text{wahr}}$, drehe das Vorzeichen von T_{BA}^{Trend} um:

$$T_{BA}^{\text{Trend}} = T_{BA}^{\text{Trend}} \cdot (-1) \quad (7.10)$$

Berechne $A_{BA,t}^{\text{Pos}}$ mithilfe von (7.3).

Schritt 6

Bestimme zufällig einen Artikel aus der Liste als Gegenartikel GA.

Schritt 7

Berechne T_{GA}^{Trend} , S_{GA}^{Saison} und $lr_{GA,t}$ wie folgt:

$$T_{GA}^{\text{Trend}} = Tf \cdot A_{BA,1}^{\text{Pos}} \cdot \text{rand} \quad (7.11)$$

Falls $T_{GA}^{\text{Trend}} > 0$, drehe das Vorzeichen von T_{GA}^{Trend} um:

$$T_{GA}^{\text{Trend}} = Tf \cdot A_{BA,1}^{\text{Pos}} \cdot \text{rand} \cdot (-1) \quad (7.12)$$

$$S_{GA}^{\text{Saison}} = Sf \cdot A_{BA,1}^{\text{Pos}} \cdot \text{rand} \quad (7.13)$$

$$A_{GA,t}^{\text{Posolr}} = A_{GA,1}^{\text{Pos}} + T_{GA}^{\text{Trend}} \cdot (t - 1) + S_{GA}^{\text{Saison}} \cdot \cos\left(\frac{2 \cdot \pi \cdot (t - 1)}{L}\right) \quad \forall t \in U \quad (7.14)$$

$$|lr_{GA,t}| = lrf \cdot A_{GA,t}^{\text{Posolr}} \cdot \text{rand} \quad \forall t \in U \quad (7.15)$$

$$lr_{GA,t} = |lr_{GA,t}| \cdot (-1), \text{ falls } \text{rand} \leq 0,5 (= |lr_{GA,t}|, \text{ sonst}) \quad \forall t \in U \quad (7.16)$$

Berechne $A_{GA,t}^{\text{Pos}}$ mithilfe von (7.3).

Schritt 8

Prüfe ob Bedingung (7.4) eingehalten wird. Falls ja, akzeptiere die Artikelzeitreihen von BA ($A_{BA,t}^{\text{Pos}}$) und GA ($A_{GA,t}^{\text{Pos}}$). Falls nicht, setze für beide eine konstante Nachfrage über alle Perioden in Höhe ihrer Anzahl Positionen in der ersten Periode ($A_{BA,1}^{\text{Pos}}$ bzw. $A_{GA,1}^{\text{Pos}}$).

Schritt 9

Reduziere S^{wahr} :

$$S^{\text{wahr}} = S^{\text{wahr}} \cdot 0,995 \quad (7.17)$$

Schritt 10

Entferne die beiden Artikel aus der Liste mit noch ungeplanten Artikeln. Wähle den nächsten noch ungeplanten Artikel als Basisartikel BA und fahre mit Schritt 4 fort. Breche das Verfahren ab, wenn alle Artikel geplant wurden.

Wenn der erste Artikel nicht zwangsläufig einen sinkenden Nachfrageverlauf haben soll, kann der Startwert in Schritt 3 auf einen Wert kleiner eins gesetzt werden. Dieses Vorgehen stellt sicher, dass einige der am häufigsten kommissionierten Artikel aus Periode 1, mit hoher Wahrscheinlichkeit stark abnehmende Nachfrageverläufe aufweisen. Im Gegenzug weisen bislang weniger häufig kommissionierte Artikel steigende Nachfrageverläufe auf. Durch die im Verlauf abnehmende Schranke, sinkt die Wahrscheinlichkeit, dass die Nachfrage der Artikel sinkt. Zu einem bestimmten Zeitpunkt ist die Wahrscheinlichkeit für eine steigende Artikelnachfrage höher. Insgesamt führt dies zu einer hohen Dynamik, da Langsamdreher zu Schnelldrehern und Schnelldreher zu Langsamdrehern werden können. Diese Wechsel haben den größten Einfluss auf ein Kommissioniersystem. Kleine Veränderungen, etwa wenn ein Artikel aus der Klasse C in die Klasse B wechselt, haben einen wesentlich geringeren Einfluss auf die Effizienz des Kommissioniersystems.

Es wurden Testdaten für zwei Szenarien erstellt. Tabelle 2 zeigt die verwendeten Problemkonstanten und die Problemparameter der beiden Szenarien.

Problemkonstanten		Symbol	Wert
Anzahl Aufträge Periode $t = 0$		A_0^{Auf}	5.000
Maximale Anzahl Positionen je Auftrag		A^{maxPos}	2
Maximale Menge je Position		M^{maxPos}	6
Saisonzykluszeit		L	12
Zugriffshäufigkeitsparameter		ZH	0,6
Anzahl Artikel		A^{Art}	6.000
Anzahl Unterperioden		A^{Unterp}	20
Untersuchungszeitraum		U	9 (+12)
Maximale monatliche Schwankung		M	2
Irregulärer Faktor		Irf	0,025
Problemparameter	Symbol	Szenario 1	Szenario 2
Trendfaktor	Tf	0,300	0,150
Saisonfaktor	Sf	0,150	0,075

Tabelle 2: Problemkonstanten und -parameter dynamische Lagerplatzvergabe

Untersuchungszeitraum sind in beiden Fällen 21 Perioden, wobei das Verfahren mithilfe der letzten neun Perioden validiert wurde. Die ersten zwölf Perioden wurden zur Bestimmung der Prognosekoeffizienten genutzt. Die klassenbasierte Lagerplatzvergabe wurde mithilfe der Daten aus Periode 1 erzeugt. Innerhalb der Klassen wurden die Artikel den Lagerplätzen zufallsbasiert zugewiesen. Das erste Szenario entspricht einer sehr dynamischen Umwelt. Dafür wurden der Trendfaktor auf 0,3 und der Saisonfaktor auf 0,15 gesetzt. Das zweite Szenario steht für eine weniger dynamische Situation. Hier wurden die beiden Parameter auf 0,15 bzw. 0,075 halbiert.

7.3 Vortest diskrete evolutionäre Partikelschwarmoptimierung

Zur Bestimmung der geeigneten Parameterkonfiguration der Algorithmen wurden umfangreiche Vortests durchgeführt. Dabei wurde ein in der Literatur übliches Vorgehen gewählt (vgl. hierzu Koch 2014, S. 122-125). Bei der Festlegung der Spannbreite der einzustellenden Parameter wurde sich an den von Tian et al. (2013, S. 488-495) gewählten Werten orientiert. Die in den Vortests berücksichtigten Spannbreiten der Parameter sind in Tabelle 3 dargestellt. Die Vortests wurden an den in Tabelle 3 dargestellten vier exemplarischen Problemklassen durchgeführt. Bei der Auswahl der Problemklassen wurde darauf geachtet, dass das gesamte in Kapitel

7.2.1 beschriebene Problemfeld möglichst gut repräsentiert wird. Je Problemklasse wurden fünf Instanzen generiert. Diese wurden in den numerischen Experimenten nicht mehr verwendet. Es wurden 243 Parameterkombinationen an den vier exemplarischen Problemklassen mit jeweils fünf Instanzen getestet.

Parameter	Symbol	Wert	
Anzahl Partikel	A^{Partikel}	{5, 10, 15}	
Anzahl Iterationen	It^{max}	{200, 350, 500}	
Schwellwert G_{best}	$S^{G_{\text{best}}}$	{0,25, 0,5, 0,75}	
Maximale Iterationen lokale Suche	It^{maxLS}	{50, 75, 100}	
Maximale Stagnationsschranke	S^{maxStag}	{20, 40, 60}	
Problem	Aufträge	Positionen	Menge
1	50	10	10
2	100	2	6
3	150	6	2
4	200	2	2

Tabelle 3: Problemklassen Vortest und Parameterwerte

Zur Bestimmung der besten Parameterkonfiguration wurde für jede Instanz der beste Zielfunktionswert aus allen Parameterkonfigurationen bestimmt. Für jede Konfiguration wurde die prozentuale Abweichung zu diesem besten Wert berechnet. Für die Validierung des DEPSO-Algorithmus wurde die Parameterkonfiguration gewählt, die durchschnittlich die niedrigste prozentuale Abweichung zu den besten Zielfunktionswerten ergab:

- Anzahl Partikel: 5
- Anzahl Iterationen: 500
- Schwellwert G_{best} : 0,5
- Maximale Anzahl Iterationen des lokalen Suchverfahrens: 100
- Maximale Stagnationsschranke: 20

Tabelle 4 zeigt einen Ausschnitt aus den Ergebnissen des Vortests.

Parameterkonfiguration					Durchschnittliche Abweichung vom besten Wert je Problemklasse
S	It^{\max}	$S^{G_{\text{best}}}$	$It^{\max LS}$	$S_{\text{max}}^{\text{Stag}}$	$\bar{\sigma}$ [%]
5	500	0,50	100	20	3,73
15	500	0,25	75	20	3,92
15	500	0,50	100	60	3,96
10	500	0,50	100	60	4,03
10	500	0,50	100	20	4,19
15	500	0,75	75	40	4,22
...

Tabelle 4: Ergebnisse Vortest

Die Verfahren wurden in Matlab R2017b implementiert und auf einem Computer mit einem 2,5 GHz Prozessor und 16 GB RAM Arbeitsspeicher durchgeführt.

7.4 Ergebnisse diskrete evolutionäre Partikelschwarmoptimierung

Der entwickelte DEPSO-Algorithmus wurde mithilfe dreier Vergleichsverfahren validiert. Bei dem ersten Verfahren findet keine Bündelung der Aufträge (Single Order Picking, SOP) statt. Das zweite Verfahren nutzt das First-Come-First-Served-Verfahren (FCFS): Aufträge werden in der Reihenfolge ihres Eintreffens solange dem offenen Batch zugewiesen, bis die Kapazitätsgrenze erreicht ist und ein neuer Batch eröffnet wird. Als drittes Vergleichsverfahren wurde der Savings-Algorithmus von Clarke & Wright, wie er in (III.) in Kapitel 6.2.2 beschrieben wurde, ausgewählt. Alle drei Verfahren werden mit der S-Shape-Routingstrategie kombiniert.

Bei allen in diesem Kapitel vorgestellten Optimierungsergebnissen der DEPSO handelt es sich um die Mittelwerte der 40 Instanzen einer Problemklasse. Abbildung 38 zeigt die erreichte prozentuale Wegstreckenreduktion durch den Einsatz der DEPSO im Vergleich zu den anderen drei Verfahren.

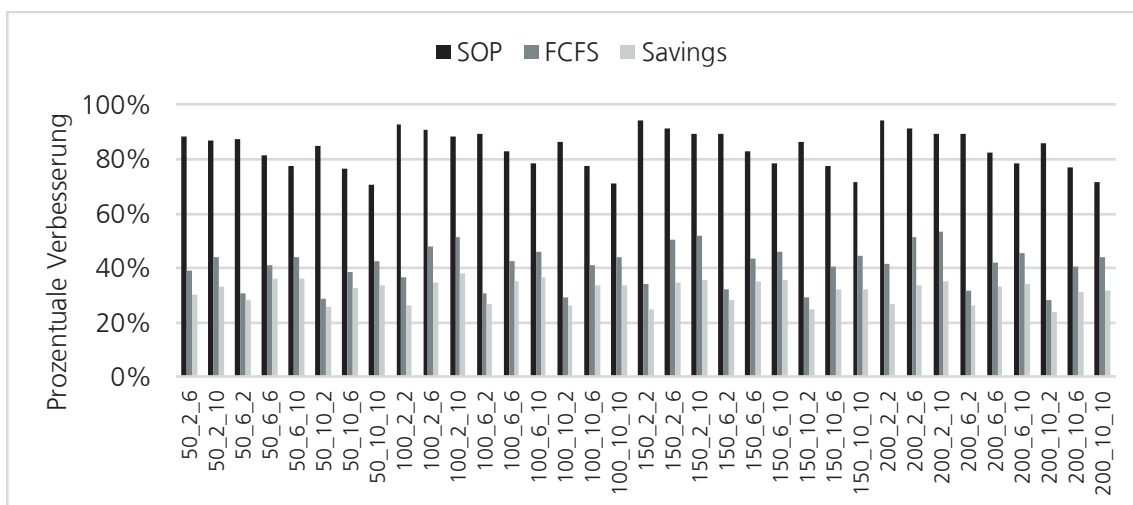


Abbildung 38: Verfahrensvergleich numerische Experimente DEPSO

Das Diagramm zeigt die hohe Leistungsfähigkeit des DEPSO-Algorithmus in allen Problemklassen und gegen jedes der drei Vergleichsverfahren. Gegenüber SOP verkürzt der DEPSO-Algorithmus den Kommissionierweg um mindestens 70,60 % und gegenüber dem FCFS um mindestens 28,26 %. Abbildung 39 zeigt die Optimierungsergebnisse im Vergleich zum Savings-Verfahren. Der DEPSO-Algorithmus verkürzt den Kommissionierweg um mindestens 24,02 %. Das Verfahren erzielt eine Wegstreckenreduktion von bis zu 38,24 %. Durchschnittlich erzielt der DEPSO-Algorithmus eine Verbesserung gegenüber dem Savings-Verfahren um 31,64 %.

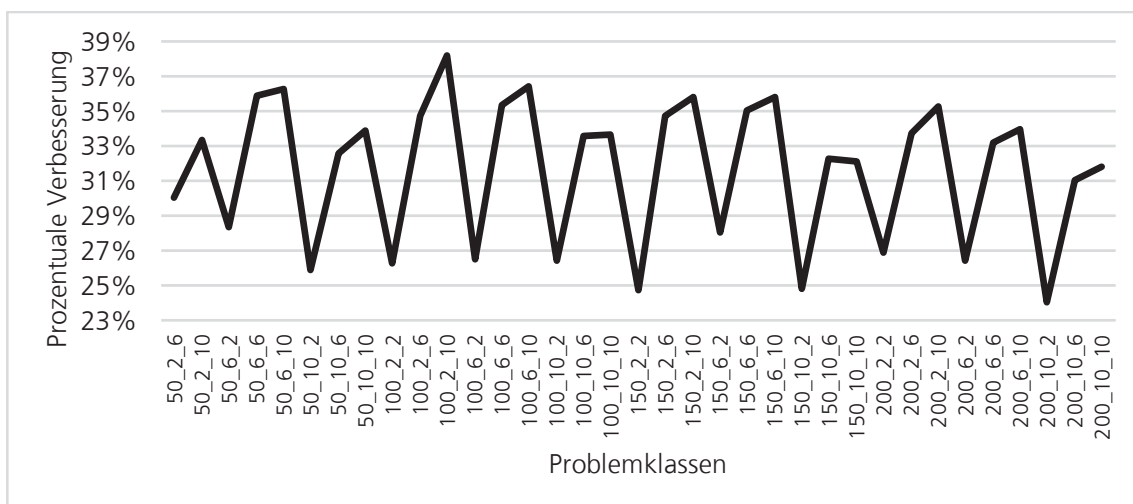


Abbildung 39: Vergleich DEPSO-Savings (numerische Experimente)

Tabelle 5 fasst die erzielten Einsparungen problemklassenübergreifend zusammen.

	SOP	FCFS	Savings
Min.	70,60 %	28,26 %	24,02 %
Ø	83,78 %	40,80 %	31,64 %
Max.	94,26 %	53,33 %	38,24 %

Tabelle 5: Übersicht Wegstreckenreduktionen DEPSO (numerische Experimente)

Im besten Fall verkürzt der DEPSO-Algorithmus den Kommissionierweg um bis zu 94,26 %. Gegenüber dem SOP erzielt die DEPSO eine Verbesserung um durchschnittlich 83,78 % und gegenüber dem FCFS um 40,80 %. Abbildung 40 zeigt den Einfluss der Anzahl Positionen und der Menge je Position auf das Optimierungsergebnis.

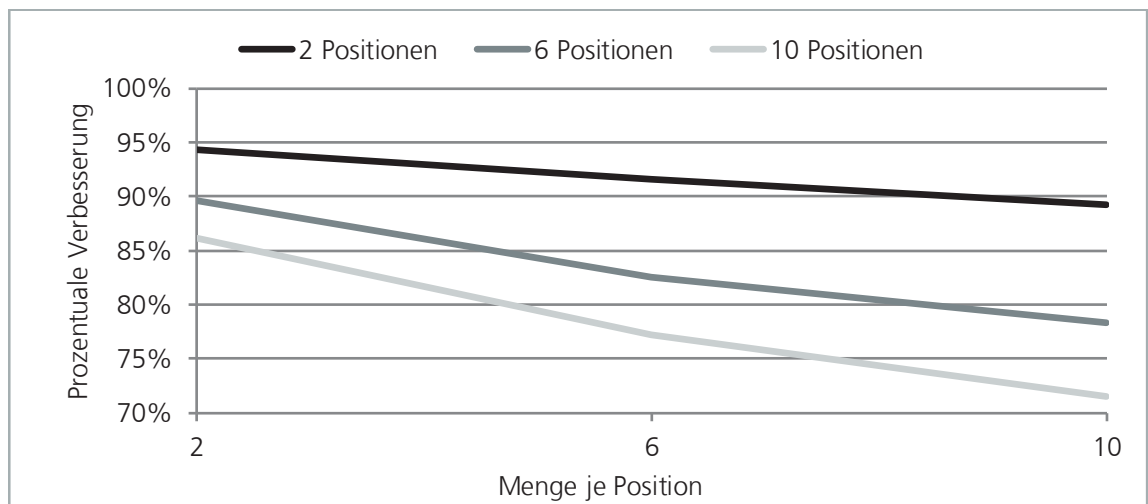


Abbildung 40: Einfluss Anzahl Positionen und Menge je Position auf das Optimierungsergebnis am Beispiel von 200 Aufträgen und SOP (numerische Experimente)

Als Vergleichsverfahren wurde hier SOP gewählt, um das Ergebnis nicht durch die Stärken und Schwächen der anderen Auftragsbildungsverfahren zu verzerren. Abbildung 40 liegen die Ergebnisse der Tests mit 200 Aufträgen zugrunde. Eine steigende Anzahl Positionen führt zu sinkenden Optimierungsergebnissen. Gleiches gilt auch für die Menge je Position. Dieser Zusammenhang erklärt auch das regelmäßige Muster in Abbildung 39. Allerdings überrascht hier zunächst die Verbesserung des Optimierungsergebnisses bei einer Zunahme der Menge je Position. Hierbei ist zu beachten, dass in Abbildung 39 ein Vergleich zwischen DEPSO und Savings-Verfahren und in Abbildung 40 ein Vergleich zwischen DEPSO und SOP stattfindet.

Wie aus Abbildung 40 deutlich wird, nimmt das erzielbare Optimierungsergebnis bei einer Zunahme der Menge sowohl bei dem Savings-Verfahren als auch bei der DEPSO ab. Dieser Effekt ist bei dem Savings-Verfahren jedoch stärker ausgeprägt als dies bei der DEPSO der Fall ist. So nimmt das Optimierungsergebnis des Savings-Verfahrens bei 200 Aufträgen und zehn Positionen um 22,95 % von 81,27 % (Menge = 2) auf 58,32 % (Menge = 10) ab. Bei der DEPSO ist im selben Fall lediglich eine Abnahme von 14,54 % zu verzeichnen. Deshalb nimmt das Optimierungsergebnis der DEPSO im Vergleich zum Savings-Verfahren bei einer Zunahme der Menge zu und nicht, wie eigentlich zu erwarten wäre, ab.

Insgesamt eignet sich das Verfahren vor allem in Anwendungsfällen mit wenigen Positionen je Auftrag und Mengen je Position. Da die Auftragsgrößen, wie in der Einleitung beschrieben, immer kleiner werden, eignet sich das Verfahren zur Lösung der zukünftigen Herausforderungen. Das Verfahren liefert jedoch auch in Situationen mit großen Aufträgen sehr gute Optimierungsergebnisse. Neben dem Optimierungsergebnis spielen die Rechenzeiten des DEPSO-Algorithmus eine wichtige Rolle. Abbildung 41 zeigt die durchschnittlichen Rechenzeiten des DEPSO-Algorithmus für die Problemklassen. Zur Lösung der Problemklasse 50_2_10 benötigt der DEPSO-Algorithmus lediglich 21,28 Sekunden. Für die Problemklasse 200_10_2 liefert das Verfahren nach 226,18 Sekunden ein Ergebnis. Damit werden die beschriebenen Optimierungsergebnisse innerhalb sehr kurzer Rechenzeiten erzielt.

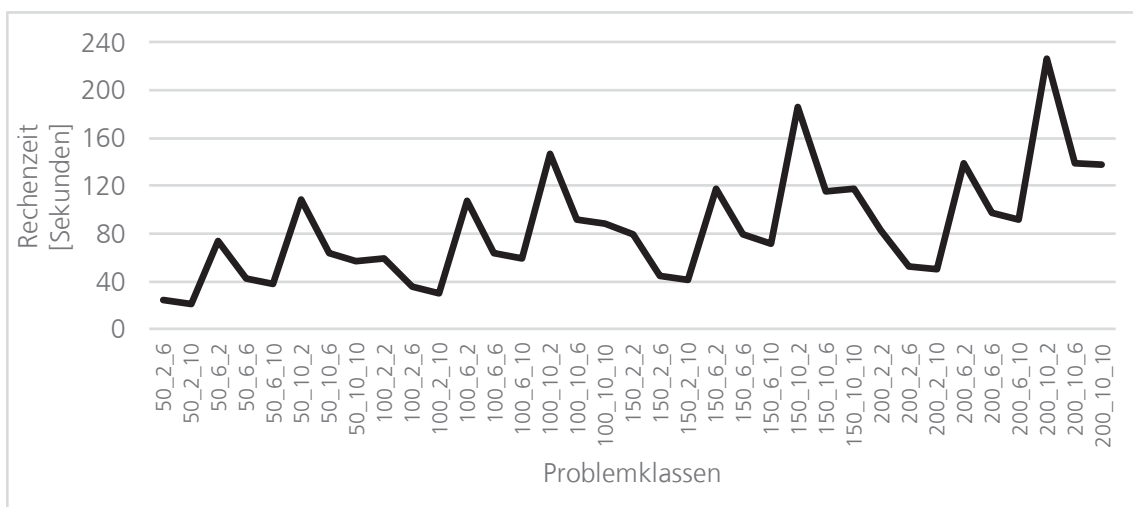


Abbildung 41: Rechenzeit DEPSO (numerische Experimente)

Die detaillierten Ergebnisse finden sich in Tabelle 7 im Anhang.

7.5 Ergebnisse dynamischer Lagerplatzvergabealgorithmus

In beiden Szenarien wurden im Zuge des dynamischen Lagerplatzvergabealgorithmus entsprechend dem Vorgehen von Kofler et al. (2015) 50 Umlagerungen geprüft. Zur Prognose der Artikelzeitreihen wurde das Verfahren von Holt-Winters für den additiven Fall gewählt. Die Prognosekoeffizienten und -werte wurden mithilfe von (3.13) – (3.16) und die Startwerte mit (3.21) – (3.23) berechnet. Der Prognosezeitraum betrug 12 Perioden. Bei der Wahl der Glättungsparameter wurde den Empfehlungen von Silver et al. (2017, S. 104) gefolgt und $\alpha = 0,19$, $\beta = 0,053$ und $\gamma = 0,1$ gewählt. Die durchschnittliche Gehgeschwindigkeit des Kommissionierers wurde auf 1 m/s festgelegt. Die physische Umlagerung eines Artikels benötigt drei Minuten, die administrative Umlagerung eine Minute. Ein Artikel muss mindestens eine Vergangenheitsperiode und die aktuelle Periode (Schwellwert $o = 2$) sowie eine Zukunftsperiode (Schwellwert $u = 1$) falsch gelagert sein und in derselben Zielklasse verbleiben, um als falsch gelagert klassifiziert und weiter untersucht zu werden. Abbildung 42 zeigt das Optimierungsergebnis für das 1. Szenario.

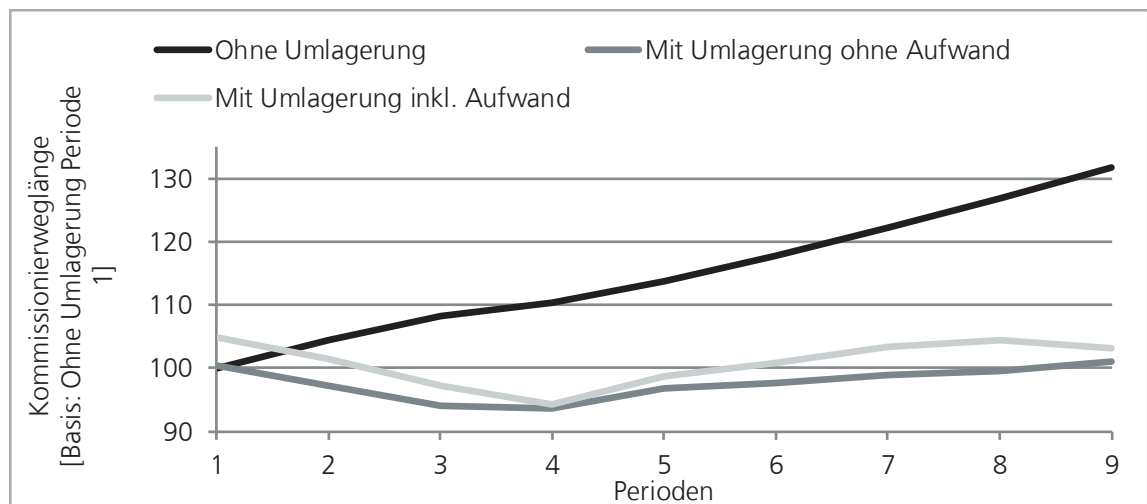


Abbildung 42: Optimierungsergebnis Szenario 1 dynamische Lagerplatzvergabe

In schwarz ist die Kommissionierweglänge abgebildet, die mithilfe des DEPSO-Algorithmus ohne Umlagerungen resultiert. Die dunkelgraue Linie steht für die Kommissionierweglänge, die sich mithilfe des DEPSO-Algorithmus und Umlagerungen ergibt. In hellgrau wird der Umlagerungsaufwand auf die Kommissionierweglänge addiert. Zur besseren Vergleichbarkeit wurden die drei Kommissionierweglängen auf die Kommissionierweglänge ohne Umlagerung in der ersten Periode (Basis = 100) normiert. Die Umlagerungen führen in allen Perioden mit Ausnahme der Periode 1, in der den Umlagerungen noch keine Wegstreckenreduktion gegenübersteht, zu erheblich kürzeren Kommissionierwegen. In Abbildung 43 sind die Ersparnisse und die Wegstreckenreduktionen den Umlagerungsaufwänden gegenübergestellt. Die Ersparnisse wurden inklusive des Umlagerungsaufwands und bezogen auf die Kommissionierweglänge ohne Umlagerungen berechnet. Der Umlagerungsaufwand bezieht sich ebenfalls auf die Kommissionierweglänge, die ohne Umlagerungen resultiert. Die Ersparnisse steigen im Zeitverlauf. Je länger der dynamische Lagerplatzvergabealgorithmus eingesetzt wird, desto höher sind die zu erwartenden Ersparnisse. Im gesamten Untersuchungszeitraum liefert das Verfahren eine Wegstreckenreduktion in Höhe von 15,02 % und eine Ersparnis in Höhe von 12,23 %.

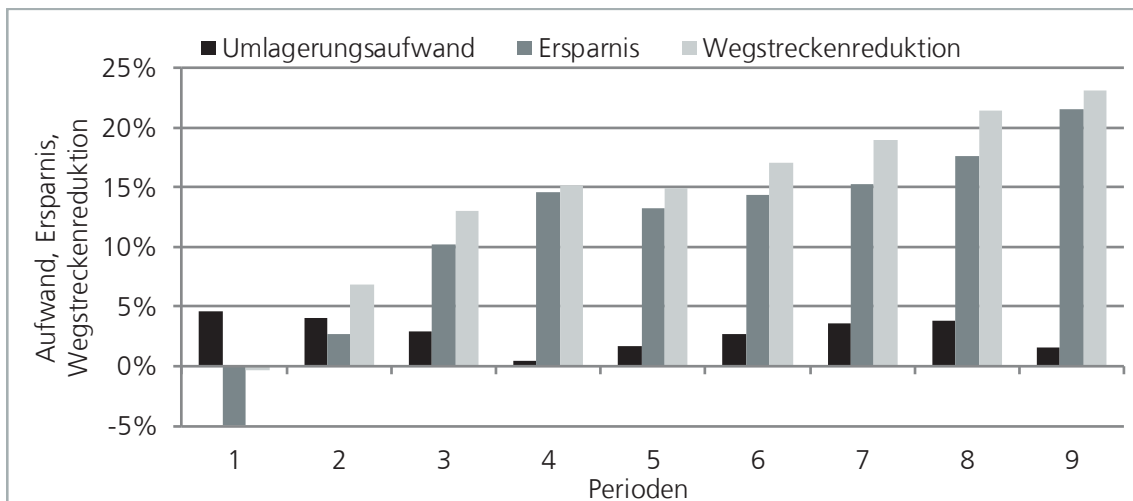


Abbildung 43: Umlagerungsaufwand, Ersparnis und Wegstreckenreduktion Szenario 1 dynamische Lagerplatzvergabe

Den hohen Ersparnissen stehen im Untersuchungszeitraum also lediglich Umlagerungsaufwände in Höhe von 2,79 % gegenüber. Dieses positive Aufwand-Nutzen-

Verhältnis resultiert aus einer sehr begrenzten Anzahl umgelagerter Artikel (Abbildung 44).

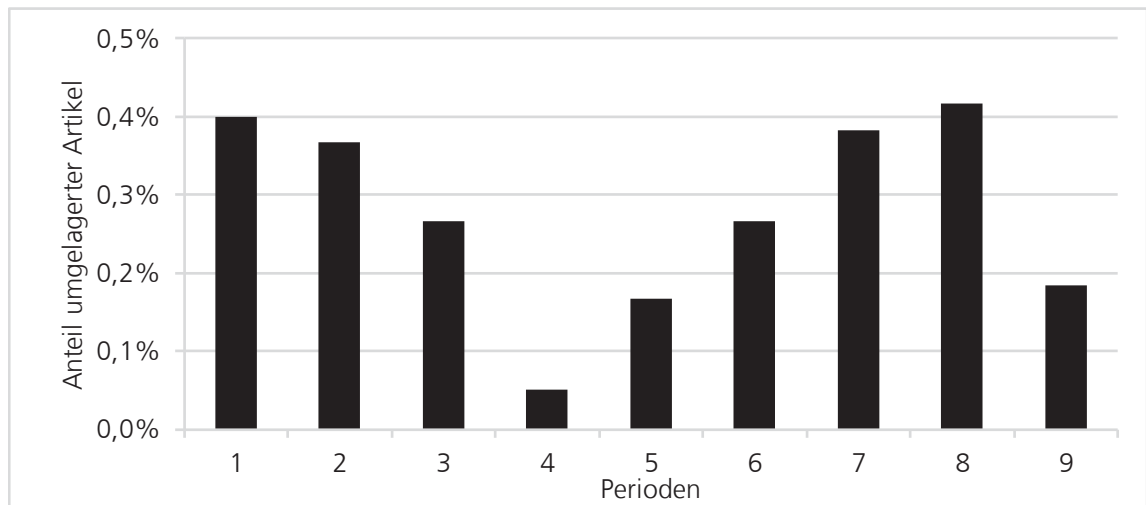


Abbildung 44: Anteil umgelagerter Artikel Szenario 1 dynamische Lagerplatzvergabe

In Periode 8 wurden mit 0,42 % der Artikel des Artikelsortiments die meisten Artikel im Untersuchungszeitraum umgelagert. Das entspricht bei 6.000 Sortimentsartikeln 25 umgelagerten Artikeln. Im Mittel wurden 0,28 % der Artikel des Sortiments umgelagert. Der Grund für die geringe Anzahl Umlagerungen ist, dass das Verfahren zunächst Artikel prüft, die starken Veränderungen unterliegen. Daraus kann der Schluss gezogen werden, dass Unternehmen sich auf die Umlagerung von Artikeln konzentrieren sollten, die zukünftig vom Langsamdreher (C-, bzw. B-Artikel) zum Schnelldreher (A-Artikel) werden. Die Ergebnisse des 1. Szenarios finden sich in den Tabellen 8 – 12 im Anhang.

Abbildung 45 zeigt das Optimierungsergebnis des zweiten Szenarios.

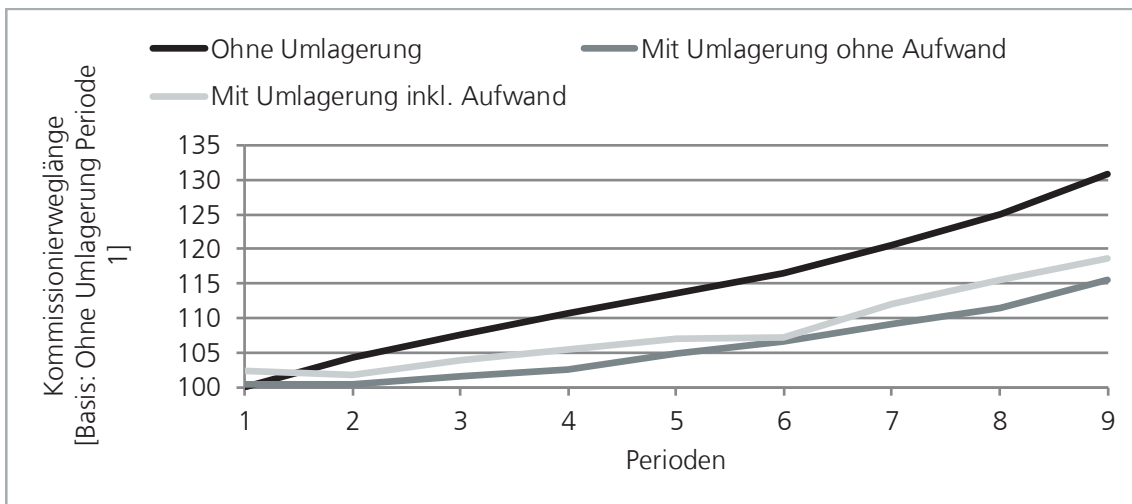


Abbildung 45: Optimierungsergebnis Szenario 2 dynamische Lagerplatzvergabe

Auch in diesem Szenario konnte der dynamische Lagerplatzvergabealgorithmus signifikante Verbesserungen im Vergleich zu dem Fall ohne Umlagerungen erzielen. Unter Berücksichtigung des Umlagerungsaufwandes entstand eine Ersparnis in Höhe von 5,37 %. Damit ist das Ergebnis niedriger als im ersten Szenario mit höherer Dynamik. Dennoch konnte in der letzten Periode 9 eine Ersparnis in Höhe von 9,35 % erzielt werden (Abbildung 46).

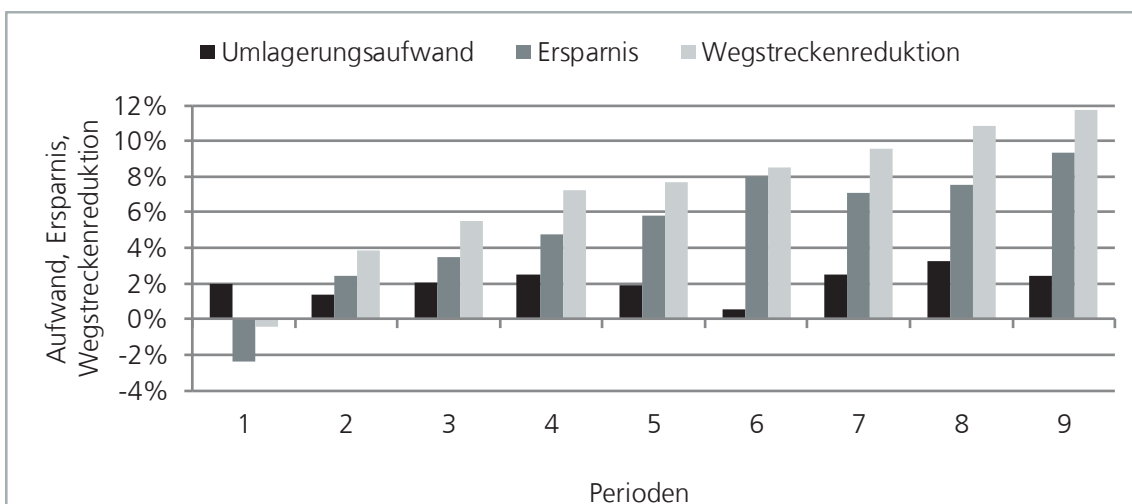


Abbildung 46: Umlagerungsaufwand, Ersparnis und Wegstreckenreduktion Szenario 2 dynamische Lagerplatzvergabe

Den Einsparungen stehen wiederum sehr niedrige Aufwände gegenüber. Im gesamten Untersuchungszeitraum betrug der Umlagerungsaufwand lediglich 2,08 % der Kommissionierweglänge, die in dieser Zeit ohne Umlagerung entstanden wäre. Der

höchste Aufwand war in Periode 8 mit 3,25 % notwendig. Umgelagert wurden zwischen 3 (0,05 %) und 20 Artikel (0,33 %) (Abbildung 47).

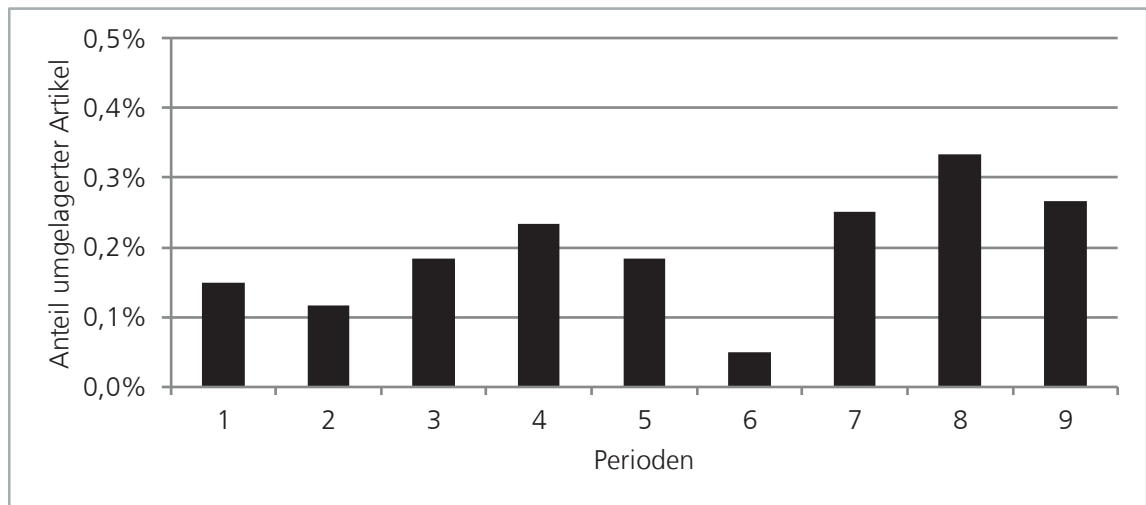


Abbildung 47: Anteil umgelagerter Artikel Szenario 2 dynamische Lagerplatzvergabe

Die Ergebnisse des 2. Szenarios finden sich in den Tabellen 13 – 17 im Anhang.

7.6 Bewertung der Ergebnisse

Der DEPSO-Algorithmus liefert unabhängig von der zugrunde liegenden Problemklasse durchgehend sehr gute Optimierungsergebnisse. Durchschnittlich kann eine Verbesserung in Höhe von 31,64 % gegenüber dem Savings-Algorithmus erzielt werden. Auch die Einsetzbarkeit des Verfahrens für Multi-Block-Layouts kann mithilfe der Ergebnisse gezeigt werden. Die Praxistauglichkeit des Verfahrens zeigt sich nicht zuletzt in den sehr niedrigen Rechenzeiten. Durch Berücksichtigung der Interdependenz der Planungsprobleme im dynamischen Lagerplatzvergabealgorithmus können im dynamischen Szenario erhebliche Einsparungen von 12,23 % erzielt werden. Der Algorithmus liefert für das dynamischere Szenario 1 bessere Optimierungsergebnisse als für das weniger dynamische Szenario 2. Das zeigt, dass der Algorithmus sehr gut zur Beherrschung der Dynamik geeignet ist. Die sehr guten Optimierungsergebnisse können mithilfe einer kleinen Anzahl an Umlagerungen realisiert werden. Es sind keine umfangreichen Lagerreorganisationen notwendig, um ein Lager auch bei einer stark schwankenden Nachfrage effizient betreiben zu können.

Durch Berücksichtigung der Umlagerungsaufwände wird die Praxistauglichkeit des Verfahrens sichergestellt. Unternehmen schrecken häufig vor Umlagerungen zurück, weil sie befürchten, dass aus den Aufwänden keine entsprechenden Einsparungen resultieren. Diese Befürchtungen können durch die Ergebnisse der numerischen Experimente ausgeräumt werden, da durchgehend sehr niedrige Umlagerungsaufwände zur Erzielung der Wegstreckenreduktion notwendig sind.

8 Fallbeispiel

Beide Verfahren wurden anhand von Kommissionieraufträgen eines Unternehmens der Elektronikbranche getestet. In Kapitel 8.1 wird zunächst der Fall beschrieben, bevor dann in Kapitel 8.2 die Ergebnisse der Untersuchung dargestellt werden. Abschließend werden diese in Kapitel 8.3 bewertet.

8.1 Fallbeschreibung

Das Unternehmen kommissioniert Bauteile aus einem Kommissionierlager für die Endmontage der Produkte. Der Untersuchung liegt das in Abbildung 48 dargestellte Kommissionierlager zugrunde. Das Beispiel wurde anonymisiert, um die Vertraulichkeit der Unternehmensinformationen sicherzustellen. Die Verfahren werden mit den realen Kommissionieraufträgen des Unternehmens getestet.

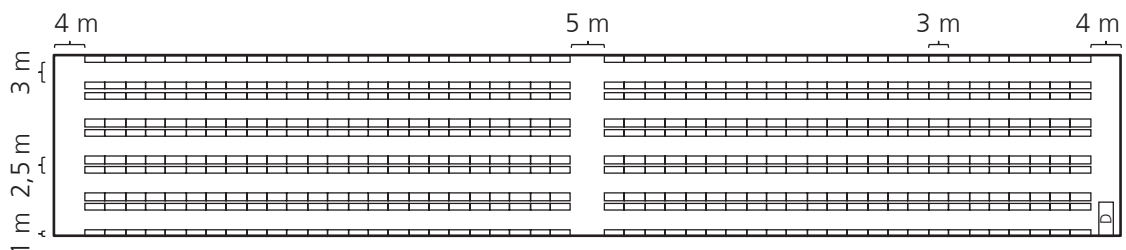


Abbildung 48: Layout Fallbeispiel

In der Mitte des Lagers befindet sich ein Quergang mit 5 m Breite. Der linke und rechte Quergang sind jeweils 4 m breit. Die fünf Kommissioniergänge sind jeweils 3 m breit. Ein Regalelement ist 1 m breit und 3 m lang. Zwei aneinander grenzende Regalreihen stehen 0,5 m voneinander entfernt. Das Depot befindet sich im rechten Quergang gegenüber dem untersten Kommissioniergang. Eine Regalreihe besteht aus 24 Regalelementen. Das Lager hat in Summe 480 Regalelemente. Ein Regalelement hat drei Lagerorte. Jeder Lagerort besteht aus zehn Lagerplätzen. Das Gesamtsystem umfasst 1.440 Lagerorte und 14.400 Lagerplätze. Das Sortiment besteht aus 11.000 Artikeln. Die Lagerplatzvergabe erfolgt klassenbasiert. Die relativen Klassengrößen sind: A = 5 %, B = 15 % und C = 80 %. Die Lagerplätze werden den Klassen

basierend auf ihrer Nähe zum Depot (Radialanordnung) zugewiesen. Die Kapazität des Kommissioniergeräts wird nicht in Gewicht, sondern in der maximalen Anzahl Positionen beschrieben. Das Gerät fasst maximal 150 Positionen. Der DEPSO-Algorithmus wurde hierfür entsprechend angepasst, so dass ein Batch nicht mehr als 150 Positionen umfassen kann. Die Gehgeschwindigkeit des Kommissionierers beträgt 1 m/s. Eine physische Umlagerung dauert drei, eine administrative Umlagerung eine Minute. Die Größe der Aufträge ist sehr unterschiedlich. Der kleinste Auftrag umfasst eine und der größte 140 Positionen. Untersucht wird ein Zeitraum von fünf Wochen. Eine Woche (= Periode) umfasst fünf Arbeitstage (= Unterperioden). Pro Tag sind bis zu 1.700 Aufträge zu kommissionieren. Die Aufträge für den nächsten Tag sind bekannt. Die Lösung für das Auftragsbildungs- und Tourenplanungsproblem ist entsprechend für den nachfolgenden Tag zu berechnen. Die Anzahl und die Zusammensetzung der Aufträge schwanken über die Zeit stark. Aktuell werden die Aufträge mit dem FCFS-Algorithmus gebündelt. Die Kommissioniertour wird mithilfe des Nearest-Neighbor-Algorithmus bestimmt.

8.2 Ergebnisse

Die Verfahren wurden in Matlab R2017b implementiert und auf einem Computer mit einem 2,5 GHz Prozessor und 16 GB RAM Arbeitsspeicher durchgeführt. Die Vortests zur Bestimmung der Parameter des DEPSO-Algorithmus wurden mithilfe von Auftragsdaten eines Tages durchgeführt, die nicht Bestandteil der Hauptuntersuchung waren. Tabelle 6 zeigt die untersuchten Parameterwerte. Im Fallbeispiel wurden andere Parameterwerte als in den numerischen Experimenten getestet, da eine größere Problemklasse zu untersuchen war.

Parameter	Symbol	Wert
Anzahl Partikel	A^{Partikel}	{5, 7}
Anzahl Iterationen	It^{max}	{300, 400}
Schwellwert G_{best}	$S^{G_{\text{best}}}$	{0,25, 0,5, 0,75}
Maximale Iterationen lokale Suche	It^{maxLS}	{50, 60, 70}
Maximale Stagnationsschranke	S^{maxStag}	{40, 50, 60}

Tabelle 6: Untersuchte Parameterwerte im Vortest des Fallbeispiels

Gewählt wurde folgende Parameterkombination:

- Anzahl Partikel: 5
- Anzahl Iterationen: 300
- Schwellwert G_{best} : 0,5
- Maximale Anzahl Iterationen des lokalen Suchverfahrens: 60
- Maximale Stagnationsschranke: 50

Am Ende jeder Woche wurden 50 Umlagerungen mithilfe des dynamischen Lagerplatzvergabealgorithmus geprüft. Als Prognoseverfahren wurde die exponentielle Glättung mit Trendkorrektur eingesetzt, da zwar ein Trend, aber keine saisonalen Schwankungen erkennbar waren. Die Prognosekoeffizienten und -werte wurden mithilfe von (3.10) – (3.12) berechnet. Die Prognose wurde für zwei Wochen erstellt. Für den Startwert \hat{a}_2 wurde jeweils der zweite vorliegende Beobachtungswert genutzt. Der Startwert \hat{b}_2 wurde aus der mit dem Glättungsparameter α gewichteten Differenz aus zweitem und erstem Beobachtungswert berechnet. Der Glättungsparameter α wurde auf 0,3 gesetzt. Für den Schwellwert σ wurde der Wert 2 (inkl. aktuelle Periode) und für u der Wert 1 gewählt.

In Abbildung 49 sind die beiden Kommissionierweglängen dargestellt, die durch den Einsatz des FCFS-Algorithmus und des DEPSO-Algorithmus berechnet wurden. Beide Kommissionierweglängen wurden auf die Kommissionierweglänge normiert, die durch den Einsatz des FCFS am ersten Tag (Basis = 100) resultiert. Basis für die Darstellung ist die Kommissionierweglänge, die mithilfe des FCFS-Algorithmus für den ersten Tag berechnet wurde. Es sind starke Schwankungen im gesamten Untersuchungszeitraum festzustellen. Der Grund hierfür ist, dass die Anzahl der Aufträge über die Zeit stark schwankt und der Aufbau der Aufträge sowie die zu kommissionierenden Artikel starken Veränderungen unterliegen. An Tag 16 wurde nicht kommissioniert.

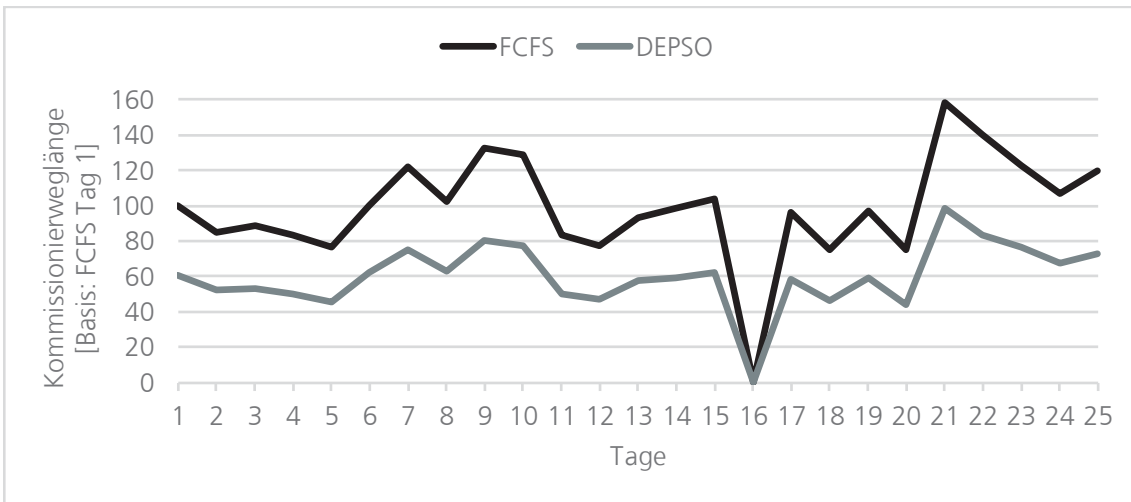


Abbildung 49: Kommissionierweglänge mit FCFS und DEPSO (Fallbeispiel)

Wie Abbildung 50 verdeutlicht, führt der Einsatz der DEPSO zu erheblichen Wegstreckenreduktionen von bis zu 40,89 % an Tag 20. Über den gesamten Untersuchungszeitraum ergeben sich Einsparungen in Höhe von 39,10 %.

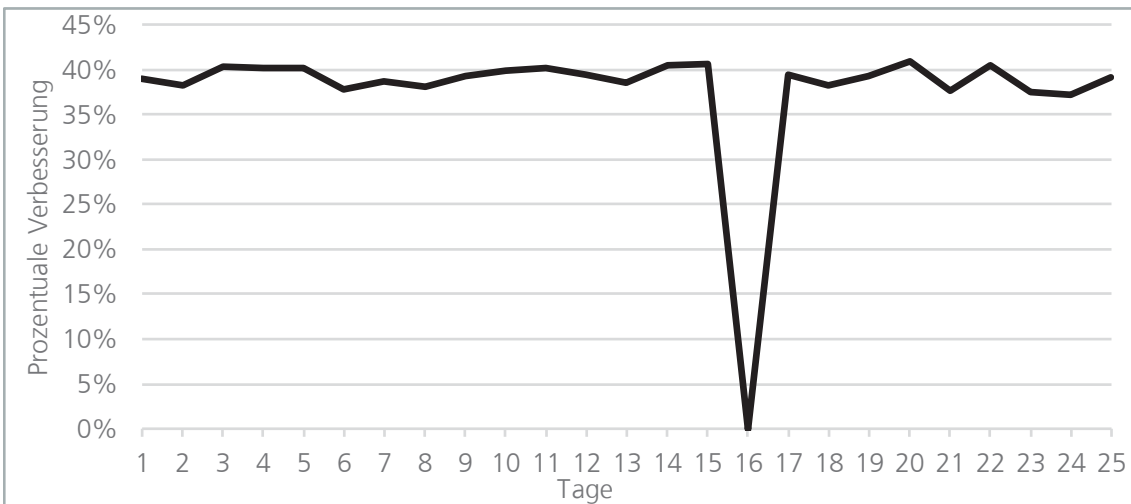


Abbildung 50: Wegstreckenreduktion durch den Einsatz des DEPSO-Algorithmus (Fallbeispiel)

Die zur Berechnung der Ergebnisse notwendigen Rechenzeiten sind in Abbildung 51 dargestellt.

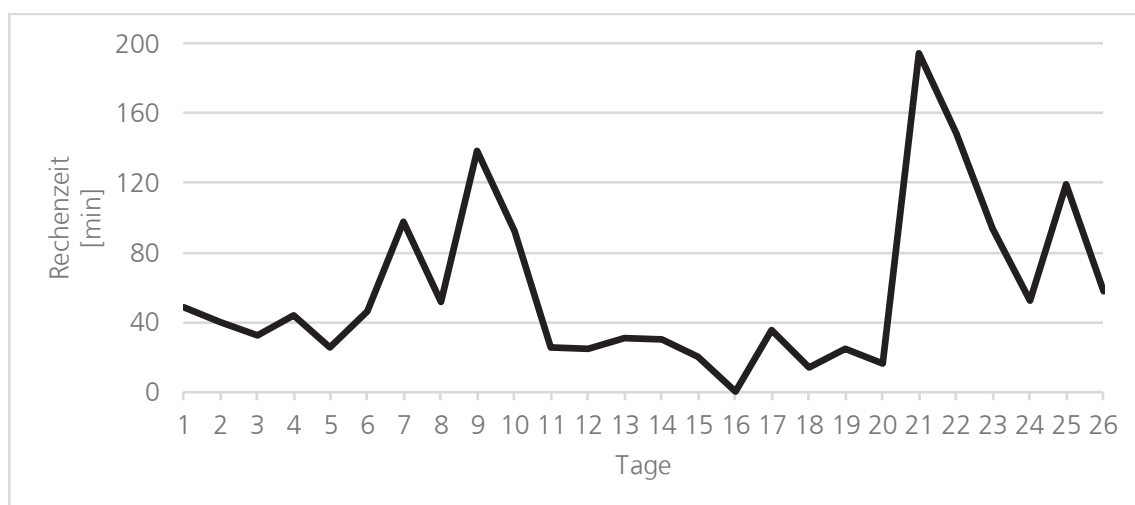


Abbildung 51: Rechenzeit DEPSO (Fallbeispiel)

Durchschnittlich benötigt der Algorithmus 57,94 Minuten, um für alle Aufträge eines Tages eine Lösung zu berechnen. Die höchsten Rechenzeiten ergaben sich an Tag 21 mit 194,39 Minuten. Am niedrigsten waren die Rechenzeiten an Tag 18 mit 13,70 Minuten. Grund für die schwankenden Rechenzeiten ist die unterschiedliche Anzahl Aufträge, die pro Tag zu kommissionieren war. Im untersuchten Fall wurde eine Lösung für alle Aufträge des jeweils nachfolgenden Tages berechnet. Die in Abbildung 51 dargestellten Rechenzeiten stellen daher kein Problem dar. In Abbildung 52 ist die bereits mit der DEPSO optimierte, Kommissionierweglänge ohne Umlagerungen, die Kommissionierweglänge nach den Umlagerungen sowie die Kommissionierweglänge nach den Umlagerungen inkl.

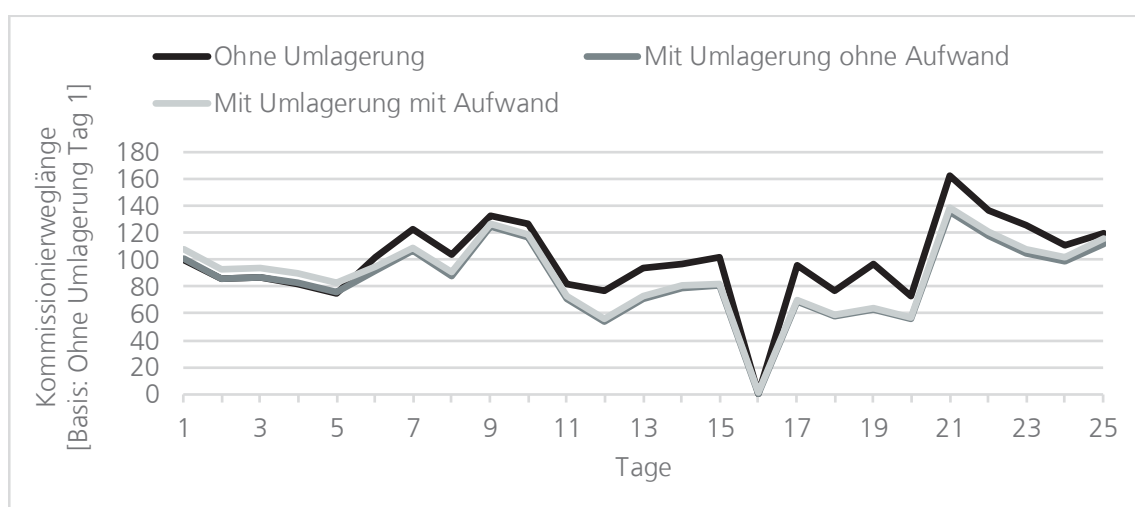


Abbildung 52: Kommissionierweglänge nach Umlagerung (Fallbeispiel)

Umlagerungsaufwand dargestellt. Zur besseren Vergleichbarkeit wurden die drei Kommissionierweglängen auf die Kommissionierweglänge ohne Umlagerung am ersten Tag (Basis = 100) normiert.

Der Umlagerungsaufwand einer Woche wurde zur übersichtlicheren Darstellung gleichmäßig auf die Wochentage verteilt. Mit Ausnahme der ersten Woche, in der dem Umlagerungsaufwand noch keine Wegstreckenreduktion gegenüberstehen kann, da in der Vorperiode noch keine Umlagerung durchgeführt wurde, wurde der Kommissionierweg nach der Umlagerung deutlich verkürzt. Die prozentualen Einsparungen werden aus Abbildung 53 ersichtlich. Der dynamische Lagerplatzvergabealgorithmus erzeugt einen bis zu 34,77 % kürzeren Kommissionierweg pro Tag als ohne Umlagerungen. Über den gesamten Untersuchungszeitraum konnte die Kommissionierweglänge um 13,66 % gesenkt werden. Dem gegenüber stehen Umlagerungsaufwände in Höhe von 3,16 %. Daraus resultiert eine Gesamtersparnis in Höhe von 10,50 % für die untersuchten fünf Wochen. Im Vergleich zum FCFS-Verfahren ohne Umlagerungen ergeben sich damit Einsparungen in Höhe von 49,60 %.

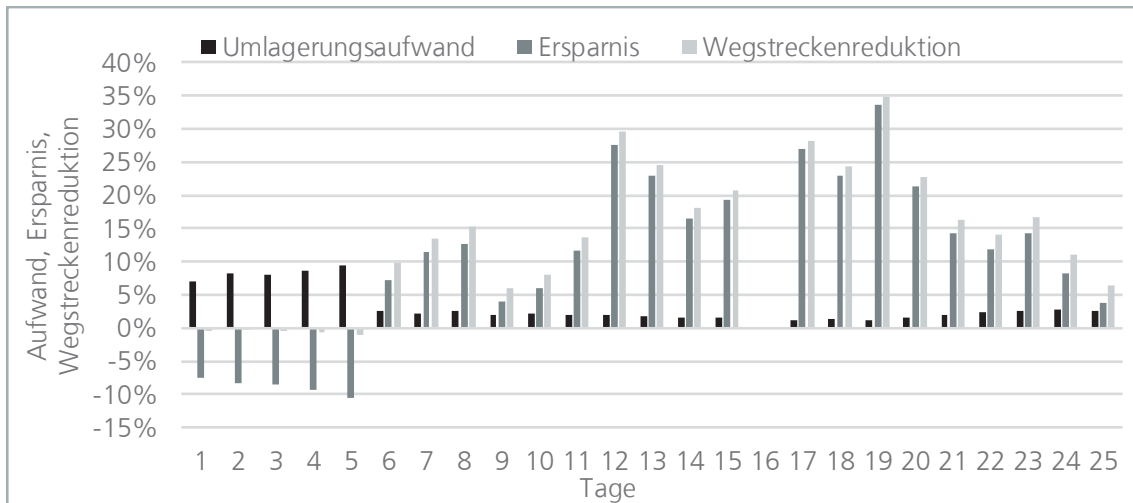


Abbildung 53: Umlagerungsaufwand, Ersparnis und Wegstreckenreduktion (Fallbeispiel)

Erreicht wurden diese Einsparungen wiederum mit einer begrenzten Anzahl umgelagerter Artikel. So wurden in Woche 1 0,3 % der Artikel des Artikelsortiments umgelagert (Abbildung 54).

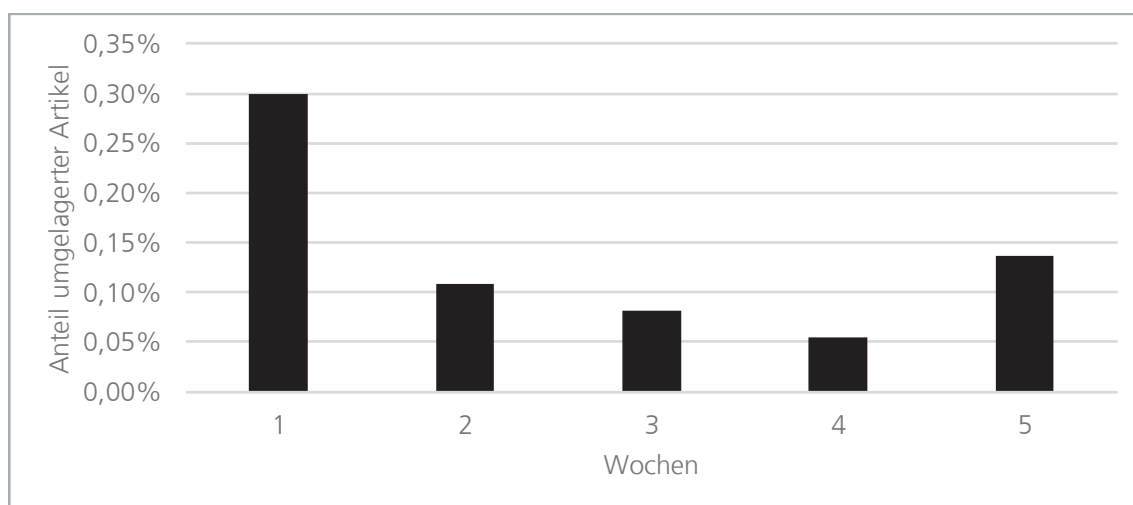


Abbildung 54: Anteil umgelagerter Artikel (Fallbeispiel)

Das entspricht 33 Artikeln. Die Ergebnisse des Fallbeispiels finden sich in den Tabellen 18 und 19 im Anhang.

8.3 Bewertung der Ergebnisse

Die entwickelten Verfahren liefern auch für reale Auftragsdaten signifikante Wegstreckenreduktionen. Der Kommissionierweg kann mithilfe des DEPSO-Algorithmus drastisch verkürzt werden. Die in den numerischen Experimenten beobachteten kurzen Rechenzeiten können bestätigt werden. So sind durchschnittlich ca. 58 Minuten für die Berechnung aller Aufträge eines Tages notwendig. Dabei ist zu beachten, dass bis zu 1.700 Aufträge mit bis zu 140 Positionen pro Tag zu berechnen waren. Die Ergebnisse können mithilfe des dynamischen Lagerplatzvergabealgorithmus weiter verbessert werden. Nach Abzug des notwendigen Umlagerungsaufwands resultiert eine zusätzliche Ersparnis in Höhe von 10,5 %. Im Vergleich zur Ausgangssituation (FCFS ohne Umlagerungen) können damit Ersparnisse von knapp 50 % realisiert werden. Damit kann die geforderte Praxistauglichkeit bestätigt werden. Die Verfahren wurden in einem ausgesprochen dynamischen Umfeld getestet. Somit kann nachgewiesen werden, dass sich das Verfahren sehr gut für den Einsatz in Situationen mit hoher Dynamik eignet. Da das Verfahren sowohl die Interdependenz

der Planungsprobleme als auch den Umlagerungsaufwand bei der Lagerplatzvergabeentscheidung berücksichtigt, konnten alle in Kapitel 1.3 formulierten Anforderungen an das Verfahren erfüllt werden. Die aus der Zielformulierung abgeleitete handlungsleitende Forschungsfrage wird durch das entwickelte Verfahren beantwortet. Dennoch können einige Erweiterungsmöglichkeiten des Verfahrens identifiziert werden. So könnten die Umlagerungsaufträge zusammen mit den Kommissionieraufträgen in die Optimierung eingehen und so der Kommissionieraufwand weiter gesenkt werden. In seiner derzeitigen Ausgestaltung berücksichtigt das Verfahren lediglich das Ziel der Wegstreckenreduktion. Eine Erweiterung um weitere Zielgrößen, wie etwa eine ergonomische Gestaltung des Systems, wäre wünschenswert. Auch könnte der Effekt mehrerer Lagerplätze für häufig zu kommissionierende Artikel auf die Optimierungsergebnisse des Verfahrens untersucht werden. Über geeignete Erweiterungen des Verfahrens könnte der Anwendungsbereich weiter vergrößert werden, so dass auch Systeme mit stark unterschiedlichen Artikel- bzw. Ladungsträger- und Lagerfachgrößen optimiert werden können.

9 Zusammenfassung

Der Trend zur zunehmenden Individualisierung führt zu einer immer volatileren Nachfrage. Unternehmen müssen Wege finden, um mit dieser gestiegenen Dynamik umzugehen. Ein effizient betriebenes Kommissioniersystem ist eine wesentliche Voraussetzung, um dem Wunsch der Kunden nach bezahlbaren individualisierten bzw. personalisierten Produkten gerecht zu werden. Da kurze und zuverlässige Lieferzeiten bei der Kaufentscheidung eine immer wichtigere Rolle spielen, müssen Artikel schnell und aufwandsarm kommissioniert werden. Ein wesentlicher Hebel sind die zurückzulegenden Kommissionierwege. Eine Senkung bewirkt nicht nur, dass Aufträge schneller bearbeitet und damit ausgeliefert werden können. Sie führt auch zu hohen Kosteneinsparungen. Aufgrund der kognitiven Fähigkeiten des Menschen und der damit verbundenen Flexibilität wird ein Großteil der Kommissioniersysteme nach wie vor manuell betrieben. Um diese Systeme effizient betreiben zu können, sind im Wesentlichen drei Planungsprobleme zu lösen: Die Bündelung von Aufträgen in sogenannten Batches (Auftragsbildungsproblem), die Planung der Kommissioniertouren (Tourenplanungsproblem) und die Vergabe der Lagerplätze (Lagerplatzvergabeproblem). Diese Probleme hängen stark voneinander ab und sollten daher gemeinsam gelöst werden. Bislang gab es hierzu aber noch kein geeignetes Lösungsverfahren.

Ziel der vorliegenden Arbeit war es daher, ein Verfahren zu entwickeln, welches die Planungsprobleme ganzheitlich löst und so zu einer signifikanten Wegstreckenreduktion führt. Neben der Interdependenz der Planungsprobleme war die Beherrschung der zunehmenden Dynamik eine weitere wesentliche Anforderung an das Verfahren. Über gezielte Umlagerungen sollte die Kundennachfrage in der Artikelanordnung abgebildet werden können. Wichtig in diesem Zusammenhang war es, sicherzustellen, dass die Umlagerungsaufwände den erzielbaren Nutzen nicht übersteigen. In einer umfangreichen Literaturrecherche konnte nachgewiesen werden, dass zur Lösung des so formulierten Forschungsbedarfes noch kein Verfahren existiert.

Da ein simultaner Lösungsansatz für praxisrelevante Problemgrößen unrealistisch ist, wurde ein iteratives Verfahren zur Lösung der drei Planungsprobleme entwickelt. Für das kombinierte Auftragsbildungs- und Tourenplanungsverfahren wurden Elemente der Schwarmintelligenz mit Bestandteilen des Evolutionary Computation kombiniert. Das Herzstück des entwickelten Verfahrens bildet ein Partikelschwarmoptimierungsalgorithmus, der im Allgemeinen sehr gut zur Lösung von kombinatorischen Optimierungsproblemen, wie dem vorliegenden Problem, geeignet ist. Schwächen bisheriger Ansätze waren deren hohe Rechenzeiten und die auf nur sehr kleine Problemgrößen beschränkte Einsetzbarkeit. Ein häufiges Problem der Partikelschwarmoptimierung ist die zu schnelle Fokussierung einer Lösung und das Verharren in einem lokalen Optimum. Um diese Schwächen zu beheben, wurde der Algorithmus um das Konzept der Mutation und um ein lokales Suchverfahren erweitert. Mithilfe des Einsatzes einer Heuristik zur Identifikation einer guten Anfangslösung, konnten die Rechenzeit des Verfahrens gesenkt und darüber hinaus das Optimierungsergebnis verbessert werden. In das Verfahren wurde ein 2opt-Algorithmus zur Bestimmung der kürzesten Kommissioniertour integriert. Dieser Algorithmus ist einfach aufgebaut und erzeugt innerhalb kurzer Rechenzeiten sehr gute Optimierungsergebnisse. Dies ist für eine kombinierte Lösung wesentlich, da der Tourenplanungsalgorithmus im Zuge des Verfahrens vielfach durchlaufen werden muss. Das so gefundene Verfahren wird als diskrete evolutionäre Partikelschwarmoptimierung (DEPSO) bezeichnet.

Zur Lösung des Lagerplatzvergabeproblems wurde ein dynamischer Lagerplatzvergabealgorithmus entwickelt. In einem dynamischen Umfeld wird es für Unternehmen zunehmend wichtig, frühzeitig zukünftige Entwicklungen zu erkennen und proaktiv zu handeln. Zur Beherrschung der Dynamik sind Prognosefähigkeiten existentiell. Daher wurden ein Prognose- und ein Schätzverfahren in das Verfahren integriert. Um die Praxistauglichkeit der Lösung sicherzustellen, wurde ein Prioritätswertverfahren entwickelt, das die Vorauswahl geeigneter Umlagerungsartikel ermöglicht. Zur Bewertung eines Umlagerungsvorschlags wird mithilfe des DEPSO-Algorithmus und dem Auftragsbestand der aktuellen Periode berechnet, ob durch die

Umlagerung der Kommissionierweg verkürzt werden kann. Die erzielte Wegstreckenreduktion wird mithilfe der Prognosedaten und des Schätzverfahrens in die Zukunft projiziert. Überwiegen die zukünftig zu erwartenden Einsparungen die notwendigen Aufwände, wird die Umlagerung durchgeführt, ansonsten verworfen. Im Ergebnis werden einerseits die veränderte Kundennachfrage in der Artikelanordnung abgebildet und andererseits die Spezifika des Auftragsbildungs- und Tourenplanungsverfahrens berücksichtigt. Interdependenz und Dynamik werden somit in einem ganzheitlichen Verfahren beherrscht.

Im Zuge von umfangreichen numerischen Experimenten konnte die Leistungsfähigkeit der entwickelten Verfahren nachgewiesen werden. Durch die detaillierte Beschreibung der Erzeugung der Testdaten wurde die Nachvollziehbarkeit des Vorgehens und der Ergebnisse sichergestellt. Mit der DEPSO lassen sich signifikante Optimierungsergebnisse innerhalb kurzer Rechenzeiten erzielen. So resultierten durch dessen Einsatz durchschnittlich ca. 32 % kürzere Kommissionierwege als mit dem Savings-Verfahren. Die Rechenzeiten bewegten sich zwischen 21 und 226 Sekunden. Eine wichtige Beobachtung im Zuge der Experimente war, dass die erzielbare Wegstreckenreduktion mit sinkender Auftragsgröße steigt. Der dynamische Lagerplatzvergabealgorithmus wurde im Zuge zweier Szenarien untersucht. Für das Szenario mit der höheren Dynamik erzeugte das Verfahren bessere Ergebnisse. Durchschnittlich konnten in diesem Szenario Wegstreckenreduktionen von ca. 15 % realisiert werden. Auch in dem weniger dynamischen Szenario wurde der Kommissionierweg durch die Umlagerungen um ca. 7 % verkürzt. Hierfür sind keine umfangreichen Lagerreorganisationen notwendig. Vielmehr konnten die Reduktionen über die Umlagerung einiger weniger Artikel realisiert werden. Das legt den Schluss nahe, dass Unternehmen sich auf die Artikel konzentrieren sollten, die zukünftig von Langsam- zu Schnelldrehern werden.

Die Praxistauglichkeit des Verfahrens konnte in einem Fallbeispiel nachgewiesen werden. Mithilfe von realen Auftragsdaten eines produzierenden Unternehmens wurden der DEPSO-Algorithmus und der dynamische Lagerplatzvergabealgorithmus

getestet. Die Daten wiesen eine sehr hohe Dynamik auf. Die DEPSO konnte die Kommissionierweglänge im Untersuchungszeitraum um ca. 39 % senken. Durchschnittlich waren hierzu lediglich knapp 58 Minuten Rechenzeit pro Tag notwendig. Der dynamische Lagerplatzvergabealgorithmus führte zu einer weiteren Verbesserung in Höhe von ca. 11 %. In Summe konnten damit Ersparnisse in Höhe von knapp 50 % realisiert werden.

Das eingangs formulierte Ziel konnte damit erreicht und die Forschungsfrage beantwortet werden. Mit dem entwickelten Verfahren lassen sich in dynamischen Situationen durch die Berücksichtigung der Interdependenz der Planungsprobleme signifikante Wegstreckenreduktionen erzielen. Die Praxistauglichkeit konnte nicht zuletzt durch die Berücksichtigung der Umlagerungsaufwände sichergestellt und im Fallbeispiel nachgewiesen werden.

Dennoch hat das Themenfeld Kommissionierung weiteren Forschungsbedarf. Der Umlagerungsprozess selbst könnte mithilfe eines kombinierten Auftragsbildungs- und Tourenplanungsverfahrens optimiert werden. Ein weiterer Aspekt wäre die Integration von Ergonomiekriterien in das Lagerplatzvergabeverfahren. Gerade in den alternden Gesellschaften der westlichen Industrienationen wird dies zu einem immer wichtigeren Aspekt. Auch könnte der Effekt mehrerer Lagerplätze für häufig zu kommissionierende Artikel untersucht werden. Das Verfahren könnte überdies für den Einsatz in Kommissioniersystemen mit stark unterschiedlichen Artikel- bzw. Ladungsträger- und Lagerfachgrößen erweitert werden. Weiterhin könnte eine Erweiterung des Verfahrens für den Einsatz in automatischen Kommissioniersystemen geprüft werden. Ein weiterer interessanter Aspekt wäre die Berücksichtigung des sogenannten „Picker Blockings“, bei dem der Effekt sich gegenseitig beeinträchtigender Kommissionierer untersucht wird.

Anhang

Problem- klasse	Vergleich DEPSO-SOP			Vergleich DEPSO-FCFS			Vergleich DEPSO-Savings		
	Min.	Ø	Max.	Min.	Ø	Max.	Min.	Ø	Max.
50_2_6	-87,50%	-88,52%	-89,37%	-32,66%	-38,98%	-48,61%	-17,12%	-30,03%	-38,54%
50_2_10	-86,04%	-87,13%	-88,25%	-36,73%	-44,00%	-48,69%	-26,54%	-33,36%	-41,96%
50_6_2	-86,23%	-87,36%	-88,48%	-24,98%	-30,74%	-36,97%	-23,22%	-28,35%	-36,82%
50_6_6	-80,18%	-81,54%	-84,61%	-36,91%	-41,09%	-48,75%	-30,71%	-35,92%	-41,48%
50_6_10	-75,49%	-77,49%	-80,20%	-38,04%	-44,20%	-50,11%	-29,99%	-36,31%	-42,50%
50_10_2	-83,83%	-84,83%	-86,56%	-18,74%	-28,61%	-34,26%	-15,85%	-25,90%	-32,07%
50_10_6	-74,69%	-76,61%	-78,83%	-34,44%	-38,75%	-43,04%	-26,33%	-32,60%	-37,59%
50_10_10	-68,30%	-70,60%	-73,90%	-38,03%	-42,37%	-46,18%	-26,95%	-33,87%	-37,96%
100_2_2	-92,29%	-92,74%	-93,43%	-31,29%	-36,46%	-43,34%	-18,53%	-26,29%	-33,78%
100_2_6	-89,76%	-90,79%	-91,41%	-40,31%	-47,77%	-53,42%	-26,00%	-34,77%	-39,34%
100_2_10	-87,43%	-88,63%	-89,56%	-45,77%	-51,33%	-54,92%	-31,86%	-38,24%	-44,05%
100_6_2	-87,92%	-89,20%	-90,26%	-24,20%	-30,78%	-35,11%	-21,29%	-26,55%	-32,89%
100_6_6	-81,34%	-82,77%	-84,61%	-38,24%	-42,66%	-47,14%	-31,32%	-35,36%	-39,40%
100_6_10	-76,62%	-78,51%	-80,30%	-41,30%	-45,74%	-49,09%	-29,87%	-36,43%	-40,70%
100_10_2	-84,70%	-86,23%	-87,76%	-24,14%	-29,22%	-35,04%	-22,49%	-26,40%	-30,49%
100_10_6	-75,48%	-77,36%	-79,54%	-38,08%	-41,29%	-45,44%	-29,73%	-33,56%	-36,36%
100_10_10	-69,20%	-71,25%	-74,83%	-41,58%	-44,12%	-46,84%	-29,62%	-33,65%	-36,71%
150_2_2	-93,74%	-94,22%	-94,64%	-29,00%	-34,15%	-39,97%	-18,52%	-24,76%	-30,73%
150_2_6	-90,30%	-91,23%	-92,65%	-46,81%	-50,26%	-55,19%	-28,10%	-34,75%	-40,20%
150_2_10	-88,50%	-89,25%	-90,16%	-45,77%	-51,97%	-55,94%	-32,30%	-35,80%	-39,95%
150_6_2	-88,74%	-89,34%	-90,06%	-28,85%	-32,23%	-36,94%	-23,05%	-28,04%	-31,86%
150_6_6	-81,32%	-82,89%	-84,07%	-38,98%	-43,43%	-46,62%	-30,10%	-35,08%	-38,98%
150_6_10	-76,54%	-78,45%	-79,67%	-43,67%	-46,16%	-48,21%	-32,03%	-35,84%	-38,94%
150_10_2	-85,47%	-86,22%	-87,04%	-24,56%	-29,21%	-35,75%	-20,73%	-24,83%	-28,74%
150_10_6	-76,02%	-77,32%	-78,65%	-37,19%	-40,74%	-43,23%	-29,21%	-32,25%	-35,59%
150_10_10	-70,28%	-71,41%	-72,88%	-42,42%	-44,38%	-47,21%	-29,71%	-32,15%	-35,38%
200_2_2	-93,83%	-94,26%	-94,65%	-36,54%	-41,36%	-45,51%	-20,65%	-26,89%	-32,32%
200_2_6	-90,69%	-91,54%	-92,10%	-47,30%	-51,42%	-55,07%	-27,60%	-33,72%	-38,08%
200_2_10	-88,43%	-89,26%	-89,95%	-50,27%	-53,33%	-57,27%	-30,25%	-35,29%	-40,23%
200_6_2	-88,69%	-89,61%	-90,15%	-27,62%	-31,43%	-35,35%	-22,90%	-26,41%	-31,27%
200_6_6	-81,17%	-82,58%	-83,94%	-38,89%	-41,96%	-44,15%	-29,02%	-33,17%	-36,30%
200_6_10	-77,19%	-78,35%	-80,04%	-43,56%	-45,24%	-48,21%	-29,40%	-34,01%	-36,77%
200_10_2	-85,42%	-86,11%	-87,13%	-25,49%	-28,26%	-31,57%	-20,97%	-24,02%	-28,02%
200_10_6	-75,60%	-77,21%	-78,16%	-37,99%	-40,34%	-42,47%	-28,11%	-31,02%	-33,45%
200_10_10	-70,31%	-71,57%	-72,62%	-41,01%	-43,94%	-46,19%	-29,88%	-31,79%	-34,19%

Tabelle 7: Optimierungsergebnisse numerische Experimente DEPSO

Unterperiode	Periode 1	Periode 2	Periode 3	Periode 4	Periode 5	Periode 6	Periode 7	Periode 8	Periode 9
1	7180	7952	8377	9307	9101	9289	9622	9748	9837
2	7557	7762	8340	8585	8658	8766	9561	10358	10708
3	8318	8639	8761	8132	8900	9062	10056	9270	10136
4	7767	7873	8347	8738	8860	9594	9372	9703	10758
5	7714	8467	8129	8819	8887	8793	9143	10395	10148
6	7479	8226	8646	8604	9088	9256	10189	9988	10113
7	7865	8006	7957	8577	8597	9375	9325	9374	10311
8	7376	8261	7747	8306	8381	9060	9944	9866	9851
9	7872	8012	8601	8794	9018	9193	9462	9648	10335
10	7752	8954	8620	8582	8166	9162	9337	10499	10526
11	8010	8196	8497	8545	8699	9310	9103	10168	10245
12	7720	8238	8862	8528	8529	9646	8919	9274	10430
13	7605	8545	8726	8867	8789	9269	9501	10033	9974
14	8155	7526	8043	8520	8701	9802	9367	9734	11076
15	7675	8348	8328	8527	9124	8765	9820	10198	10382
16	7913	7620	8016	8546	9264	9164	9381	9894	10272
17	7930	8226	9109	8670	9169	9071	9702	10075	9958
18	8089	7971	8581	8022	8647	9251	9286	10000	9718
19	7712	7929	8582	8571	8828	8462	9570	9400	10067
20	8160	7950	8358	8791	9940	9060	9655	9899	10255
Gesamt	155849	162701	168627	172031	177346	183350	190315	197524	205100

Tabelle 8: Dynamischer Lagerplatzvergabealgorithmus Szenario 1: Kommissionierweglänge DEPSO ohne Umlagerung [in LE]

Periode	1	2	3	4	5	6	7	8	9
Umlagerungsaufwand	7091	6643	4831	852	3022	4930	6931	7434	3214

Tabelle 9: Dynamischer Lagerplatzvergabealgorithmus Szenario 1: Umlagerungsaufwand [in LE]

Periode	1	2	3	4	5	6	7	8	9
Umgelagerte Artikel	24	22	16	3	10	16	23	25	11

Tabelle 10: Dynamischer Lagerplatzvergabealgorithmus Szenario 1: Anzahl umgelagerter Artikel

Unterperiode	Periode 1	Periode 2	Periode 3	Periode 4	Periode 5	Periode 6	Periode 7	Periode 8	Periode 9
1	112,07	151,56	153,68	123,75	150,10	144,60	185,14	214,88	162,64
2	183,65	180,28	177,78	205,27	210,86	193,86	200,82	241,71	191,25
3	190,07	229,97	172,48	184,46	201,93	184,23	237,16	210,70	234,78
4	165,74	160,34	201,71	202,99	164,25	208,19	198,01	208,19	224,90
5	173,54	158,08	170,56	186,05	180,33	227,99	179,07	214,38	223,97
6	168,30	182,94	184,46	181,11	192,78	198,27	234,01	218,52	214,94
7	173,57	149,53	165,65	192,35	192,25	209,58	224,51	222,27	233,64
8	178,15	179,35	157,09	172,79	187,62	201,11	187,74	219,58	254,04
9	180,45	182,63	202,61	197,89	189,82	191,90	235,63	196,45	200,70
10	185,19	205,35	183,83	202,81	179,71	187,48	198,14	220,25	215,52
11	162,39	180,90	207,43	166,47	178,77	204,55	177,94	184,69	218,54
12	156,84	205,38	165,08	192,22	151,36	183,49	214,59	208,32	235,30
13	159,27	170,78	191,97	154,07	191,60	196,02	184,94	230,47	232,25
14	167,23	167,08	194,70	227,56	206,44	189,11	198,21	240,73	262,99
15	150,17	160,26	176,15	184,14	197,79	172,58	192,02	228,72	240,12
16	170,71	146,23	187,65	160,94	203,95	223,64	228,29	211,28	210,39
17	158,61	172,36	219,44	173,64	208,99	181,45	184,03	222,96	201,63
18	180,04	174,99	176,14	181,33	181,86	176,09	199,38	234,25	216,82
19	153,83	177,41	176,21	170,11	176,63	212,54	204,15	221,70	250,69
20	194,43	165,70	224,90	167,76	202,31	211,37	199,78	229,60	206,57

Tabelle 11: Dynamischer Lagerplatzvergabealgorithmus Szenario 1: Rechenzeit [in Sekunden]

Unterperiode	Periode 1	Periode 2	Periode 3	Periode 4	Periode 5	Periode 6	Periode 7	Periode 8	Periode 9
1	7306	7317	6962	7645	7540	7497	7438	7178	7288
2	7647	7422	7302	7280	7381	7300	7864	8073	8602
3	8279	8004	7464	7079	7663	7703	8202	7549	7993
4	7970	7418	7191	7397	7581	7611	7836	7776	8122
5	7754	7835	7130	7613	7561	7114	7545	8285	7597
6	7353	8076	7831	7466	7566	7736	8376	7569	7816
7	7761	7505	7043	7038	7270	7749	7755	7619	7789
8	7526	7522	6784	7323	7386	7428	8134	7526	7527
9	7866	7221	7430	7453	7594	7868	7519	7436	8231
10	7788	8583	7459	7267	6673	7884	7697	7940	7988
11	8048	7562	7359	7227	7412	7838	7074	7887	7697
12	7680	7351	7538	7516	7419	7946	7230	7466	7815
13	7693	7711	7647	7549	7502	7572	7367	8117	7525
14	7986	7189	6816	7147	7670	8087	7668	7686	8090
15	7778	7762	7192	7092	7527	7176	7884	8110	8108
16	8028	7308	7002	7522	7776	7755	7557	7755	7915
17	7978	7642	7862	7199	7768	7411	7995	8135	7826
18	8000	7355	7409	6903	7568	7668	7471	7700	7707
19	7924	7530	7712	7032	7583	7028	7741	7546	7920
20	8070	7307	7474	7262	8336	7704	7904	7930	8024
Gesamt	156435	151620	146607	146010	150776	152075	154257	155283	157580

Tabelle 12: Dynamischer Lagerplatzvergabealgorithmus Szenario 1: Kommissionierweglänge nach Umlagerung ohne Aufwand [in LE]

Unterperiode	Periode 1	Periode 2	Periode 3	Periode 4	Periode 5	Periode 6	Periode 7	Periode 8	Periode 9
1	7384	7414	8291	8403	8030	8571	9152	8813	8887
2	7747	7649	7322	7838	9036	8273	9631	9323	9194
3	7091	8016	8269	8283	8498	8404	8748	8989	9981
4	7248	7978	7857	7880	8417	8604	8461	8726	9300
5	7298	7761	8494	8444	8191	8742	9276	8944	9840
6	7783	7946	7774	8142	8225	8675	8525	9358	9532
7	7603	7664	8018	8186	8095	8984	9107	9343	9890
8	7820	8073	7458	8321	8367	8717	9123	8681	8722
9	7148	7435	7982	8662	9034	8447	9688	9050	9822
10	6995	7533	7451	8467	8147	8831	8689	9058	9489
11	7593	7667	7826	8043	8560	8737	9007	9212	10107
12	7376	7388	8032	7484	8219	8318	8849	9615	9980
13	7447	7318	7521	8248	8392	8303	9220	9515	9816
14	7497	8095	7901	7659	8618	8626	8988	9275	9728
15	7445	7791	8161	8218	8279	8001	8328	9732	9574
16	6550	7741	8139	8403	8554	8855	8783	9449	9789
17	7054	7565	8196	8468	8439	8054	8664	9050	9883
18	7155	7730	7515	8074	8019	9147	8611	9865	9967
19	7158	7634	8419	7899	8506	8498	8644	8850	9815
20	7956	7358	7880	7887	7715	8938	8171	9268	9538
Gesamt	147348	153756	158506	163009	167341	171725	177665	184116	192854

Tabelle 13: Dynamischer Lagerplatzvergabealgorithmus Szenario 2: Kommissionierweglänge DEPSO ohne Umlagerung [in LE]

Periode	1	2	3	4	5	6	7	8	9
Umlagerungsaufwand	2875	2150	3239	4106	3173	916	4417	5988	4622

Tabelle 14: Dynamischer Lagerplatzvergabealgorithmus Szenario 2: Umlagerungsaufwand [in LE]

Periode	1	2	3	4	5	6	7	8	9
Umgelagerte Artikel	9	7	11	14	11	3	15	20	16

Tabelle 15: Dynamischer Lagerplatzvergabealgorithmus Szenario 2: Anzahl umgelagerter Artikel

Unterperiode	Periode 1	Periode 2	Periode 3	Periode 4	Periode 5	Periode 6	Periode 7	Periode 8	Periode 9
1	138,36	192,84	181,90	198,75	172,44	174,21	187,23	197,52	213,32
2	167,04	180,16	202,65	162,93	204,16	204,52	193,06	214,79	203,27
3	152,98	174,03	178,30	212,70	179,04	206,93	187,52	198,15	238,73
4	141,63	160,22	180,55	163,51	211,38	159,67	226,95	216,93	231,81
5	143,41	161,78	187,40	192,99	178,57	223,62	190,66	180,61	234,39
6	161,41	179,60	163,97	170,41	174,54	189,79	191,54	190,93	181,39
7	183,98	190,73	202,03	183,23	171,93	204,40	210,37	199,60	247,57
8	150,26	177,87	170,21	159,14	160,79	185,78	208,66	204,53	205,76
9	147,37	156,39	166,16	161,50	205,61	152,22	230,22	198,83	233,53
10	156,66	164,91	171,51	177,29	168,92	185,37	166,23	208,15	213,59
11	167,56	169,72	157,77	191,73	197,82	195,71	188,11	222,57	235,40
12	142,02	183,77	159,57	193,51	148,08	196,25	217,74	218,68	202,94
13	173,65	165,51	199,56	159,27	170,82	239,11	192,51	202,23	197,68
14	144,05	168,65	176,28	165,25	190,89	188,40	172,64	198,71	243,82
15	138,74	166,88	204,53	217,82	197,67	159,99	167,37	187,50	243,59
16	161,24	221,18	166,51	177,65	166,44	165,27	173,43	210,45	244,88
17	140,65	155,66	154,22	186,95	169,21	180,84	189,57	228,65	188,71
18	152,08	190,06	161,79	151,05	161,34	197,34	179,70	205,22	227,38
19	158,76	174,88	183,47	161,74	174,86	173,63	213,62	199,14	202,94
20	179,39	135,05	167,95	187,11	183,07	198,95	191,81	225,42	172,32

Tabelle 16: Dynamischer Lagerplatzvergabealgorithmus Szenario 2: Rechenzeit [in Sekunden]

Unterperiode	Periode 1	Periode 2	Periode 3	Periode 4	Periode 5	Periode 6	Periode 7	Periode 8	Periode 9
1	7403	7047	8054	7581	7564	7986	8343	8016	8087
2	7635	7249	6960	7263	8180	7320	8771	8649	8183
3	7002	7518	7685	7645	7687	7891	7875	8242	8690
4	7221	7713	7490	7361	7615	7798	7627	7698	8044
5	7310	7484	7895	7848	7750	8056	8348	8146	8395
6	7859	7732	7289	7661	7591	7867	7732	8288	8360
7	7469	7654	7562	7420	7483	8261	8385	8551	8577
8	7673	7644	7267	7479	7860	7957	8242	7731	7834
9	7370	7027	7530	7928	8137	7651	8600	8353	8620
10	7062	7142	7081	7878	7550	8010	8211	7958	8619
11	7626	7479	7385	7501	7915	8107	8034	8180	8928
12	7423	7208	7536	6851	7670	7516	8097	8662	8738
13	7567	7198	6940	7682	7656	7683	8529	8615	8544
14	7888	7913	7801	7216	7888	7996	8009	8163	8647
15	7382	7672	7343	7604	7790	7506	7655	8569	8628
16	6702	7333	7760	8015	7812	8023	7655	8292	8732
17	7246	7116	7802	7784	8058	7479	7623	7823	8618
18	7141	7394	7165	7407	7311	8035	7747	8466	8702
19	7060	7236	7788	7529	7932	7976	7767	7685	9006
20	7928	7074	7403	7517	7026	7999	7460	8098	8245
Gesamt	147967	147833	149736	151170	154475	157117	160710	164185	170197

Tabelle 17: Dynamischer Lagerplatzvergabealgorithmus Szenario 2: Kommissionierweglänge nach Umlagerung ohne Aufwand [in LE]

Woche	Wochentag	Tag	FCFS	DEPSO ohne Umlagerung	DEPSO nach Umlagerung	Umlagerungs- aufwand
30	1	1	51027	31109	31219	2192
30	2	2	43216	26692	26720	2192
30	3	3	45272	27019	27122	2192
30	4	4	42763	25602	25785	2192
30	5	5	38996	23327	23585	2192
31	1	6	51027	31747	28605	840
31	2	7	62413	38317	33115	840
31	3	8	52087	32297	27352	840
31	4	9	67655	41135	38660	840
31	5	10	65608	39419	36214	840
32	1	11	42605	25502	22030	496
32	2	12	39581	23961	16851	496
32	3	13	47682	29307	22095	496
32	4	14	50471	30083	24638	496
32	5	15	53134	31588	25016	496
33	1	16	0	0	0	344
33	2	17	49050	29720	21358	344
33	3	18	38488	23802	17995	344
33	4	19	49432	30013	19578	344
33	5	20	38239	22602	17445	344
34	1	21	80848	50425	42218	980
34	2	22	71668	42657	36612	980
34	3	23	62677	39208	32658	980
34	4	24	54733	34409	30607	980
34	5	25	61150	37274	34909	980
Gesamt			1259822	767215	662380	24257

Tabelle 18: Fallstudie: Kommissionierweglänge und Umlagerungsaufwand [in Meter]

Periode	1	2	3	4	5
Umgelagerte Artikel	33	12	9	6	15

Tabelle 19: Fallstudie: Anzahl umgelagerter Artikel

Literaturverzeichnis

- Adolph et al. 2016** Adolph, Siri, Kübler, Patrick, Metternich, Joachim & Abele, Eberhard. 2016. Overall commissioning effectiveness: systematic identification of value added shares in material supply. *Procedia CIRP* **41**, S. 562-567. DOI: 10.1016/j.procir.2015.12.039
- Albareda-Sambola et al. 2009** Albareda-Sambola, Maria, Alonso-Ayuso, Antonio, Molina, Elisenda & de Blas, Clara Simon. 2009. Variable Neighborhood Search for Order Batching in a Warehouse. *Asia Pacific Journal of Operational Research* **26** (5), S. 655-683. DOI: 10.1142/S0217595909002390
- Ang et al. 2012** Ang, Marcus, Lim, Yun Fong & Sim, Melvyn. 2012. Robust Storage Assignment in Unit-Load Warehouses. *Management Science* **58** (11), S. 2114-2130. DOI: 10.1287/mnsc.1120.1543
- Azadnia et al. 2013** Azadnia, Amir Hossein, Taheri, Shahrooz, Ghadimi, Pezhman, Saman, Muhammad Zameri Mat & Wong, Kuan Yew. 2013. Order Batching in Warehouses by Minimizing Total Tardiness: A Hybrid Approach of Weighted Association Rule Mining and Genetic Algorithms. *The Scientific World Journal* 2013, S. 1-13. DOI: 10.1155/2013/246578
- Bäck et al. 1993** Bäck, Thomas & Schwefel, Hans-Paul. 1993. An Overview of Evolutionary Algorithms for Parameter Optimization. *Evolutionary Computation* **1** (1), S. 1-23. DOI: 10.1162/evco.1993.1.1.1
- Bamberg et al. 2008** Bamberg, Günter, Coenenberg, Adolf G. & Krapp, Michael. 2008. *Betriebswirtschaftliche Entscheidungslehre*. 14., überarb. Aufl. München: Vahlen. ISBN 9783800635061.
- Barr et al. 1995** Barr, Richard S, Golden, Bruce L., Kelly, James P., Resende, Mauricio G.C. & Stewart Jr., William R. 1995. Designing and reporting on computational experiments with heuristic methods.

Journal of heuristics **1** (1), S. 9-32.
DOI: 10.1007/BF02430363

Bauernhansl 2015

Bauernhansl, Thomas. 2015.
Die Produktion der Zukunft zwischen Mass Personalization und Mass Sustainability: (R)Evolution in der Produktion.
In: *Innovationstag 2015: Innovationsideen für die Produktion von morgen*, Linz, 10. März 2015, 40 Folien.

Bauernhansl 2017

Bauernhansl, Thomas. 2017.
Die Vierte Industrielle Revolution - Der Weg in ein wertschaffendes Produktionsparadigma.
In: Vogel-Heuser, Birgit, Bauernhansl, Thomas & ten Hompel, Michael (Hrsg.): *Handbuch Industrie 4.0 Bd. 4. Allgemeine Grundlagen*.
2. Aufl.
Wiesbaden: Springer Vieweg, S. 1-32.
ISBN 9783662532546.
DOI: 10.1007/978-3-662-53254-6_1

Battini et al. 2015

Battini, Daria, Calzavara, Martina, Persona, Alessandro & Sgarbossa, Fabio. 2015.
Order picking system design: the storage assignment and travel distance estimation (SA&TDE) joint method.
International Journal of Production Research **53** (4), S. 1077-1093.
DOI: 10.1080/00207543.2014.944282

Behnisch et al. 2017

Behnisch, Pia, Glock, Christoph, Grosse, Eric & Ries, Jörg. 2017.
Auf dem Weg zum Warehouse 4.0? Zum aktuellen Stand der Automatisierung in der Lagerhaltung.
In: Glock, Christoph & Grosse, Eric (Hrsg.): *Warehousing 4.0. Technische Lösungen und Managementkonzepte für die Lagerlogistik der Zukunft*.
Lauda-Königshofen: B + G Wissenschaftsverlag, S. 53-74.
ISBN 9783944325071.

Bessant et al. 2002

Bessant, John, Knowles, David, Briffa, Greg & Francis, David. 2002.
Developing the agile Enterprise.
International Journal of Technology Management **24** (5-6), S. 484-497.
DOI: 10.1504/IJTM.2002.003066

Bindi et al. 2007

Bindi, F., Manzini, R., Pareschi, A. & Regattieri, A. 2007.
Similarity coefficients and clustering techniques for the correlated assignment problem in warehousing systems.

- In: Proceedings of 19th International Conference on Production Research, Valparaiso, 29. Juli - 2. August 2007, S. 1-6.
- Birattari et al. 2001** Birattari, Mauro, Paquete, Luis, Stützle, Thomas & Varrentrapp, Klaus. 2001. Classification of Metaheuristics and Design of Experiments for the Analysis of Components. *Tech. Rep. AIDA-01-05*.
- Blum et al. 2003** Blum, Christian & Roli, Andrea. 2003. Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison. *ACM Computing Surveys* **35** (3), S. 268-308. DOI: 10.1145/937503.937505
- Blum et al. 2011** Blum, Christian, Puchinger, Jakob, Raidl, Günther R. & Roli, Andrea. 2011. Hybrid metaheuristics in combinatorial optimization: A survey. *Applied Soft Computing* **11** (6), S. 4135-4151. DOI: 10.1016/j.asoc.2011.02.032
- Bogon 2013** Bogon, Tjorben. 2013. *Agentenbasierte Schwarmintelligenz*. Wiesbaden: Springer Vieweg. ISBN 9783658022921. Trier, Univ. Diss., 2012.
- Bormann 1975** Bormann, Peter. 1975. *Adaptive Verfahren bei der Prognose durch exponentielle Glättung*. München, Univ. Diss., 1975.
- Bottani et al. 2012** Bottani, Eleonora, Cecconi, Margherita, Vignali, Giuseppe & Montanari, Roberto. 2012. Optimisation of storage allocation in order picking operations through a genetic algorithm. *International Journal of Logistics Research and Applications* **15** (2), S. 127-146. DOI: 10.1080/13675567.2012.694860
- Boussaïd et al. 2013** Boussaïd, Ilhem, Lepagnot, Julien & Siarry, Patrick. 2013. A survey on optimization metaheuristics. *Information Sciences* **237**, S. 82-117. DOI: 10.1016/j.ins.2013.02.041
- Brockhoff 2005** Brockhoff, Klaus. 2005. Prognosen. Bea, Franz Xaver, Friedl, Birgit & Schweitzer, Marcell (Hrsg.): *Allgemeine Betriebswirtschaftslehre - Band 2: Führung*.

9., neubearb. u. erw. Aufl.
Stuttgart: UTB, S. 759-799.
ISBN 9783825210823.

Cambazard et al. 2018

Cambazard, H. & Catusse, N. 2018.
Fixed-parameter algorithms for rectilinear Steiner tree
and rectilinear traveling salesman problem in the
plane.
European Journal of Operational Research **270** (2),
S. 419-429.
DOI: 10.1016/j.ejor.2018.03.042

Carlo et al. 2010

Carlo, Hector J. & Giraldo, German E. 2010.
Optimizing the Rearrangement Process in a Dedicated
Warehouse.
Progress in Material Handling Research, S. 39-48.
Verfügbar: <http://www.mhi.org/downloads/learning/cicmhe/colloquium/2010/carlo.pdf>. Zugriff:
28.04.2016.

Carnap 1931

Carnap, Rudolf. 1931.
Die physikalische Sprache als Universalsprache der
Wissenschaft.
Erkenntnis **2** (1), S. 432-465.
DOI: 10.1007/BF02028172

Carnap 1934

Carnap, Rudolf. 1934.
Logische Syntax der Sprache.
Wien: Springer-Verlag.
ISBN 9783662253755.

Caron et al. 1998

Caron, Franco, Marchet, Gino & Perego, Alessandro.
1998.
Routing policies and COI-based storage policies in
picker-to-part systems.
International Journal of Production Research **36** (3),
S. 713-732.
DOI: 10.1080/002075498193651

Chan et al. 2011

Chan, Felix T.S. & Chan, Hing K. 2011.
Improving the productivity of order picking of a man-
ual-pick and multi-level rack distribution warehouse
through the implementation of class-based storage.
Expert Systems with Applications **38** (3), S. 2686-
2700.
DOI: 10.1016/j.eswa.2010.08.058

Charon et al. 1993

Charon, Irène & Hudry, Olivier. 1993.
The noising method: a new method for combinatorial
optimization.
Operations Research Letters **14** (3), S. 133-137.
DOI: 10.1016/0167-6377(93)90023-A

- Chatfield et al. 1988** Chatfield, Chris & Yar, Mohammad. 1988. Holt-Winters Forecasting: Some Practical Issues. *Journal of the Royal Statistical Society. Series D (The Statistician)* **37** (2), S. 129-140.
DOI: 10.2307/2348687
- Chen et al. 2005** Chen, Mu-Chen & Wu, Hsiao-Pin. 2005. An association-based clustering approach to order batching considering customer demand patterns. *Omega* **33** (4), S. 333-343.
DOI: 10.1016/j.omega.2004.05.003
- Chen et al. 2011** Chen, Lu, Langevin, André & Riopel, Diane. 2011. A tabu search algorithm for the relocation problem in a warehousing system. *International Journal of Production Economics* **129** (1), S. 147-156.
DOI: 10.1016/j.ijpe.2010.09.012
- Chen et al. 2013** Chen, Fangyu, Wang, Hongwei, Qi, Chao & Xie, Yong. 2013. An ant colony optimization routing algorithm for two order pickers with congestion considerations. *Computers & Industrial Engineering* **66** (1), S. 77-85.
DOI: 10.1016/j.cie.2013.06.013
- Chen et al. 2015** Chen, Tzu-Li, Cheng, Chen-Yang, Chen, Yin-Yann & Chan, Li-Kai. 2015. An efficient hybrid algorithm for integrated order batching, sequencing and routing problem. *International Journal of Production Economics* **159**, S. 158-167.
DOI: 10.1016/j.ijpe.2014.09.029
- Cheng et al. 2015** Cheng, Chen-Yang, Chen, Yin-Yann, Chen, Tzu-Li & Yoo, John Jung-Woon. 2015. Using a hybrid approach based on the particle swarm optimization and ant colony optimization to solve a joint order batching and picker routing problem. *International Journal of Production Economics* **170**, S. 805-814.
DOI: 10.1016/j.ijpe.2015.03.021
- Chiang et al. 2011** Chiang, David Ming-Huang, Lin, Chia-Ping & Chen, Mu-Chen. 2011. The adaptive approach for storage assignment by mining data of warehouse management system for distribution centres. *Enterprise Information Systems* **5** (2), S. 219-234.
DOI: 10.1080/17517575.2010.537784

- Choy et al. 2014** Choy, K.L., Ho, G.T.S. & Lee, C.K.H. 2014.
A RFID-based storage assignment system for enhancing the efficiency of order picking.
Journal of Intelligent Manufacturing **28** (1), S. 111-129.
DOI: 10.1007/s10845-014-0965-9
- Christofides et al. 1973** Christofides, Nicos & Colloff, I. 1973.
The Rearrangement of Items in a Warehouse.
Operations Research **21** (2), S. 577-589.
DOI: 10.1287/opre.21.2.577
- Clarke et al. 1964** Clarke, Geoff & Wright John W. 1964.
Scheduling of Vehicles from a Central Depot to a Number of Delivery Points.
Operations Research **12** (4), S. 568-581.
DOI: 10.1287/opre.12.4.568
- Cornuéjols et al. 1985** Cornuéjols, Gérard, Fonlupt, Jean & Naddef, Denis. 1985.
The traveling salesman problem on a graph and some related integer polyhedra.
Mathematical Programming **33** (1), S. 1-27.
DOI: 10.1007/BF01582008
- Coyle et al. 2003** Coyle, John J., Bardi, Edward, J. & Langley Jr., C. John. 2003.
The Management of Business Logistics. A Supply Chain Perspective.
7. Aufl.
Mason: South-Western College Pub.
ISBN 9780324007510.
- Das et al. 2008** Das, Swagatam, Abraham, Ajith & Konar, Amit. 2008.
Swarm Intelligence Algorithms in Bioinformatics.
In: Kelemen, Arpad, Abraham, Ajith & Chen, Yuehui (Hrsg.): *Computational Intelligence in Bioinformatics*.
Berlin, Heidelberg: Springer-Verlag, S. 113-147.
ISBN 9783540768036.
DOI: 10.1007/978-3-540-76803-6_4
- de Koster et al. 1998** de Koster, René & van der Poort, Edo. 1998.
Routing order pickers in a warehouse: a comparison between optimal and heuristic solutions.
IEE Transactions **30** (5), S. 469-480.
DOI: 10.1080/07408179808966487
- de Koster et al. 1999** de Koster, René de; van der Poort, E.S. & Wolters, M. 1999.
Efficient orderbatching methods in warehouses.
International Journal of Production Research **37** (7),

- S. 1479-1504.
DOI: 10.1080/002075499191094
- de Koster et al. 2007**
de Koster, René de; Le-Duc, Tho & Roodbergen, Kees Jan. 2007.
Design and control of warehouse order picking: A literature review.
European Journal of Operational Research **182** (2), S. 481-501.
DOI: 10.1016/j.ejor.2006.07.009
- Domschke et al. 2015**
Domschke, W., Drexl, A., Klein, R. & Scholl, A. 2015.
Einführung in das Operations Research.
9., überarb. u. verb. Aufl.
Berlin, Heidelberg: Springer Gabler.
ISBN 9783662482162.
- Dorigo 1996**
Dorigo, Marco. 1996.
Ant System: Optimization by a Colony of Cooperating Agents.
IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, Part B (Cybernetics) **26** (1), S. 29-41.
DOI: 10.1109/3477.484436
- Elsayed et al. 1989**
Elsayed, E. A. & Unal, O. I. 1989.
Order batching algorithms and travel-time estimation for automated storage/retrieval systems.
International Journal of Production Research **27** (7), S. 1097-1114.
DOI: 10.1080/00207548908942610
- Ene et al. 2012**
Ene, Seval & Öztürk, Nursel. 2012.
Storage location assignment and order picking optimization in the automotive industry.
International Journal of Advanced Manufacturing Technology **60** (5-8), S. 787-797.
DOI: 10.1007/s00170-011-3593-y
- Englert et al. 2007**
Englert, Matthias, Röglin, Heiko & Vöcking, Berthold. 2007.
Worst case and probabilistic analysis of the 2-Opt algorithm for the TSP.
Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics, New Orleans, Louisiana, 7. - 9. Januar 2007, S. 1295-1304.
ISBN 9780898716245.
- Erlach 2010**
Erlach, Klaus. 2010.
Wertstromdesign - Der Weg zur schlanken Fabrik.
2., bearb. u. erw. Aufl.

Heidelberg: Springer.
ISBN 9783540898665.

Farmer et al. 1986

Farmer, J. Doyne, Packard, Norman H. & Perelson, Alan S. 1986.
The immune system, adaptation, and machine learning.
Physica D: Nonlinear Phenomena **22** (1-3), S. 187-204.
DOI: 10.1016/0167-2789(86)90240-X

Feo et al. 1989

Feo, Thomas A. & Resende, Mauricio, G.C. 1989.
A probabilistic heuristic for a computationally difficult set covering problem.
Operations Research Letters **8** (2), S. 67-71.
DOI: 10.1016/0167-6377(89)90002-3

Fogel et al. 1966

Fogel, Lawrence Jerome, Owens, Alvin J. & Walsh, Michael John 1966.
Artificial Intelligence Through Simulated Evolution.
New York: John Wiley & Sons.
ISBN 9780471265160.

Franses 1998

Franses, Philip Hans. 1998.
Time series models for business and economic forecasting.
Cambridge: Cambridge University Press.
ISBN 9780521586412.

Frazelle 2002

Frazelle, Edward. 2002.
World-Class Warehousing and Material Handling.
New York: McGraw-Hill.
ISBN 9780071785594.

Frerichs et al. 1980

Frerichs, Walter & Kübler, Knut. 1980.
Gesamtwirtschaftliche Prognoseverfahren.
München: Vahlen.
ISBN 3800608235.

Gademann et al. 2005

Gademann, Noud & van de Velde, Steef. 2005.
Order batching to minimize total travel time in a parallel-aisle warehouse.
IIE Transactions **37** (1), S. 63-75.
DOI: 10.1080/07408170590516917

Garfinkel 2005

Garfinkel, Maurice. 2005.
Minimizing Multi-zone Orders in the Correlated Storage Assignment Problem.
Atlanta, Univ. Diss., 2005.

Gerdes et al. 2004

Gerdes, Ingrid, Klawonn, Frank & Kruse, Rudolf. 2004.
Evolutionäre Algorithmen: Genetische Algorithmen,

- Strategien und Optimierungsverfahren, Beispielanwendungen.*
Wiesbaden: Vieweg.
ISBN 9783528055707.
- Gibson et al. 1992** Gibson, David R. & Sharp, Gunter P. 1992.
Order batching procedures.
European Journal of Operational Research **58** (1),
S. 57-67.
DOI: 10.1016/0377-2217(92)90235-2
- Glaser 1986** Glaser, Horst. 1986.
Material- und Produktionswirtschaft.
3. Aufl.
Düsseldorf: VDI-Verlag.
ISBN 3184007359.
- Glock et al. 2011** Glock, Christoph H. & Grosse, Eric H. 2011.
Lagerplatzvergabe in U-förmigen Kommissionierlagern.
Zeitschrift für wirtschaftlichen Fabrikbetrieb **106** (1-2),
S. 65-68.
DOI: 10.3139/104.110510
- Glover 1986** Glover, Fred. 1986.
Future Paths for Integer Programming and Links to Artificial Intelligence.
Computers & operations research **13** (5), S. 533-549.
DOI: 10.1016/0305-0548(86)90048-1
- Glover 1998** Glover, Fred. 1998.
A template for scatter search and path relinking. In:
Hao, J.K., Lutton, E., Ronald, E., Schoenauer, M. & Snyers, D. (Hrsg.): *Artificial Evolution. Lecture Notes in Computer Science.*
Berlin, Heidelberg: Springer.
ISBN 9783540641698.
DOI: 10.1007/BFb0026589
- Glover et al. 1986** Glover, Fred & McMillan, Claude. 1986.
The general employee scheduling problem: An integration of MS and AI.
Computers & Operations Research **13** (5), S. 563-573.
DOI: 10.1016/0305-0548(86)90050-X
- Glover et al. 1993** Glover, Fred, Taillard, Eric & Werra, Dominique de. 1993.
A user's guide to tabu search.
Annals of Operations Research **41** (1), S. 3-28.
DOI: 10.1007/BF02078647

- Götze 2000** Götze, Wolfgang. 2000.
Grafische und empirische Techniken des Business-Forecasting.
München, Wien: Oldenbourg.
ISBN 9783486255140.
- Grosse et al. 2013** Grosse, Eric H., Glock, Christoph H. & Jaber, Mohammad Y. 2013.
The effect of worker learning and forgetting on storage reassignment decisions in order picking systems.
Computers & Industrial Engineering **66**, S. 653-662.
DOI: 10.1016/j.cie.2013.09.013
- Grosse et al. 2014** Grosse, Eric H., Glock, Christoph H. & Ballester-Ripoll, Rafael. 2014.
A Simulated Annealing Approach for the Joint Order Batching and Order Picker Routing Problem with Weight Restrictions.
International Journal of Operations and Quantitative Management **20** (2), S. 65-83.
- Gu et al. 2007** Gu, Jinxiang, Goetschalckx, Marc & McGinnis, Leon F. 2007.
Research on warehouse operation: A comprehensive review.
European Journal of Operational Research **177** (1), S. 1-21.
DOI: 10.1016/j.ejor.2006.02.025
- Gu et al. 2010** Gu, Jinxiang, Goetschalckx, Marc & McGinnis, Leon F. 2010.
Research on warehouse design and performance evaluation: A comprehensive review.
European Journal of Operational Research **203** (3), S. 539-549.
DOI: 10.1016/j.ejor.2009.07.031
- Gudehus 2012** Gudehus, Timm. 2012.
Logistik 2. Netzwerke, Systeme und Lieferketten.
4. Aufl.
Berlin, Heidelberg: Springer-Verlag.
ISBN 9783642293764.
- Hansmann 1983** Hansmann, Karl-Werner. 1983.
Kurzlehrbuch Prognoseverfahren.
Wiesbaden: Gabler Verlag.
ISBN 9783409134446.
- Harrison 1965** Harrison, P.J. 1965.
Short-Term Sales Forecasting.

- Journal of the Royal Statistical Society. Series C (Applied Statistics)* **14** (2/3), S. 102-139.
DOI: 10.2307/2985338
- Hembecker et al. 2007**
Hembecker, Fernanda, Lopes, Heitor, S. & Godoy, Walter Jr. 2007.
Particle Swarm Optimization for the Multidimensional Knapsack Problem, International Conference on Adaptive and Natural Computing Algorithms, Warschau, 11.-14. April 2007.
Berlin, Heidelberg: Springer Verlag, S. 358-365.
ISBN 9783540716181.
DOI: 10.1007/978-3-540-71618-1_40
- Henn et al. 2010**
Henn, Sebastian, Koch, Sören, Dörner, Karl F., Strauss, Christine & Wäscher, Gerhard. 2010.
Metaheuristics for the Order Batching Problem in Manual Order Picking Systems.
Business Research **3** (1), S. 82-105.
DOI: 10.1007/BF03342717
- Henn et al. 2011**
Henn, Sebastian, Koch, Sören & Wäscher, Gerhard. 2011.
Order Batching in Order Picking Warehouses: A Survey of Solution Approaches.
Magdeburg, Univ., Working Paper Series, 1, S. 1-30.
Verfügbar: http://www.fww.ovgu.de/fww_media/femm/femm_2011/2011_01-EGOTEC-phrn72tev8ic4js28lkrko7ah5fgf0l5.pdf. Zugriff: 08.03.2016.
- Henn et al. 2012**
Henn, Sebastian & Wäscher, Gerhard. 2012.
Tabu search heuristics for the order batching problem in manual order picking systems.
European Journal of Operational Research **222** (3), S. 484-494.
DOI: 10.1016/j.ejor.2012.05.049
- Hillis 1990**
Hillis, Daniel W. 1990.
Co-evolving parasites improve simulated evolution as an optimization procedure.
Physica D: Nonlinear Phenomena **42** (1-3), S. 228-234.
DOI: 10.1016/0167-2789(90)90076-2
- Ho et al. 2006**
Ho, Y.-C., Tseng, Y.-Y. 2006.
A study on order-batching methods of order-picking in a distribution centre with two cross-aisles.
International Journal of Production Research **44** (17), S. 3391-3417.
DOI: 10.1080/00207540600558015

- Ho et al. 2008** Ho, Ying-Chin, Su, Teng-Sheng & Shi, Zhi-Bin. 2008. Order-batching methods for an order-picking warehouse with two cross aisles. *Computers & Industrial Engineering* **55** (2), S. 321-347.
DOI: 10.1016/j.cie.2007.12.018
- Holland 1975** Holland, John H. 1975. *Adaption in natural and artificial systems*. Ann Arbor: Univ. of Michigan Press. ISBN 9780472084609.
- Hou et al. 2010** Hou, Jiang-Liang, Yu-Jen Wu & Ya-Jung Yang. 2010. A model for storage arrangement and re-allocation for storage management operations. *International Journal of Computer Integrated Manufacturing* **23** (4), S. 369-390.
DOI: 10.1080/09511921003642154
- Hsieh et al. 2011** Hsieh, Ling-Feng & Huang, Yi-Chen. 2011. New batch construction heuristics to optimise the performance of order picking systems. *International Journal of Production Economics* **131** (2), S. 618-630.
DOI: 10.1016/j.ijpe.2011.02.006
- Hsu et al. 2005** Hsu, Chih-Ming, Chen, Kai-Ying, Chen, Mu-Chen. 2005. Batching orders in warehouses by minimizing travel distance with genetic algorithms. *Computers in Industry* **56** (2), S. 169-178.
DOI: 10.1016/j.compind.2004.06.001
- Hüttner 1986** Hüttner, Manfred. 1986. *Prognoseverfahren und ihre Anwendung*. Berlin, New York: de Gruyter. ISBN 3110108267.
- Jäger et al. 2013** Jäger, Jens Michael, Kluth, Andreas, Sauer, Marcus & Schatz, Anja. 2013. Komplexitätsbewirtschaftung. Die neue Managementdisziplin in Produktion und Supply Chain. *Zeitschrift für wirtschaftlichen Fabrikbetrieb* **108** (5), S. 341-343.
DOI: 10.3139/104.110942
- Jaikumar et al. 1990** Jaikumar, Ramchandran & Solomon, Marius M. 1990. Dynamic Operational Policies in an Automated Warehouse. *IIE Transactions* **22** (4), S. 370-376.
DOI: 10.1080/07408179008964191

- Jarvis et al. 1991** Jarvis, Jay M. & McDowell, Edward D. 1991. Optimal Product Layout in an Order Picking Warehouse. *IIE Transactions* **23** (1), S. 93-102. DOI: 10.1080/07408179108963844
- Kallrath 2013** Kallrath, Josef. 2013. *Gemischt-ganzzahlige Optimierung: Modellierung in der Praxis. Mit Fallstudien aus Chemie, Energiewirtschaft, Papierindustrie, Metallgewerbe, Produktion und Logistik*. 2., überarb. und erw. Aufl. Wiesbaden: Springer Spektrum. ISBN 9783658006907.
- Karaboga 2005** Karaboga, Dervis. 2005. An idea based on honey bee swarm for numerical optimization. *Technical Report - TR06, Erciyes University*, S. 1-10. Verfügbar: <http://www-lia.deis.unibo.it/Courses/SistInt/articoli/bee-colony1.pdf>. Zugriff: 20.06.2016.
- Kashan et al. 2013** Kashan, Ali Husseinzadeh, Kashan, Mina Husseinzadeh & Karimiyan, Somayyeh. 2013. A particle swarm optimizer for grouping problems. *Information Science* **252**, S. 81-95. DOI: 10.1016/j.ins.2012.10.036
- Kennedy et al. 1995** Kennedy, James & Russell, Eberhart. 1995. Particle swarm optimization. *Proceedings of 1995 IEEE International Conference on Neural Networks, Perth, 27. November - 1. Dezember 1995*, S. 1942-1948.
- Kirkpatrick et al. 1983** Kirkpatrick, Scott, Gelatt, C. Daniel & Vecchi, Mario P. 1983. Optimization by Simulated Annealing. *Science* **220** (4598), S. 671-680. DOI: 10.1126/science.220.4598.671
- Koch 2012** Koch, Susanne. 2012. *Logistik. Eine Einführung in Ökonomie und Nachhaltigkeit*. Berlin, Heidelberg: Springer Vieweg. ISBN 9783642152894.
- Koch 2014** Koch, Sören. 2014. *Genetische Algorithmen für das Order Batching-Problem in manuellen Kommissioniersystemen*. Wiesbaden: Springer Gabler. ISBN 9783658053451. Magdeburg, Univ. Diss., 2013.

- Koch et al. 2016** Koch, Sören, Wäscher, Gerhard. 2016.
A grouping genetic algorithm for the Order Batching Problem in distribution warehouses.
Journal of Business Economics **86** (1-2), S. 131-153.
DOI: 10.1007/s11573-015-0789-x
- Kofler 2015** Kofler, Monika. 2015.
Optimising the storage location assignment problem under dynamic conditions.
Linz, Univ. Diss., 2014.
- Kofler et al. 2010** Kofler, Monika, Beham, Andreas, Wagner, Stefan, Affenzeller, Michael & Reitingner, Clemens. 2010.
Reassigning Storage Locations in a Warehouse to Optimize the Order Picking Process.
Proceedings of the 22th European Modeling and Simulation Symposium EMSS, Fes, 13.-15. Oktober 2010.
Verfügbar: https://www.researchgate.net/profile/Monika_Kofler2/publication/230701443_Reassigning_Storage_Locations_in_a_Warehouse_to_Optimize_the_Order_Picking_Process/links/55b6a9b308ae9289a08bbfa1.pdf. Zugriff: 28.04.2016.
- Kofler et al. 2011** Kofler, Monika, Beham, Andreas, Wagner, Stefan, Affenzeller, Michael & Achleitner, Werner 2011.
Re-Warehousing vs. Healing: Strategies for Warehouse Storage Location Assignment.
3rd IEEE International Symposium on Logistics and Industrial Informatics, Budapest, 25.-27. August 2011, S. 77-82.
DOI: 10.1109/LINDI.2011.6031124
- Kofler et al. 2015** Kofler, Monika, Beham, Andreas, Wagner, Stefan & Affenzeller, Michael. 2015.
Robust Storage Assignment in Warehouses with Correlated Demand.
In: Borowik, G., Chaczko, Z., Jacak, W. & Łuba, T. (Hrsg.): *Computational Intelligence and Efficiency in Engineering Systems.*
Heidelberg, New York, Dordrecht, London: Springer-Verlag, S. 415-428.
ISBN 9783319157207.
DOI: 10.1007/978-3-319-15720-7_29
- Koza 1992** Koza, John R. 1992.
Genetic Programming - On the Programming of Computers by Means of Natural Selection.
Cambridge: MIT Press.
ISBN 9780262527910.

- Kubicek 1977**
Kubicek, H. 1977.
Heuristische Bezugsrahmen und heuristisch angelegte Forschungsdesigns als Elemente einer Konstruktionsstrategie empirischer Forschung. In: Köhler, R. (Hrsg.): *Empirische und handlungstheoretische Forschungskonzeptionen in der Betriebswirtschaftslehre*. Stuttgart: Poeschel, S. 1-37.
ISBN 3791002147.
- Kühne et al. 2011**
Kühne, Annegret & Wenger, Wolf. 2011.
Mengenprognose mit dem Holt-Winters-Verfahren am Beispiel des monatlichen Energiebedarfs von Industrieunternehmen. In: Wenger, Wolf, Geiger, Martin Josef & Kleine, Andreas (Hrsg.): *Business Excellence in Produktion und Logistik. Festschrift für Prof. Dr. Walter Habenicht*. Wiesbaden: Springer Fachmedien, S. 317-336.
ISBN 9783834966889.
DOI: 10.1007/978-3-8349-6688-9_17
- Kulak et al. 2012**
Kulak, Osman, Sahin, Yusuf & Taner, Mustafa Ege-men. 2012.
Joint order batching and picker routing in single and multiple-cross-aisle warehouses using cluster-based tabu search algorithms.
Flexible Services and Manufacturing Journal **24** (1), S. 52-80.
DOI: 10.1007/s10696-011-9101-8
- Lam et al. 2007**
Lam, Hoang Thanh, Nicolaevna, Popova Nina & Quan, Nguyen Thoi Minh. 2007.
A heuristic particle swarm optimization.
Proceedings of the 9th annual conference on Genetic and evolutionary computation, ACM, London, 7.-11. Juli 2007.
DOI: 10.1145/1276958.1276988
- Le-Duc et al. 2005**
Le-Duc, Tho & de Koster, René. 2005.
Travel distance estimation and storage zone optimization in a 2-block class-based storage strategy warehouse.
International Journal of Production Research **43** (17), S. 3561-3581.
DOI: 10.1080/00207540500142894
- Le-Duc et al. 2007**
Le-Duc, Tho & de Koster, René. 2007.
Travel time estimation and order batching in a 2-block warehouse.
European Journal of Operational Research **176** (1), S. 374-388.
DOI: 10.1016/j.ejor.2005.03.052

- Lee 2007** Lee, Tsung-Ying. 2007. Optimal spinning reserve for a wind-thermal power system using EIPSO. *IEEE Transactions on Power Systems* **22** (4), S. 1612-1621. DOI: 10.1109/TPWRS.2007.907519
- Li et al. 2016** Li, Jiayi, Moghaddam, Mohsen & Nof, Shimon Y. 2016. Dynamic storage assignment with product affinity and ABC classification - a case study. *The International Journal of Advanced Manufacturing Technology* **84** (9-12), S. 2179-2194. DOI: 10.1007/s00170-015-7806-7
- Lian et al. 2008** Lian, Zhigang, Gu, Xingsheng & Jiao, Bin. 2008. A novel particle swarm optimization algorithm for permutation flow-shop scheduling to minimize makespan. *Chaos, Solitons & Fractals* **35** (5), S. 851-861. DOI: 10.1016/j.chaos.2006.05.082
- Lin et al. 2016** Lin, Chun-Cheng, Kang, Jia-Rong, Hou, Chung-Chih & Cheng, Chen-Yang. 2016. Joint order batching and picker Manhattan routing problem. *Computers & Industrial Engineering* **95**, S. 164-174. DOI: 10.1016/j.cie.2016.03.009
- Liu 2004** Liu, Chiun-Ming. 2004. Optimal Storage Layout and Order Picking for Warehousing. *International Journal of Operations Research* **1** (1), S. 37-46. DOI: 10.1.1.402.1087
- Liu et al. 2008** Liu, D.S., Tan, K.C., Huang, S.Y., Goh, C.K. & Ho, W.K. 2008. On solving multiobjective bin packing problems using evolutionary particle swarm optimization. *European Journal of Operational Research* **190** (2), 357-382. DOI: 10.1016/j.ejor.2007.06.032
- Makridakis et al. 1980** Makridakis, Spyros, Reschke, Hasso & Wheelwright, Steven C. 1980. *Prognosetechniken für Manager*. Wiesbaden: Springer Fachmedien. ISBN 9783409960816.
- März et al. 2011** März, Lothar & Weigert, Gerald. 2011. Simulationsgestützte Optimierung.

- In: März, L., Krug, W., Rose, O. & Weigert, G. (Hrsg.): *Simulation und Optimierung in Produktion und Logistik*. Heidelberg, Dordrecht, London, New York: Springer, S. 3-12.
ISBN 9783642145360.
DOI: 10.1007/978-3-642-14536-0_1
- Mantel et al. 2007**
Mantel, Ronald J., Schuur, Peter C. & Heragu, Sunderesh S. 2007.
Order oriented slotting: a new assignment strategy for warehouses.
European Journal of Industrial Engineering **1** (3), S. 301-316.
DOI: 10.1504/EJIE.2007.014689
- Manzini 2012**
Manzini, Riccardo. 2012.
Foreword.
In: Manzini, Riccardo (Hrsg.): *Warehousing in the Global Supply Chain. Advanced Models, Tools and Applications for Storage Systems*. London, Dordrecht, Heidelberg, New York: Springer-Verlag, S. vii-ix.
ISBN 9781447122746.
- Manzini et al. 2007**
Manzini, Riccardo, Gamberi, Mauro, Persona, Alessandro & Regattieri, Alberto. 2007.
Design of a class based storage picker to product order picking system.
The International Journal of Advanced Manufacturing Technology **32** (7-8), S. 811-821.
DOI: 10.1007/s00170-005-0377-2
- Manzini et al. 2015**
Manzini, R, Accorsi, R., Gamberi, M. & Penazzi, S. 2015.
Modeling class-based storage assignment over life cycle picking patterns.
International Journal of Production Economics **170**, S. 790-800.
DOI: 10.1016/j.ijpe.2015.06.026
- Martin 2014**
Martin, Heinrich. 2014.
Transport- und Lagerlogistik. Planung, Struktur, Steuerung und Kosten von Systemen der Intralogistik. Wiesbaden: Springer Vieweg.
ISBN 9783658031435.
- Matusiak et al. 2014**
Matusiak, Marek, de Koster, René, Kroon, Leo & Saarinen, Jari. 2014.
A fast simulated annealing method for batching precedence-constrained customer orders in a warehouse.
European Journal of Operational Research **236** (3),

S. 968-977.

DOI: 10.1016/j.ejor.2013.06.001

Mayer et al. 2009

Mayer, Stephan H., Thiry, Erik & Frank, Cay-Bernhard. 2009.

Supply chain excellence amidst the global economic crisis: 6th European A.T. Kearney/ELA logistics study 2008/2009.

Darmstadt: Technische Universität, Institut für Betriebswirtschaftslehre.

ISBN 9783924606596.

Meller et al. 2009

Meller, Russell D. & Gue, Kevin R. 2009.

The Application of New Aisle Designs for Unit-Load Warehouses, *Proceedings of the NSF CMMI Engineering Research and Innovation Conference*, Honolulu, 22.-25. Juni 2009.

Meneghetti et al. 2013

Meneghetti, Antonella & Monti, Luca. 2013.

Multiple-weight unit load storage assignment strategies for energy-efficient automated warehouses.

International Journal of Logistics Research and Applications **17** (4), S. 304-322.

DOI: 10.1080/13675567.2013.861896

Menéndez et al. 2017

Menéndez, Borja, Bustillo, Manuel, Pardo, Eduardo, G. & Duarte, A. 2017.

General Variable Neighborhood Search for the Order Batching and Sequencing Problem.

European Journal of Operational Research **263** (1), S. 82-93.

DOI: 10.1016/j.ejor.2017.05.001

Mertens et al. 2012

Mertens, Peter & Rässler, Susanne. 2012.

Prognoserechnung - Einführung und Überblick. In: Mertens, Peter & Rässler, Susanne (Hrsg.): *Prognoserechnung*.

7., wesentl. überarb. u. erw. Aufl.

Berlin, Heidelberg: Springer-Verlag, S. 3-10.

ISBN 9783790827972.

DOI: 10.1007/978-3-7908-2797-2_1

Mladenović et al. 1997

Mladenović, Nenad. & Hansen, Pierre. 1997.

Variable Neighborhood Search.

Computers & Operations Research **24** (11), S. 1097-1100.

DOI: 10.1016/S0305-0548(97)00031-2

Möller 2011

Möller, Klaus. 2011.

Increasing warehouse order picking performance by sequence optimization.

Procedia Social and Behavioral Sciences **20**, S. 177-

185.
DOI: 10.1016/j.sbspro.2011.08.023
- Moon et al. 2001**
Moon, G. & Kim, G.-P. 2001.
Effects of relocation to AS/RS storage location policy with production quantity variation.
Computers & Industrial Engineering **40** (1), S. 1-13.
DOI: 10.1016/S0360-8352(00)00005-X
- Mühlenbein et al. 1996**
Mühlenbein, H. & Paaß, G. 1996.
From recombination of genes to the estimation of distributions I. Binary parameters. In: Voigt, H.M., Ebeling, W., Rechenberg, I. & Schwefel, H.P. (Hrsg.): *Parallel Problem Solving from Nature*. Berlin, Heidelberg: Springer. ISBN 9783540706687.
DOI: 10.1007/3-540-61723-X_982
- Muppani et al. 2008**
Muppani, Venkata Reddy & Adil, Gajendra Kumar. 2008.
A branch and bound algorithm for class based storage location assignment.
European Journal of Operational Research **189** (2), S. 492-507.
DOI: 10.1016/j.ejor.2007.05.050
- Muralidharan et al. 1995**
Muralidharan, R., Linn, R. J. & Pandit R. 1995.
Shuffling heuristics for the storage location assignment in an AS/RS.
International Journal of Production Research **33** (6), S. 1661-1672.
DOI: 10.1080/00207549508930234
- Nave 2009**
Nave, Markus. 2009.
Einführung und Grundlagen.
In: Pulverich, Michael & Schietinger, Jörg (Hrsg.): *Handbuch Kommissionierung - Effizient Picken und Packen*. München: Heinrich Vogel, S. 16-29.
ISBN 9783574260940.
- Neuhäuser 2013**
Neuhäuser, Daniel. 2013.
Ein Ansatz zur simulationsgestützten Planung und Bewertung von Lagerreorganisationsmaßnahmen am Beispiel des Lebensmitteleinzelhandels. Aachen: Shaker. ISBN 9783844015935.
Stuttgart, Univ. Diss., 2012.
- Nickel et al. 2011**
Nickel, Stefan, Stein, Oliver & Waldmann, Karl-Heinz. 2011.
Operations Research.

Berlin, Heidelberg: Springer.
ISBN 9783642226236.

Nyhuis 2008

Nyhuis, Peter. 2008.
Entwicklungsschritte zu Theorien der Logistik.
In: Nyhuis, Peter (Hrsg.): *Beiträge zu einer Theorie der Logistik*.
Berlin, Heidelberg: Springer-Verlag, S. 1-17.
ISBN 9783540756415.

Önüt et al. 2008

Önüt, Semih, Tuzkaya, Umut R. & Doğaç, Bilgehan.
2008.
A particle swarm optimization algorithm for the multiple-level warehouse layout design problem.
Computers & Industrial Engineering **54** (4), S. 783-799.
DOI: 10.1016/j.cie.2007.10.012

Osman et al. 1996

Osman, Ibrahim H. & Laporte, Gilbert. 1996.
Metaheuristics: A bibliography.
Annals of Operations Research **63** (5), S. 511-623.
DOI: 10.1007/BF02125421

o.V. 2013

o.V. 2013.
Kompakt Lexikon Management. 2.000 Begriffe nachschlagen, verstehen, anwenden.
Wiesbaden: Springer Gabler.
ISBN 9783658030247.

Pan et al. 2009

Pan, Jason Chao-Hsien & Wu, Ming-Hung. 2009.
A study of storage assignment problem for an order picking line in a pick-and-pass warehousing system.
Computers & Industrial Engineering **57** (1), S. 261-268.
DOI: 10.1016/j.cie.2008.11.026

Pan et al. 2015

Pan, Jason Chao-Hsien, Shih, Po-Hsun, Wu, Ming-Hung & Lin, Jyh-Horng. 2015.
A storage assignment heuristic method based on genetic algorithm for a pick-and-pass warehousing system.
Computer & Industrial Engineering **81**, S. 1-13.
DOI: 10.1016/j.cie.2014.12.010

Pansart et al. 2018

Pansart, L., Catusse, N. & Cambazard, H. 2018.
Exact algorithms for the order picking problem.
Computers & Operations Research **100**, S. 117-127.
DOI: 10.1016/j.cor.2018.07.002

Passino 2002

Passino, Kevin M. 2002.
Biomimicry of Bacterial Foraging for Distributed Optimization and Control.

- IEEE control systems* **22** (3), S. 52-67.
DOI: 10.1109/MCS.2002.1004010
- Pazour et al. 2015**
Pazour, Jennifer A., Carlo, Héctor J. 2015.
Warehouse reshuffling: Insights and optimization.
Transportation Research Part E: Logistics and Transportation Review **73**, S. 207-226.
DOI: 10.1016/j.tre.2014.11.002
- Petersen 1999**
Petersen, Charles G. 1999.
The impact of routing and storage policies on warehouse efficiency.
International Journal of Operations & Production Management **19** (10), S. 1053-1064.
DOI: 10.1108/01443579910287073
- Petersen et al. 1999**
Petersen, Charles G. & Schmenner, Roger W. 1999.
An Evaluation of Routing and Volume-based Storage Policies in an Order Picking Operation.
Decision Sciences **30** (2), S. 481-501.
DOI: 10.1111/j.1540-5915.1999.tb01619.x
- Petersen et al. 2004a**
Petersen, Charles G. & Aase, Gerald. 2004.
A comparison of picking, storage, and routing policies in manual order picking.
International Journal of Production Economics **91** (1), S. 11-19.
DOI: 10.1016/j.ijpe.2003.09.006
- Petersen et al. 2004b**
Petersen, Charles G., Siu, Charles & Heiser, Daniel R. 2004.
Improving order picking performance utilizing slotting and golden zone storage.
International Journal of Operations & Production Management **25** (10), S. 997-1012.
DOI: 10.1108/01443570510619491
- Pfohl 2010**
Pfohl, Hans-Christian. 2010.
Logistiksysteme. Betriebswirtschaftliche Grundlagen.
8., neu bearb. u. aktual. Aufl.
Berlin, Heidelberg: Springer-Verlag.
ISBN 9783642041624.
- Pierre et al. 2003**
Pierre, Benjamin, Vannieuwenhuyse, Bart, Dominanta, Denis & van Dessel, Hendrik. 2003.
Dynamic ABC Storage Policy in Erratic Demand Environments.
Jurnal Teknik Industri **5** (1), S. 1-12.
- Quintanilla et al. 2015**
Quintanilla, Sacramento, Pérez, Ángeles, Ballestín, Francisco & Lino, Pilar 2015.
Heuristic algorithms for a storage location assignment problem in a chaotic warehouse.

- Engineering Optimization* **47** (10), S. 1405-1422.
DOI: 10.1080/0305215X.2014.969727
- Rao et al. 2013a** Rao, Subir S. & Adil, Gajendra K. 2013.
Optimal class boundaries, number of aisles, and pick list size for low-level order picking systems.
IIE Transactions **45** (12), S. 1309-1321.
DOI: 10.1080/0740817X.2013.772691
- Rao et al. 2013b** Rao, Subir S. & Adil, Gajendra K. 2013.
Class-based storage with exact S-shaped traversal routing in low-level picker-to-part systems.
International Journal of Production Research **52** (16), S. 4979-4996.
DOI: 10.1080/00207543.2013.784419
- Ratliff et al. 1983** Ratliff, H. Donald & Rosenthal Arnon S. 1983.
Order-Picking in a Rectangular Warehouse: A Solvable Case of the Traveling Salesman Problem.
Operations Research **31** (3), S. 507-521.
DOI: 10.1287/opre.31.3.507
- Rechenberg 1973** Rechenberg, Ingo. 1973.
Evolutionsstrategie - Optimierung technischer Systeme nach Prinzipien der biologischen Evolution.
Stuttgart-Bad Cannstatt: Friedrich Frommann Verlag.
ISBN 9783772803734.
- Reschke 2013** Reschke, Volker. 2013.
Lagerplatzvergabe in Person-zur-Ware-Kommissioniersystemen.
Aachen: Shaker Verlag.
Berichte aus der Logistik.
ISBN 9783844018189.
Magdeburg, Univ. Diss., 2012.
- Reynolds 1994** Reynolds, R.G. 1994.
An introduction to cultural algorithms. In: Sebalk, A.V. & Fogel, L.J. (Hrsg.): *Proceedings of the Third Annual conference on Evolutionary Programming, San Diego, 24.-26. Februar 1994*.
River Edge: World Scientific Publishing.
ISBN 9789810218102.
- Rinne et al. 2002** Rinne, Horst & Specht, Katja. 2002.
Zeitreihen: statistische Modellierung, Schätzung und Prognose.
München: Vahlen.
ISBN 3800628775.
- Rößler et al. 2016** Rößler, Irene & Ungerer, Albrecht. 2016.
Statistik für Wirtschaftswissenschaftler. Eine anwendungsorientierte Darstellung.

- 5., aktual. Aufl.
Berlin, Heidelberg: Springer Gabler.
ISBN 9783662484128.
- Roll 2003**
Roll, Martin. 2003.
Fallstudien als Instrument der Controllingforschung.
Ein Überblick zu ihrer Einordnung, Beurteilung und
Durchführung.
Controlling und Management **47** (5), S. 315-317.
DOI: 10.1007/BF03254197
- Roodbergen 2001**
Roodbergen, Kees Jan. 2001.
Layout and routing methods for warehouses.
Rotterdam: Selbstverlag.
ERIM PH.D. series Research in Management 4, TRAIL
Thesis Series; 2001/3.
ISBN 9058920054.
Rotterdam, Univ., Diss., 2001.
- Roodbergen et al. 2001a**
Roodbergen, Kees Jan & de Koster, René. 2001.
Routing methods for warehouses with multiple cross
aisles.
International Journal of Production Research **39** (9),
S. 1865-1883.
DOI: 10.1080/00207540110028128
- Roodbergen et al. 2001b**
Roodbergen, Kees Jan & de Koster, René. 2001.
Routing order pickers in a warehouse with a middle
aisle.
European Journal of Operational Research **133** (1),
S. 32-43.
DOI: 10.1016/S0377-2217(00)00177-6
- Rouwenhorst et al. 2000**
Rouwenhorst, B., Reuter, B., Stockrahm, G.J., van
Houtum, R.J. & Mantel, W.H.M. 2000.
Warehouse design and control: Framework and litera-
ture review.
European Journal of Operational Research **122** (3),
S. 515-533.
DOI: 10.1016/S0377-2217(99)00020-X
- Sadiq et al. 1996**
Sadiq, Malik, Landers, Thomas L. & Tylor, Don G.
1996.
An assignment algorithm for dynamic picking sys-
tems.
IIE transactions **28** (8), S. 607-616.
DOI: 10.1080/15458830.1996.11770706
- Sadri et al. 2006**
Sadri, Javad. & Suen, Ching Y. 2006.
A genetic binary particle swarm optimization model.
Evolutionary Computation, IEEE Congress, Vancouver,

16.-21. July 2006, S. 656-663.
DOI: 10.1109/CEC.2006.1688373

Schanz 1987

Schanz, Günther. 1987.
Wissenschaftstheoretische Grundfragen der Führungsforschung. In: Kieser, Alfred, Reber, Gerhard & Wunderer, Rolf. (Hrsg.): *Handwörterbuch der Führung - Enzyklopädie der Betriebswirtschaftslehre - Band 10*. Stuttgart: C.E. Poeschel Verlag.
ISBN 3791080288.

Scholl et al. 1997

Scholl, Armin; Klein, Robert & Jürgens, Christian. 1997.
Bison: A fast hybrid procedure for exactly solving the one-dimensional bin packing problem.
Computers & Operations Research **24** (7), S. 627-645.
DOI: 10.1016/S0305-0548(96)00082-2

Scholz et al. 2015

Scholz, André & Wäscher, Gerhard. 2015.
A Solution for the Joint Order Batching and Picker Routing Problem in a Two-Block Layout.
Magdeburg, Univ., Working Paper Series, 4.
Verfügbar: http://www.mansci.ovgu.de/mansci_media/publikationen/2012/FEMM+_+A+Solution+Approach+for+the+Joint+Order+Batching+and+Picker+Routing+Problem+in+a+Two_Block+Layout-p-1456.pdf. Zugriff: 08.03.2016.

Scholz et al. 2017a

Scholz, André & Wäscher, Gerhard. 2017.
Order Batching and Picker Routing in manual order picking systems: the benefits of integrated routing.
Central European Journal of Operations Research **25** (2), S. 491-520.
DOI: 10.1007/s10100-017-0467-x

Scholz et al. 2017b

Scholz, André, Schubert, Daniel & Wäscher, Gerhard. 2017.
Order picking with multiple pickers and due dates - Simultaneous solution of Order Batching, Batch Assignment and Sequencing, and Picker Routing Problems.
European Journal of Operational Research **263** (2), S. 461-478.
DOI: 10.1016/j.ejor.2017.04.038

Schröder 2012

Schröder, Michael. 2012.
Einführung in die kurzfristige Zeitreihenprognose und Vergleich der einzelnen Verfahren. In: Mertens, Peter & Rässler, Susanne (Hrsg.): *Prognoserechnung*. 7., wesentl. überarb. u. erw. Aufl.
Berlin, Heidelberg: Springer-Verlag, S. 11-45.

- ISBN 9783790827972.
DOI: 10.1007/978-3-7908-2797-2_2
- Schuhr 2012**
Schuhr, Roland. 2012.
Einführung in die Prognose saisonaler Zeitreihen mit Hilfe exponentieller Glättungstechniken und Vergleich der Verfahren von Holt/Winters und Harrison. In: Mertens, Peter & Rässler, Susanne (Hrsg.): *Prognoserechnung*.
7., wesentl. überarb. u. erw. Aufl.
Berlin, Heidelberg: Springer-Verlag, S. 47-73.
ISBN 9783790827972.
DOI: 10.1007/978-3-7908-2797-2_3
- Schwede et al. 2011**
Schwede, Christian, Klingebiel, Katja, Pauli, Thomas & Wagenitz, Axel. 2011.
Simulationsgestützte Optimierung für die distributionorientierte Auftragsreihenfolgeplanung in der Automobilindustrie.
In: März, L., Krug, W., Rose, O. & Weigert, G. (Hrsg.): *Simulation und Optimierung in Produktion und Logistik - Praxisorientierter Leitfaden mit Fallbeispielen*.
Heidelberg, Dordrecht, London, New York: Springer Verlag, S. 151-170.
ISBN 9783642145360.
DOI: 10.1007/978-3-642-14536-0_13
- Sedighizadeh et al. 2009**
Sedighizadeh, Davoud. & Masehian, Ellips. 2009.
Particle swarm optimization methods, taxonomy and applications.
International Journal of Computer Theory and Engineering **1** (5), S. 486-502.
- Silver et al. 2017**
Silver, Edward A., Pyke, David F. & Thomas, Douglas J. 2017.
Inventory and Production Management in Supply Chains.
4. Aufl.
Boca Raton: CRC Press.
ISBN 9781466558618.
- Simon 2008**
Simon, Dan. 2008.
Biogeography-Based Optimization.
IEEE Transactions on Evolutionary Computation **12** (6), S. 702-713.
DOI: 10.1109/TEVC.2008.919004
- Sooksaksun et al. 2012**
Sooksaksun, Natanaree & Kachitvichyanukul, Voratas. 2012.
A class-based storage warehouse design using a particle swarm optimisation algorithm.
International Journal of Operational Research **13** (2),

S. 219-237.

DOI: 10.1504/IJOR.2012.045188

Shqair et al. 2014

Shqair, Maram, Altarazi, Safwan & Al-Shihabi, Sameh 2014.

A statistical study employing agent-based modeling to estimate the effects of different warehouse parameters on the distance traveled in warehouses.

Simulation Modelling Practice and Theory **49**, S. 122-135.

DOI: 10.1016/j.simpat.2014.08.002

Stadtler 2007

Stadtler, Helmut. 2007.

Laudatio auf Prof. Dr. Wolfgang Domschke.

Gesellschaft für Operations Research-Jahrestagung, 5.-7. September, Saarbrücken.

Verfügbar: <http://www.gor-ev.de/auszeichnungen-der-gor/wissenschaftspreis-der-gor/laudatio-auf-prof-dr-wolfgang-domschke>. Zugriff: 18.12.2017.

Storn et al. 1997

Storn, Rainer & Price, Kenneth. 1997.

Differential Evolution - A Simple and Efficient Heuristic for global Optimization over Continuous Spaces.

Journal of Global Optimization **11** (4), S. 341-359.

DOI: 10.1023/A:1008202821328

Stützle 1998

Stützle, Thomas G. 1998.

Local Search Algorithms for Combinatorial Problems - Analysis, Improvements and New Applications.

Darmstadt, Univ. Diss, 1998.

Suhl et al. 2013

Suhl, Leena & Mellouli, Taieb. 2013.

Optimierungssysteme - Modelle, Verfahren, Software, Anwendungen.

3., korr. u. aktual. Aufl.

Berlin, Heidelberg: Springer Gabler.

ISBN 9783642389375.

Talbi 2009

Talbi, El-Ghazali. 2009.

Metaheuristics - From design to implementation.

Hoboken: John Wiley & Sons.

ISBN 9780470278581.

Teece 1998

Teece, David J. 1998.

Capturing Value from Knowledge Assets: The New Economy, Markets for Know-how, and Intangible Assets.

California Management Review **40** (3), S. 55-79.

DOI: 10.2307/41165943

Teece et al. 1997

Teece, David J., Pisano, Gary & Shuen, Amy. 1997.

Dynamic Capabilities and Strategic Management.

Strategic Management Journal **18** (7), S. 509-533.

- Tempelmeier 2006** Tempelmeier, Horst. 2006.
Material-Logistik. Modelle und Algorithmen für die Produktionsplanung und -steuerung in Advanced Planning-Systemen.
6., Aufl.
Berlin, Heidelberg: Springer-Verlag.
ISBN 9783540284277.
- ten Hompel et al. 2010** ten Hompel, Michael & Schmidt, Thorsten. 2010.
Warehouse Management. Organisation und Steuerung von Lager- und Kommissioniersystemen.
4., neu bearb. Aufl.
ISBN 9783642031854.
- ten Hompel et al. 2011** ten Hompel, Michael, Sadowsky, Volker & Beck, Maria. 2011.
Kommissionierung. Materialflusssysteme 2 - Planung und Berechnung der Kommissionierung in der Logistik.
Berlin, Heidelberg: Springer-Verlag.
ISBN 9783540299400.
- Theys et al. 2010** Theys, Christophe, Bräysy, Olli, Dullaert, Wout & Raa, Birger. 2010.
Using a TSP heuristic for routing order pickers in warehouses.
European Journal of Operational Research **200** (3), S. 755-763.
DOI: 10.1016/j.ejor.2009.01.036
- Thommen 2008** Thommen, Jean-Paul. 2008.
Lexikon der Betriebswirtschaft. Managementkompetenz von A bis Z.
4., überarb. u. erw. Aufl.
Zürich: Versus Verlag.
ISBN 9783039090501.
- Tian et al. 2013** Tian, Ye., Liu, Dayou, Yuan, Donghiu & Wang, Kunhao. 2013.
A discrete PSO for two-stage assembly scheduling problem.
The International Journal of Advanced Manufacturing Technology **66** (1-4), S. 481-499.
DOI: 10.1007/s00170-012-4343-5
- Tomczak 1992** Tomczak, Torsten. 1992.
Forschungsmethoden in der Marketingwissenschaft - Ein Plädoyer für den qualitativen Forschungsansatz.
Marketing ZfP **14** (2), S. 77-87.
DOI: 10.15358/0344-1369-1992-2-77

- Tompkins et al. 2010** Tompkins, James A., White, John A., Bozer, Yavuz A. & Tanchoco, J.M.A. 2010.
Facilities Planning.
4. Aufl.
Hoboken: John Wiley & Sons.
ISBN 9780470444047.
- Tsai et al. 2008** Tsai, C.-Y., Liou, J.J.H. & Huang, T.-M. 2008.
Using a multiple-GA method to solve the batch picking problem: considering travel distance and order due time.
International Journal of Production Research **46** (22),
S. 6533-6555.
DOI: 10.1080/00207540701441947
- Tsamis et al. 2015** Tsamis, Nikolaos, Giannikas, Vaggelis, McFarlane, Duncan, Lu, Wenrong & Strachan, James. 2015.
Adaptive Storage Location Assignment for Warehouses Using Intelligent Products.
In: Borangiu, Theodor, Thomas, André & Trentesaux, Damien (Hrsg.): *Service Orientation in Holonic and Multi-agent Manufacturing*.
Heidelberg, New York, Dordrecht, London: Springer,
S. 271-279.
ISBN 9783319151595.
DOI: 10.1007/978-3-319-15159-5_25
- Tyssen 2012** Tyssen, Matthias. 2012.
Zukunftsorientierung und dynamische Fähigkeiten. Corporate Foresight in Unternehmen der Investitionsgüterindustrie.
Wiesbaden: Springer Fachmedien.
ISBN 9783834938961.
Wiesbaden, Univ. Diss, 2011.
- Ulrich 1984** Ulrich, Hans. 1984.
Die Betriebswirtschaftslehre als anwendungsorientierte Sozialwissenschaft. In: Ulrich, Hans, Dyllick, Thomas & Probst, Gilbert J.B. (Hrsg.): *Management*.
Bern: Haupt, S. 168-199.
ISBN 9783258034461.
- Ulrich 2001** Ulrich, Hans. 2001.
Systemorientiertes Management: Das Werk von Hans Ulrich.
Bern: Haupt.
ISBN 9783258063591.
- Ulrich et al. 1976** Ulrich, Peter & Hill, Wilhelm. 1976.
Wissenschaftstheoretische Grundlagen der Betriebswirtschaftslehre (Teil 1).

- Wirtschaftswissenschaftliches Studium: Zeitschrift für Ausbildung und Hochschulkontakt* **5** (7), S. 304-309.
- Valle et al. 2017** Valle, Christiano Arbex; Beasley, John E. & Cunha, Alexandre Salles da. Optimally solving the joint order batching and picker routing problem. *European Journal of Operational Research* **262** (3), S. 817-834.
DOI: 10.1016/j.ejor.2017.03.069
- van den Berg 1999** van den Berg, Jeroen P. 1999. A literature survey on planning and control of warehousing systems. *IIE Transactions* **31** (8), S. 751-762.
DOI: 10.1023/A:1007606228790
- van Gils et al. 2017** van Gils, Teun, Ramaekers, Katrien, Caris, An & de Koster, René. 2017. Designing Efficient Order Picking Systems by Combining Planning Problems: State-of-the-art Classification and Review. *European Journal of Operational Research* (2017), S. 1-34.
DOI: 10.1016/j.ejor.2017.09.002
- van Nieuwenhuysse et al. 2009** van Nieuwenhuysse, Inneke & de Koster, René. 2009. Evaluating order throughput time in 2-block warehouses with time window batching. *International Journal of Production Economics* **121** (2), S. 654-664.
DOI: 10.1016/j.ijpe.2009.01.013
- VDI 1994** Verein Deutscher Ingenieure. 1994. *VDI-Richtlinie 3590, Blatt 1: Kommissioniersysteme - Grundlagen*. Berlin: Beuth Verlag.
- Vogel 2015** Vogel, Jürgen. 2015. *Prognose von Zeitreihen. Eine Einführung für Wirtschaftswissenschaftler*. Wiesbaden: Springer Fachmedien. ISBN 9783658068370.
- Voudouris 1997** Voudouris, Christos. 1997. *Guided local search for combinatorial optimisation problems*. Essex, Univ. Diss., 1997.
- Wäscher 2004** Wäscher, Gerhard. 2004. Order Picking: A Survey of Planning Problems and Methods. In: Dyckhoff, Harald, Lackes, Richard & Reese, Joachim (Hrsg.): *Supply Chain Management and Reverse*

Logistics.

Berlin, Heidelberg: Springer, S. 323-347.

ISBN 9783540248156.

DOI: 10.1007/978-3-540-24815-6_15

Walter-Busch 1984

Walter Busch, E. 1984.

Methodische Aspekte des Interdisziplinaritätsproblems
in der systemorientierten Managementlehre.

In: Ulrich, H. (Hrsg.): *Grundlegung einer allgemeinen
Theorie der Gestaltung, Lenkung, und Entwicklung
zweckorientierter sozialer Systeme.*

Univ. St. Gallen, 1984, S. 226-252.

Weber 1990

Weber, Karl 1990.

Wirtschaftsprognostik.

München: Vahlen.

ISBN 9783800613854.

Wei et al. 2002

Wei, Chengjian, He, Zhenya, Zhang, Yifeng & Pei,
Wenjiang. 2002.

*Swarm directions embedded in fast evolutionary pro-
gramming.*

Evolutionary Computation. Proceedings of the 2002
Congress on IEEE, Honolulu, 12.-17. Mai 2002,
S. 1278-1283.

DOI: 10.1109/CEC.2002.1004427

Weicker 2015

Weicker, Karsten. 2015.

Evolutionäre Algorithmen.

3., überarb. u. erw. Aufl.

Wiesbaden: Springer Vieweg.

ISBN 9783658099589.

Werners 2013

Werners, Brigitte. 2013.

Grundlagen des Operations Research.

3., überarb. Aufl.

Berlin: Springer Gabler.

ISBN 9783642401015.

Wiendahl 2011

Wiendahl, Hans-Hermann. 2011.

Auftragsmanagement der industriellen Produktion.

Grundlagen, Konfiguration, Einführung.

Berlin, Heidelberg: Springer-Verlag.

ISBN 9783642191480.

Won et al. 2005

Won, J. & Olafsson, S. 2005.

Joint order batching and order picking in warehouse
operations.

International Journal of Production Research **43** (7),
S. 1427-1442.

DOI: 10.1080/00207540410001733896

- Yin 2006**
Yin, Peng-Yeng. 2006.
Genetic particle swarm optimization for polygonal approximation of digital curves.
Pattern Recognition and Image Analysis **16** (2),
S. 223-233.
DOI: 10.1134/S105466180602009X
- Yu et al. 2008**
Yu, Mengfei & de Koster, René. 2008.
Performance approximation and design of pick-and-pass order picking systems.
IIE Transactions **40** (11), S. 1054-1069.
DOI: 10.1080/07408170802167613
- Zhiming et al. 2008**
Zhiming, Liu, Cheng, Wang & Jian, Li. 2008.
Solving constrained optimization via a modified genetic particle swarm optimization.
Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, and multimedia and workshop, Adelaide, 21.-23. Januar 2008, Artikel Nummer 49.
ISBN 9789639799196.
- Zimmermann 2008**
Zimmermann, Hans-Jürgen. 2008.
Operations Research - Methoden und Modelle. Für Wirtschaftsingenieure, Betriebswirte und Informatiker.
2., aktual. Aufl.
Wiesbaden: Friedr. Vieweg & Sohn Verlag.
ISBN 9783834804556.
- Zobel 2005**
Zobel, Alexander. 2005.
Agilität im dynamischen Wettbewerb. Basisfähigkeit zur Bewältigung ökonomischer Turbulenzen.
Wiesbaden: Deutscher Universitäts-Verlag/GWV Fachverlage.
ISBN 9783322812162.
Potsdam, Univ. Diss., 2005.
- Zohm 2004**
Zohm, Frederik. 2004.
Management von Diskontinuitäten. Das Beispiel der Mechatronik in der Automobilzulieferindustrie.
Wiesbaden : Deutscher Universitätsverlag.
ISBN 9783844018189.
Aachen, Univ. Diss., 2003.

Kurze und zuverlässige Lieferzeiten werden bei Kaufentscheidungen immer wichtiger. Effizient betriebene Kommissioniersysteme werden damit zu einem entscheidenden Faktor. Der Wunsch der Kunden nach individualisierten Produkten führt zu sinkenden Auftragsgrößen, einer zunehmend volatilen Nachfrage und damit zu steigenden Herausforderungen für die Kommissionierung. So führt die gestiegene Dynamik der Kundennachfrage ohne Umlagerungen zu einer sinkenden Effizienz. Um Aufträge möglichst schnell und aufwandsarm kommissionieren zu können, sind drei Planungsprobleme zu lösen: Die Vergabe der Lagerplätze, die Bündelung von Aufträgen in sogenannten Batches und die Planung der Kommissioniertouren. Diese Probleme sind stark voneinander abhängig, werden bislang aber separat gelöst. Für die gemeinsame Lösung der Probleme gibt es noch kein geeignetes Verfahren. In dieser Arbeit wird ein Verfahren vorgestellt, das die drei Probleme ganzheitlich mithilfe von Schwarmintelligenz- und evolutionären Algorithmen löst und die zukünftige Dynamik der Kundennachfrage mittels Prognose antizipiert.

ISBN 978-3-8396-1504-1



FRAUNHOFER VERLAG