

TECHNISCHE UNIVERSITÄT MÜNCHEN

Lehrstuhl für Betriebswissenschaften und Montagetechnik am
Institut für Werkzeugmaschinen und Betriebswissenschaften (*iwb*)

Dynamische Routenzugoptimierung bei kurzfristigen Materialabrufen

Ulrich Stephan Teschemacher

Vollständiger Abdruck der von der Fakultät für Maschinenwesen der Technischen
Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender:

Prof. Dr.-Ing. Michael Zäh

Prüfer der Dissertation:

1. Prof. Dr.-Ing. Gunther Reinhart

2. Prof. Dr.-Ing. Johannes Fottner

Die Dissertation wurde am 26.02.2019 bei der Technischen Universität München
eingereicht und durch die Fakultät für Maschinenwesen am 28.05.2019 angenom-
men.

Geleitwort der Herausgeber

Die Produktionstechnik ist für die Weiterentwicklung unserer Industriegesellschaft von zentraler Bedeutung, denn die Leistungsfähigkeit eines Industriebetriebes hängt entscheidend von den eingesetzten Produktionsmitteln, den angewandten Produktionsverfahren und der eingeführten Produktionsorganisation ab. Erst das optimale Zusammenspiel von Mensch, Organisation und Technik erlaubt es, alle Potentiale für den Unternehmenserfolg auszuschöpfen.

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

Ziel der Forschungsarbeiten des iwv ist die ständige Verbesserung von Produktentwicklungs- und Planungssystemen, von Herstellverfahren sowie von Produktionsanlagen. Betriebsorganisation, Produktions- und Arbeitsstrukturen sowie Systeme zur Auftragsabwicklung werden unter besonderer Berücksichtigung mitarbeiterorientierter Anforderungen entwickelt. Die dabei notwendige Steigerung des Automatisierungsgrades darf jedoch nicht zu einer Verfestigung arbeitsteiliger Strukturen führen. Fragen der optimalen Einbindung des Menschen in den Produktentstehungsprozess spielen deshalb eine sehr wichtige Rolle.

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

Gunther Reinhart

Michael Zäh

Danksagung

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

Mein besonderer Dank gilt meinem Doktorvater Herrn Prof. Dr.-Ing. Gunther Reinhart und Herrn Prof. Dr.-Ing. Michael F. Zäh für ihre stets großzügige Unterstützung und wohlwollende Förderung im Rahmen meiner Forschungstätigkeit. Weiterhin gilt mein Dank Herrn Prof. Dr.-Ing. Johannes Fottner, Leiter des Lehrstuhls für Fördertechnik Materialfluss Logistik der Technischen Universität München, für die Übernahme des Kofereferats sowie die aufmerksame Durchsicht meiner Arbeit.

Der für mich entscheidende Erfolgsfaktor bei der Erstellung meiner Dissertation waren die zahlreichen Diskussionen in fachlichen und persönlichen Belangen mit wohlwollenden und erfahrenen Kollegen. Besonders bedanken möchte ich mich in diesem Rahmen bei Tobias Steinhäuser, Michael Niehues, Benedikt Sager, Cosima Stocker, Sebastian Schindler und Jan-Fabian Meis, die mir bei zahlreichen Fragestellungen mit Rat und Tat zur Seite standen und von denen ich viel für mich lernen konnte. Den vier Erstgenannten danke ich außerdem für die gründliche Durchsicht meiner Dissertation und die zahlreichen konstruktiven Anmerkungen, die meine Arbeit erheblich vorangebracht haben. Auch allen von mir im Rahmen meiner Promotionszeit betreuten Studierenden bin ich für die wertvollen fachlichen Beiträge zu meiner Dissertation zu großem Dank verpflichtet.

Des weiteren danke ich den hier nicht namentlich genannten Experten aus der Industrie, die es mir ermöglicht haben, meine Forschungsergebnisse in der Praxis zu evaluieren. Zusätzlich standen sie mir mit fundiertem und umfangreichem praktischen Expertenwissen für die Absicherung der Ergebnisse zur Seite.

Mein Weg und damit letztendlich die vorliegende Arbeit wäre ohne meine Familie und meine Freunde nicht möglich gewesen. Insbesondere meinen Eltern Lilli und Peter danke ich für die immerwährende Unterstützung bei all meinen Vorhaben, womit sie mir letztendlich die Grundlage für das Promotionsvorhaben geschafft

haben. Auch meinen beiden Geschwistern Sonja und Tobias möchte ich in diesem Rahmen für die viele schöne gemeinsame Zeit danken und wünsche ihnen viel Erfolg für die eigenen Promotionsvorhaben.

Liebe Cosima, Du hast mich durch zahlreiche fachliche Diskussionen an vielen Stellen vorangebracht und auch durch die besonders genaue Durchsicht der Dissertation nachhaltig zur Qualität meiner Arbeit beigetragen. Besonders wichtig war für mich jedoch deine unerschütterliche positive Grundhaltung und die emotionale Unterstützung. Eine Doktorarbeit lässt sich ohne die Unterstützung und Rücksichtnahme durch die Partnerin nicht schreiben – für deinen Beitrag und deine Hilfe möchte ich mich daher ganz besonders bedanken. Ich wünsche dir viel Erfolg bei deiner Doktorarbeit und freue mich auf eine gemeinsame Zukunft.

München, im Juli 2019

Ulrich Teschemacher

Inhalt

Abkürzungsverzeichnis	IX
Verzeichnis der Formelzeichen	XI
1 Einleitung	1
1.1 Ausgangssituation	1
1.2 Problemstellung	3
1.3 Zielsetzung	6
1.4 Spezifizierung des Untersuchungsbereichs	8
1.5 Wissenschaftstheoretische Einordnung	9
1.6 Methodischer Rahmen	12
1.7 Aufbau der Arbeit	14
2 Grundlagen	17
2.1 Innerbetriebliche Logistik	17
2.1.1 Aufgabenbereiche und Zielsetzung	17
2.1.2 Konzepte zur innerbetrieblichen Logistiksteuerung	20
2.2 Optimierungsmodelle im Unternehmensumfeld	22
2.2.1 Zielstellung und Einsatzbereiche	22
2.2.2 Mathematische Modellierung	24
2.2.3 Lösungsmethoden	25
2.3 Tourenplanungsprobleme	26
2.3.1 Charakteristik von Tourenplanungsproblemen	26
2.3.2 Modellierung	27
2.3.3 Tourenplanungsprobleme mit Zeitfenstern	29
2.3.4 Tourenplanungsprobleme mit variablen Zeitanteilen	31
2.3.5 Sonstige Erweiterungen von Tourenplanungsproblemen	32
2.3.6 Zielgrößen	33
2.3.7 Komplexitätstheoretische Betrachtung	35

3	Anforderungen	37
3.1	Übergeordnete Anforderungen	37
3.2	Spezifische Anforderungen	38
4	Lösungsverfahren für Tourenplanungsprobleme	43
4.1	Allgemeines	44
4.2	Übersicht über Lösungsverfahren	46
4.2.1	Heuristiken	46
4.2.2	Meta-Heuristiken	50
4.2.3	Hybride Ansätze	61
4.3	Analyse der Eignung für die innerbetriebliche Logistik	62
4.3.1	Vergleich der Lösungsqualität	63
4.3.2	Rechenzeit und Skalierbarkeit	64
4.3.3	Analyse der Umsetzungsfähigkeit notwendiger Restriktionen	66
4.3.4	Zusammenfassung der Lösungsverfahren	67
4.4	Forschungsdefizit	67
5	Dynamische Routenplanung	69
5.1	Erweiterung des VRP um zeitpunktabhängige Servicezeiten	69
5.1.1	Modellierung der Servicezeiten	69
5.1.2	Ermittlung der Parameter	72
5.1.3	Modellierung des Depots	74
5.2	Entwicklung des Ameisenalgorithmus für Logistikprobleme	75
5.2.1	Biologische Herkunft	75
5.2.2	Funktionsweise des Ameisenalgorithmus	76
5.2.3	Unterschiede zwischen künstlichen und realen Ameisen	79
5.2.4	Abwandlungen des Ameisenalgorithmus	81
5.2.5	Berücksichtigung von Nebenbedingungen	83
5.3	Verwendung des Algorithmus für die Tourenplanung	85
5.4	Zyklische Wiederverwendung logistischer Ressourcen	87
5.5	Berücksichtigung zeitlich veränderlicher Einflussgrößen	89
5.5.1	Adaption der Zielfunktion	90
5.5.2	Berücksichtigung in der heuristischen Information	90
5.5.3	Veränderung des Verhältnisses zwischen Pheromonkonzentration und heuristischem Wert	92

5.6	Priorisierung kurzfristiger Abrufe	93
5.6.1	Szenarien für kurzfristige Priorisierung	93
5.6.2	Berücksichtigung der Priorisierung	94
5.7	Lokale Suchverfahren zur Verbesserung der Lösung	96
5.7.1	Einfügeverfahren	96
5.7.2	Lokale Optimierungsmaßnahmen	98
5.8	Aufbau des Algorithmus zur Logistikoptimierung	100
5.9	Einbettung des Modells in das Produktionsumfeld	105
5.9.1	Schritt 1: Statische Parameter ermitteln	106
5.9.2	Schritt 2: Dynamische Informationen berücksichtigen	106
5.9.3	Schritt 3: Lösungen ermitteln	106
5.9.4	Schritt 4: Fahraufträge disponieren	107
5.9.5	Schritt 5: Informationen für die Fahrer bereitstellen	107
5.10	Zusammenfassung	108
6	Prädiktion der notwendigen Kapazitäten	109
6.1	Methoden zur Vorhersage künftiger Kapazitätsbedarfe	109
6.2	Auswahl der Methode zur Bedarfsprognose	111
6.3	Vorhersage künftiger Abrufe	112
6.4	Wahrscheinlichkeitstheoretische Betrachtung	113
6.5	Ableitung von Maßnahmen aus der Kapazitätsprognose	116
6.5.1	Ableiten der zu erwartenden Kapazitätsbedarfe	117
6.5.2	Abmindern von Kapazitätsspitzen	118
6.6	Zusammenfassung	119
7	Verifikation und praktische Validierung	121
7.1	Entwicklung des Softwaretools	122
7.1.1	Funktionalität des Programms	123
7.1.2	Methodisches Vorgehen bei der Entwicklung	125
7.2	Simulation dynamischer Testfälle	126
7.2.1	Entwicklung dynamischer Benchmarks	126
7.2.2	Beurteilung der Leistungsfähigkeit	127
7.3	Evaluierung der Prognosefähigkeit	129
7.4	Praktische Anwendung	132
7.4.1	Ausgangssituation des Anwenders	132
7.4.2	Ablauf der Logistikprozesse	133

Inhalt

7.4.3	Integration der Steuerung	134
7.4.4	Parametrierung auf die physischen Gegebenheiten des Anwenders	134
7.4.5	Anwendung der Steuerung	135
7.4.6	Analyse der Simulationsergebnisse	135
7.5	Identifikation wirtschaftlich relevanter Einsatzgebiete	136
7.6	Kritische Bewertung der Forschungsergebnisse	138
8	Zusammenfassung und Ausblick	143
8.1	Zusammenfassung	143
8.2	Ausblick	144
	Literatur	147
	Anhang	181
A.1	Einflüsse auf den wirtschaftlichen Betrieb von dynamischen Logistiksystemen	181
A.2	Klassifizierung von Tourenplanungsproblemen	182
A.3	Getroffene Annahmen für die Kapazitätsprognose	183
A.4	Betreute Studienarbeiten	184

Abkürzungsverzeichnis

ACO	Ant Colony Optimization
ACS	Ant Colony System
AS	Ant System
CVRP	Capacitated Vehicle Routing Problem
EAS	Elitist Ant System
ERP	Enterprise Ressource Planning
GRASP	Greedy Randomized Adaptive Search Procedure
INCOSE	International Council on Systems Engineering
JIT	Just In Time
KNN	Künstliche Neuronale Netze
LMS	Least Mean Squares
MMAS	Max-Min Ant System
NB	Nebenbedingung
OR	Operations Research
PPS	Produktionsplanung und -steuerung
PSO	Partikelschwarmoptimierung
RZ	Routenzug
SA	Simulated Annealing
SOFM	Self-Organizing Feature Maps
TDVRP	Time Dependend Vehicle Routing Problem
VND	Variable Neighborhood Descend
VNS	Variable Neighborhood Search
VRPTW	Vehicle Routing Problem with Time Windows
VRP	Vehicle Routing Problem
TSP	Traveling Salesman Problem

Verzeichnis der Formelzeichen

Schreibweisen

Vektoren und Matrizen sind in der vorliegenden Arbeit durch fettgedruckte Klein- bzw. Großbuchstaben dargestellt.

Griechische Buchstaben

α	Koeffizient für Einfluss der Pheromone (ACO)
β	Koeffizient für Einfluss der heuristischen Information (ACO)
η	Heuristischer Wert (ACO)
λ	Länge der Kette beim Austausch bei lokalen Suchverfahren
μ	Erwartungswert
$\Phi(x)$	Standardnormalverteilung
ρ	Verdunstungsfaktor der Pheromone (ACO)
σ	Standardabweichung
τ	Pheromonintensität
τ_0	Initiale Pheromonkonzentration
$\Delta\tau_{ij}^k$	Menge an Pheromonen für die Pheromonablage (ACO)
Ψ	Lösung
Ψ^0	Initiale Lösung
Ψ^i	Iterationsbeste Lösung
Ψ^{g^b}	Global beste Lösung
Ψ^k	Lösung der Ameise k
ξ	Verdunstungsrate (ACS)
ζ	Faltungsparemeter

Lateinische Buchstaben

d	Weglänge
\bar{d}_a	Durchschnittliche Kapazitätsbelegung durch eine Bestellung
$d_{i-1 \rightarrow i}$	Distanz zwischen Station $i - 1$ und i
$F(\mathbf{x})$	Zielfunktion
f	Fehler der LMS-Methode
$f(x)$	Dichtefunktion der Normalverteilung
$f_{v,i}$	Faktor für die variable Umladezeit an Station i
$g(\mathbf{x})$	Nebenbedingungen
$h(x)$	Rechteckfunktion
i	Laufvariable, Index
J	Zufällig nach Übergangsregel ausgewählte Kante
j	Mögliche nächste Station (ACS)
K	Menge der noch zu besuchenden Stationen (ACO)
k	Laufvariable, Index, Nummerierung der Ameisen (ACO)
L_{ψ^k}	Bewertung der Lösung Ψ^k nach der Zielfunktion (i. d. R. Dauer)
$L_{\psi^{g^b}}$	Bewertung der Lösung Ψ^{g^b} nach der Zielfunktion (i. d. R. Dauer)
m	Laufvariable, Index
n	Laufvariable, Index
N	Liste aller Stationen
N_{min}	Mindestanzahl Stichproben
n_n	Zähler für das Nicht-enthalten-Sein einzelner Stationen
n_p	Anzahl Parameter
$\mathcal{N}(\mu, \sigma^2)$	Normalverteilung mit Erwartungswert μ und der Varianz σ^2
p	Wahrscheinlichkeit für die Annahme einer Lösung (VNS, ACO)
q	Gleichmäßig verteilte Zufallszahl (ACO)
q_0	Parameter für Entscheidungsgrenzwert (ACO)
R^2	Bestimmtheitsmaß
$S_{i,j}$	Savingsfaktor für das direkte Verbinden von Station i und j
T	Temperatur (SA)
$t_{aktuell}$	Aktueller Zeitpunkt in der Tour
$t_{B,i}$	Zeitlicher Eingang der Bestellung i

$t_{B,n+1}$	Bestellzeitpunkt der Bestellung $n + 1$
$t_{B,p}$	Prognostizierter Bestellzeitpunkt der Bestellung p
$t_{c,i}$	Konstante Umladezeit an Station i
$t_{e,i}$	Spätester Ablieferzeitpunkt (Ende des Zeitfensters) an Station i
$t_{Komm.}$	Zeit zum Kommissionieren vor Abfahrt
t_F	Fahrtzeit
$t_{F,i-1 \rightarrow i}$	Fahrtzeit von Station $i - 1$ zu i
t_{ges}	Gesamtzeit der Tour
t_i	Ankunftszeit an Station i
$t_{letztmglich,i}$	Letztmöglicher Ablieferzeitpunkt an Station i
$t_{s,i}$	Servicezeit an Station i
t_W	Wartezeit
t_0	Startzeitpunkt der Tour
$t_{0,m}$	Startzeitpunkt der Tour m
Δt	Erwartungswert für den zeitlichen Abstand zweier Bestellungen
v	Geschwindigkeit des Routenzugs
\mathbf{x}	Variablenvektor
Δz	Wert der Verschlechterung (SA)

1 Einleitung

1.1 Ausgangssituation

Bedingt durch die Globalisierung konkurriert ein Großteil der heutzutage in Deutschland produzierten Güter international in gesättigten Märkten. Eine am Markt verfügbare Angebotsvielfalt mit ähnlichen Innovationen drängt Unternehmen dazu, sich durch immer kundenindividuellere Produkte Alleinstellungsmerkmale zu verschaffen, somit jedoch auch eine größere Anzahl an Produkten in verhältnismäßig kleinen Stückzahlen anzubieten (WESTKÄMPER & LÖFFLER 2016, S. 52 ff.; SCHUH 2012, S. 2). Ebenso wird häufig der Produktlebenszyklus verkürzt (WIENDAHL 2014, S. 99; SCHUH 2012, S. 2), um sich immer mit den aktuellsten Produkten am Markt behaupten zu können.

All diese Veränderungen führen aus Sicht eines produzierenden Unternehmens dazu, dass sich immer regelmäßiger der Bedarf für Anpassungen am Produktionssystem ergibt. Zudem ist eine hohe Stückzahlflexibilität für eine wirtschaftliche Produktion wichtig, da sich bei neuen Produkten trotz intensiver Marktforschung häufig nur mit geringer Zuverlässigkeit voraussagen lässt, wie sich der Absatz entwickeln wird (GÜNTNER 2007, S. 31). Zur Bewältigung dieser Herausforderungen eingesetzte flexible Konzepte in der Produktionsplanung und -steuerung (PPS) basieren häufig auf der Forderung, noch möglichst kurzfristig Auftragsreihenfolgen oder Maschinenbelegungen anpassen zu können (vgl. z. B. NIEHUES 2017, S. 69). Auch neuartige Fabrikkonzepte, wie beispielsweise die Inselfertigung bzw. Modularisierung und Flexibilisierung von Montagebereichen, welche bisher in Linie angeordnet sind, basieren grundlegend auf Flexibilisierung und somit dem Ermöglichen kurzfristiger Umplanungen (AUDI 2016).

Diese gestiegenen Anforderungen an die Flexibilität bei Produktionssystemen sind in Konsequenz auch für die Materialbereitstellung relevant, da eine Umplanung der

1 Einleitung

Fertigungsabläufe nur dann möglich ist, wenn die Verfügbarkeit der dafür erforderlichen Materialien sichergestellt werden kann. Sofern die Produktionsanlagen und -prozesse auf die nicht genau prognostizierbaren Stückzahlen sowie die ebensowenig vorhersehbare Verteilung der Variantenkonfigurationen reagieren können, ist die Logistik somit ebenfalls unmittelbar von diesen Schwankungen betroffen (BUSSE 2013, S. 6). Eine gute Skalierbarkeit der Logistik ist daher essenziell für den langfristig wirtschaftlichen Betrieb der Produktion. Erst durch das reibungslose Zusammenspiel aller Teilaspekte können die Potenziale der neuen Konzepte voll ausgeschöpft werden.

Daher ist auch die Logistik gefordert, sich flexibel aufzustellen und auf Änderungen kurzfristig reagieren zu können (GÜNTNER & BOPPERS 2013, S. 293). In Konsequenz ist es von Vorteil, wenn sich die Logistik möglichst flexibel an die Produktion anpassen kann und somit wenige zusätzliche Restriktionen an den Aufbau des Produktionssystems gestellt werden müssen.

Neben den Anforderungen hinsichtlich der erforderlichen Flexibilität führen immer schon existierende Erschwernisse wie bspw. Überkapazitäten durch Volumenschwankungen in diesem wirtschaftlichen Umfeld zu einer schwierigen Konkurrenzsituation und setzen somit die agierenden Unternehmen unter erheblichen Kostendruck (SCHUH 2012, S. 2). Um den Herausforderungen in Hinblick auf die Kosten gerecht zu werden, sind im gesamten Unternehmen große Anstrengungen notwendig, um sämtliche Prozesse auf maximale Effizienz zu trimmen. Vor allem die produzierenden Bereiche in Unternehmen sind stark kostengetrieben und unterliegen deshalb im Regelfall permanenten Verbesserungsprogrammen (J. SINGH & H. SINGH 2015, S. 76 ff.). Die genaue Betrachtung der Kosten zeigt, dass ein erheblicher Teil der Produktionskosten auf die Logistik entfällt. SEECK (2010, S. 10) beziffert diesen Anteil an den Gesamtkosten eines produzierenden Unternehmens typischerweise auf zwischen 10 % und 20 %, davon wiederum entsteht ungefähr die Hälfte durch die fabrikinterne Logistik (GÜNTHER 2006). Die Logistik ist damit einer der größten Kostentreiber in der Produktion, was sich auch daran erkennen lässt, dass bei Fabrikplanungsprojekten für die Layoutplanung die Minimierung des Transportaufkommens häufig als oberste Zielgröße herangezogen wird (SCHOLZ 2010, S. 3-4; S. SINGH & SHARMA 2006, S. 425 f.).

Die fabrikinterne Logistik zeichnet sich somit zum einen als elementarer Befähiger für flexible Produktionssysteme aus, zum anderen ist sie ein wichtiger Stellhebel auf

dem Weg zum kosteneffizienten, wirtschaftlichen Fertigungsbetrieb. Die Bedeutung von Flexibilität und Wirtschaftlichkeit sowie die Wege zur Erreichung ebendieser werden deshalb in den folgenden Kapiteln im Detail betrachtet.

1.2 Problemstellung

Für die innerbetriebliche Logistik sind wie eingangs dargelegt die beiden Zielgrößen *Flexibilität*¹ und *Wirtschaftlichkeit* relevant, welche in vielen Fällen in Konkurrenz zueinander stehen. Die Flexibilität, d. h. die Möglichkeit zur schnellen Anpassung auf Änderungen am Produktionssystem und dessen Prozessen sowie die kurzfristige Reaktion auf Auftragseingänge und deren ordnungsgemäße Auslieferung, ist essenziell, um mit den dynamischer werdenden Produktionsprozessen Schritt halten zu können (vgl. LIEBEROTH-LEDEN ET AL. 2017, S. 484 ff.). Zum anderen gilt es nach wie vor, im Kostenoptimum zu arbeiten – eine Verbesserung der Flexibilität auf Kosten der Wirtschaftlichkeit ist in der Regel nur in geringem Umfang akzeptabel. Die Notwendigkeit effizienter Prozesse zeigt sich u. a. am logistischen Zieldreieck in Abbildung 1.1, welches den in der Logistik vorherrschenden Konflikt zwischen dem gleichzeitigen Wunsch nach schnellen Auslieferungen, hoher Qualität und somit wenigen durch die Logistik induzierten Produktionsstörungen sowie niedrigen Kosten verdeutlicht (ARNDT 2015, S. 81). Eine Verbesserung von zwei Zielen lässt sich fast immer durch die Verschlechterung des dritten erzielen (z. B. Verbesserung von Qualität und Lieferzeit durch deutlich erhöhte Kosten), eine Verbesserung sämtlicher Zielgrößen ist jedoch im Regelfall nur durch effizientere Prozesse realisierbar.²

Aufgrund ihrer hohen Kapazität sowie der damit einhergehenden Effizienz und Wirtschaftlichkeit lässt sich in den letzten Jahren eine steigende Verbreitung von Routenzügen als innerbetriebliches Transportmittel beobachten (GÜNTNER ET AL. 2002, S. 12; SCHMIDT ET AL. 2016). Deren Steuerung erfolgt jedoch häufig nach dem

¹ Wie in Abschnitt 1.1 beschrieben umfasst dies im vorliegenden Kontext u. a. die Wandlungsfähigkeit, die Stückzahlflexibilität sowie die Reihenfolgeflexibilität.

² Entgegen der häufigen Interpretation des logistischen Zieldreiecks, dass zwei Ziele nur immer durch eine Verschlechterung des dritten Ziels ermöglichen lassen, ist eine gleichzeitige Verbesserung aller drei Ziele durchaus erreichbar. Da es möglich ist, mit einer Einzelmaßnahme alle drei Zielgrößen auf einmal zu verschlechtern, bewirkt das Rückgängigmachen ebendieser Handlung die Verbesserung aller drei Zielgrößen. Selbiges ist auch dann möglich, wenn zuvor nicht eine bewusst verschlechternde Maßnahme durchgeführt wird.

1 Einleitung

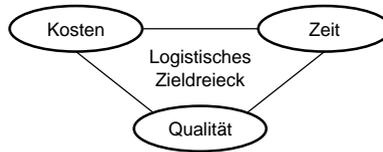


Abbildung 1.1: Das logistische Zieldreieck (ARNDT 2015, S. 81)

sogenannten *Milkrun-Prinzip* auf Basis starrer Routen und teilweise fest getakteter Abfahrtszeiten (KLUG 2018, S. 253 ff.). Die spontane und flexible Anpassung von Touren ist auf dieser Basis nicht möglich. Eine 2017 durchgeführte Studie zum Einsatz von Routenzugsystemen zur Produktionsversorgung zeigt, dass sich 94% der befragten Unternehmen durch eine dynamische Steuerung Vorteile erwarten, diese Konzepte derzeit im praktischen Umfeld jedoch noch kaum anzutreffen sind (LIEB ET AL. 2017, S. 63-71). Die Forschung auf diesem Gebiet ist daher von bedeutendem wirtschaftlichen Interesse.

Nach LIEB ET AL. (2017, S. 63) sind im Bereich der Logistik diverse Schritte für den reibungslosen Betrieb notwendig: Dies umfasst u. a. die Zusammenfassung einzelner Aufträge zu Touren, deren Reihenfolgebildung und Terminierung sowie die Bereitstellung der richtigen Kapazitäten. Je nach gewähltem Konzept können einzelne dieser Schritte bereits vorab durchgeplant sein (LIEB ET AL. 2017, S. 63). Im Gegensatz zum klassischen Milkrun-Prinzip sollte bei unregelmäßigen Materialabrufen im Hinblick auf die maximale operative Flexibilität den Fahrten jedoch bevorzugt kein fester Fahrplan zugrunde liegen, da andernfalls bei vielen spontanen Änderungen oder Abweichungen kein kurzfristiges Reagieren mehr möglich ist und somit keine zuverlässige Materialversorgung gewährleistet werden kann. Vor allem in größeren Bereichen ist eine technische Unterstützung zur dynamischen Planung der Touren und Zeiten erforderlich, da in derartigen Szenarien eine manuelle Steuerung selbst für Experten mit langjähriger Erfahrung in der Regel zu komplex wird.

Um auf die eingangs erläuterten Veränderungen am Produktionssystem und die sich dadurch verändernden Stellen für die Ablieferung von Material in der Logistik flexibel reagieren zu können, darf eine Produktionssystemanpassung keinen übermäßigen Aufwand in der Logistik erzeugen, da andernfalls die Wirtschaftlichkeit solcher Anpassungen durch die Folgeaufwände negativ beeinflusst wird. Das Gesamtsystem

muss so konzipiert sein, dass die automatisierte Routenplanung Änderungen der Umgebung ohne weitere manuelle Eingriffe selbstständig übernimmt und einplant.

Neben der bereits aufgezeigten wirtschaftlichen Relevanz, ist das Problem auch aus wissenschaftlicher Sicht von Interesse. Zahlreiche praktische Anforderungen sind in bisher vorhandenen theoretischen wissenschaftlichen Lösungsansätzen nicht vollumfänglich berücksichtigt und sollen im Rahmen dieser Arbeit verbessert werden. So muss beim *Einbehälter-Kanban* zur effizienten Ausnutzung vorhandener Flächen in Werken bei Anlieferung eines großen Ladungsträgers im Regelfall der alte Behälter direkt wieder abtransportiert werden, da am Arbeitsplatz nicht ausreichend Platz für beide vorhanden ist³ (GUDEHUS 2012, S. 450). Ist der alte Ladungsträger noch nicht leer, so muss der Inhalt des alten Ladungsträgers manuell in den neuen umgeladen werden, wodurch ein erheblicher Zeitaufwand entsteht. Der Füllstand und somit die Umladezeit ist abhängig von der Zeitspanne, um welche die Lieferung zu früh geliefert wurde. Würde der Behälter stehengelassen, so muss in einer weiteren Tour des Routenzuges ein Platz für den Rücktransport freigehalten werden, was zu erheblichen Einbußen in der Kapazitätsausnutzung führen würde. Die effizienteste Route zeichnet sich daher nicht ausschließlich durch die geringste Distanz aus, sondern durch die kürzeste notwendige Zeit zur vollständigen Abarbeitung (LIEB ET AL. 2017, S. 66). Zur Ermittlung dieser Zeit müssen u. a. zeitpunktabhängige Umladezeiten berücksichtigt werden. Eine solche Tour hat ggf. eine Verlängerung der zurückgelegten Distanzen zugunsten effizienterer Umladevorgänge zur Folge. Die Behebung der genannten wissenschaftlichen Lücken schafft hier die Basis für die wirtschaftliche Konzeption der Steuerung.

Viele derzeitige Steuerungsansätze scheitern an diesen Herausforderungen. Existierende dynamische Ansätze – sowohl kommerziell verfügbare als auch aktuelle Erkenntnisse aus der Forschung – finden sich selten im praktischen Einsatz (LIEB ET AL. 2017, S. 64). Potenzielle Gründe dafür sind die Komplexität bei der Einführung, komplexere Prozesse im Betrieb sowie durch die Steuerung entstehende Einschränkungen im Falle von Umplanungen. Im unternehmerischen Umfeld sind daher häufig Routenzüge mit festem Fahrplan oder die manuelle Verteilung der Fahraufträge anzutreffen (LIEB ET AL. 2017, S. 71). Diese praktischen und theore-

³ Beim 1:1-Tausch des Einbehälter-Kanban erfolgt die Auslösung einer neuen Bestellung bei Unterschreitung eines zuvor definierten Mindestbestands (manuell oder automatisiert).

1 Einleitung

tischen Defizite zeigen die Notwendigkeit der Forschung an adaptiven, sich selbst steuernden Logistikprozessen (vgl. auch ABELE & REINHART 2011, S. 140).

Gerade im Hinblick auf die in vielen Unternehmen heutzutage gute vorhandene Datenlage ist die Entwicklung von darauf aufbauenden wirtschaftlich einsetzbaren Konzepten lohnenswert. Viele Daten, und somit Informationen über den aktuellen Produktionszustand, werden aufgezeichnet und können über ihren originären Erfassungszweck hinaus für die Verwendung in anderen Bereichen herangezogen werden (MCKINSEY 2015, S. 12). VOGEL-HEUSER ET AL. (2017, S. 10 ff.) sehen eine durchgängige Datenverfügbarkeit und -transparenz als Grundlage für eine effiziente Steuerung der Routenzuglogistik. Diese erst in jüngster Zeit erfüllte Voraussetzung für eine verbesserte und dynamische Logistiksteuerung bleibt bislang weitgehend ungenutzt und soll im Rahmen dieser Arbeit als Grundlage verwendet werden. In Anlehnung an die Konzepte aus dem Bereich *Industrie 4.0* wird für derartige Problemstellungen auch der Begriff *Logistik 4.0* herangezogen, im Rahmen derer versucht wird, kognitive Logistikprozesse und -systeme zu ermöglichen, die lernfähig, intelligent und autark entscheidungsfähig sind (LIEBEROTH-LEDEN ET AL. 2017, S. 452; BOUSONVILLE 2017, S. 5).

1.3 Zielsetzung

Auf Grundlage der erläuterten Problemstellung leitet sich das Forschungsziel dieser Arbeit ab. Zielsetzung ist die Erhöhung der Wirtschaftlichkeit der innerbetrieblichen Logistik – im Speziellen liegt der Fokus dabei auf dem dynamischen, intralogistischen Materialtransport mittels Routenzügen. Dafür müssen zum einen durch vorausschauende Planung Kapazitäten eingespart werden, zum anderen dient das dynamischere Planen und Steuern der Routenzüge als Befähiger, um eine bessere Auslastung der Transportmittel zu realisieren. Zu diesem Zweck muss zur Unterstützung der Mitarbeiter ein Werkzeug in Form eines Programms entwickelt werden, welches die operative Planung übernimmt. Diese Umsetzung darf trotz fest integrierter Abläufe nicht die eingangs geforderte Flexibilität in der Logistik gefährden.

Im Fokus stehen Fabriken, die in Großserie produzieren. Die Materialbestellungen, welche die Produktionsbereiche vom Lager abrufen, sind dabei nicht langfristig

vorhersehbar, weisen jedoch eine gewisse Regelmäßigkeit auf. Im Rahmen von Expertengesprächen wurden die folgenden praktischen Zielsetzungen für die operative Logistiksteuerung identifiziert und werden im Weiteren betrachtet:

Hohe Prozesssicherheit: Der beschriebene Prozess muss so konzipiert sein, dass auch bei geringfügigen Änderungen kein manuelles Einschreiten erforderlich wird.

Echtzeitfähigkeit der Tourenberechnung: Die Touren müssen so schnell berechnet werden, dass bei neuen Bestellungen oder Veränderungen am Gesamtsystem (z. B. Störungen) ohne großen Zeitversatz ein neues, optimiertes Vorgehen für alle Transportmittel generiert werden kann.

Hohe Effizienz: Das Ziel der Wirtschaftlichkeit kann nur durch einen effizienten Einsatz sämtlicher Ressourcen und somit einen möglichst geringen gesamthafte Ressourceneinsatz erreicht werden.

Hohe Fehlertoleranz: Da im praktischen Einsatz mit Fehlbedienungen zu rechnen ist, muss das System mit diesen umgehen können und entsprechende Möglichkeiten zur Korrektur anbieten.

Unabhängigkeit von spezifischem Mitarbeiterwissen: Das Konzept soll unabhängig von der Mitarbeitererfahrung immer dieselbe hohe Ergebnisqualität erzielen.

Skalierbarkeit: Auch bei einer Vergrößerung oder Verkleinerung des Betrachtungsbereichs muss die Wirtschaftlichkeit gewährleistet sein.

Flexibilität des Gesamtsystems: Änderungen an den Produktionsanlagen oder -prozessen dürfen für das Logistikkonzept nicht zu kostspieligen Anpassungen führen.

Zusammengefasst führt die Erfüllung sämtlicher o. g. Zielsetzungen zu einem System, welches in der Lage ist, mittels intelligenter, kurzfristiger Steuerung der Routenzüge dynamisch auf Veränderungen zu reagieren. Diese können dabei von spontaner oder langfristiger Natur sein – die Logistiksteuerung muss sich hierfür ohne großen Aufwand anpassen können.

Eine ausschließliche Suche nach möglichst kurzen Strecken ist hierfür wie beschrieben nicht ausreichend. Zum einen müssen zusätzliche Besonderheiten bei zu früher

1 Einleitung

oder zu später Anlieferung berücksichtigt werden. Zum anderen kann eine Zusammenfassung der Fahrten auf weniger Fahrzeuge dazu führen, dass diese in Summe länger ausgelastet sind, dafür jedoch Fahrzeuge zumindest temporär eingespart werden können. Dies hilft in Bezug auf die geforderte Wirtschaftlichkeit – falls sich die Fahrzeuge für längere Zeiträume einsparen lassen, können durch deren Einsatz in anderen Bereichen auch finanzielle Einsparungen erzielt werden. Das Reduzieren der notwendigen Fahrzeuganzahl hat demnach Priorität über der Reduzierung der Fahrtzeiten.

Das zu entwickelnde Gesamtsystem für die operative Logistiksteuerung ist zusätzlich so zu konzipieren, dass die Anforderungen an IT-Schnittstellen so gering wie möglich sind, um keine zusätzlichen Hürden für die Einführung eines derartigen Systems zu erstellen. Auch die erforderlichen Schritte zur erstmaligen Einrichtung sollen aus demselben Grund aufwandsarm gehalten werden.

1.4 Spezifizierung des Untersuchungsbereichs

Um den Bereich potenziell gültiger Lösungen vorab einzugrenzen und so eine zielgerichtete Untersuchung bestehender Ansätze im Stand der Wissenschaft zu ermöglichen, wird der zu betrachtende Bereich in diesem Kapitel spezifiziert.

MARTIN (2014, S. 3) unterteilt die Logistik in die Bereiche *Beschaffungslogistik*, *Produktionslogistik*, *Distributionslogistik* und *Entsorgungslogistik*. Im Rahmen dieser Arbeit liegt der Fokus auf der fabrikinternen Produktionslogistik, wenngleich große Teile des Konzepts auch für sämtliche der genannten Logistikbereiche geeignet sind. Das Gesamtsystem soll dabei aufgrund der steigenden Verbreitung (siehe Abschnitt 1.1) mit einem Fokus auf Routenzüge ausgelegt werden, sämtliche anderen Transportmittel in der Logistik jedoch ebenfalls unterstützen.

Wie in Abschnitt 1.3 dargelegt, muss das Konzept skalierbar und flexibel anpassbar sein. Daher werden sowohl starre Vorausplanungen, die nur mit hohem Aufwand anpassbar wären, sowie manuelles Planen, was bei einer Komplexitätssteigerung schnell an die Grenzen stößt, explizit aus dem Betrachtungsbereich für mögliche Lösungen ausgeschlossen.

Das Konzept soll für ein dynamisches Umfeld einsetzbar sein, d. h. Bereiche mit klar vorterminierten Materialbedarfen sollen durch den Algorithmus zwar ebenfalls optimierbar sein, allerdings muss der Ansatz vor allem mit erst kurzfristig verfügbaren Bestellungen umgehen können. Dies umfasst auch Strategien für den Fall, dass selbst unter optimaler Ausnutzung sämtlicher verfügbarer Ressourcen eine rechtzeitige Anlieferung nicht mehr für alle Bestellungen möglich ist.

Geplantes Einsatzfeld für das Konzept sind Großserienproduktionen, in denen durch verschiedene Varianten Schwankungen in den Bedarfen auftreten können. Geringere Stückzahlen, bei denen eine dynamisch optimierte Logistik zwar ebenfalls vorteilhaft wäre, der Aufwand der Datenerhebung für die Logistik den Effizienzgewinn jedoch um ein Vielfaches übersteigen würde, werden aus dem Untersuchungsbereich ausgeschlossen.

1.5 Wissenschaftstheoretische Einordnung

Nach der inhaltlichen Einordnung der Forschungsschwerpunkte in den vorangegangenen Abschnitten wird im Folgenden vorgestellt, in welche wissenschaftstheoretische Denkweise die vorliegende Arbeit eingebettet ist. Dies ermöglicht dem Leser zum einen die Rekonstruktion und Beurteilung der Ergebnisse und soll zum anderen dabei helfen, die Grenzen der Arbeit verstehen und einschätzen zu können.

Die Wissenschaft ist grundsätzlich als eine Interpretation der Wirklichkeit zu verstehen, „die endgültige Wahrheit“ kann von ihr nicht geliefert werden. Vielmehr hat die Wissenschaft zur Aufgabe, eine fundierte Interpretation der Realität zu bieten (STÄHLER 2001, S. 13). ULRICH & HILL (1976, S. 305 f.) unterscheiden in ihren Überlegungen zur Wissenschaftssystematik zwischen Formal- und Realwissenschaften, wie dies in Abbildung 1.2 gezeigt ist. Die Formalwissenschaften (z. B. Philosophie, Logik, Mathematik) beschäftigen sich mit nicht real existierenden Objekten und sind daher nur auf ihre logische Wahrheit überprüfbar. Die Realwissenschaften, welche sich mit der Beschreibung, Erklärung und Gestaltung wahrnehmbarer Wirklichkeitsausschnitte beschäftigen, werden zusätzlich dem Kriterium der faktischen Wahrheit⁴

⁴ Im Gegensatz zur logischen Wahrheit kann bei der faktischen Wahrheit geprüft werden, ob sie im Hinblick auf das real existierende Problem Sinn ergibt (vgl. ULRICH & HILL 1976).

1 Einleitung

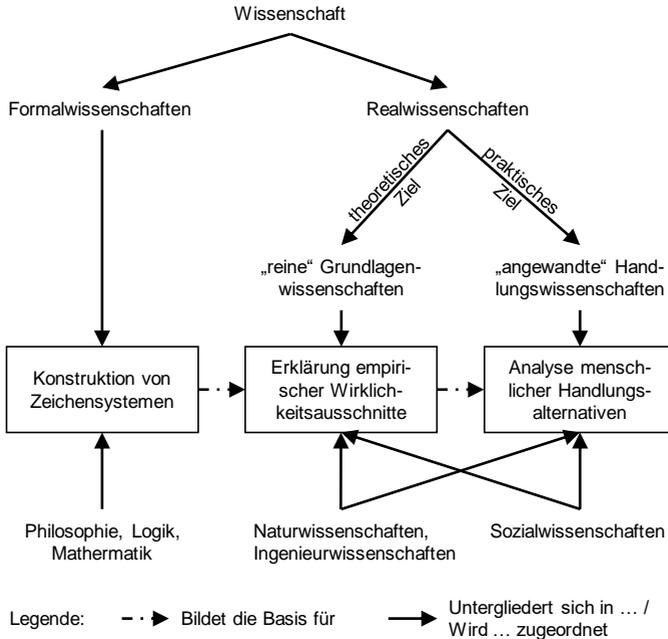


Abbildung 1.2: Wissenschaftssystematik nach ULRICH & HILL (1976, S. 305)

unterworfen. Ohne Formalwissenschaften sind Realwissenschaften nicht möglich (ULRICH & HILL 1976, S. 305 f.).

ULRICH & HILL (1976, S. 305 f.) unterteilen Realwissenschaften wiederum in Grundlagenwissenschaften („reine“ Wissenschaften) und Handlungswissenschaften („angewandte“ Wissenschaften). Fokus der Grundlagenwissenschaften ist die Erklärung existierender Phänomene. Demgegenüber versuchen die angewandten Wissenschaften (auch Anwendungsforschung genannt), menschliche Handlungsalternativen zur Gestaltung sozialer und technischer Systeme zu analysieren. Elementarer Bestandteil jeglicher Anwendungsforschung ist daher das Aufzeigen möglicher Einsatzgebiete sowie die Evaluation an praktischen Problemen (ULRICH & HILL 1976, S. 305 f.). Da die vorliegende Arbeit der Anwendungsforschung zuzuordnen ist, wird besonderer Wert auf die praktische Anwendbarkeit sowie die Validierung ebendieser gelegt (siehe Kapitel 7).

1.5 Wissenschaftstheoretische Einordnung

Das Vorgehen in der Forschung zwischen Grundlagenforschung und Anwendungsforschung unterscheidet sich erheblich, wie Tabelle 1.1 aufzeigt. Die Tatsache, dass praktische Probleme gelöst werden sollen und die gefundene Lösung an diesen Problemen gemessen werden muss, führt zu Vorgehensweisen, die sich von denen der Grundlagenforschung zum Teil grundsätzlich unterscheiden.

Tabelle 1.1: *Tabellarischer Vergleich von Grundlagen- und Anwendungsforschung* (STÄHLER 2001, S. 14; SPOUN 1998, S. 163)

Merkmale	Grundlagenforschung	Anwendungsforschung
Entstehung der Probleme	in der Wissenschaft	in der Praxis
Abgrenzung der Probleme	durch Theoriezusammenhang	durch Praxiszusammenhang
Forschungsziele	Theorie-Entwicklung und -prüfung, Erklärung der bestehenden Wirklichkeit	Entwerfen möglicher Wirklichkeiten
Angestrebte Aussagen	deskriptiv, wertfrei	normativ, wertend
Forschungsregulativ	Wahrheit	Nützlichkeit
Fortschrittskriterium	Allgemeingültigkeit, Bestätigungsgrad, Erklärungskraft, Prognosekraft von Theorien	Praktische Problemlösungskraft von Modellen und Methoden
Ablauf	Sequenziell: Erhebung, Auswertung, Interpretation	Zyklischer, iterativer Lernprozess: Problem und Ziele bestimmen, Handlungsplan aufstellen, Handlung realisieren, evaluieren, ggf. Handlungsplan modifizieren etc.

ULRICH & HILL (1976, S. 305 f.) beschreiben drei zu berücksichtigende Aspekte des realwissenschaftlichen Denkens: Den *Entdeckungszusammenhang*, den *Begründungszusammenhang* sowie den *Verwendungszusammenhang*.

Der *Entdeckungszusammenhang* beschreibt die Problematik, dass eine gefundene theoretische Konzeption abhängig von der Abgrenzung des Objektbereichs, der konkreten Problemstellung, von Grundbegriffen und Variablen zur Erfassung des Objektbereichs sowie der Aufstellung und Systematisierung von Arbeitshypothesen

1 Einleitung

sein kann. Um diesem Problem entgegenzutreten und bezüglich der Systemgrenzen Transparenz zu schaffen, werden in Kapitel 3 aus den zuvor identifizierten Herausforderungen die zu erfüllenden Anforderungen abgeleitet und aus dem Betrachtungsbereich auszuschließende Bereiche beschrieben. Eine erste Abgrenzung des Betrachtungsbereichs erfolgte bereits in Abschnitt 1.4.

Der *Begründungszusammenhang* beschreibt die Fragestellung, inwiefern aus einzelnen Beobachtungen auf die Allgemeinheit geschlossen werden kann. Durch das Aufstellen logischer Denkschlüsse, sowohl beim Formulieren der Anforderungen (Kapitel 3) als auch bei der Betrachtung der gewinnbringenden Einsatzbereiche (Abschnitt 7.5), kann bei diesem Aspekt Klarheit geschaffen werden.

Der *Verwendungszusammenhang* entspricht der Frage nach der Verwendung bzw. dem Zweck der wissenschaftlicher Aussagen. Anwendungsforschung steht hierbei grundsätzlich vor dem Konflikt, dass Lösungen zu wirtschaftlichen oder gesellschaftlichen Problemstellungen erwartet werden, der Mehrwert dieser jedoch erst nach Abschluss der Forschungstätigkeit quantifiziert werden kann. Im Rahmen dieser Arbeit werden sämtliche in Kapitel 3 aufgestellten Anforderungen daher in Abschnitt 7.6 kritisch hinterfragt und bewertet.

Die grundsätzliche Herausforderung jeglicher Wissenschaft – der Anspruch auf Objektivität – ist auch für Natur- und Ingenieurwissenschaften von elementarer Bedeutung, welcher ein einzelner Forscher alleine nicht zur Gänze nachkommen kann (POPPER 2017, S. 221). Als Ausweg nennt POPPER (2017, S. 221) die Diskussion und die gegenseitige Kritik unter Wissenschaftlern⁵. Die Thesen der vorliegenden Arbeit wurden deshalb auf nationalen und internationalen Konferenzen zur Diskussion gestellt, aufgrund ihres Charakters als anwendungsbezogene Forschungsarbeit wurden die Konzepte zusätzlich mit potenziellen Anwendern diskutiert und evaluiert.

1.6 Methodischer Rahmen

Aufgrund der Tatsache, dass zur erfolgreichen Erreichung der beschriebenen Ziele viele Einzelkomponenten nahtlos ineinander greifen müssen, wurde das Vorgehen

⁵ POPPER (2017) formuliert in seiner Ausarbeitung 27 Thesen zur Logik der Sozialwissenschaften. Die hier genannten Aussagen beziehen sich auf die elfte und zwölfte These.

zur Umsetzung des Forschungsvorhabens nach Konzepten des Systems Engineering⁶ durchgeführt. Dabei wurde auf das vom International Council on Systems Engineering (INCOSE) entwickelte Vorgehen *SIMILAR Process*⁷ gesetzt. Dieses ist in Abbildung 1.3 dargestellt.

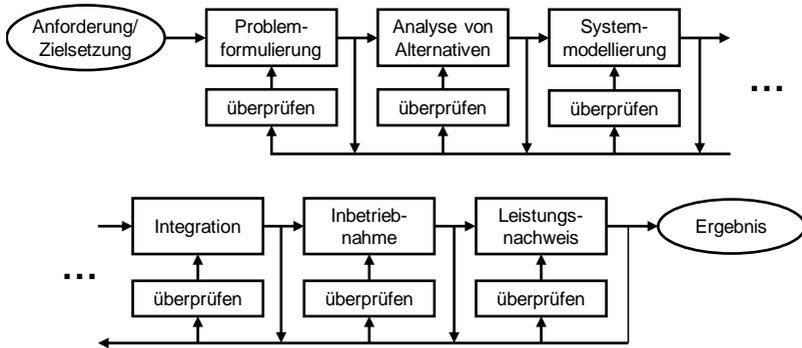


Abbildung 1.3: Vorgehensmodell SIMILAR-Process (in Anlehnung an BAHILL & GISSING 1998, S. 518)

Im ersten Schritt der Methode, der *Problemformulierung*, wird beschrieben, welche Funktion das System erfüllen muss bzw. welche Schwachstellen verbessert werden müssen. Dabei ist eine Beschreibung sämtlicher Anforderungen erforderlich, wobei nur das „Was?“ beschrieben wird, nicht das „Wie?“. Die *Analyse von Alternativen* stellt sicher, dass aus allen theoretisch denkbaren Möglichkeiten die aussichtsreichste Option für die weitere Entwicklung zugrunde gelegt wird. Sind im Laufe des Prozesses neue Daten verfügbar, so muss diese Analyse erneut durchgeführt werden. In den Ingenieurwissenschaften ist im Anschluss in den meisten Fällen eine *Systemmodellierung* erforderlich, welche auf unterschiedliche Art und Weise erfolgen kann (z. B. Flussdiagramme, analytische Formulierung oder Computersimulation). Diese Modelle reduzieren die Realität auf die für das betrachtete Problem wesentli-

⁶ Systems Engineering bezeichnet das Design von technischen Systemen über ihren gesamten Produktlebenszyklus hinweg. Besonderes Augenmerk liegt dabei auf der interdisziplinären Betrachtung der Prozesse. (WINZER 2016, S. 2 ff.)

⁷ Der Name *SIMILAR Process* ist ein Akronym, welches sich aus den englischen Begriffen State the Problem (sinngemäß: Problemformulierung), Investigate Alternatives (sinngemäß: Analyse von Alternativen), Model the System (sinngemäß: Systemmodellierung), Integrate (sinngemäß: Integration), Launch the System (sinngemäß: Inbetriebnahme oder Pilotierung), Assess Performance (sinngemäß: Validierung bzw. Leistungsnachweis) und Re-evaluation (sinngemäß: wiederholte Überprüfung) zusammensetzt.

1 Einleitung

chen Aspekte und offenbaren so Engstellen sowie schlecht aufeinander abgestimmte Aktivitäten. Außerdem ist anhand der Modelle eine erste Plausibilitätsprüfung der Anforderungen möglich. Da Systeme im Regelfall nicht vollkommen autark arbeiten können, sondern mit anderen Komponenten interagieren müssen, wird im Rahmen der *Integration* das Zusammensetzen der einzelnen Untersysteme durchgeführt. Ein möglichst gutes Zusammenspiel dieser Untersysteme muss schon in den vorhergehenden Planungsschritten berücksichtigt worden sein und wird in diesem Prozessschritt auf die Probe gestellt. Bei der *Inbetriebnahme* wird das Gesamtsystem in den Produktiveinsatz geführt, je nach Entwicklungsziel in ein produktives Szenario oder in ein Testszenario zur Bewertung der Eigenschaften unter realen Bedingungen. Zuletzt wird im Schritt *Leistungsnachweis* geprüft und gemessen, ob das entwickelte Konzept die erwarteten Eigenschaften und Vorteile aufweist. Parallel zu jedem dieser Schritte ist zu *überprüfen*, ob die Ergebnisse im Hinblick auf die zu erreichenden Ziele nützlich und ausreichend sind. Die Erkenntnis daraus wird zur Entscheidung über entsprechende Konsequenzen und notwendige Änderungen des Vorgehens herangezogen. Das regelmäßige Überprüfen ist essenziell und sollte nach jedem errungenen Ergebnis durchgeführt werden. (BAHILL & GISSING 1998, S. 518; INCOSE o.D.)

Sowohl die Forschungsmethodik als auch der Aufbau der vorliegenden Arbeit ist an den Ablauf des SIMILAR-Process angelehnt. Im folgenden Abschnitt werden die daraus abgeleiteten Kapitel vorgestellt. Die einzelnen Unterkapitel basieren jeweils auf davon unabhängigen, untergeordneten Methoden.

1.7 Aufbau der Arbeit

Die vorliegende Arbeit ist in acht Kapitel untergliedert, deren Aufbau in Abbildung 1.4 zusammengefasst ist und auf den Schritten aus Abschnitt 1.6 basiert. Zu Beginn dieses Kapitels wird die Ausgangssituation und die Motivation für die Thematik dargestellt und anschließend eine grundsätzliche Zielsetzung sowie eine Spezifizierung des Betrachtungsraums vorgenommen.

Kapitel 2 legt die theoretischen Grundlagen als logische Basis und zum Verständnis der Arbeit. Zunächst werden die Herausforderungen in der innerbetrieblichen Logistik erläutert. Daran anschließend erfolgt eine Beschreibung des Vorgehens bei der

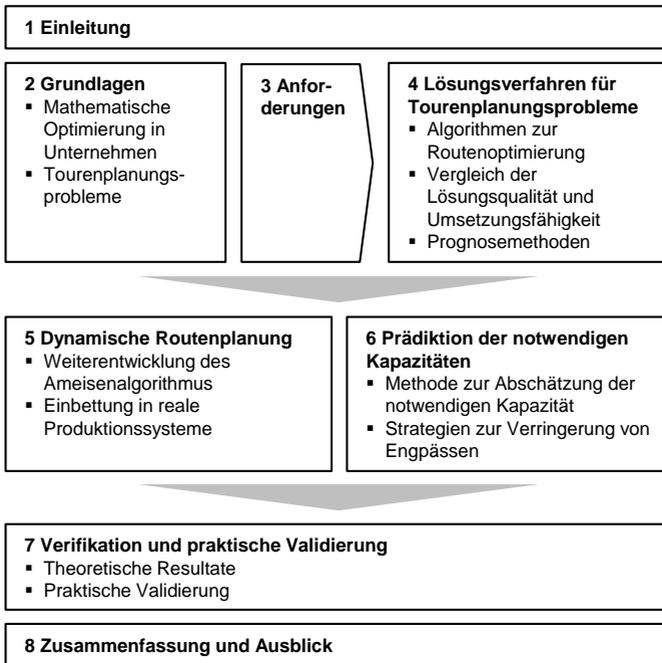


Abbildung 1.4: Aufbau der Arbeit

Anwendung mathematischer Optimierungsmodelle im Unternehmensumfeld. Dieses Wissen wird dann konkret auf den Anwendungsfall der Tourenplanungsprobleme projiziert.

Auf dieser Basis werden in Kapitel 3 die Anforderungen definiert, welche von einem Konzept zur dynamischen Routenzugoptimierung erfüllt sein müssen. Anhand dieser Anforderungen werden die in Kapitel 4 vorgestellten existierenden Lösungsverfahren verglichen und deren Eignung für die Problemstellung bewertet. Relevant sind dabei Algorithmen zur Reihenfolgeoptimierung aus den beiden Gruppen *Heuristiken* und *Meta-Heuristiken*. Die einzelnen Algorithmen werden jeweils kurz in ihrer Funktionsweise vorgestellt, zudem wird auf deren Einsatzmöglichkeiten hingewiesen. Das Kapitel schließt mit einer Bewertung der Algorithmen hinsichtlich Lösungsqualität und Umsetzungsfähigkeit, aus welcher das Forschungsdefizit abgeleitet wird.

1 Einleitung

Aus den Analysen in Kapitel 4 geht der Ameisenalgorithmus als der für den Anwendungsfall geeignetste Algorithmus hervor. Daher wird in Kapitel 5 und Kapitel 6 der Ameisenalgorithmus um die zuvor identifizierten Anforderungen erweitert. Kapitel 5 startet mit der Modellierung des erweiterten Tourenplanungsproblems und geht auf den notwendigen Vorgang der Parametrierung ein. Anschließend wird die Funktionsweise des Ameisenalgorithmus erläutert und die für die Einsetzbarkeit in logistischen Anwendungen notwendigen Erweiterungen zur Realisierung erarbeitet. Das Kapitel schließt mit der Beschreibung der Einbettung des entwickelten Konzepts in das Produktionsumfeld.

In Kapitel 6 wird auf Basis der Analyse von Vergangenheitsdaten ein statistisches Modell zur Prädiktion der notwendigen Kapazitäten entwickelt. Zum Abschluss werden verschiedene Vorgehensweisen für das Ableiten von Handlungen aus der Kapazitätsprognose vorgestellt, falls in dieser die prognostizierten Kapazitätsbedarfe die tatsächlich vorhandenen Kapazitäten überschreiten sollten.

Die Verifikation und die praktische Validierung der vorgestellten Konzepte erfolgt in Kapitel 7. Dabei wird zunächst eine Evaluation des Algorithmus anhand bestehender Testdatensätze durchgeführt, im Anschluss wird das Gesamtkonzept anhand realer Daten simuliert und dessen Leistungsfähigkeit nachgewiesen. Essenziell für die Einschätzung der praktischen Einsetzbarkeit ist die Vorstellung des Anwendungsbeispiels bei einem Automobilkonzern. Den Abschluss des Kapitels bilden die Betrachtung der gewinnbringenden Einsatzgebiete sowie die kritische Bewertung der erzielten Forschungsergebnisse.

Alle Inhalte und Erkenntnisse sind in Kapitel 8 abschließend zusammengefasst. Zusätzlich wird ein Ausblick auf weitere potenzielle Forschungstätigkeiten im Rahmen der betrachteten Fragestellungen gegeben.

2 Grundlagen

Im vorangegangenen Kapitel wurden die Rahmenbedingungen und die Notwendigkeit effizienter innerbetrieblicher Logistikprozesse aufgezeigt. Das folgende Kapitel legt darauf aufbauend die Grundlagen für das Verständnis der weiteren Arbeit. Dazu wird in Abschnitt 2.1 zunächst dargestellt, welche Aufgabenbereiche und Zielsetzung für die innerbetriebliche Logistik von Relevanz sind, um im Weiteren aufzuzeigen, welche Möglichkeiten zur Umsetzung dieser Ziele aus technischer Sicht denkbar sind. In Abschnitt 2.2 werden Modellierungsmöglichkeiten sowie grundsätzliche Methoden zur Darstellung und Berechnung von mathematischen Optimierungsansätzen im Unternehmensumfeld gezeigt. Diese werden in Abschnitt 2.3 konkret auf das Tourenplanungsproblem angewandt, welches die Grundlage für die Ausführungen in dieser Arbeit bildet.

2.1 Innerbetriebliche Logistik

2.1.1 Aufgabenbereiche und Zielsetzung

Die *innerbetriebliche Logistik*, auch *Intralogistik* oder *Standortlogistik* genannt, umfasst alle Transporte zwischen dem Wareneingang, den internen Quellen und Senken sowie dem Warenausgang (GUDEHUS 2012, S. 5). Die Beschaffungslogistik, die Distributionslogistik sowie die Entsorgungslogistik werden zur *außerbetrieblichen Logistik* gezählt, auch *Extralogistik* genannt. Die Bereiche sind in Abbildung 2.1 dargestellt und voneinander abgegrenzt.

Aufgabe der innerbetrieblichen Logistik ist es, firmeninterne Logistikleistungen anzubieten und diese auszuführen (GUDEHUS 2012, S. 5). Dieser Aufgabenbereich umfasst im Bereich der langfristigen strategischen Planung u. a. Entscheidungen

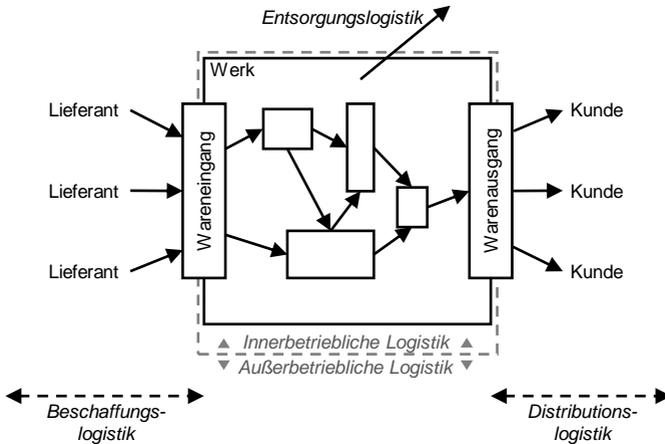


Abbildung 2.1: Abgrenzung der innerbetrieblichen Logistik (in Anlehnung an GUDEHUS 2012, S. 5)

über Steuerungssysteme¹, Netzwerkplanung² und Layouts. Die mittelfristige operative Planung befasst sich mit Personalbestand, Arbeitszeiten, groben Mengenflüssen sowie Transportmitteln und -wegen. In der kurzfristigen operativen Planung erfolgt schlussendlich die zeit- und mengengenaue Steuerung für die Ausführung der Prozesse (vgl. SEITZ 2016, S. 15). Wie in Abschnitt 1.4 dargelegt, sind im Rahmen dieser Arbeit die Prozesse rund um die innerbetriebliche Materialbereitstellung von Interesse, konkret die kurzfristige operative Planung sowie die davon betroffenen Teile der mittelfristigen operativen Planung. Die folgenden Ausführungen beschränken sich daher auf diese Teilbereiche.

Ziel der innerbetrieblichen Materialbereitstellung ist die termingetreue Auslieferung sämtlicher Materialien an die Stationen innerhalb einer Fabrik mit möglichst hoher Effizienz (GUDEHUS 2012, S. 69 ff.). Charakteristisch für die innerbetriebliche Logistik ist, dass die Erfüllung einer gewissen Aufgabe gefordert wird, deren Überer-

¹ Einen umfassenden Überblick über die Konzeption von schlanken Logistiksystemen inkl. deren Steuerung liefert DURCHHOLZ (2014).

² Für detaillierte Ausführungen zum Thema Netzwerkplanung sei z. B. auf SAGER (2019) verwiesen. Eine integrierte Betrachtung der beiden Probleme der Produktionsprozessplanung und der Planung von Transportprozessen zwischen verschiedenen Standorten findet sich in EHM (2017).

füllung³ jedoch keinen Mehrwert bringt. Es handelt sich daher um ein klassisches Problem nach dem Minimierungsprinzip, d. h. der geforderte Nutzen steht fest (alle Stationen müssen ohne den Aufbau produktionsnaher Lagerflächen termingerecht mit Material versorgt werden), wohingegen der Ressourceneinsatz so weit wie möglich minimiert werden soll (WERNERS 2013, S. 1). Diese Schwerpunktsetzung zeigt sich auch daran, dass das *Transportieren* von Gegenständen prinzipiell als eine der sieben Verschwendungsarten⁴ aus Sicht des Lean-Management gilt (ÖNO 2013, S. 54) und somit so weit wie möglich reduziert werden sollte. Für die Bereitstellung zur richtigen Zeit am richtigen Ort ist ein Kunde zwar durchaus bereit zu zahlen, dies gilt jedoch nur für dem Kunden wertbringende Logistik – fabrikinterne Logistik bringt für den Kunden keinen Mehrwert und kann daher nicht als wertschöpfende Tätigkeit gezählt werden (DURCHHOLZ 2014, S. 48 ff.). Dennoch ist der Transport eine zwingend notwendige Tätigkeit zur Aufrechterhaltung der Produktionsfähigkeit einer Fabrik. Demnach gilt ganz allgemein der Grundsatz, die Aufwände für die innerbetrieblichen Transporte auf das Minimum zu reduzieren. Als Richtlinie zur Gestaltung schlanker Logistikprozesse lassen sich die sieben Arten der Verschwendung nach ÖNO (2013) wie in Abbildung 2.2 dargestellt auch für die Logistik adaptieren (GÜNTNER & BOPPERT 2013, S. 46).

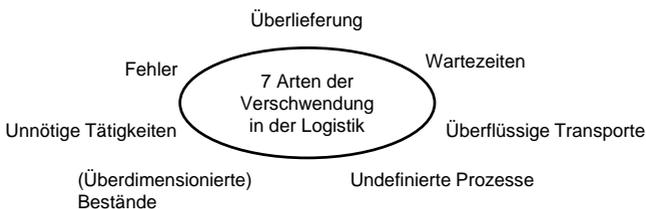


Abbildung 2.2: Die sieben Arten der Verschwendung in der Logistik (nach GÜNTNER & BOPPERT 2013)

Obwohl sie keinen direkten Beitrag zur Wertschöpfung leistet, ist die Leistung der Logistik dennoch ein wesentlicher Hebel für die Prozesse eines produzierenden

³ Übererfüllung wie z. B. eine zu frühe Anlieferung, Belieferung mit mehr Material als bestellt, etc. bieten keinen Mehrwert für den jeweiligen Besteller, können für die logistischen Abteilungen jedoch zu nennenswerter Mehrarbeit führen.

⁴ Die sieben Verschwendungsarten nach ÖNO (2013, S. 54) sind Überproduktion, Bestände, Wartezeiten, Transport, unnötige Bearbeitungsschritte, Bewegungen sowie Ausschuss und Nacharbeit (vgl. hierzu auch DOMBROWSKI & MIELKE 2015, S. 33)

2 Grundlagen

Unternehmens (PFOHL 2004, S. 7 ff.). ARNOLD (2008, S. 8) definiert die Logistikleistung über die vier Faktoren Lieferzeit, Liefertreue, Lieferqualität und Lieferflexibilität. ARNDT (2015, S. 79) fügt diesen noch die beiden zusätzlichen Faktoren Fehlerquote und Beanstandungsquote hinzu.⁵ Eine möglichst gute Erfüllung dieser Ziele ist zur Erreichung der gewünschten Wirtschaftlichkeit und der damit eng verknüpften Effizienz (vgl. Kapitel 1) essenziell.

Die Logistikleistung sowie die Versorgungssicherheit lässt sich durch eine hohe Lagerkapazität, möglichst nahe an den Stellen des Bedarfs, einfach erhöhen. Unter anderem wegen Flächen-, Bestands- und Lagerkosten werden von der innerbetrieblichen Logistik jedoch möglichst geringe Lagerbestände angestrebt⁶ – ein Ziel, welches mit vielen anderen Bereichszielen eines Unternehmens konkurriert (ARNDT 2015, S. 80). Viele dieser Zielkonflikte lassen sich durch erhöhte finanzielle Aufwände beseitigen, da dies ist jedoch bei gewinnorientierten Unternehmen nicht gewollt ist, bietet sich als Ausweg eine Effizienzsteigerung der Prozesse an, um ihre Leistung und die Wirtschaftlichkeit zu steigern (vgl. Abschnitt 1.1).

2.1.2 Konzepte zur innerbetrieblichen Logistiksteuerung

Nach der Betrachtung der prinzipiellen Aufgaben der innerbetrieblichen Logistik wird im Folgenden auf die Möglichkeit zur Steuerung ebendieser eingegangen. Aus KLENK & GALKA (2013, S. 388) sowie VDI-RICHTLINIE 5586-1 (2016) und VDI-RICHTLINIE 5586-2 (2016) ergeben sich hierfür drei grundsätzlich verschiedene Konzepte: Die Bildung starrer Routen, die freie Einteilung sowie die algorithmische Optimierung.

Das Konzept der starren Routen basiert auf fest definierten Touren, die im Voraus festgelegt werden. Stationen bzw. Materialabrufe sind dabei unmittelbar einer Route

⁵ Es gibt auch noch andere Bewertungskriterien bzw. Modelle zur Definition der relevanten Anforderungen wie bspw. die „6 r der Logistik“ nach WANNENWETSCH (2014, S. 68): das richtige Material, zum richtigen Zeitpunkt, in der richtigen Menge, in der richtigen Qualität, am richtigen Ort, zu den richtigen Kosten (WANNENWETSCH 2014, S. 68).

⁶ In der Produktion wird häufig versucht, das Lager vollständig zu eliminieren, höchstens kleine „Synchronisationspuffer“ werden toleriert. Dies steht im starken Kontrast zur überbetrieblichen Distributionslogistik, in der die Bedeutung von Lagern unumstritten ist. In vielen Fällen führt diese Eliminierung der Lager in der Produktion jedoch nur zu einer Verschiebung der Lagerplätze. (ARNOLD 2008, S. 19)

zugeordnet. Dabei sind unterschiedliche Strategien denkbar: Entweder der Routenzug fährt exakt die geplante Strecke, unabhängig davon, ob einzelne Stationen beliefert werden müssen. Alternativ kann der Routenzug auch nur exakt die Punkte anfahren, an denen tatsächlich Lieferungen notwendig sind und dazwischen Abkürzungen nehmen. Die Zuordnung der Stationen zur Tour sowie deren Reihenfolge entsprechen dabei trotzdem der vorab definierten Planung (RICHTER 2005). Zusätzlich kann unterschieden werden, ob die Touren getaktet bzw. nach einem festen Fahrplänen gestartet werden, oder ob neue Fahrten erst bei Bedarf ausgelöst werden (KLENK & GALKA 2013, S. 388).

Bei der freien Einteilung entscheidet ein Routenzugfahrer oder ein Fahrdienstleiter situativ, welche Bestellungen durch welches Fahrzeug ausgeliefert werden. Dabei wird nicht auf fixe Routen zurückgegriffen, sondern es werden je nach aktueller Auftragslage Routen generiert, welche die Bestellungen möglichst gut zusammenfassen. Je besser der Routenzugfahrer bzw. der Fahrdienstleiter über aktuell anstehende und in Kürze eintreffende Bestellungen informiert ist, desto höher ist die Wahrscheinlichkeit, dass eine gesamtheitlich betrachtet effiziente Tour gebildet wird. Zwei weitere Aspekte beeinflussen die Qualität der Routen: Zum einen hängt sie stark von der Erfahrung des Mitarbeiters ab, zum anderen wird es für Mitarbeiter mit großem Logistikbereich und entsprechend vielen Aufträgen ohne technische Hilfsmittel zunehmend schwieriger, den Überblick zu behalten. Für große Bereiche ist es bei der freien Einteilung daher schwer, die gewünschte Effizienz zu realisieren.

Bei der Routenzugsteuerung durch algorithmische Optimierung entspricht das Vorgehen dem der freien Einteilung, nur dass die aktuellen Stationen bzw. Bestellungen, der zur Verfügung stehende Fuhrpark sowie die Positionen und Strecken im Logistiknetz zur Tourenbildung an einen Algorithmus übergeben werden. Dieser errechnet daraus den zum aktuellen Zeitpunkt optimalen Fahrplan. Der Algorithmus kann dabei auch erwartete künftige Verfügbarkeiten oder bereits absehbare Bestellungen mit einbeziehen. Zahlreiche Algorithmen zur Lösung des zugrundeliegenden mathematischen Optimierungsproblems⁷ sind bekannt, die unterschiedliche Schwerpunkte adressieren. Vorteilhaft ist die Tatsache, dass bei effizienten Algorithmen im

⁷ Bei dem hier relevanten Optimierungsproblem handelt es sich um das sogenannte Tourenplanungsproblem, welches in Abschnitt 2.3 im Detail beschrieben wird.

2 Grundlagen

Gegensatz zur manuellen Einteilung auch bei großen Problemstellungen noch mit guten Lösungen gerechnet werden kann.

Der überwiegende Anteil der heute eingesetzten Routenzugsysteme operiert mit statischen Routen (LIEB ET AL. 2017, S. 64). Bei geschickter Auslegung lassen sich bei konstanten Bedarfen hiermit äußerst effiziente Systeme entwickeln (vgl. A. MEYER 2017). Nachteilig sind jedoch die eingeschränkten Reaktionsmöglichkeiten auf Schwankungen in der Auftragsmenge oder auf unterschiedliche Auslastungen der einzelnen Routen. Das entgegengesetzte Extrem, bei dem die Fahrten manuell generiert und Aufträge zu Touren zusammengefasst werden, ist zwar flexibel anpassbar, jedoch für große Bereiche kaum einsetzbar, da die Problemkomplexität die menschliche Kapazität ab einem gewissen Zeitpunkt überschreitet. Diese Konzepte kommen daher, wie in Abschnitt 1.4 bereits dargelegt, zur Umsetzung nicht in Frage. Der Fokus der Untersuchungen in den folgenden Kapiteln liegt somit auf der algorithmischen Optimierung und deren Integration in das Produktionsumfeld. Die Grundlagen hierfür werden im nächsten Abschnitt gelegt.

2.2 Optimierungsmodelle im Unternehmensumfeld

2.2.1 Zielstellung und Einsatzbereiche

Der Bereich Operations Research (OR) beschäftigt sich mit praxisnahen, komplexen Problemstellungen in Planungsprozessen zur Vorbereitung einer möglichst guten Entscheidung durch die Anwendung mathematischer Methoden⁸ (DOMSCHKE ET AL. 2015, S. 1). Somit umfasst OR alle Einsatzfelder in Unternehmen, in denen mithilfe mathematischer Methoden eine Bewertung verschiedener Handlungsmöglichkeiten erfolgt. Die Bedeutung des OR in der Praxis nimmt permanent zu, weil durch den weltweiten Wettbewerb einerseits die Komplexität der Probleme wächst, andererseits die Anforderungen an die Qualität der Lösungen permanent steigen (LÜBBECKE ET AL. 2014, S. VI).

⁸ Die Ursprünge des OR kommen aus dem militärischen Bereich. Erst später wurden die Methoden auf zivile Einsatzzwecke übertragen, wie z. B. für die Bestimmung der optimalen Zeitpunkte im Versand. (ELLINGER ET AL. 2003, S. 1)

Grundsätzlich können zwei verschiedene Arten von Optimierungsproblemen unterschieden werden. Bei Maximierungsproblemen soll bei gegebenen Ressourcen der Nutzen maximiert werden, während bei Minimierungsproblemen der Nutzen von vornherein festgelegt ist, aber der dafür notwendige Ressourceneinsatz minimiert werden soll (WERNERS 2013, S. 1). In der Logistikoptimierung handelt es sich, wie in Abschnitt 2.1.1 dargelegt, klassischerweise um Minimierungsprobleme, da die zu erfüllende Aufgabe feststeht, während der Einsatz an Ressourcen (Fahrzeuge, Personal, Energie, etc.) möglichst gering sein soll.

Die durchzuführenden Schritte beim Einsatz von Optimierungsmodellen im Unternehmensumfeld werden durch das Vorgehen nach DOMSCHKE ET AL. (2015, S. 1 f.) beschrieben:

1. *Erkennung und Analyse des Problems*: Identifikation des Handlungsbedarfs und Erkennen von Handlungsmöglichkeiten.
2. *Bestimmung von Zielen und Handlungsmöglichkeiten*: Identifikation des Ziels, Abstraktion und Erstellung eines deskriptiven Modells.
3. *Mathematisches Modell*: Entwicklung des mathematischen Modells auf Basis des deskriptiven Modells (unter Beachtung von Restriktionen).
4. *Datenbeschaffung*: Beschaffung von Daten aus Prognosen bzw. Vergangenheitsdaten.
5. *Lösungsfindung*: Mithilfe eines Algorithmus wird eine für die gegebene mathematische Formulierung besonders gut geeignete Alternative als Lösung ausgewählt.
6. *Bewertung der Lösung*: Kritische Analyse der Lösung sowie Betrachtung, inwieweit Fehler bei der Abstraktion bzw. Modellbildung das Ergebnis verfälscht haben.

Dieses Vorgehensmodell stellt eine Spezifizierung der Schritte aus dem Vorgehen in Abschnitt 1.6 dar. Der Aufbau der vorliegenden Arbeit entspricht daher dem von DOMSCHKE ET AL. (2015, S. 1–2) empfohlenen Vorgehen für Optimierungsprobleme.

2.2.2 Mathematische Modellierung

Ein mathematisches Modell besteht mindestens aus einer Alternativenmenge und einer Zielfunktion zur Bewertung dieser Menge (DOMSCHKE ET AL. 2015, S. 4). Das heißt, dass für ein gegebenes Problem eine Vielzahl an Alternativen existiert, die als prinzipielle Lösungsmöglichkeit in Betracht gezogen werden können. Welche dieser Lösungen die optimale Lösung für das Problem ist, wird durch die Zielfunktion ermittelt, die eine Bewertung einer jeden Alternative ermöglicht. Die Zielfunktion beschreibt mathematisch, wie sich die einzelnen Parameter auf die Bewertung der Lösung auswirken⁹. Je nach Anwendungsfall ist dabei das Minimum oder das Maximum dieser Funktion das Ziel (vgl. Abschnitt 2.2.1).

Zusätzlich kann durch Nebenbedingungen der gültige Lösungsraum eingeschränkt werden, wodurch für einzelne Parameter deren Gültigkeitsbereich festgelegt wird. Dabei sind vielfältige Möglichkeiten zur Beschränkung möglich, so können u. a. Bereiche festgelegt werden (z. B. „größer als 4“ oder „zwischen 1 und 7“) oder auch vordefinierte Ergebnisbereiche (z. B. „entweder 0 oder 1“) gefordert werden. (WINSTON & GOLDBERG 2004, S. 51 ff.)

Ein Optimierungsmodell besteht daher in seiner mathematischen Formulierung aus den folgenden Bestandteilen (GERDTS & LEMPIO 2011, S. 15; DOMSCHKE ET AL. 2015, S. 4 f.):

$$\text{Maximiere (oder Minimiere) die Zielfunktion } F(\mathbf{x}) \quad (2.1)$$

unter den Nebenbedingungen

$$g_i(\mathbf{x}) \begin{cases} \geq \\ = \\ \leq \end{cases} 0 \quad \text{für } i = 1, \dots, m \quad (2.2)$$

Bedeutung der Symbole:

- \mathbf{x} Variablenvektor mit den Komponenten x_1, \dots, x_n
- $F(\mathbf{x})$ Zielfunktion für den Variablenvektor \mathbf{x}
- $g_i(\mathbf{x})$ Nebenbedingungen

⁹ Im Unternehmensumfeld beschreibt die Zielfunktion i. d. R. wirtschaftliche Zusammenhänge, häufig handelt es sich um den Einfluss der einzelnen Stellgrößen auf die Gesamtkosten.

Die Einträge in \mathbf{x} sind dabei die Entscheidungsvariablen. Jede Konfiguration aus Entscheidungsvariablen muss das Modell vollständig beschreiben (WINSTON & GOLDBERG 2004, S. 49 ff.). Die Menge aller Kombinationen von \mathbf{x} , welche sämtliche Nebenbedingungen erfüllen, wird als *zulässige Menge* bezeichnet. Eine Lösung, die alle Nebenbedingungen erfüllt, heißt *zulässige Lösung* (GERDTS & LEMPIO 2011, S. 15).

Für viele praktische Problemstellungen sind mehrere Zielfunktionen nötig¹⁰, die kombiniert werden können und so das gesamthafte Optimum gefunden werden kann (MURATA & ITAI 2008, S. 99; WATANABE & SAKAKIBARA 2008, S. 113). Die Ziele können dabei zueinander komplementär, konkurrierend oder neutral sein, u. U. sind sie auch in unterschiedlichen Einheiten angegeben (DOMSCHKE ET AL. 2015, S. 61; ZIMMERMANN & GUTSCHE 1991, S. 21 f.).

Im Rahmen der Modellbildung muss entschieden werden, wie diese Zielfunktionen in Zusammenhang stehen. So ist bei gleichberechtigten Zielsetzungen die lineare Zusammensetzung zu einer sogenannten *multikriteriellen Zielfunktion* denkbar, die einen mathematisch definierten Kompromiss zwischen Erreichung der einzelnen Zielfunktionen beschreibt. Eine andere Möglichkeit ist eine *hierarchische Zielfunktion*, wobei zunächst nach der einen, dann nach der anderen Zielfunktion optimiert wird. Damit hat die erste Zielfunktion Priorität über die zweite. (ZIMMERMANN & GUTSCHE 1991, S. 42 ff.; IRNICH ET AL. 2014, S. 22)

2.2.3 Lösungsmethoden

Die Suche nach der besten Konfiguration zur Maximierung (bzw. Minimierung) der Zielfunktion $F(\mathbf{x})$ unter Berücksichtigung aller Nebenbedingungen $g_i(\mathbf{x})$ (vgl. Abschnitt 2.2.2) erfolgt im Regelfall durch einen Algorithmus. Für einfache Problemstellungen können dabei sogenannte *exakte Lösungsverfahren* eingesetzt werden, welche zuverlässig die optimale Lösung ermitteln können. Für komplexe Probleme

¹⁰ Beispiele für Problemstellungen mit mehreren Zielfunktionen sind u. a. Optimierungen im Unternehmensumfeld, bei denen zwar als primäre Größe der Gewinn optimiert wird, andere Ziele wie bspw. Flexibilität, Sicherheit, Umweltschutz oder Durchlaufzeiten jedoch ebenso möglichst zufriedenstellend erfüllt sein sollen.

2 Grundlagen

steigt der Rechenaufwand allerdings so stark an, dass *Näherungsverfahren* zur Annäherung herangezogen werden müssen (DOMSCHKE ET AL. 2015, S. 133 ff.). Hier bieten sich Heuristiken und Meta-Heuristiken an, welche die optimale Lösung nur annähern können und demnach meistens nicht das Optimum erreichen. Je nach Anwendungsfall ist der signifikante Geschwindigkeitsvorteil der Heuristiken bzw. Meta-Heuristiken jedoch ein so entscheidender Vorteil, dass auf den Einsatz von exakten Verfahren verzichtet wird. (CACERES-CRUZ ET AL. 2014)

Die Verfahren können zum Teil problemspezifisch sein, weshalb Details zur Problemkomplexität in Abschnitt 2.3.7 direkt anhand des vorliegenden Problems erläutert werden. Eine detaillierte Vorstellung verschiedener Lösungsalgorithmen in Bezug auf das im Rahmen der Arbeit wichtige Tourenplanungsproblem (siehe folgender Abschnitt) findet sich in Kapitel 4.

2.3 Tourenplanungsprobleme

2.3.1 Charakteristik von Tourenplanungsproblemen

Um die innerbetrieblichen Logistikprozesse durch die mathematischen Methoden aus Abschnitt 2.2 optimieren zu können, ist eine für Algorithmen interpretierbare Beschreibung der Situation erforderlich. Die Optimierung der Touren für Routenzüge in innerbetrieblichen Logistiksystemen kann dabei als sogenanntes Tourenplanungsproblem (engl. Vehicle Routing Problem (VRP)) modelliert werden (GYULAI ET AL. 2013, S. 2). Grundlage dafür ist das Problem des Handlungsreisenden (engl. Traveling Salesman Problem (TSP)), bei dem ein Reisender die Reihenfolge für den Besuch mehrerer Städte so wählen muss, dass die zurückgelegte Strecke minimal wird. Zum Abschluss muss er zu seinem Ausgangsort zurückkehren. (KORTE & VYGEN 2012, S. 593 ff.)

Das auf dem TSP basierende VRP ist eine erstmals von DANTZIG & RAMSER (1959) beschriebene Problemstellung. Es definiert im Wesentlichen die Aufgabe, eine vorhandene Menge an Aufträgen (z. B. Kundenaufträge oder Bestellungen) so in Reihenfolge zu bringen und zu einzelnen Touren zusammenzufassen, dass eine bestimmte Zielfunktion (z. B. Minimierung der Distanz) optimal erfüllt wird. Dabei wird jede Tour von einem Transportmittel bedient (ARNOLD 2008, S. 49; WENGER 2010, S. 40).

Zusätzlich wird bei VRPs die maximale Kapazität der Transportmittel berücksichtigt.¹¹ Für jedes Transportmittel wird dabei die maximale Zuladung definiert, die in einer einzelnen Tour ausgefahren werden kann und nicht überschritten werden darf. Für die Lösung des Problems ergibt sich hierdurch die Notwendigkeit der Unterteilung der Transportaufträge auf unterschiedliche Touren. (ARNOLD 2008, S. 49; IRNICH ET AL. 2014, S. 3)

Wie in Abschnitt 2.2.2 beschrieben, kann auch das VRP diversen Restriktionen unterliegen, wie beispielsweise zeitlichen Beschränkungen an einzelnen Stationen, Restriktionen bei der Kombination von Beladungen oder maximalen Tourenzeiten und -distanzen (WENGER 2010, S. 40). Die notwendigen Nebenbedingungen ergeben sich dabei aus dem gegebenen Problem.

Aufgrund vielfältiger Möglichkeiten zur Erweiterung der Zielfunktion sowie den aus der Praxis abgeleiteten zu berücksichtigenden Nebenbedingungen existieren zahlreiche verschiedene Tourenplanungsprobleme, welche in den folgenden Abschnitten erläutert werden. Diese decken die praktischen Anforderungen in unterschiedlichen Bereichen ab (WEISE ET AL. 2009, S. 4). Daraus lassen sich bei Bedarf weitere VRPs ableiten, wobei auch Kombinationen möglich sind. Durch die steigende Anzahl an Nebenbedingungen wächst jedoch auch die Komplexität des Modells, wodurch sich der Aufwand zur Lösung des Problems erhöht¹². Eine möglichst pragmatische Modellierung ohne zu große Vereinfachung der Realität sollte daher das Modellierungsziel sein. Eine Taxonomie und Eingruppierung der Tourenplanungsprobleme findet sich in Anhang A.2.

2.3.2 Modellierung

Zur mathematischen Lösung des Tourenplanungsproblems ist eine eindeutige Beschreibung notwendig, die alle relevanten Informationen beinhaltet. Unabhängig von der internen Speicherverwaltung des gewählten Lösungsalgorithmus, welche

¹¹ Obwohl es für die kapazitätsbeschränkte Version des VRP eine eigene Bezeichnung gibt (Capacitated Vehicle Routing Problem (CVRP)), wird die Kapazitätsbeschränkung i. d. R. auch in der Grundform des VRP berücksichtigt.

¹² Erhöhte Komplexität des Modells kann die Entscheidung bzgl. einer Algorithmusklasse beeinflussen, wie dies in Abschnitt 2.2.3 dargelegt wurde. Die Problemkomplexität wird in Abschnitt 2.3.7 detailliert betrachtet.

2 Grundlagen

je nach Implementierung unterschiedliche Formen annehmen kann, wird zur Modellierung des Problems i. d. R. eine Darstellung als Graph gewählt (IRNICH ET AL. 2014, S. 4 ff.).

Ein Graph ist eine abstrakte Struktur aus der Graphentheorie, die eine Menge von Objekten sowie deren Zusammenhänge untereinander beschreibt. Die Objekte werden dabei als Knoten bezeichnet, die Verbindungen zwischen den Knoten als Kanten. Eine Kante verbindet dabei stets zwei Knoten und lässt sich bei Bedarf durch eine sogenannte Gewichtung parametrieren (z. B. Distanz zwischen den Knoten). Ist definiert, welcher der beiden Knoten der Anfangs- bzw. Endknoten ist, so wird dies als gerichteter Graph bezeichnet. Ist die Richtung nicht festgelegt, so handelt es sich um einen ungerichteten Graphen. (TURAU & WEYER 2015, S. 19 ff.)

Ein VRP wird durch vier Komponenten beschrieben (LABADIE ET AL. 2016, S. 7 f.), die im Graph dargestellt werden:

1. *Die zu besuchenden Stationen*¹³, die eine gewisse Nachfragemenge einfordern. Im Graphen werden diese als Knoten dargestellt.
2. *Das logistische Netzwerk*, welches für die Belieferung der Stationen verwendet wird. Es wird durch die Kanten im Graph dargestellt. Die Gewichtung der Kanten stellt hierbei die Distanz¹⁴ zwischen zwei Knoten dar.
3. *Eine Fahrzeugflotte*, welche die Aufträge abarbeitet.
4. *Das Depot*, normalerweise der Start- und Endpunkt der Touren, welches ebenfalls als Knoten dargestellt wird.

Die Modellierung ist in Abbildung 2.3 visuell dargestellt. Jeder Knoten enthält dabei die Informationen über die zu beliefernde Station. Die spezifischen Angaben (Materialtyp, Menge, Zeitfenster für die Belieferung, Umladedauern, etc.) für die

¹³ In der Literatur finden sich je nach adressiertem Anwendungsgebiet unterschiedliche Bezeichnungen für den selben Sachverhalt. Die Begriffe *Stadt*, *Ort*, *Ziel*, *Kunde*, *Auftrag* und *Station* bezeichnen im Kontext der Tourenplanungsprobleme immer einen Ablieferort, der in eine Tour eingeplant werden muss. Aufgrund des Einsatzes im Fabrikumfeld wird in dieser Arbeit einheitlich der Begriff *Station* verwendet.

¹⁴ Je nach gewählter Zielgröße kann die Gewichtung auch anders definiert sein, z. B. als Kosten, Aufwände, Zeiten, etc.

Belieferung sind hier ebenfalls hinterlegt. Diese Informationen müssen nicht zwangsläufig statisch sein, sie können auch durch Funktionen beschrieben werden und demnach von anderen Parametern wie z. B. der Zeit abhängig sein.

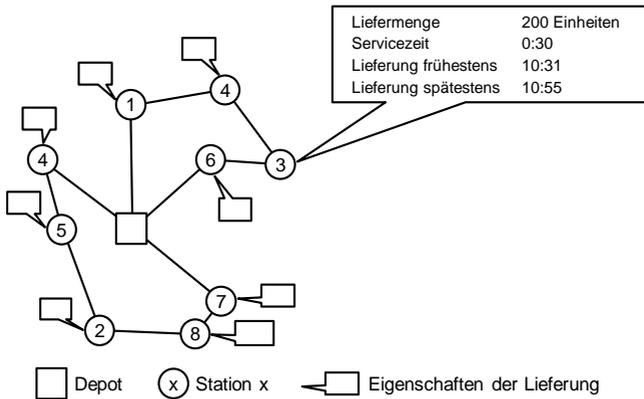


Abbildung 2.3: Modellierung des VRP als Graph

Die Distanzen auf den Kanten lassen sich zusätzlich zur Darstellung im Graphen in einer sogenannten Adjazenzmatrix formulieren, bei der die Distanzen als Werte eingetragen sind. Für nicht mögliche Verbindungen werden dabei in der Matrix die entsprechenden Felder leer gelassen¹⁵. Im Falle eines ungerichteten Graphen ist diese Matrix symmetrisch, siehe Abbildung 2.4. Bei innerbetrieblichen Logistiksystemen können sich gerichtete Kanten beispielsweise durch Einbahnstraßen ergeben. Das Formulieren der Adjazenzmatrix ist für die numerische Lösung der mathematischen Problemstellung von Vorteil. (TURAU & WEYER 2015, S. 30 ff.)

2.3.3 Tourenplanungsprobleme mit Zeitfenstern

Das Tourenplanungsproblem mit Zeitfenstern (engl. Vehicle Routing Problem with Time Windows (VRPTW)) ist eine Erweiterung des VRP, bei dem jede Station inner-

¹⁵ TURAU & WEYER (2015, S. 33) schlagen alternativ für nicht mögliche Felder beliebige Werte vor, die nicht im Definitionsbereich der Kantengewichtung liegen, z. B. 0 oder ∞ . Im Falle dieser Belegungen muss jedoch bei den nachfolgenden Rechenschritten darauf geachtet werden, dass nicht trotzdem mit diesen Werten weitergerechnet wird.

2 Grundlagen

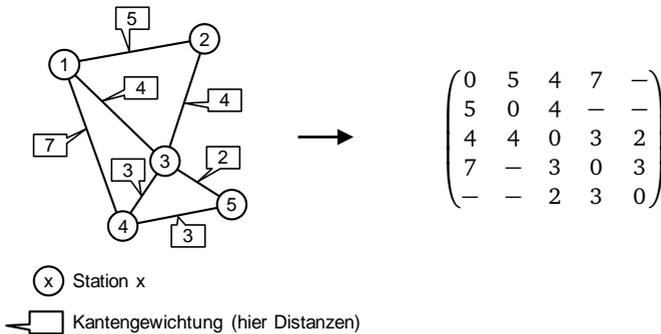


Abbildung 2.4: Ableitung der Adjazenzmatrix aus dem gewichteten Graphen

halb eines gegebenen Zeitfensters beliefert werden muss (DESAULNIERS ET AL. 2014, S. 131; MATTFELD & VAHRENKAMP 2014, S. 303; WEISE ET AL. 2009, S. 4).

Dabei wird unterschieden in harte und weiche Zeitfenster. Erreicht ein Fahrzeug im Falle eines harten Zeitfensters¹⁶ eine Station zu früh, so muss es bis zum Beginn des Zeitfensters warten, um das Material auszuliefern. Ob diese Wartezeit die Erfüllung der Zielfunktion verschlechtert, hängt von deren Formulierung ab (DESAULNIERS ET AL. 2014, S. 131). Bei Verletzung des Zeitfensters nach hinten ist die Lösung ungültig und muss verworfen werden. Bei weichen Zeitfenstern¹⁷ können diese hingegen in beide Richtungen überschritten werden, dies wird der Tour mit zusätzlichen „Strafkosten“¹⁸ angerechnet, gefährdet jedoch nicht die Gültigkeit der Lösung (MATTFELD & VAHRENKAMP 2014, S. 303). Zeitfenster können auch eine nur einseitige Begrenzung haben, z. B. nur einen spätesten Liefertermin (DESAULNIERS ET AL. 2014, S. 131).

Bei Problemen mit ausschließlich weichen Zeitfenstern sind alle gültigen Lösungen des Standardproblems ohne Zeitfenster auch mit Zeitfensterrestriktionen gültig. Für den Lösungsalgorithmus ist somit das Auffinden gültiger Lösungen verhältnismä-

¹⁶ Ein Beispiel für ein hartes Zeitfenster sind z. B. die aus dem Alltag bekannten Ladenöffnungszeiten.

¹⁷ Weiche Zeitfenster treten beispielsweise bei der Belieferung von Kunden bei Paketdiensten auf. Dem Kunden wird ein Zeitfenster vorgegeben, das Logistikunternehmen versucht, innerhalb dieses Zeitraums zu liefern. Wird das Zeitfenster verletzt, so führt dies zwar ggf. zu Kundenbeschwerden, hat jedoch keine weiteren Konsequenzen für das Logistikunternehmen.

¹⁸ Der Wert der Zielfunktion wird dabei aktiv durch die Strafkosten verschlechtert, auch wenn dies die eigentliche mathematische Formulierung der Zielfunktion nicht so bewerten würde.

ßig einfach. Insbesondere Meta-Heuristiken, welche auf eine Vielzahl von gültigen Lösungen angewiesen sind, können derartig formulierte Problemstellungen besser lösen. Harte Zeitfenster verkleinern hingegen den potenziellen Lösungsraum (MATTFELD & VAHRENKAMP 2014, S. 304 f.), wobei sich viele Anforderungen aus der Praxis nicht ohne die Verwendung ebendieser modellieren lassen. Als Ausweg kann unter Umständen eine Anpassung des realen Problems (z. B. Verhandlung mit Kunden oder Lieferanten entlang der Supply Chain etc.) angedacht werden (MATTFELD & VAHRENKAMP 2014, S. 305). Neben mathematischen Vorteilen bei der Lösung des Problems kann dies auch dabei helfen, Prozesse im Unternehmen zu flexibilisieren und zu vereinfachen. Im Allgemeinen bleibt jedoch kein anderer Weg als die steigende Komplexität und damit auch die steigende Rechenzeit zu akzeptieren.¹⁹

2.3.4 Tourenplanungsprobleme mit variablen Zeitanteilen

Die nötige Zeit zur kompletten Abarbeitung einer Tour setzt sich aus zahlreichen Einzelbestandteilen zusammen. So wird die Zeit in Fahrtzeit und Servicezeit an der Station unterteilt, welche im Falle der Routenzuglogistik wiederum in Bestandteile wie Absteigen, Abladen, Aufladen, etc. aufgeteilt werden kann (BUCHHOLZ & CLAUSEN 2009, S. 162). In den bisher beschriebenen Modellen²⁰ wurden sämtliche dieser Bestandteile als konstant betrachtet, was jedoch für viele Problemstellungen keine hinreichend präzise Modellierung darstellt.

Unterschiedliche Komponenten können dabei variabel sein. Ein viel betrachtetes Beispiel ist die Berücksichtigung zeitabhängiger Verkehrseinflüsse²¹, welche vor allem in extralogistischen Systemen relevant sind (DONATI ET AL. 2008). Auch BEKTAŞ ET AL. (2014, S. 329 ff.) betrachten dynamische Elemente, allerdings werden hier nur schwankende Streckenzeiten und ungewisse Fahrzeugverfügbarkeiten in Betracht gezogen. Bei zeitpunktabhängigen Servicezeiten ist hingegen die Zeit an der Station abhängig vom Ankunftszeitpunkt. Derartige Zeiten sind in der Übersicht

¹⁹ Eine zusätzliche Möglichkeit der Modellierung führen LAU ET AL. (2003, S. 563 ff.) ein, indem für zu späte Belieferung und damit das Verletzen der Zeitfenster Strafpunkte auf die Zielfunktion aufgeschlagen werden. Bei entsprechender Wahl der Höhe der Strafpunkte führt die Optimierung hier wiederum zu einer Lösung ohne Verletzung der Zeitfenster.

²⁰ vgl. Abschnitt 2.3.2 und 2.3.3

²¹ Häufig ist bei Staus in Abhängigkeit der Zeit eine Vorhersage aufgrund von Erfahrungswerten möglich (vgl. DONATI ET AL. 2008).

2 Grundlagen

von BRAEKERS ET AL. (2016, S. 303 ff.) erwähnt, erfahren jedoch in der Forschung deutlich weniger Beachtung²².

Zusammengefasst können die Zeiten in der Modellierung entweder statisch, zeitpunktabhängig (als formelmäßiger Zusammenhang vom Zeitpunkt t des Erreichens der Station im Rahmen einer Lösung), dynamisch (veränderlich über den Zeitpunkt der Berechnung) oder stochastisch sein. Diese Zusammenhänge müssen in der Optimierung berücksichtigt werden. (CACERES-CRUZ ET AL. 2014, S. 32:18)

2.3.5 Sonstige Erweiterungen von Tourenplanungsproblemen

Zusätzlich zu den bereits genannten Erweiterungen gibt es eine Reihe zusätzlicher praxisrelevanter Modelle. Klassischerweise wird beim VRP ein abgegrenzter einmaliger Zeitraum betrachtet, d.h. die Touren werden zusammengestellt und alle gleichzeitig gestartet – das VRP ist damit komplett abgearbeitet. Im Gegensatz dazu können Fahrzeuge nach ihrer Rückkehr für die nächste Tour zur Verfügung stehen, sofern dies durch die Zeitfenster der Stationen möglich ist. Wiederkehrende Fahrzeuge sind unter der Bezeichnung *Multi-Trip* in vereinzelt Modellierungen des VRP berücksichtigt (IRNICH ET AL. 2014, S. 15; BATTARRA ET AL. 2009, S. 304 ff.; CATTARUZZA ET AL. 2014, S. 833 ff.). Außerdem gibt es dynamische Tourenplanungsprobleme, bei denen während der Berechnung der optimalen Lösung neue Stationen hinzugefügt werden können (RICHTER 2005, S. 55).

Allen Modellierungen ist gleich, dass ihnen das sogenannte *Non-Passing-Theorem* zugrunde liegt (JAEGERE ET AL. 2014, S. 13). Dies beschreibt, dass selbst durch variable Zeitannteile ein später losfahrendes Fahrzeug das früher gestartete nicht überholen kann. Somit kommt ein früher losfahrendes Fahrzeug bei der gleichen Route immer vor dem später gestarteten an. In Konsequenz lohnt sich das Herausögern der Abfahrt nur, falls die Hoffnung auf bessere Tourkombinationen besteht – andernfalls gilt allgemein, dass eine frühere Abfahrt auch eine frühere Rückkehr bedeutet.

²² Für die Modellierung der fabrikinternen Routenzuglogistik werden die zeitpunktabhängigen Servicezeiten benötigt, die Modellierung und Erweiterung ist Inhalt von Abschnitt 5.1.

2.3.6 Zielgrößen

In vorhergehenden Abschnitten wurden die notwendigen Nebenbedingungen beschrieben. Diese schränken zwar den gültigen Lösungsraum ein, geben jedoch keine Information über die Optimalität einer Lösung. Zu diesem Zweck wird die sogenannte Zielfunktion (siehe Abschnitt 2.2.2) eingeführt. Je nach Wahl der Zielfunktion kann jeder Punkt im Lösungsraum bzw. jede beliebige Reihenfolge der Stationen zur optimalen Lösung werden. Daher ist zur Ermittlung der optimalen Lösung zusätzlich die Kenntnis über die für die Problemstellung entscheidende Zielgröße notwendig, d. h. eine Formel zur Bewertung der Güte der Lösung (KALLRATH 2013, S. 13 f.). Mögliche Zielgrößen sind unter anderem

- die Minimierung der Transportkosten,
- die Minimierung der Transportzeit eines einzelnen Fahrzeugs,
- die Minimierung der Gesamttransportzeit der Flotte,
- die Minimierung der Zeiten, die ein Material geladen ist,
- die Minimierung der Flottengröße,
- die Optimierung hin zu einer gleichmäßigen Auslastung,
- die Minimierung der Routenlängen (KARA ET AL. 2008, S. 93) oder
- die Minimierung der Zeit bis zur Auslieferung (MURATA & ITAI 2008, S. 99).

Zahlreiche weitere Zielfunktionen kommen in Frage und müssen in Abhängigkeit des Anwendungsfalls ausgewählt werden.

Als Zielgröße für die innerbetrieblich Logistik kommen verschiedene Größen in Frage. Diese werden in den folgenden Unterabschnitten im Detail vorgestellt. Die Auswahl ist dabei nicht abschließend, je nach Details des Anwendungsfalls können auch gänzlich andere Zielgrößen relevant sein und ausgewählt werden. Auch die Kombinationen aus mehreren verschiedenen Größen ist möglich (siehe Abschnitt 2.2.2).

Minimierung der Distanz: Die Optimierung nach der Distanz ist die klassische Optimierungsgröße beim TSP. Bei dieser Zielgröße werden ausschließlich die zurückgelegten Fahrtstrecken berücksichtigt.²³ Damit ist diese Optimierungsgröße auch einsetzbar, wenn keine weiteren Informationen wie bspw. Fahrgeschwindigkeit oder Umladezeiten verfügbar sind.

²³ Je nach Anwendungsszenario kann bei dieser Zielgröße die letzte Strecke vernachlässigt werden, dies ist zum Beispiel bei Kurierdiensten wie beim Ausliefern einer Pizza der Fall (KARA ET AL. 2008, S. 91).

Für das VRP ist sie praktisch nur einsetzbar, wenn der Zeitanteil für das Fahren deutlich größer ist, als der für die restlichen Tätigkeiten. Speziell in der innerbetrieblichen Logistik können diese jedoch nicht vernachlässigt werden. Dennoch beruhen fast alle Benchmarkinstanzen zum Vergleich von Lösungsalgorithmen auch beim VRP auf dieser Zielgröße.

Die Minimierung nach der Distanz entspricht außerdem annäherungsweise der Minimierung nach der eingesetzten Energie, da der Energieverbrauch in den meisten Szenarien nahezu proportional zur zurückgelegten Strecke ist (KARA ET AL. 2008, S. 93).

Minimierung der Zeit Wird nach der Zeit optimiert, ist es das Ziel, sämtliche Transportaufträge in möglichst kurzer Zeit abzuarbeiten. Dabei werden in der Regel alle zur Verfügung stehenden Fahrzeuge genutzt (CARIĆ ET AL. 2008, S. 15). Im Gegensatz zur Optimierung nach der Distanz müssen hier u. a. auch Umladezeiten, Wartezeiten oder Ausfallzeiten berücksichtigt werden, da andernfalls zu kurze Gesamtzeiten ermittelt werden.

Weil einfache VRPs eine linear mit der Distanz verknüpfte Fahrzeit annehmen, entspricht die Optimierung nach der Zeit in diesem Fall der Optimierung nach der Distanz. Erst bei zeitabhängigen Komponenten wie Staus oder Zeitfenstern ergibt sich ein Unterschied gegenüber der Optimierung nach der Distanz. Die Komplexität des Problems steigt durch diese Erweiterungen erheblich.

Minimierung der notwendigen Transportkapazität Da für jeden Materialabruf der spätestmögliche Liefertermin bekannt ist, bringt eine frühere Anlieferung keine Vorteile, sondern im Normalfall eher Nachteile durch Wartezeiten oder erhöhte Umladezeiten (siehe Abschnitt 1.2). Es ist daher nicht zwingend notwendig, immer so schnell wie möglich zu liefern, wie dies bei der Optimierung nach der Zeit der Fall wäre, denn unter Umständen lässt sich mit einem Teil der vorhandenen Transportmöglichkeiten bereits der gesamte Auftragsbestand abarbeiten. Freie Ressourcen können dann anderweitig verwendet werden und erlauben einen wirtschaftlicheren Betrieb der Logistik.

Die Minimierung nach der notwendigen Transportkapazität entspricht ansatzweise einer Minimierung der Kosten (CARIĆ ET AL. 2008, S. 15). Allerdings wird dabei zwar der Ressourceneinsatz und somit die laufenden Kosten minimiert, auf die Fixkosten wird jedoch keine Rücksicht genommen. Deshalb ist

zur Realisierung wirtschaftlicher Vorteile neben der Optimierung der Touren auch eine Optimierung der angrenzenden Prozesse (z. B. flexible Einteilung von Personal und Fahrzeugen) erforderlich.²⁴

2.3.7 Komplexitätstheoretische Betrachtung

Da bei kombinatorischen Problemen eine endliche Anzahl an Lösungen existiert, ist es prinzipiell möglich, alle Lösungen zu vergleichen und daraus die beste Lösung auszuwählen. Dieses Verfahren nennt sich *vollständige Enumeration*, ist für größere Problemstellungen jedoch praktisch nicht durchführbar, da die Rechenzeit ins Unermessliche ansteigt. (KISTNER 2003, S. 184)

Aufgrund dieser Problematik lassen sich Optimierungsprobleme in *praktisch lösbare* und *praktisch nicht lösbare* Probleme unterteilen:

Praktisch lösbar heißt, dass ein sogenannter polynomieller Algorithmus zur Lösung des Problems existiert. Unter einem polynomiellen Algorithmus wird ein Algorithmus verstanden, bei dem sich die Zeitkomplexität mit einer polynomiellen Formel ermitteln lässt²⁵. Das Problem gehört dann der Klasse P an. (BRONŠTEJN ET AL. 2013, S. 1191 ff.)

Für Probleme, deren Zeitkomplexität stärker als polynomiell ansteigt, spricht man von nicht-polynomieller Zeitkomplexität und ordnet sie daher der Klasse NP zu. Das heißt, dass eine exakte Lösung außer bei extrem kleinen Problemen *praktisch nicht möglich* ist. (BRONŠTEJN ET AL. 2013, S. 1198 ff.)

Lösungsmethoden, die den gesamten Lösungsraum abdecken, sind für Probleme der Klasse NP zu langsam. Um dennoch Lösungen zu ermitteln, müssen daher Näherungsverfahren eingesetzt werden, welche zwar nicht die optimale Lösung garantieren können, im Regelfall jedoch zumindest in die Nähe des Optimums kommen.

²⁴ Das Problem der remanenten Fixkosten wird im Rahmen dieser Arbeit im langfristigen Planungshorizont in Form der Prognose berücksichtigt (siehe Kapitel 6).

²⁵ Bildlich betrachtet entspricht das der Fragestellung, um wie viel sich der Rechenaufwand erhöht, wenn das zu lösende Problem vergrößert wird. Lässt sich dieser Zuwachs durch eine polynomielle Formel beschreiben, ist das Problem praktisch lösbar, andernfalls gilt es als praktisch nicht lösbar.

2 Grundlagen

Das VRP zählt zu den Problemen der Klasse NP (LENSTRA & KAN 1981, S. 223). Damit lässt sich eine exakte Lösung für größere Probleminstanzen nicht mehr finden. Beim VRPTW ist bereits das Ermitteln einer gültigen initialen Lösung NP-schwer (SAVELSBERGH 1985, S. 299). Um die geforderte dynamische Reaktionsfähigkeit dennoch sicherstellen zu können, muss zur Lösung daher auf Näherungsverfahren gesetzt werden (DITTES 2015, S. 83 ff.).

3 Anforderungen

3.1 Übergeordnete Anforderungen

Wie in Abschnitt 1.3 beschrieben, ist das Ziel der Arbeit die Erhöhung der Leistungsfähigkeit von innerbetrieblichen Logistiksystemen zur Ermöglichung dynamisch reaktionsfähiger Systeme. Daraus lassen sich auf Basis der in Kapitel 2 geschaffenen Grundlagen Anforderungen an das zu entwickelnde Gesamtkonzept definieren.

Unter der grundlegenden Annahme, dass die übergeordnete vorliegende Problemklasse in der Praxis effizient lösbar ist, lassen sich axiomatisch vier grundlegende Anforderungen an Lösungsmethoden definieren, die für anwendungsnahe Forschung gültig sind (vgl. hierzu auch Abschnitt 1.5):

Richtigkeit der Ergebnisse: Die grundlegende Anforderung ist, dass die ermittelten Ergebnisse richtig sein müssen. Dies ist an zwei Stellen relevant: Bei der Modellierung, welche die praktischen Besonderheiten hinreichend genau abbilden muss sowie bei der Findung der optimalen Lösung, die zu Ergebnissen führen muss, welche möglichst optimal sind und die Effekte der Realität möglichst genau beschreiben.

Anwendbarkeit im industriellen Kontext: Das Konzept zur Lösung der Problemklasse muss praktisch einsetzbar sein. Daraus lässt sich beispielsweise ableiten, dass der Aufwand zur Erstellung von Schnittstellen zu existierenden IT-Systemen möglichst gering sein muss oder dass das System eine ausreichende Flexibilität zur Anpassung an sich verändernde Fabrikstrukturen aufweist.

Wirtschaftlichkeit: Die Lösungsmethode muss für das Unternehmen in der Praxis einen Mehrwert generieren, der die Kosten überkompensiert. Ziel muss die Reduktion von kostenrelevanten Größen sein.

3 Anforderungen

Allgemeingültigkeit: Das Konzept muss allgemeingültig sein, d. h. es muss mit geringen Adaptionen für verschiedene Szenarien einsetzbar und gültig sein.

Der Erfüllungsgrad einzelner dieser Anforderungen kann je nach Lösung unterschiedlich sein. Ob dieser ausreichend ist, muss im Kontext der Problemstellung betrachtet und bewertet werden.

3.2 Spezifische Anforderungen

Wie in Abschnitt 2.1.2 gezeigt, empfiehlt sich für die vorliegende Problemstellung die Entwicklung eines dafür angepassten Algorithmus. Im Folgenden werden daher Anforderungen an derartige Lösungsverfahren formuliert, die sich aus den übergeordneten Anforderungen aus Abschnitt 3.1 ableiten lassen. Einige davon sind spezifisch für algorithmische Implementierungen, andere sind auch für klassische Steuerungsansätze gültig.

Pünktlichkeit: Alle Lieferungen müssen pünktlich abgeliefert werden, da in der Fabrik ansonsten Stillstände drohen. Je nach Platzverfügbarkeit ist eine zu frühe Anlieferung zwar ineffizient, jedoch i. d. R. unkritisch in Bezug auf Produktionsausfälle.

Dynamische Reaktionsfähigkeit: Eingehende Materialanfragen müssen hinreichend schnell bearbeitet werden, so dass auch kurzfristige Bestellungen noch rechtzeitig ausgeliefert werden. Dies vermeidet die Notwendigkeit langer Vorausplanungen, was die Flexibilität einschränken würde.

Fehlertoleranz: Im praktischen Betrieb treten regelmäßig ungeplante Situationen auf, die bei der Logistikplanung berücksichtigt werden müssen. Dazu gehört beispielsweise die Reaktionsfähigkeit auf Fehler der Fahrer bzw. auf Transportanweisungen von außerhalb des Standardprozesses (z. B. Eilaufträge), die Reaktion auf etwaige ungeplante Eingriffe an undefinierten Stellen im Prozess sowie allgemein eine möglichst hohe Toleranz gegenüber Abweichungen von der ursprünglichen Planung. Auch die Berücksichtigung bzw. Berichtigung von Fehlern (z. B. durch zu kurzfristige Bestellungen, durch Beladung und Mitnahme des falsche Behälters, etc.) muss ohne großen Zusatzaufwand möglich sein (BRUNGS 2012, S. 71). Ebenso ist die Reaktion auf eigenständige, spontane,

manuelle Fehlerbehebungen erforderlich, da somit das Erfahrungswissen und die Problemlösekompetenz der Mitarbeiter trotz automatisierten Vorgehens noch in vollem Umfang genutzt werden können.

Effizienz: Für eine hohe Effizienz ist die Vermeidung von Verschwendung wichtig. Deshalb müssen alle Prozesse möglichst schlank und effizient sein. Dies umfasst u. a. die Vermeidung unnötiger Transporte, falscher Transporte, verlängerter Umladezeiten, Stillstände und Wartezeiten (siehe die sieben Arten der Verschwendung in Abschnitt 2.1.1). Direkt damit zusammenhängend sind ein geringer Ressourceneinsatz sowie eine hohe Auslastung der bestehenden Flotte. Somit wird der gesamte Auftragsbestand mit einer minimalen Anzahl an Fahrzeugen durchgeführt, nicht benötigte Ressourcen werden vorab für entsprechende Zeitintervalle freigestellt.

Prognosefähigkeit: Zur Planung der notwendigen Kapazitäten müssen Kapazitätsbedarfe vorhersehbar sein. Sollten nicht handhabbare Kapazitätsspitzen auftreten, müssen diese so früh wie möglich bekannt sein, um gegebenenfalls Gegenmaßnahmen rechtzeitig ergreifen zu können.

Integrierbarkeit in bestehende Lösungen: Um die Hürden bei der Einführung nicht unnötig anzuheben, ist der Bedarf an Eingangsdaten so gering wie möglich zu halten. Der Aufwand durch die Schnittstellenprogrammierung zu anderen Softwarelösungen kann andernfalls zu Kosten führen, welche den Einsatz des gesamten Systems in Frage stellen können. Zusätzlich muss die Steuerung in der Lage sein, auf eine Überregelung durch den Anwender reagieren zu können.

Skalierbarkeit hinsichtlich Problemgröße: Die Lösungsmethodik muss auch für große Probleme noch innerhalb akzeptabler Zeiten Lösungen finden.

Erweiterbarkeit: Zukünftige in der Logistik zu berücksichtigende Eigenschaften, wie bspw. der Einsatz anders gearteter Transportmittel, müssen im Konzept abbildbar sein. Eine entsprechende Erweiterbarkeit ist daher bereits im Voraus zu berücksichtigen. Restriktionen, die aus der Produktion an die Logistik gestellt werden, müssen ebenfalls als Nebenbedingung darstellbar sein.

Korrektheit: Die ermittelten Pläne müssen in der prognostizierten Zeit durchführbar sein, d. h. die Zeitbausteine des Ergebnisses müssen mit denen der realen

3 Anforderungen

Tourdurchführung übereinstimmen. Dazu ist es wichtig, dass bei der Lösungserstellung alle relevanten Faktoren berücksichtigt werden, was wiederum bedeutet, dass der Algorithmus mit den real vorhandenen, als Nebenbedingung formulierten, Restriktionen der Fertigung agieren können muss. Für die speziellen Anforderungen durch die Zielstellung ist es erforderlich, dass der Algorithmus mit variablen Zeitanteilen umgehen kann und diese in die Ermittlung der optimalen Tour mit einbezieht.

Unabhängigkeit von Mitarbeitererfahrung: Die Logistiksteuerung muss unabhängig von der Erfahrung, vom Wissensstand sowie von der aktuellen Tagesform der Mitarbeiter gute und reproduzierbare Ergebnisse generieren.

Rechenzeit: Der Algorithmus muss innerhalb von kurzen Rechenzeiten zu guten Ergebnissen kommen. Diese Anforderung gilt auch für große Systeme. Wie kurz die Rechenzeit in der praktischen Anwendung sein muss, ergibt sich aus den genauen Rahmenbedingungen des Anwendungsfalls. Eine geringe Rechenzeit ist einer der wichtigsten Befähiger für eine gute Skalierbarkeit.

Lösungsqualität: Das Erreichen des Optimums ist wünschenswert, jedoch nicht zwingend erforderlich, solange die Lösung hinreichend nahe am Optimum liegt. Die Möglichkeit zur Parametrierung zwischen guter Lösungsqualität und schneller Rechenzeit ist im Bezug auf die breite Einsetzbarkeit und damit die Allgemeingültigkeit sowie die Anwendbarkeit im industriellen Kontext ein wichtiger Bestandteil.

Die spezifischen Anforderungen lassen sich den übergeordneten zuordnen. Diese Zuordnung ist in Tabelle 3.1 dargestellt. Zusätzlich lässt sich der Tabelle entnehmen, welche Anforderungen für welchen Teil des Konzepts relevant sind. So muss das Gesamtkonzept schlussendlich alle Anforderungen erfüllen, wohingegen für den Optimierungsalgorithmus nur eine Teilmenge davon relevant ist. Die Erfüllung der verbleibenden Anforderungen muss durch die passende Integration des Algorithmus sichergestellt werden. Für den Algorithmus ergeben sich somit die Anforderungen *Skalierbarkeit hinsichtlich der Problemgröße, Erweiterbarkeit, Rechenzeit und Lösungsqualität*. Die Erfüllung dieser Anforderungen durch den Algorithmus ist eine zwingende Voraussetzung dafür, dass das Gesamtkonzept sämtlichen Anforderungen gerecht werden kann. In Kapitel 4, welches verschiedene Lösungsverfahren vergleicht, sind daher nur die vier hier aufgeführten Anforderungen von Relevanz,

Tabelle 3.1: Zuordnung der Anforderungen zu den übergeordneten Anforderungen

	Zugeordnet zu ...				Relevant für ...	
	Richtigkeit der Ergebnisse	Anwendbarkeit im industriellen Kontext	Wirtschaftlichkeit	Allgemeingültigkeit	Algorithmus	Gesamtkonzept
Pünktlichkeit	●	●	●			●
Dynamische Reaktionsfähigkeit	●	●	●			●
Fehlertoleranz	●	●	●			●
Effizienz			●			●
Prognosefähigkeit			●			●
Skalierbarkeit hins. Problemgröße		●		●	●	●
Integrierbarkeit		●		●		●
Erweiterbarkeit		●		●	●	●
Korrektheit	●	●		●		●
Unabh. von Mitarbeitererfahrung		●	●			●
Rechenzeit		●	●		●	●
Lösungsqualität	●	●	●		●	●

Legende:

- Anforderung hat keinen Einfluss auf die Erfüllung der übergeordneten Anforderung
- Anforderung zur Erfüllung der Hauptanforderungen essenziell

wohingegen in Kapitel 5 und Kapitel 6 die vollständige Liste an Anforderungen berücksichtigt werden muss.

Basierend auf diesen Anforderungen werden in den folgenden Kapiteln die Algorithmusstrukturen zur Lösung des Tourenplanungsproblems im Kontext der fabrik-internen Materialbereitstellung geschaffen. Trotz der technischen Möglichkeit zur präzisen Vorgabe von exakten Navigationsanweisungen sei darauf hingewiesen, dass es in der Praxis nicht von Vorteil ist, diese im Detail vorzugeben. Für automatische fahrerlose Transportfahrzeuge ist dies zwar nötig, bei menschlich gesteuerten Systemen werden jedoch die kognitiven Fähigkeiten der Fahrer genutzt, um lokal kleine Verbesserungen an der Route vorzunehmen. Dies kann beispielsweise durch kurzfristige bauliche Veränderungen oder gesperrte Wege im Werk von Vorteil sein.

4 Lösungsverfahren für Tourenplanungsprobleme

Wie in Abschnitt 2.1.2 dargelegt, wird in dieser Arbeit eine algorithmische Optimierung zur Verbesserung der innerbetrieblichen Logistik angestrebt. Auf Basis dieser grundlegenden Ausrichtung sowie der in Kapitel 3 identifizierten Anforderungen an ein Konzept zur dynamischen Logistiksteuerung wird in diesem Kapitel der Stand der Wissenschaft hinsichtlich unterschiedlicher Algorithmen zur Optimierung dargestellt.

Die Analyse kommerzieller Konzepte für die fabrikinterne Routenzuglogistik zeigt, dass einige Anbieter Lösungen zur dynamischen Steuerung von Routenzügen anbieten (z. B. *FLX-TLS Transport- / Staplerleitsystem* von FLEXUS oder *RoutMan* von LOGSOL). Die Auffassung des Begriffs „dynamisch“ ist bei diesen Systemen jedoch nicht so umfassend wie in der vorliegenden Arbeit. So werden Routen schon dann als dynamisch bezeichnet, wenn der Abfahrtszeitpunkt keinem starren System folgt, trotzdem aber immer exakt dieselbe Strecke zurückgelegt wird. Im Kontext der vorliegenden Arbeit erfolgt bei dynamischen Routen hingegen auch die Zusammenstellung sowie die Streckenführung für jede Tour individuell.

Kern des in dieser Arbeit entwickelten Konzepts ist ein Algorithmus zur Generierung der Touren, welcher in eine übergeordnete Struktur eingebettet wird. Aus wissenschaftlicher Sicht stellt der Algorithmus neben der Art und Weise der Einbettung in das Logistiksystem einen der Schwerpunkte der Forschungsarbeit dar. Unter diesem Gesichtspunkt ist es erforderlich, im Folgenden die zugrundeliegende Algorithmik im Detail zu betrachten. Deshalb werden in diesem Kapitel zunächst die Algorithmen zum Lösen von Tourenplanungsproblemen vorgestellt und die für die fabrikinterne Materialbereitstellung relevanten Teilprobleme beschrieben und analysiert. Die zur Anwendung des Algorithmus erforderliche Einbettung in das Gesamtkonzept erfolgt nach der Weiterentwicklung des Algorithmus in Abschnitt 5.3.

4.1 Allgemeines

Zur Lösung des Tourenplanungsproblems (siehe Abschnitt 2.3) existieren zahlreiche Lösungsansätze. Wie in Abschnitt 2.2.3 bereits kurz eingeführt, wird dabei unterschieden in *exakte Lösungsverfahren*, welche garantiert die optimale Lösung ermitteln sowie in *Näherungsverfahren*, die in deutlich kürzerer Zeit eine gute Lösung ermitteln, jedoch nicht zwingend das Optimum erreichen. Im Folgenden soll ein Überblick über bestehende Methoden gegeben werden, welche in Abbildung 4.1 aufgelistet sind.

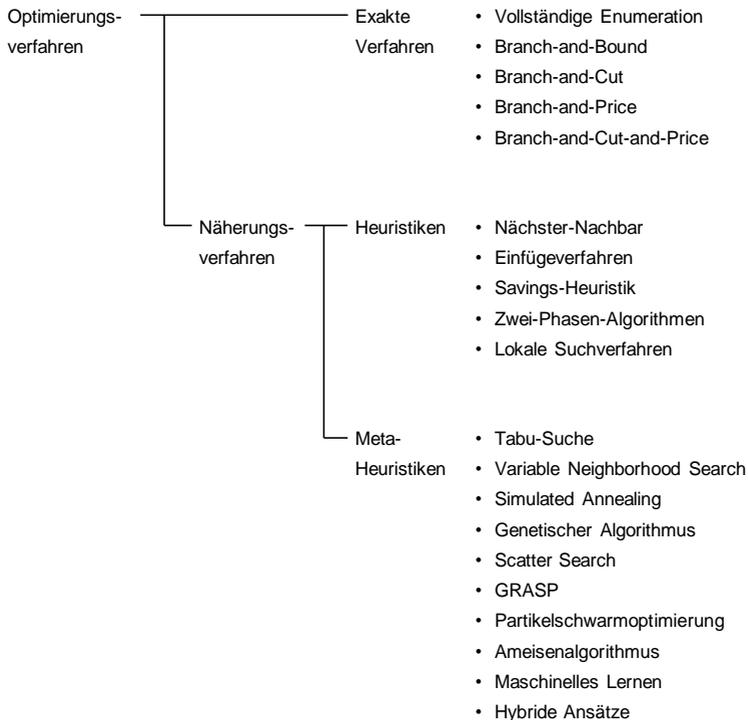


Abbildung 4.1: Klassifizierung der für die Tourenplanung relevanten Optimierungsverfahren (in Anlehnung an CACERES-CRUZ ET AL. 2014, S. 32:7)

Exakte Lösungsverfahren ermitteln zu einem gegebenen Problem die optimale Lösung. TALBI (2009, S. 19) führt hierzu diverse Lösungsverfahren an, begonnen bei

der vollständigen Enumeration bis hin zu den etablierten Branch-and-X-Algorithmen (Branch-and-Bound, Branch-and-Cut, Branch-and-Price, Branch-and-Cut-and-Price). Zur besseren Beherrschung der Problemgröße erfolgt in einigen Fällen eine Zerlegung in kleinere Probleme (FESTA 2014, S. 3). Die Komplexität großer mathematischer Kombinatorikprobleme führt dennoch dazu, dass eine Lösung unter praktischen Gesichtspunkten nicht mehr möglich ist, da die Berechnung zu lange dauern würde.¹ Daher sind exakte Lösungsverfahren trotz permanenter Fortschritte bei den Algorithmen nur bei geringen Problemgrößen einsetzbar (BALDACCI ET AL. 2012, S. 6).²

Näherungsverfahren nähern sich an die optimale Lösung zu einem Kombinatorikproblem an, können dabei allerdings nicht garantieren, dass das Optimum tatsächlich erreicht wird. Im Regelfall sind die Ergebnisse jedoch nahe am Optimum. Da sie auch für große Probleminstanzen eingesetzt werden können, haben sie in den letzten Jahren vermehrt Aufmerksamkeit in der Forschung erfahren (TALBI 2009, S. 23). Näherungsverfahren werden in zwei Kategorien untergliedert: zum einen problemspezifische Heuristiken und zum anderen verallgemeinerte Meta-Heuristiken. Problemspezifische Heuristiken sind auf die spezifische Problemstellung (in diesem Fall das Tourenplanungsproblem) angepasste Verfahren, die gezielt Eigenschaften des Problems zur Lösungsfindung nutzen. Im Gegensatz dazu sind Meta-Heuristiken abstrahierte Problemlösungsverfahren, die für verschiedene Problemstellungen einsetzbar sind. (LABADIE ET AL. 2016, S. 39)

Einfache Heuristiken sind beispielsweise die Nächste-Nachbar-Heuristik (immer Auswahl des nahegelegensten Nachbarknotens), das Einfügeverfahren (Einfügen der übrigen Knoten in die bestehenden Touren) sowie der Zwei-Phasen-Algorithmus (Zerlegung des Problems in Teilbereiche und anschließend Lösung der Teilprobleme) (LABADIE ET AL. 2016, S. 16 ff.). Heuristiken rechnen in der Regel sehr schnell, sind jedoch im Bezug auf Lösungsqualität für die meisten Anwendungsfälle nicht

¹ Für das TSP, welches von der Komplexität etwas geringer ist, weil die Zuordnung von Fahrzeugen zu Touren entfällt, gibt es für ein Problem mit n Städten $\frac{(n-1)!}{2}$ mögliche Lösungen. Werden $n = 50$ Städte betrachtet, so ergibt das über $3 \cdot 10^{62}$ Möglichkeiten. Selbst unter der sehr optimistischen Annahme, dass pro Sekunde eine Million Möglichkeiten berechnet werden können, ergibt sich eine Rechenzeit von $9,6 \cdot 10^{48}$ Jahren.

² Dies gilt vor allem für Probleme der Klasse NP (vgl. Abschnitt 2.3.7). Je nach Anforderung an die Lösungsgeschwindigkeit sind jedoch auch Probleme der Klasse P nicht exakt lösbar. Eine Bewertung, ob eine exakte Lösung sinnvoll ist, muss von Anwendungsfall zu Anwendungsfall individuell durchgeführt werden.

4 Lösungsverfahren für Tourenplanungsprobleme

ausreichend. Sie finden daher hauptsächlich zur Entwicklung initialer Lösungen für andere Algorithmen (z. B. bei vielen Meta-Heuristiken) Anwendung. Eine detaillierte Beschreibung relevanter Heuristiken findet sich in Abschnitt 4.2.1.

Meta-Heuristiken basieren häufig auf Phänomenen aus der Natur (TALBI 2009, S. 25 ff.). Im Gegensatz zu Heuristiken sind sie nicht problemspezifisch, sondern stellen eine abstrahierte Lösungsmöglichkeit für ähnlich strukturierte Probleme dar. Eine Auswahl an Meta-Heuristiken sind beispielsweise das Simulated Annealing (Übertragung des Abkühlprozesses bei Werkstoffen auf die Optimierung), die Tabu-Suche (Erweitertes lokales Suchverfahren, bei dem bereits erprobte Lösungen in einer Tabu-Liste zwischengespeichert werden), der Genetische Algorithmus (Weiterentwicklung der Lösungen durch Mutation und Vererbung) und der Ameisenalgorithmus (Simulation eines Ameisenschwarms) (LABADIE ET AL. 2016, S. 39 ff.). Zahlreiche weitere Verfahren existieren und sind in Abschnitt 4.2.2 beschrieben. Obwohl Meta-Heuristiken die optimale Lösung nicht garantieren können, sind sie aufgrund der schnellen Ermittlung der Lösungen die am häufigsten eingesetzte Klasse an Lösungsmethoden.

Im Folgenden werden in Abschnitt 4.2.1 die Funktionsweise einiger relevanter Heuristiken vorgestellt, welche als wichtiger Bestandteil in vielen Algorithmen eingesetzt werden. Im Anschluss liegt der Fokus in Abschnitt 4.2.2 auf Meta-Heuristiken, welche für die gegebene Problemklasse die am häufigsten eingesetzte Algorithmusgruppe darstellt. Ein kurzer Einblick in hybride Algorithmen in Abschnitt 4.2.3 schließt den Abschnitt ab.

4.2 Übersicht über Lösungsverfahren

4.2.1 Heuristiken

Heuristiken sind problemspezifisch, d. h. sie werden direkt für das zu lösende Problem entwickelt. Sie nutzen zur Lösungsfindung vorab bekannte Eigenschaften des zu lösenden Problems aus. Die meisten Heuristiken können hinsichtlich der Lösungsqualität nicht mit Meta-Heuristiken konkurrieren, sind jedoch dafür in der Lage,

sehr schnell Lösungen zu generieren. Daher sind sie oft als Eröffnungsverfahren³ zur Generierung initialer Lösungen oder als lokale Suchverfahren eingebettet in andere Algorithmen anzutreffen.

Nächste-Nachbar-Heuristik

Die Nächste-Nachbar-Heuristik gehört zu den Heuristiken, die nur eine einzige Lösung generieren ohne diese mit anderen Möglichkeiten zu vergleichen (LABADIE ET AL. 2016, S. 16). Zur Generierung der Lösungen wird von der aktuellen Station die nächstgelegene Station ausgewählt und als nächstes Element in die Tour eingeplant. Die Heuristik liefert für die ersten Stationen gute Ergebnisse, die Vervollständigung der Touren erfordert jedoch zum Schluss sehr suboptimale Verbindungen zur Integration der verbliebenen Stationen. Zudem kann sich durch die Wahl des Startpunktes die Lösungsqualität unvorhersehbar ändern. (DITTES 2015, S. 86)

Einfügeverfahren

Einfügeverfahren starten mit einer sehr kurzen Tour, die in der Regel nur aus zwei Stationen besteht. Dann werden nach und nach alle weiteren Stationen jeweils so zur bestehenden Tour hinzugefügt, dass sich durch den zusätzlichen „Umweg“ die Tourlänge so wenig wie möglich verlängert. Es gibt unterschiedliche Strategien, wie die initiale Tour konstruiert wird, wie die nächste hinzuzufügende Station ausgewählt wird und wo diese eingefügt wird (LABADIE ET AL. 2016).

Die Lösungsqualität lässt sich bei Einfügeverfahren zwar nicht genau vorhersehen, weil wie bei jedem Näherungsverfahren die Gefahr besteht, in einem lokalen Optimum zu landen, allerdings kann zumindest bewiesen werden, dass bei Einfügen mit der kürzesten Wegstreckenverlängerung die Strecke maximal doppelt so lange ist, wie die der optimalen Tour (WENDT 1995, S. 17).

³ Viele komplexere Algorithmen benötigen zum Start der Optimierung eine initiale gültige Lösung, von der aus nach besseren Lösungen gesucht werden kann. Als Eröffnungsverfahren wird der Bestandteil eines Algorithmus bezeichnet, der diese erste initiale Lösung generiert.

Savings-Heuristik

Die Savings-Heuristik (deutsch: Sparalgorithmus) wurde von CLARKE & WRIGHT (1964) direkt für das VRP entwickelt und basiert auf der Ermittlung von Einsparungen durch das Ersetzen von Kanten (VIDAL ET AL. 2013, S. 4).

Zu Beginn werden alle zu beliefernden Stationen mit jeweils einer individuellen Tour beliefert und sind daher mit je einem Hin- und einem Rückpfeil mit dem Depot verbunden. Im nächsten Schritt werden je zwei Touren miteinander kombiniert. Um zu ermitteln, welche Kombinationen von Stationen die optimale neue Tour ergeben, wird immer für zwei Stationen der sogenannte *Savings-Wert* errechnet (BRÄYSY & GENDREAU 2005). Dieser Wert gibt an, um wie viel kürzer die gesamte Tourlänge wird, wenn die beiden Stationen in einer gemeinsamen Tour beliefert werden. Mit der Kantenlänge d ermittelt sich der Wert für die Stationen i und j durch $S_{i,j} = d_{i,rück} + d_{j,hin} - d_{i,j}$ (RICHTER 2005, S. 45 ff.).

Für die gültige Kombination mit dem höchsten Savings-Wert wird wie in Abbildung 4.2 für jede Station jeweils eine Kante zum Depot entfernt sowie eine neue zwischen den beiden Stationen eingefügt. Im Anschluss wird erneut für alle Stationspaare, die noch über direkte Kanten zum Depot verfügen, der Savings-Wert errechnet und ein neues Paar optimiert (RICHTER 2005, S. 45 ff.).

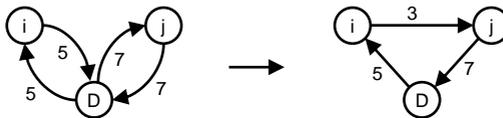


Abbildung 4.2: Schematisches Vorgehen der Savings-Heuristik (in Anlehnung an RICHTER 2005, S. 46)

Aufgrund der einfachen Nachvollziehbarkeit und der guten Implementierbarkeit erfreut sich die Savings-Heuristik nach wie vor großer Beliebtheit (CACERES-CRUZ ET AL. 2014, S. 32:9).

Zwei-Phasen-Algorithmen

Die grundlegende Idee von Zwei-Phasen-Algorithmen ist die Rückführung des VRP auf das TSP. Somit muss ein Algorithmus nur noch die Reihenfolge der Stationen

optimieren, ein anderer übernimmt davon unabhängig die Zuteilung zu Routen. Die beiden Probleme einzeln zu betrachten senkt die Komplexität erheblich. Zur Umsetzung existieren zwei Möglichkeiten:

Die Mehrzahl der Algorithmen clustert zunächst die einzelnen Stationen zu Bereichen, für welche im nachfolgenden Schritt einzeln die Touren konstruiert werden (*Cluster-first, route second*). Ein Beispiel ist der *Sweep*-Algorithmus, der ausgehend von einer frei gewählten Station um das Depot herum die Stationen in Segmente einteilt. Für diese so zusammengefassten Stationen werden dann Routen wie beim TSP berechnet. Häufig ist das letzte zusammengestellte Segment kleiner als die übrigen, wodurch die Lösungsgüte herabgesetzt wird, außerdem ist das Verfahren für durch Nebenbedingungen eingeschränkte Tourenplanungsprobleme nur schwer einsetzbar. (LABADIE ET AL. 2016, S. 19)

Das umgekehrte Vorgehen ist ebenso möglich (*Route-first, cluster-second*). Dazu wird zuerst eine lange Tour mit allen Stationen gebildet, die im Anschluss jeweils nach Erreichen der Kapazitätsgrenze in Einzeltouren geschnitten wird (BEASLEY 1983).

Da in Zwei-Phasen-Algorithmen grundsätzlich beliebige Algorithmen verwendet werden können, ist auch eine Integration der Meta-Heuristiken aus Abschnitt 4.2.2 möglich. Dies konnte in den letzten Jahren die Leistungsfähigkeit einiger Algorithmen signifikant steigern. (vgl. PRINS 2004; HAGHIGHI ET AL. 2009)

Lokale Suchverfahren

Lokale Suchverfahren sind Verfahren zur Verbesserung einer bestehenden Lösung und eignen sich daher nicht als Eröffnungsverfahren. Sie suchen in der direkten Umgebung einer initialen Lösung Ψ^0 , die z. B. durch eine Heuristik erzeugt sein kann, nach verbesserten Lösungen Ψ' in der direkten Umgebung der bisherigen Lösung. Dabei wenden sie sogenannte *Moves* an, um ähnliche Lösungen zu ermitteln. Wird dadurch eine verbesserte Lösung Ψ' gefunden, so wird Ψ durch diese ersetzt und von dieser Lösung aus weiter gesucht. (LABADIE ET AL. 2016, S. 23 ff.)

Der einfachste Move ist die Verschiebung einer Station, die an einer anderen Stelle der Tour oder in eine andere Tour wieder eingefügt wird (Abbildung 4.3a). Alternativ können auch zwei Stationen gegeneinander ausgetauscht werden. Zusätzlich gibt es den Kantenaustausch, bei dem k Kanten von einer Route entfernt und mit k

4 Lösungsverfahren für Tourenplanungsprobleme

anderen Kanten verbunden werden (S. LIN & KERNIGHAN 1973, S. 502), wie in Abbildung 4.3b dargestellt. Der λ -Austausch verschiebt wahlweise eine Kette mit der Länge von λ Stationen, oder vertauscht wie in Abbildung 4.3c zwei Ketten mit einer Länge von λ Stationen (LABADIE ET AL. 2016, S. 23 ff.; OSMAN 1993, S. 427).

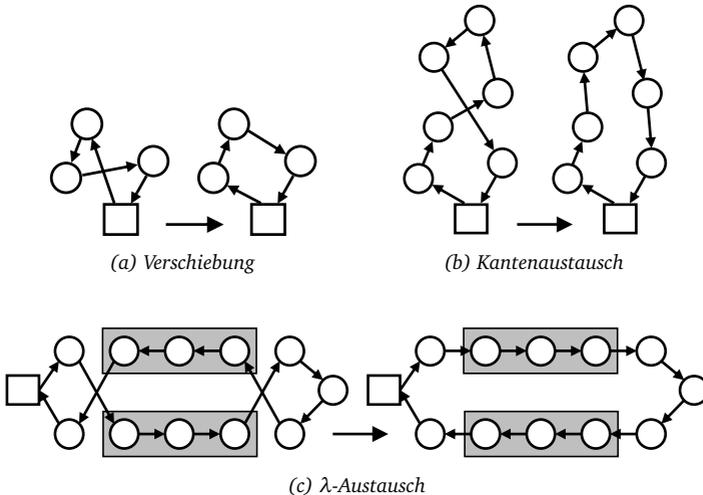


Abbildung 4.3: Moves für lokale Suchverfahren

Lokale Suchverfahren sind selten als eigenständiger Algorithmus anzutreffen, vielmehr werden sie in den meisten Meta-Heuristiken zur Verbesserung der ermittelten Lösungen eingesetzt (z. B. MONTEMANNI ET AL. 2005; TURAN ET AL. 2017; ARMAS & MELIÁN-BATISTA 2015).

4.2.2 Meta-Heuristiken

Im Gegensatz zu Heuristiken sind Meta-Heuristiken a priori von einem konkreten Problem unabhängig. Sie definieren eine Folge von durchzuführenden Schritten, die teilweise problemspezifisch angepasst werden können. Um nicht in lokalen Optima zu enden, akzeptieren viele Meta-Heuristiken auch eine temporäre Verschlechterung des aktuellen Lösungswerts.

In den folgenden Abschnitten sind die für das VRP relevanten Meta-Heuristiken in ihrer grundlegenden Funktionsweise erläutert. Für Details zu den Lösungsmethoden sei auf die dort jeweils angegebene Literatur verwiesen.

Tabu-Suche

Die Tabu-Suche ist ein von GLOVER (1986) entwickeltes iteratives meta-heuristisches Lösungsverfahren für komplexe Probleme, welches zur Klasse der lokalen Suchverfahren⁴ gehört (STEPANENKO 2008, S. 37). Seine Grundideen finden sich in vielen anderen Heuristiken und Meta-Heuristiken wieder.

Zu Beginn der Lösungsfindung wird eine Initial-Lösung generiert (zufällig oder mit einer dafür geeigneten Heuristik), von der aus die eigentliche Optimierung startet. Im nächsten Schritt werden in der Nachbarschaft der aktuellen Lösung neue Lösungen konstruiert, wovon die beste ausgewählt wird, auch falls diese schlechter sein sollte, als die aktuelle Lösung. Zusätzlich wird die vorhergehende Lösung auf die Tabu-Liste gesetzt, womit sie in kommenden Zügen nicht mehr als Lösung in Frage kommt („tabu ist“). Dies verhindert das Verharren in einem lokalen Optimum. (GLOVER 1990, S. 76 ff.)

Befindet sich der Algorithmus in einem lokalen Optimum, so sind alle umliegenden Lösungen schlechter als die aktuelle. Die Tabu-Suche wählt in diesen Fällen die Lösung mit der geringsten Verschlechterung aus. Im nächsten Schritt wird ebenso vorgegangen – da die vorangegangenen Lösungen jeweils bereits auf der Tabu-Liste stehen, bewegt sich der Algorithmus langsam vom lokalen Optimum weg bis wieder mit der Standardvorgehensweise nach neuen Optima gesucht werden kann. Die Tabu-Suche wird in der Regel nach einer bestimmten Anzahl an Iterationen oder nach einer maximalen Anzahl an Iterationen ohne Verbesserung der bisher besten ermittelten Lösung beendet. (STEPANENKO 2008, S. 37 ff.)

Es existieren zahlreiche Möglichkeiten zur Optimierung des Algorithmus. Beispielsweise kann in einen Kurzzeit- und einen Langzeitspeicher unterschieden werden, womit sich präziser steuern lässt, welche Züge bei der Entscheidung für eine Lösung

⁴ Lokale Suchverfahren können als wie Abschnitt 4.2.1 als Heuristiken ausgelegt sein oder wie hier in Form einer abstrahierten Meta-Heuristik.

4 Lösungsverfahren für Tourenplanungsprobleme

erlaubt sind. Auch ein „Häufigkeiten-Speicher“, der bei einer Rückkehr zu einer bereits geprüften Lösung verhindert, dass der gleiche Pfad wieder eingeschlagen wird, ist eine von vielen Erweiterungsmöglichkeiten. Eine umfassende Übersicht dazu geben PIRIM ET AL. (2008, S. 5 ff.).

Die Tabu-Suche wird in mehreren verschiedenen Varianten auf das VRP angewendet. HO & SZETO (2014) wenden die Tabu-Suche auf ein Tourenplanungsproblem an, bei dem Ausgleichstransporte für das Bike-Sharing in Großstädten optimiert werden. LEUNG ET AL. (2011) formulieren ein um starke Nebenbedingungen erweitertes Tourenplanungsproblem und führen zur Beschleunigung eine Bestrafung bestimmter Areale des Suchraums ein. Die Erhöhung der Rechengeschwindigkeit ist auch das Ziel von SZYMON & DOMINIK (2013), die mehrere Tabu-Suchen parallel auf für parallele Berechnungen optimierten Grafikkartenprozessoren implementieren und somit erhebliche Geschwindigkeitsvorteile generieren können. Die Anwendungsbeispiele zeigen, dass die Verwendung innerhalb eines dynamischen Systems sowohl von der Rechengeschwindigkeit, als auch von der Lösungsqualität her prinzipiell denkbar ist.

Variable Neighborhood Search

Auch die Variable Neighborhood Search (VNS) (deutsch: Variable Nachbarschaftssuche) ist eine auf der lokalen Suche basierende Meta-Heuristik zum Lösen kombinatorischer Optimierungsprobleme. Sie durchsucht benachbarte Lösungen systematisch, um sowohl zu lokalen Optima zu finden, als auch aus diesen wieder zu entkommen. (MLADENović & HANSEN 1997)

Der VNS nach HANSEN & MLADENović (2005, S. 213) liegen drei grundlegende Erkenntnisse zugrunde:

- ERKENNTNIS 1: Ein lokales Minimum in einer Nachbarschaftsstruktur ist nicht zwangsläufig auch eines einer anderen Nachbarschaftsstruktur.
- ERKENNTNIS 2: Ein globales Minimum ist ein lokales Minimum hinsichtlich aller Nachbarschaftsstrukturen.
- ERKENNTNIS 3: Für viele Probleme liegen die lokalen Minima hinsichtlich einer oder mehrerer Nachbarschaftsstrukturen relativ nahe zusammen.

Die letzte Beobachtung – welche rein empirischer Natur ist – impliziert, dass ein lokales Optimum oft Informationen über das globale Optimum beinhaltet. Dies können beispielsweise einige Parameter mit demselben Wert in beiden Lösungen sein, wobei im Regelfall jedoch nicht ersichtlich ist, welche der Parameter diese sind. (HANSEN & MLADENOVIC 2009, S. 3976)

Zur Durchführung wird zunächst mit einer initialen Lösung Ψ^0 gestartet, welche beispielsweise durch eine Heuristik ermittelt werden kann. Von dieser Lösung können Nachbarschaften bestimmt werden, d. h. Lösungsräume, in denen die Lösungen in einigen Parametern Ähnlichkeit mit Ψ^0 haben. Zunächst wird mit einer sehr kleinen Nachbarschaft begonnen und dort nach neuen Optima gesucht. Die Nachbarschaftsräume werden sukzessive vergrößert, falls keine bessere Lösung gefunden werden kann. Im Falle einer besseren Lösung wird wieder mit einem sehr kleinen Nachbarschaftsraum von dieser Lösung aus gestartet.

Bei der einfachsten Variante, der sogenannten Variable Neighborhood Descend, wird aus dem betrachteten Nachbarschaftsraum jeweils immer die beste Lösung ausgewählt. Die klassische VNS wählt hingegen stochastisch ein Element aus, um zu verhindern, dass sich der Algorithmus zyklisch im Kreis bewegt. Für die so ausgewählte Lösung Ψ' wird ein lokales Optimierungsverfahren⁵ angewendet, das Ergebnis wird in Ψ'' gespeichert. Falls Ψ'' eine bessere Lösung darstellt als Ψ , so wird diese Lösung dort abgespeichert und die nächste Iteration gestartet. Falls nicht, wird der Raum der Nachbarschaften entsprechend vergrößert und dort analog verfahren. (HARTL 2014)

Ergibt sich nach einer vordefinierten Anzahl an Iterationen keine Verbesserung mehr, so kann das sogenannte *Shaking* (vgl. BELHAIZA ET AL. 2014, S. 271) eingesetzt werden, bei dem zufällig aus einer größeren Nachbarschaft eine Lösung ausgewählt wird, unabhängig davon, ob sie eine Verbesserung oder Verschlechterung darstellt. Alternativ löst das *skewed VNS* (HANSEN & MLADENOVIC 2005, S. 225 f.) diese Herausforderung, indem auch geringfügig schlechtere Lösungen akzeptiert werden, sofern sich diese von der aktuellen Lösung ausreichend stark unterscheiden. Somit wird erreicht, dass der Algorithmus auch noch in Bereichen des Lösungsraums nach

⁵ Unter lokalen Optimierungsverfahren werden einfache Heuristiken verstanden, die ggf. die Lösung verbessern können. Dies kann beispielsweise die zufällige oder gezielte Vertauschung zweier Elemente sein (siehe lokale Suchverfahren, Abschnitt 4.2.1).

4 Lösungsverfahren für Tourenplanungsprobleme

dem globalen Optimum sucht, die weiter von dem lokalen Optimum entfernt sind. (HANSEN & MLADENOVIC 2005, S. 225 f.; HARTL 2014)

Für das Tourenplanungsproblem werden verschiedene Ansätze in Veröffentlichungen diskutiert. LABADIE ET AL. (2012) lösen ein Problem, bei dem innerhalb eines gegebenen Zeitfensters möglichst viele Stationen eines Netzes besucht werden müssen, TURAN ET AL. (2017) beschäftigen sich mit einem Tourenplanungsproblem mit stochastischen Bedarfen. Dynamische Probleme werden sowohl von ARMAS & MELIÁN-BATISTA (2015) betrachtet, welche als Besonderheit die Zeitfenster als schwache Nebenbedingungen formulieren, als auch von C. LIN ET AL. (2014), welche ein praktisches Kurierproblem lösen, bei dem der Fahrer erst nach der Ankunft an einem Zielort über das nächste Ziel informiert wird. In allen Fällen eignet sich der Algorithmus gut zur Lösung, da auch innerhalb von kurzen Rechenzeiten akzeptable Lösungen generiert werden können.

Simulated Annealing

Das Simulated Annealing (SA) (deutsch: *Simulierte Abkühlung*) ist eine an den physikalischen Prozess der Abkühlung geschmolzener Metalle angelehnte Meta-Heuristik. Während beim Abkühlen zu Beginn noch häufige Gefügeumwandlungen stattfinden, stellt sich bei ausreichend langsamer Abkühlung ein Gleichgewicht mit minimaler Energie ein. (FINK 2017)

Zu Beginn wird eine initiale Lösung benötigt, von der aus wie bei der VNS in der Nachbarschaft nach besseren Lösungen gesucht wird. Ist eine gefundene Lösung besser, so wird diese sofort akzeptiert. Im Falle einer Verschlechterung wird sie mit einer gewissen Wahrscheinlichkeit u. U. trotzdem angenommen. Diese Wahrscheinlichkeit ergibt sich in Analogie zur Abkühlung über die Formel $P = e^{-\Delta z/T}$, wobei Δz dem Wert der Verschlechterung entspricht und T einem der Temperatur entsprechenden Parameter (EGLESE 1990, S. 1 f.; VOSS 2009, S. 2065). Solange die Temperatur T zu Beginn der Optimierung hoch ist, besteht demnach eine hohe Wahrscheinlichkeit, eine Verschlechterung zu akzeptieren. Je länger das Optimierungsverfahren läuft, desto weniger wird außerhalb des lokalen Suchraums gesprungen (PARDALOS & MAVRIDOU 2009, S. 3592).

Für ein Tourenplanungsproblem unter speziellen Nebenbedingungen für die Beladung von Anhängern setzen S. LIN ET AL. (2011) SA erfolgreich ein, jedoch mit zu hohen Rechenzeiten für den dynamischen Einsatz. Diesem Problem entgegen WANG ET AL. (2015) mit der Parallelisierung des Algorithmus, was die Rechenzeiten zwar signifikant verringert, die grundsätzliche Problematik der zu hohen Rechenzeiten jedoch nicht gänzlich lösen kann.

Genetischer Algorithmus

Die Klasse der Evolutionären Algorithmen bildet das biologische Prinzip der Vererbung nach. In der ursprünglichen Entwicklungsform unterscheiden sich der Evolutionäre und der Genetische Algorithmus in ihrer Darstellungsform der Lösungskombinationen. Mit zunehmender Weiterentwicklung verschwimmen die Unterschiede jedoch immer weiter, so dass in vielen Fällen mittlerweile für beide Repräsentationen die Bezeichnung Genetischer Algorithmus verwendet wird. (MICHALEWICZ 1996, S. 8 f.; NIEHUES 2017, S. 42)

Zunächst müssen für das Problem mehrere Lösungen in Form von Individuen generiert werden, welche durch die Aneinanderreihung von *Genen* innerhalb eines *Chromosoms* dargestellt sind. Dies geschieht wahlweise zufällig oder durch initiale Heuristiken (siehe Abschnitt 4.2.1). Im nächsten Schritt wird mittels der sog. *Selektion* ein Mutter- sowie ein Vaterindividuum durch den Vergleich der Fitness-Funktion⁶ jeweils zweier zufällig ausgewählter Individuen gewählt. Die Fitness beschreibt dabei die Güte der Lösung – übertragen auf die Biologie stellt sie die Überlebenswahrscheinlichkeit des Individuums dar. In der anschließenden *Kombination* werden die Gene von Mutter und Vater vermischt und ergeben somit neue Gene für das Kind. Im nächsten Iterationsschritt, der *Mutation*, wird, abgebildet durch verschiedene Wahrscheinlichkeiten, eine zufällige Modifikation der Gene des Kindes durchgeführt. Anschließend startet der Algorithmus mit einer erneuten *Selektion* in die nächste Iteration. Das Vorgehen wird solange iterativ fortgesetzt, bis ein anfangs definiertes Abbruchkriterium erfüllt ist (z. B. Rechenzeit oder Anzahl an Iterationen). (RIMSCHA 2014, S. 5; GERDES ET AL. 2004, S. 36 ff.)

⁶ Die Fitness-Funktion beschreibt denselben Sachverhalt wie die im OR eingeführte Zielgröße in Abschnitt 2.2.2.

4 Lösungsverfahren für Tourenplanungsprobleme

Zahlreiche Strategien zur Abwandlung des Grundalgorithmus wurden entwickelt, wie z. B. der Elitismus, bei dem das Individuum mit dem besten Fitnesswert einer Population in der nächsten Generation erneut in die Population übernommen wird oder der statische Austausch, bei dem pro Generation ein oder zwei Individuen durch neue ersetzt werden. Diese Erweiterungen führen nicht zwangsläufig zu einer besseren Lösung, vielmehr muss problemspezifisch entschieden werden, welche Abwandlung des Algorithmus die beste Lösung erzielen kann. (MITCHELL 1998, S. 328)

Anwendungsfälle für Genetische Algorithmen im Bereich der Tourenplanung liefern beispielsweise HAGHANI & JUNG (2005), die ein dynamisches Tourenplanungsproblem mit zeitpunktabhängigen Streckenzeiten modellieren. KEPAPTSOGLU ET AL. (2015) wenden den Genetischen Algorithmus ebenfalls auf ein dynamisches Problem an, wobei Wetterprognosen für die Abschätzung der Geschwindigkeiten von Containerschiffen genutzt werden. Ein dynamisches Just-In-Time-Szenario wird in NOVAES ET AL. (2015) modelliert und gelöst, dabei kann durch die geschickte Einplanung zusätzlicher Lastwagen die Wirtschaftlichkeit des Systems erhöht werden. Die Leistungsfähigkeit liegt zusammengefasst im Mittelfeld der hier vorgestellten Algorithmen, sowohl bei der Lösungsqualität als auch bei der Berechnungsgeschwindigkeit müssen Abstriche in Kauf genommen werden. Dennoch zeigen die Anwendungsfälle, dass ein Einsatz in dynamischen Systemen unter bestimmten Rahmenbedingungen denkbar ist.

Scatter Search

Die Scatter Search gehört wie der genetische Algorithmus ebenfalls zur Klasse der Evolutionären Optimierungsverfahren, unterscheidet sich jedoch von den Genetischen Algorithmen grundlegend durch die Art und Weise der Modifikation der Populationen (WEICKER 2015, S. 172 ff.). Im Gegensatz zum Genetischen Algorithmus erfolgen die Veränderungen hier allerdings nicht zufällig (GLOVER ET AL. 2000, S. 653 f.).

Zu Beginn der Scatter Search wird eine große Menge an Startlösungen erzeugt. Dabei ist zum einen wichtig, dass diese möglichst hochqualitativ sind, was durch

eine Verbesserungsmethode⁷ sichergestellt werden kann. Zum anderen ist jedoch ebenso entscheidend, dass diese eine möglichst hohe Streuung aufweisen, so dass ein großer Bereich abgesucht wird (MARTÍ ET AL. 2006, S. 360 ff.). In den folgenden Schritten werden erfolgversprechende Lösungen – das sind wahlweise besonders gute Lösungen oder besonders weit verstreute Lösungen – für die weiteren Lösungsschritte abgespeichert. Aus diesem Set an Lösungen werden dann Bestandteile der Lösungen extrahiert, aus denen im Folgenden wiederum neue Lösungen zusammengesetzt werden können. Dieses neue Zusammensetzen der Bestandteile kann entweder zufällig passieren oder es wird durch Linearkombinationen bzw. problemspezifische Mechanismen umgesetzt (GLOVER ET AL. 2000, S. 90). Die neuen Lösungen werden ebenfalls wieder durch eine Verbesserungsmethode lokal optimiert und die besten Lösungen (gute Erfüllung der Zielfunktion oder weite Streuung) für die weiteren Iterationen verwendet. (REGO & LEÃO 2005, S. 3 f.; DU & SWAMY 2016)

MARTÍ ET AL. (2005, S. 277) stellen fest, dass die Scatter Search durch ihr systematischeres Vorgehen schneller zu guten Lösungen kommt als der Genetische Algorithmus, sich die Lösungsqualität jedoch nur geringfügig verbessert. ZHANG ET AL. (2012) stellen für ein VRP fest, dass die Scatter Search vor allem bei großen Problemstellungen gegenüber dem Genetischen Algorithmus überlegen ist. Sowohl BELFIORE & YOSHIKAZAKI (2009) als auch TANG ET AL. (2010) setzen das Lösungsverfahren erfolgreich zur Lösung praktischer Tourenplanungsprobleme mit Kapazitätsrestriktionen ein, haben dabei jedoch keine Echtzeitanforderungen zu erfüllen.

Greedy Randomized Adaptive Search Procedure

Greedy-Algorithmen („gierige Algorithmen“) sind Algorithmen, die sich bei jeder Entscheidung immer für die Möglichkeit des zum aktuellen Zeitpunkt größten Gewinns entscheiden, dadurch jedoch in Konsequenz anfällig für das Verbleiben in lokalen Optima sind (CORMEN 2009, S. 414). Die Greedy Randomized Adaptive Search Procedure (GRASP) nach FEO & RESENDE (1989) verfolgt den Ansatz, diese Greedy-Algorithmen so zu randomisieren, dass trotzdem eine möglichst global optimale Lösung gefunden wird.

⁷ Siehe lokale Suchverfahren, Abschnitt 4.2.1.

4 Lösungsverfahren für Tourenplanungsprobleme

Dazu sind zwei Strategien denkbar. Entweder werden aus dem gesamten Lösungsraum möglichst zufällig mehrere Lösungen ausgewählt, woraus dann durch den Greedy-Algorithmus die Beste gewählt wird. Alternativ können mittels des Greedy-Algorithmus die besten Lösungen des Lösungsraums in die nähere Auswahl genommen werden und dann daraus zufällig eine Lösung ausgewählt werden. Dieses Verfahren kann iterativ wiederholt werden, wobei eine Schwachstelle des ursprünglichen Algorithmus ist, dass von einer zur nächsten Iteration kein Lerneffekt auftritt. Daher wurden Abwandlungen der GRASP entwickelt, welche diesen Lernprozess in den Algorithmus integrieren (z. B. PRAIS ET AL. 1998).

Für Tourenplanungsprobleme wird GRASP von FERRER ET AL. (2016) für die Koordination der Versorgung nach Katastrophen eingesetzt, wobei als Zielgröße die möglichst gleichmäßige Versorgung gewählt wird. PINO ET AL. (2012) wenden den Algorithmus auf ein Tourenplanungsproblem mit Zeitfenstern an, um die Anzahl nicht rechtzeitig beliefelter Kunden zu senken. LABADIE ET AL. (2016, S. 43) stellen zusammenfassend fest, dass die Lösungsqualität nicht mit anderen Algorithmen konkurrieren kann, es jedoch viele Ansätze zur Kombination mit anderen Algorithmen gibt.

Partikelschwarmoptimierung

Die Partikelschwarmoptimierung (PSO) geht auf die Arbeiten von EBERHART & KENNEDY (1995) zurück und ist dem Vorbild des natürlichen Schwarmverhaltens nachgebildet.

Individuen eines Schwarms folgen gewissen Gesetzmäßigkeiten bei ihren Bewegungen. Zum einen versuchen sie, bei der Gruppe zu bleiben und demnach konsequenterweise auch der Gruppenbewegung zu folgen, zum anderen halten sie einen Mindestabstand zu anderen Individuen ein (REYNOLDS 1987, S. 25 ff.). Dieses Verhalten wird auf die Optimierung angewendet und bewirkt, dass ein Raum durch die einzelnen Individuen systematisch abgesucht wird, durch deren Mindestabstand jedoch die Wahrscheinlichkeit für das Verbleiben in lokalen Minima deutlich reduziert wird. HEPPNER & GRENANDER (1990, S. 234) fügen der Modellierung des Schwarms noch Regionen hinzu, die für die Individuen des Schwarms besonders attraktiv sind, so dass der Schwarm durch die Umgebung ebenfalls beeinflusst wird. Aufgrund

ihrer Verteilung auf viele Einzelindividuen lässt sich die PSO gut parallelisieren und somit die Effizienz der Berechnung steigern (DALI & BOUAMAMA 2015).

Die erste Anwendung des PSO für VRP stammt von AI & KACHITVICHYANUKUL (2007), die den Algorithmus für die Lösung des kapazitätsbeschränkten Tourenplanungsproblems verwenden. M.-C. CHEN ET AL. (2016) lösen ein erweitertes Tourenplanungsproblem mithilfe eines um lernende Mechanismen erweiterten PSO, NOROUZI ET AL. (2015) wenden die Partikelschwarmoptimierung auf ein VRP mit konkurrierenden Routen an. In den meisten Fällen erzielt die PSO gute Ergebnisse, kann jedoch in Bezug auf die Rechenzeit keine neuen Maßstäbe setzen (LABADIE ET AL. 2016, S. 91; NOROUZI ET AL. 2015).

Ameisenalgorithmus

Der Ameisenalgorithmus ist eine von der Futtersuche der Ameisen inspirierte Meta-Heuristik und geht auf die Arbeiten von COLORNI ET AL. (1992) zurück. Ameisenkolonien sind ohne Zuhilfenahme optischer Anhaltspunkte in der Lage, den kürzesten Weg zwischen Bau und Futterquelle zu finden (DORIGO ET AL. 1996, S. 29 ff.).

Zunächst muss das VRP für die Verwendung im Ameisenalgorithmus (englisch: Ant Colony Optimization (ACO)) als Graph formuliert werden (siehe Abschnitt 2.3.2). Zu Beginn starten mehrere Ameisen vom Knoten des Depots und suchen dann entlang der Kanten eine möglichst optimale Route, um alle Stationen zu beliefern. Nach der Lösungskonstruktion folgt die Ablage von Duftstoffen (sogenannten Pheromonen) auf den Kanten. Dabei können unterschiedliche Strategien zum Einsatz kommen: Wahlweise werden die Pheromone für jede Lösung in Relation zur Lösungsqualität platziert, alternativ kann auch immer nur die beste Lösung zur Ablage von Pheromonen verwendet werden (Elitismusstrategie). Danach startet die zweite Iteration, wobei die Pheromone auf den Kanten bereits zum effizienteren Finden einer guten Lösung verwendet werden. Um das Verharren innerhalb lokaler Optima zu verhindern, kann auf den viel verwendeten Kanten der Pheromonwert reduziert werden um das Erkunden neuer Kanten zu forcieren. (DORIGO ET AL. 1996, S. 29 ff.; DORIGO & GAMBARDELLA 1997a, S. 53 f.)

4 Lösungsverfahren für Tourenplanungsprobleme

Optional können in jeder Iteration vor dem Pheromonupdate sog. *Daemon Actions* ausgeführt werden, d. h. beispielsweise die Optimierung der Lösung durch lokale Optimierungsverfahren (DORIGO & STÜTZLE 2004, S. 38).

Aufgrund ihrer dem VRP sehr ähnlichen Modellierung des Problems gibt es zahlreiche Anwendungsfälle des ACO. So setzen GAJPAL & ABAD (2009) und MONTEMANNI ET AL. (2005) den Ameisenalgorithmus jeweils für dynamische Tourenplanungsprobleme ein, bei denen die Aufträge erst während der Durchführung eintreffen. DONATI ET AL. (2008) sowie BALSEIRO ET AL. (2011) beschreiben eine Implementierung des ACO, der dynamische Zeitanteile bei den Streckenzeiten berücksichtigt. SCHYNS (2015) betrachtet ein dynamisches System an einem Flughafen, bei dem die Lieferungen möglichst immer schon zu Beginn des Zeitfensters ausgeführt werden sollten, um möglichst reaktionsfähig zu bleiben. Die Rechenzeiten sind in einem für dynamische Probleme akzeptablen Rahmen, speziell der Ansatz von SCHYNS (2015) wurde explizit für ein dynamisches Problem entworfen.

Maschinelles Lernen

Auf dem Gebiet des maschinellen Lernens werden derzeit große Fortschritte gemacht. Für die Lösung von Tourenplanungsproblemen sind v. a. Künstliche Neuronale Netze (KNN) relevant, die vom Aufbau, der Organisation und der Funktionalität vom menschlichen Hirn inspiriert sind. Ausgangspunkt der Entwicklung war die Erkenntnis, dass das Gehirn trotz unterlegener Rechenleistung im Vergleich zu heutigen Computern in vielen Gebieten mit Leichtigkeit deutlich überlegene Ergebnisse erzielt (STEINHAUS 2015, S. 3). KNN haben sich in vielen Teilgebieten etabliert und erreichen dort gute Ergebnisse (ERTEL 2016, S. 9 ff.).

Ein neuronales Netz besteht aus mehreren Schichten von Neuronen, die untereinander verbunden sind. Der ersten Schicht werden die Eingangsparameter übergeben, die Neuronen verarbeiten diese Eingangsdaten und geben das Ergebnis an alle nachfolgenden Neuronen weiter, welche wiederum eine Weiterverarbeitung und die Weitergabe der Ergebnisse veranlassen. Die Art und Weise, wie die Parameter im Neuron verarbeitet werden, sowie die Intensität der Weitergabe an die nachfolgenden Neuronen ist dabei a priori noch nicht festgelegt, dies wird durch den selbstlernenden Charakter der Netze mit der Zeit optimiert.

Im Bereich des VRP sind vor allem die von HOPFIELD (1982) vorgeschlagenen Hopfield-Netze zur Lösung anzutreffen. Im Gegensatz zum allgemeinen Aufbau der Netze gibt es hier nur eine Schicht an Neuronen, wobei der Ausgang eines jeden Neurons mit dem Eingang sämtlicher anderer Neuronen verbunden ist. Ein zweiter erfolgreicher Ansatz sind die sogenannten Self-Organizing Feature Maps (SOFM) (deutsch: Selbstorganisierende Karten). Beide Verfahren sind primär für das TSP geeignet, es muss demnach in einem vorgelagerten Schritt durch einen separaten Algorithmus eine Zuweisung der Stationen zu den einzelnen Touren gemacht werden.⁸

GHAZIRI & OSMAN (2006) wenden eine SOFM auf ein VRP an, bei dem sowohl Ablieferungen als auch Abholungen berücksichtigt werden mussten. CRÉPUT ET AL. (2007) und HAGHIGHI ET AL. (2009) lösen das Tourenplanungsproblem mit Zeitfenstern, indem sie zunächst die Stationen in Segmente einteilen und dann Lösungen für die einzelnen Teilprobleme errechnen. Die mit KNN erreichten Ergebnisse sind dabei zum Teil nah an den besten bekannten Lösungen, allerdings muss nach CRÉPUT ET AL. (2007) für die guten Ergebnisse eine hohe Rechenzeit in Kauf genommen werden. Auf dem Gebiet des maschinellen Lernens sind derzeit enorme Entwicklungsschritte zu beobachten, so dass sich in Zukunft sehr großes Potenzial aus diesen Algorithmen erwarten lässt.

4.2.3 Hybride Ansätze

Viele der in den vorhergehenden Abschnitten beschriebenen Meta-Heuristiken lassen sich mit dem Ziel kombinieren, die Vorteile mehrerer Verfahren gleichzeitig zu nutzen.

Eine Möglichkeit ist dabei die Aufteilung der Aufgabenbereiche auf unterschiedliche Algorithmen. So lösen z. B. A.-L. CHEN ET AL. (2006) das Zuordnungsproblem zu Routen mit einer Partikelschwarmoptimierung, während das Tourenplanungsproblem für die einzelnen Touren mit Simulated Annealing gelöst wird. Analog dazu finden sich viele Beispiele, etwa YU ET AL. (2011) (Genetischer Algorithmus +

⁸ vgl. Cluster-first, Route-second-Algorithmen in Abschnitt 4.2.1.

4 Lösungsverfahren für Tourenplanungsprobleme

Tabu-Suche) oder beliebige Algorithmen in Zusammenhang mit der Sweep-Heuristik (siehe Abschnitt 4.2.1).

Die Alternative dazu ist die Integration oder die Verkettung zweier Meta-Heuristiken bzw. Heuristiken. So werden beispielsweise zahlreiche Heuristiken zur lokalen Optimierung innerhalb von Meta-Heuristiken verwendet (LABADIE ET AL. 2016, S. 15 ff.). Die Kombination mehrerer Meta-Heuristiken ist ebenfalls möglich, Beispiele hierfür sind KALAYCI & KAYA (2016), die den Ameisenalgorithmus in die Variable Neighborhood Search integrieren oder CHANG & YEN (2012), welche die Generierung der Ausgangslösungen für die Scatter-Search durch die Tabu-Suche verbessern. Die Kombinationen können zu besseren Ergebnissen sowie kürzeren Rechenzeiten führen, wobei einerseits in der praktischen Anwendung nur geringe Verbesserungen möglich sind und andererseits beides nicht garantiert werden kann. Zudem muss bei der Implementierung für beide Verfahren die entsprechende Parametrierung erfolgen, womit eine höhere Sensitivität auf Änderungen der Problemstellung und somit erhöhter Aufwand einher geht (KALAYCI & KAYA 2016, S. 174).

4.3 Analyse der Eignung für die innerbetriebliche Logistik

Nach der Vorstellung der prinzipiell einsetzbaren Lösungsverfahren werden diese im Hinblick auf deren Lösungsqualität, Rechenzeit und Erweiterbarkeit verglichen. Das *No Free Lunch*-Theorem (sinngemäß auf deutsch: *Nichts ist umsonst*) nach WOLPERT & MACREADY (1997) besagt zu dieser Zielsetzung, dass es kein Optimierungsverfahren geben kann, das für alle Anwendungsfälle gleichermaßen überlegen ist. Die Bevorteilung einzelner Algorithmen entsteht demnach erst durch die Einschränkung auf spezifische Anwendungsfälle.

In Kapitel 3 wurden für den Algorithmus vier relevante Anforderungen definiert. Für gute Lösungen ist eine Abwägung zwischen den Anforderungen *Lösungsqualität* (Abschnitt 4.3.1) sowie *Rechenzeit* und *Skalierbarkeit* (Abschnitt 4.3.2) aus Abschnitt 3.2 erforderlich. Damit der Algorithmus überhaupt für die Problemstellung einsetzbar ist, muss hingegen dessen *Erweiterbarkeit* sichergestellt sein, was in Abschnitt 4.3.3 analysiert wird.

4.3.1 Vergleich der Lösungsqualität

Zahlreiche Veröffentlichungen ziehen zum Vergleich der Leistungsfähigkeit der Algorithmen allgemein anerkannte Benchmarks heran. Die am häufigsten verwendete Gruppe sind die sog. *Solomon*-Benchmarks nach SOLOMON (1987). Diese Benchmarks sind nicht perfekt für den Anwendungsfall der fabrikinternen Logistik, da sie die zusätzlichen Anforderungen aus Kapitel 3 zum Großteil nicht abbilden. Dennoch lassen sich aus dem so gewonnenen Vergleich der Algorithmen einige Erkenntnisse ableiten. Zusätzlich sind die in Veröffentlichungen bzgl. Lösungsqualität und Rechenzeit gemachten Angaben nur schwer zu vergleichen, da zum einen unterschiedliche Hardware zur Optimierung eingesetzt wurde und zum anderen die Leistungsfähigkeit auch stark von der Softwareumgebung sowie der Qualität der Implementierung abhängt (SILBERHOLZ & GOLDEN 2010, S. 635). Nichtsdestotrotz kann das Vergleichen dieser Werte eine ungefähre Tendenz über die Leistungsfähigkeit angeben. Die Werte sind in Tabelle 4.1 aus den einzelnen Veröffentlichungen zusammengetragen.

Die Solomon-Benchmarkinstanzen sind in verschiedene Klassen eingeteilt (siehe Abbildung 4.4), für die jeweils wiederum mehrere unterschiedliche Zeitfenster definiert sind. Für die gegebene Zielsetzung sind vor allem die zufällig verteilten Stationen der Klasse R (*Random*) sowie die zufällig mit lokaler Clusterung verteilten Stationen der Klasse RC (*Random/Cluster*) besonders interessant. Die stark zu Clustern zusammengefassten Szenarien der Klasse C (*Cluster*) entsprechen nur entfernt der innerhalb von Fabriken anzutreffenden Realität, da die Stationen in Fabriken im Regelfall über die gesamte Fläche verteilt sind. Eine zusätzliche Unterteilung findet in die Gruppen 1 und 2 statt. Die Benchmarks der Gruppe 1 zeichnen sich durch einen kurzen Planungshorizont und kleine Zeitfenster aus, während in Gruppe 2 ein langer Planungszeitraum sowie lange Zeitfenster charakteristisch sind. Für die Bewertung der Lösungsqualität sind daher vorrangig die Benchmarks R1 und RC1 von Interesse.

Im Bezug auf die Abweichung vom Optimum kommen nach Tabelle 4.1 prinzipiell die Tabu-Suche, die VNS, die Scatter Search, der Ameisenalgorithmus sowie das Maschinelle Lernen in Frage (vgl. Tabelle 4.1). Detaillierte Abwägungen zwischen diesen Algorithmen sollten aufgrund der unscharfen Datenlage (s. o.) zwar nicht ge-

4 Lösungsverfahren für Tourenplanungsprobleme

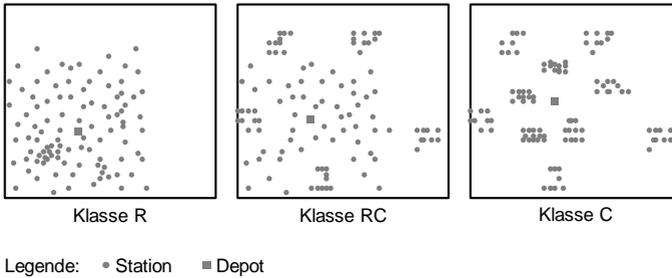


Abbildung 4.4: Unterschiedliche Anordnungen der Stationen im Raum bei den Problemklassen R, RC und C

troffen werden, die übrigen Algorithmen weichen jedoch so stark von der optimalen Lösung ab, dass ein Einsatz nicht sinnvoll erscheint.

4.3.2 Rechenzeit und Skalierbarkeit

Eine optimal geplante Tour ist wertlos, wenn deren Berechnung zu lange benötigt, um nach Abschluss noch aktuell zu sein. Auch für den Vergleich der *Rechenzeiten* können die Werte aus Tabelle 4.1 herangezogen werden. Die Rechenzeiten für die Problemklassen R1 und RC1 zeigen, dass nur Simulated Annealing, GRASP, der Genetische Algorithmus und der Ameisenalgorithmus in Frage kommen.

Dabei ist allerdings zu berücksichtigen, dass der Vergleich der Rechenzeiten nicht unabhängig von der Lösungsqualität durchgeführt werden sollte. Zwar erreichen Algorithmen wie bspw. GRASP sehr gute Rechenzeiten, kommen dabei jedoch nicht annähernd an die optimale Lösung heran. Unter Berücksichtigung der Rechenzeiten wird klar, dass einzig der Ameisenalgorithmus sowohl akzeptable Rechenzeiten als auch eine gute Lösungsqualität erreicht. Die guten Ergebnisse des Maschinellen Lernens müssen hingegen nach GHAZIRI & OSMAN (2006, S. 110) mit sehr hohen Rechenzeiten erkaufte werden, so dass diese auch ausgeschlossen werden müssen.

Die *Skalierbarkeit hinsichtlich der Problemgröße* ist gegeben, wenn eine Vergrößerung des zu lösenden Problems die Rechenzeit möglichst nur geringfügig verlängert. Angaben hierzu lassen sich den Vergleichen nicht entnehmen. Durch die Fokussierung

4.3 Analyse der Eignung für die innerbetriebliche Logistik

Tabelle 4.1: Gegenüberstellung der Lösungsverfahren anhand der Solomon-Benchmarks

	Abweichung vom Optimum für R1 [%]	Rechenzeit für R1 [s]	Abweichung vom Optimum für C1 [%]	Rechenzeit für C1 [s]	Abweichung vom Optimum für R1 [%]	Rechenzeit für R1 [s]	Abweichung vom Optimum für C1 [%]	Rechenzeit für C1 [s]	Abweichung vom Optimum für R1 [%]	Rechenzeit für R1 [s]	Abweichung vom Optimum für C2 [%]	Rechenzeit für C2 [s]	Abweichung vom Optimum für R2 [%]	Rechenzeit für R2 [s]	Abweichung vom Optimum für C2 [%]	Rechenzeit für C2 [s]	Abweichung vom Optimum für R2 [%]	Rechenzeit für R2 [s]
Branch-and-Cut-and-Price nach BETTINELLI ET AL. (2011) ^a	7,76	354	2,48	1453,	9,54	-	14,9	-	12,65	-	9,44	-	14,9	-	12,65	-	14,9	-
Tabu-Suche nach TAILLARD ET AL. (1997)	4,67	2296	0,45	2926	4,71	1877	11,2	3372	0,92	3275	19	1933	11,2	3372	0,92	3275	19	1933
Variable Neighbourhood Search nach BRÄVSY (2003) ^b	4,30	7542	0,20	-	3,94	6132	5,2	6432	0,50	-	8,78	3114	5,2	6432	0,50	-	8,78	3114
Simulated Annealing nach CHANG & RUSSELL (1996)	10,82	145	12,81	136,2	10,45	117	22	144	13,44	68,4	38,32	131,4	22	144	13,44	68,4	38,32	131,4
Greedy Randomized Adaptive Search Procedure nach KONTO-RAVDIS & BARD (1995)	12,44	73,1	0,07	9,0	11,89	72,5	23,7	116	0,4	13,5	34,8	127,5	23,7	116	0,4	13,5	34,8	127,5
Genetischer Algorithmus nach BERGER & BARAKOUI (2004)	120	0,27	120	8,58	120	16,67	120	2,36	120	32,34	120	120	120	2,36	120	32,34	120	120
Scatter Search nach RUSSELL & CHIANG (2006)	2,46	2387	0,20	1064,4	3,29	2625	2,61	2335	0,48	1328,4	9,8	1883	2,61	2335	0,48	1328,4	9,8	1883
Partikelschwarmoptimierung nach GONG ET AL. (2012)	7,16	1649	3,60	878,2	4,44	1458	14,1	2323	4,35	681	17,16	2287	14,1	2323	4,35	681	17,16	2287
Ameisenalgorithmus nach GAMBARDELLA ET AL. (1999) ^c	2,93	600	0,20	600	2,91	600	2,58	600	0,94	600	10,88	600	2,58	600	0,94	600	10,88	600
Künstliches Neuronales Netz nach HAGHIGHI ET AL. (2009) ^d	0,46	-	0,22	-	-	-	0,06	-	0,45	-	-	-	0,06	-	0,45	-	-	-

^a Nicht alle Probleme konnten gelöst werden, die Werte gelten für die gelösten Probleme.

^b Werte gelten für RVNS (1) Konfiguration.

^c Werte gelten für den Versuch mit 600 Sekunden Rechenzeit.

^d Nicht alle Testinstanzen wurden getestet; Werte gelten für die betrachteten Probleme; die Rechenzeit des Verfahrens für große Probleme ist sehr lang (GHAZIRI & OSMAN 2006, S. 110).

auf Meta-Heuristiken ist eine prinzipbedingte Skalierbarkeit jedoch gegeben, solange die Rechenzeit nicht für kleine Probleminstanzen schon zu hoch ist.

4.3.3 Analyse der Umsetzungsfähigkeit notwendiger Restriktionen

Nachdem Lösungsqualität und Rechenzeit für Standardfälle in den vorangegangenen Abschnitten untersucht wurden, soll im Folgenden untersucht werden, welche der funktionalen Anforderungen aus Abschnitt 3.2 von den Algorithmen bereits unterstützt werden bzw. inwieweit diese nachgerüstet werden können.

Wichtigste Anforderung ist die *Korrektheit* (Abschnitt 3.2), welche von keinem der Algorithmen in hinreichendem Maße abgedeckt ist, da an keiner Stelle zeitpunktabhängige Servicezeiten berücksichtigt werden. Inwieweit die Möglichkeit der Erweiterung der Algorithmen um diese Eigenschaften besteht, wird im folgenden Abschnitt behandelt.

Für die Tabu-Suche sowie die VNS ist eine Erweiterung der Zielfunktion um diese Bestandteile kein Problem. Da sich durch die veränderlichen Zeitanteile allerdings während der Lösungskonstruktion permanent die Umgebung und damit die Lage der lokalen bzw. globalen Optima ändert, ist mit Einbußen bei der Lösungsgeschwindigkeit zu rechnen. Die Scatter Search hat durch ihren weiten Suchbereich hier den Vorteil, leichter aus diesen Strukturen ausbrechen zu können. Allerdings kann sie, wie auch Tabu-Suche und VNS, die veränderlichen Servicezeiten nur in der Zielfunktion bewerten. Beim Ameisenalgorithmus kann zusätzlich zur Berücksichtigung in der Zielfunktion bereits bei der Lösungskonstruktion in der Kanten-Heuristik eine Berücksichtigung erfolgen. Dadurch wird der Effekt abgemildert, dass sich Änderungen an der Reihenfolge auf die Lage der Optima und somit die Lösungsgeschwindigkeit auswirken.

Die *Mehrfachverwendung von Fahrzeugen* ist in allen vier oben betrachteten Algorithmen in der Wissenschaft betrachtet. Die Kombination mit den anderen Anforderungen erfordert hier zwar eine Neubetrachtung der daraus ableitbaren Erkenntnisse, die prinzipielle Umsetzbarkeit ist jedoch für alle näher betrachteten Algorithmen gegeben.

Die Anforderung *Prognosefähigkeit* erfüllt keiner der Algorithmen, was sich auf die meist theoretischen Anwendungsfälle der Veröffentlichungen zurückführen lässt. Zur

Prognose muss der Algorithmus daher in ein übergeordnetes Vorgehen eingegliedert werden, womit sich die zukünftigen Bedarfe mit der zu erwartenden Tourenplanung kombinieren lassen.

Aufgrund der Fokussierung auf Meta-Heuristiken ist eine prinzipielle Erweiterbarkeit per se gegeben, durch die notwendigen Erweiterungen kann deren breite Einsetzbarkeit jedoch eingeschränkt werden. Dies ist unabhängig vom gewählten Algorithmus bei der Entwicklung zu berücksichtigen.

4.3.4 Zusammenfassung der Lösungsverfahren

Die Erkenntnisse aus den vorangegangenen Abschnitten sind in Tabelle 4.2 zusammengefasst. Dabei sind nur die Anforderungen aufgelistet, deren Erfüllung in Abschnitt 3.1 als direkt relevant für den Algorithmus bestimmt wurden.

Die Zusammenfassung zeigt, dass Lösungsalgorithmen entweder bei der Lösungsqualität und Skalierbarkeit gut sind oder sich hingegen gut erweitern lassen. Der Ameisenalgorithmus qualifiziert sich hier durch gute Eigenschaften bzgl. aller Anforderungen als aussichtsreichste Variante. Bei der Rechenzeit und der Lösungsqualität liegt der Algorithmus auf einem guten Level. Der eigentliche Vorteil wird jedoch durch die im Vergleich bessere Erweiterbarkeit erreicht, die sowohl in der Zielfunktion als auch in der heuristischen Funktion erfolgen kann und somit eine bessere und schnellere Konvergenz bei Anpassungen erwarten lässt. Bestätigt wird diese Tendenz durch die Untersuchungen von LACKNER (2004, S. 191-206), der populationsbasierten Meta-Heuristiken die beste Leistungsfähigkeit bei dynamischen Szenarien attestiert.

4.4 Forschungsdefizit

Die Analyse des Stand der Technik und Forschung führt zu der Erkenntnis, dass keiner der Ansätze die vollständige Liste der Anforderungen an Algorithmen aus Kapitel 3 vollständig erfüllt.

Die Möglichkeit zur Berücksichtigung variabler Servicezeiten wird beim derzeitigen Stand der Wissenschaft nicht hinreichend abgedeckt. Demnach ist die Anforderung

4 Lösungsverfahren für Tourenplanungsprobleme

Table 4.2: Bewertung des Erfüllungsgrads der für die Algorithmen relevanten Anforderungen

	Lösungsqualität	Rechenzeit	Skalierbarkeit	Erweiterbarkeit
Tabu-Suche	●	•	●	•
VNS	●	•	●	•
Simulated Annealing	•	•	•	•
Genetischer Algorithmus	•	●	•	●
Scatter Search	●	•	●	●
GRASP	•	●	•	●
PSO	•	•	●	●
Ameisenalgorithmus	●	●	●	●
Maschinelles Lernen	●	•	•	●

Legende:

- Anforderung nicht erfüllt und nur schwer zu integrieren
- Anforderung nicht erfüllt, Implementierung mit größerem Aufwand vermutlich möglich
- Anforderung nicht erfüllt, Implementierung scheint sehr realistisch
- Anforderung erfüllt

der Richtigkeit im Allgemeinen ebenfalls nicht gegeben. Auch die Beobachtung, dass in aktuellen Lösungen i. d. R. Fahrzeuge nur eine einzige Tour fahren, wiederkehrende Fahrzeuge somit nicht betrachtet werden, ist im Weiteren zu beachten. Erweiterungen des Algorithmus müssen mit dem Ziel einer kurzen Rechenzeit sowie einer akzeptablen Lösungsqualität vorgenommen werden. Zusätzlich ist darauf zu achten, dass auch bei Fehlern in den Eingangsdaten ein tolerantes Verhalten bei manuellen Eingriffen oder Abweichungen vom Plan gewährleistet ist. Bei den Algorithmen hat der Ameisenalgorithmus aus Sicht der Algorithmusstruktur sowie den bisherigen Ergebnissen aus dem Vergleich der Algorithmen die besten Potenziale, weshalb im Folgenden in Kapitel 5 dieser Algorithmus zur weiteren Entwicklung als Grundlage verwendet wird.

Die Prognosefähigkeit bei keinem der bisherigen Ansätze gegeben. Dies soll unabhängig von der dynamischen Tourenplanung in Kapitel 6 erfolgen.

5 Dynamische Routenplanung

Das vorhergehende Kapitel offenbart den Forschungsbedarf auf dem Gebiet der innerbetrieblichen Logistik zur Verbesserung der dynamischen Fähigkeiten, da keines der untersuchten Konzepte die Anforderungen aus Kapitel 3 vollständig erfüllt. Im Folgenden wird daher ein Gesamtkonzept erarbeitet, welches diese Defizite adressiert. Dazu wird in Abschnitt 5.1 zunächst die zugrundeliegende mathematische Modellierung um die relevanten Bestandteile ergänzt, d.h. die formelle Beschreibung des Problems definiert. Danach werden nach einer detaillierten Analyse des Ameisenalgorithmus die einzelnen Systemkomponenten schrittweise ergänzt. Das Kapitel schließt in Abschnitt 5.9 mit einem methodischen Vorgehen zur Einbettung des Gesamtkonzepts in die Infrastruktur der bestehenden Logistik.

5.1 Erweiterung des VRP um zeitpunktabhängige Servicezeiten

5.1.1 Modellierung der Servicezeiten

Tourenplanungsprobleme, wie sie in Abschnitt 2.3 beschrieben wurden, enthalten in ihrer ursprünglichen Form keine variablen Zeitanteile. Schwankungen in der Fahrtzeit sind jedoch in vielen Anwendungsfällen, wie z. B. bei Navigationssystemen mit Berücksichtigung von Stauinformationen, eine relevante Größe. Deshalb können im zeitpunktabhängigen Tourenplanungsproblem (Time Dependent Vehicle Routing Problem (TDVRP)) vom Startzeitpunkt einer Fahrt abhängige Fahrtzeiten berücksichtigt werden.

Die Berücksichtigung von zeitpunktabhängigen Servicezeiten ist hingegen nur für spezielle Anwendungsfälle von Bedeutung und wird daher in den Standardproblemen

5 Dynamische Routenplanung

nicht berücksichtigt.¹ Für die innerbetriebliche Logistik sind sie jedoch ein wichtiger Baustein, weshalb im Folgenden das Tourenplanungsproblem um zeitpunktabhängige Servicezeiten erweitert wird.²

Um diese zu berücksichtigen, wird aus der statischen Servicezeit $t_{s,i}$ an Station i die variable Servicezeit $t_{s,i}(t_i)$ zur Ankunftszeit t_i an Station i . Im allgemeinen Fall kann $t_{s,i}(t_i)$ durch beliebige Formeln beschrieben werden. Für die innerbetriebliche Logistik ergibt sich der Zusammenhang mit den Bestandteilen aus Abbildung 5.1 zu

$$t_{s,i}(t_i) = t_{c,i} + f_{v,i} \cdot (t_{e,i} - t_i) \quad (5.1)$$

mit

- $t_{c,i}$ Konstante Umladezeit an Station i
- $f_{v,i}$ Faktor für die variable Umladezeit
- $t_{e,i}$ Spätester Ablieferzeitpunkt (Ende des Zeitfensters) an Station i
- t_i Ankunftszeit an Station i

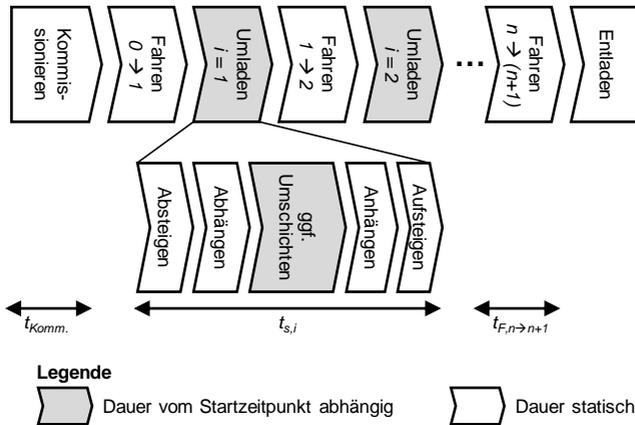


Abbildung 5.1: Zeitliche Bestandteile einer Routenzugfahrt (aufbauend auf BUCHHOLZ & CLAUSEN 2009, S. 162)

¹ Die Möglichkeit variabler Servicezeiten wird zwar von BRAEKERS ET AL. (2016, S. 303 ff.) als theoretische Möglichkeit erwähnt, erfährt jedoch keine weitere Beachtung bei der Umsetzung.

² Da im innerbetrieblichen Logistikverkehr keine prognostizierbaren Staus auftreten, wird die Fahrtzeit im Folgenden als statisch betrachtet. Das hier entwickelte Modell inkl. des Algorithmus ist jedoch mit den entsprechenden Erweiterungen auch für variable Fahrtgeschwindigkeiten einsetzbar.

5.1 Erweiterung des VRP um zeitpunktabhängige Servicezeiten

Für den Fall einer Anlieferung zum letztmöglichen Zeitpunkt $t_i = t_{e,i}$ ergibt sich daraus nur die konstante Umladezeit $t_{s,i}(t_i) = t_{c,i}$, d. h. in diesem Fall die Zeiten für das Absteigen, Abhängen, Austauschen, Anhängen und wieder Aufsteigen auf das Fahrzeug. Je früher die Anlieferung erfolgt, desto größer wird bei einem 1:1-Tausch der Behälter der Faktor $(t_{e,i} - t_i)$, aus welchem sich durch Multiplikation mit dem Parameter $f_{v,i}$ die variable Umladezeit ermittelt, d. h. die Zeit für das Umschichten des alten Behälterinhalts in den neuen Behälter.

Eine zu späte Anlieferung wird nicht akzeptiert, da dies einen Stillstand des entsprechenden Produktionsabschnitts zur Folge hätte. Es handelt sich daher in diesem Fall um ein einseitiges Zeitfenster. Dieses wichtige Kriterium muss als harte Randbedingung im Algorithmus berücksichtigt werden.

Die Gesamtzeit der Fahrt setzt sich aus den Bestandteilen der Elemente aus Abbildung 5.1 zusammen und lässt sich bis zur Abfahrt an der letzten Station für alle n Stationen durch

$$t_{ges} = t_{Komm.} + \sum_{i=1}^n [t_{F,i-1 \rightarrow i} + t_{s,i}(t_i)] \quad (5.2)$$

mit

$t_{Komm.}$	Zeit zum Kommissionieren vor Abfahrt
n	Anzahl Stationen
$t_{F,i-1 \rightarrow i}$	Fahrtzeit von Station $i - 1$ zu Station i
$t_{s,i}(t_i)$	Servicezeit an Station i zum Ankunftszeitpunkt t_i

berechnen. Die Fahrtzeit kann durch den Zusammenhang

$$t_{F,i-1 \rightarrow i} = d_{i-1 \rightarrow i} \cdot v \quad (5.3)$$

mit der Distanz $d_{i-1 \rightarrow i}$ zwischen den Stationen $i - 1$ und i sowie der Geschwindigkeit v ermittelt werden³.

³ Die Geschwindigkeit v kann je nach Fabrikumfeld auch von den äußeren Rahmenbedingungen abhängen (z. B. Breite der Fahrstraße, geladene Ladungsträger, etc.). Dies lässt sich durch eine Modifikation von Gleichung 5.3 integrieren. Das Modell sowie der Algorithmus sind im Hinblick auf diese Modifikation entwickelt.

5 Dynamische Routenplanung

Zur Ermittlung der Ankunftszeit t_i müssen für jede Station die bis dorthin angefallenen Zeiteile aufsummiert und zum Startzeitpunkt t_{Start} addiert werden. Für die erste Station ergibt sich daher die Ankunftszeit

$$t_1 = t_{Start} + t_{Komm.} + t_{F,0 \rightarrow 1}, \quad (5.4)$$

für die zweite Station ist der Zusammenhang unter Verwendung von Gleichung 5.1 durch

$$\begin{aligned} t_2 &= t_1 + t_{s,i}(t_1) + t_{F,1 \rightarrow 2} \\ &= t_{Start} + t_{Komm.} + t_{F,0 \rightarrow 1} \\ &\quad + t_{c,1} + f_{v,1} \cdot (t_{e,1} - t_{Start} - t_{komm.} - t_{F,0 \rightarrow 1}) + t_{F,1 \rightarrow 2} \end{aligned} \quad (5.5)$$

zu ermitteln. Die Ankunftszeiten für die nachfolgenden Stationen $t_3 \dots t_n$ lassen sich analog herleiten. Der Term wird mit jeder weiteren Station jedoch immer unhandlicher und durch die gegenseitige Beeinflussung der Umladezeiten entsprechend schwer analytisch lösbar.

5.1.2 Ermittlung der Parameter

Die Distanzmatrix mit den Distanzen $d_{i \rightarrow j}$ kann aus dem Werkslayout abgeleitet werden. Die Parameter für die durchschnittliche Fahrgeschwindigkeit v , die konstante Umladezeit t_c sowie den Faktor f_v für die variable Umladezeit müssen hingegen anwendungsfallsspezifisch durch Testläufe ermittelt werden, wofür der folgende Abschnitt die Methodik beschreibt.

Zur Berechnung sämtlicher Eigenschaften und Zeiten an den Stationen, wie sie in Abschnitt 5.1.1 beschrieben wurden, müssen diverse Parameter gesetzt werden. Eine exakte Bestimmung der tatsächlichen Werte dafür ist in der Praxis jedoch in den meisten Fabriken nicht möglich, wobei sowohl technische als auch organisatorische Probleme im Wege stehen. Selbst wenn einzelne der Parameter durch direkte Messungen ermittelt werden könnten (z. B. die Geschwindigkeit v), ist dies nicht empfehlenswert, weil im Einsatz in der Produktion permanente externe Einflüsse auf das Routenzugsystem einwirken und somit Abweichungen von einem unter Laborbedingungen gemessenen Wert auftreten würden. Die Werte müssen deshalb

5.1 Erweiterung des VRP um zeitpunktabhängige Servicezeiten

möglichst präzise anhand von im Realbetrieb aufgenommenen Messungen ermittelt werden.

Eine Möglichkeit zur Ermittlung der Parametergrößen ist die Regressionsanalyse, womit der Einfluss der Parameter so gewählt werden kann, dass eine möglichst hohe Übereinstimmung mit der Realität eintritt (BACKHAUS ET AL. 2016, S. 64 ff.). Abbildung 5.2 zeigt die Arbeitsschritte zur Durchführung einer Regressionsanalyse.

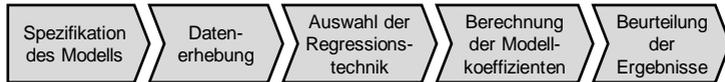


Abbildung 5.2: Arbeitsschritte zur Durchführung einer Regressionsanalyse (in Anlehnung an URBAN & MAYERL 2011, S. 22 f.)

Der erste Schritt des Vorgehens, die Spezifikation des Modells, wurde in Abschnitt 5.1.1 bereits vorgestellt.

Im anschließenden Schritt der Datenerhebung müssen anwendungsfallspezifisch die notwendigen Daten erhoben werden. Dafür werden über eine hinreichend große Anzahl an Touren Messungen der Gesamtdauer durchgeführt. Mit dem Wissen über die zurückgelegte Strecke innerhalb der jeweiligen Tour lassen sich daraus in den folgenden Schritten die unbekannt Parameter ermitteln.

Da sämtliche zu ermittelnden Parameter linear in die Gleichung eingehen, lässt sich dem Modell im Rahmen der Auswahl der Regressionstechnik Linearität unterstellen (URBAN & MAYERL 2011, S. 202 ff.). Als Möglichkeit zur Parameteridentifikation bietet sich für lineare Regressionsmodelle die Methode der kleinsten Quadrate an (engl. Least Mean Squares (LMS)) (PAPAGEORGIU ET AL. 2015, S. 133 ff.). Dafür wird zu jedem Messwert (in diesem Fall die Tourdauer) über die aufgestellte Formel (vgl. Abschnitt 5.1.1) ein Wert berechnet. Die Methode versucht nun, die Parameter in der Formel so zu wählen, dass wie in Gleichung 5.6 die Summe der quadrierten Differenzen zwischen errechnetem und experimentellem Wert möglichst klein wird:⁴

$$\min f = \frac{1}{2} \sum_{i=1}^m (t_{i,\text{gemessen}} - t_{i,\text{errechnet}})^2 \quad (5.6)$$

⁴ Zur numerischen Lösung dieses Gleichungssystems eignet sich beispielsweise der in MATLAB integrierte Befehl `lsqnonlin`.

5 Dynamische Routenplanung

Die Methode der kleinsten Quadrate liefert nur dann verwertbare Ergebnisse, wenn gewisse Kennzahlen eingehalten werden, die sicherstellen, dass die Realität durch das Modell hinreichend genau abgebildet wird. Dies ist beispielsweise das Bestimmtheitsmaß R^2 , welches den Anteil der Varianz der Zielgröße durch das Regressionsmodell erklärt, oder der Varianz-Inflationsfaktor, welcher eine Maßzahl dafür ist, wie stark sich die Einflussfaktoren gegenseitig beeinflussen. Detaillierte Informationen zur Berechnung dieser Koeffizienten finden sich bei URBAN & MAYERL (2011, S. 232).

Sind alle Anforderungen an das Regressionsmodell erfüllt, so können nach URBAN & MAYERL (2011, S. 22 f.) die Modellkoeffizienten berechnet werden. Im Anschluss daran muss das Ergebnis dahingehend beurteilt werden, ob Verfälschungen am Regressionsmodell vorliegen oder ob die Parameter hinreichend genau bestimmt werden konnten.

Von grundlegendem Interesse bei der Ermittlung der Parameter ist die zu erwartende Genauigkeit. Dabei lässt sich feststellen, dass mit steigender Größe der Stichprobe die Genauigkeit der durch die Regressionsanalyse ermittelten Parameter besser wird. GREEN (1991) gibt für die Stichprobengröße die Faustformel

$$N_{min} = 50 + 8n_p \quad (5.7)$$

an, mit der die empfohlene Mindestanzahl N_{min} an Stichproben in Abhängigkeit der Anzahl n_p der zu ermittelnden Parameter errechnet werden kann.⁵

5.1.3 Modellierung des Depots

Das Depot stellt den Ausgangspunkt der Touren im VRP dar und entspricht im vorliegenden Anwendungsfall dem Lager. Bei Algorithmen für kombinatorische Probleme ist zu berücksichtigen, dass diese im Rahmen der Lösungsfindung versuchen, die einzelnen Elemente in eine möglichst gute Reihenfolge zu bringen. Das Depot im VRP kommt dabei in jeder Tour zweimal vor (als Start und Endpunkt). Zusätzlich

⁵ Die tatsächliche Anzahl notwendiger Messungen hängt stark von der Beschaffenheit des Problems und den Messungen ab. So kommen AUSTIN & STEYERBERG (2015, S. 627 ff.) auf eine deutlich niedrigere Empfehlung von mindestens 2 Messungen pro unabhängiger Variable. Die hier angegebene Formel kann daher nur als grober Richtwert gesehen werden. Zudem kann mit verhältnismäßig ungenauen Parametern gestartet werden und während der ersten Touren im Realbetrieb mit weiteren Messungen die Stichprobengröße Stück für Stück erweitert werden.

5.2 Entwicklung des Ameisenalgorithmus für Logistikprobleme

optimieren die meisten Algorithmen alle Touren auf einmal, d. h. sie bilden eine lange Verkettung aller Stationen und trennen diese dann an den eingeplanten Depots in die separaten Touren auf. Für diese Vorgehensweise vereinfacht es die Findung von Lösungen erheblich, wenn das Depot dupliziert wird. So kann jedes der duplizierten Depots ein einziges Mal eingeplant werden, wodurch im Algorithmus keine Sonderbehandlung für Depots implementiert sein muss. Die Anzahl der duplizierten Depots muss dabei um eins höher sein als die Anzahl der Touren. Die Modellierung ist exemplarisch für zwei Touren und somit drei Depots in Abbildung 5.3 dargestellt.

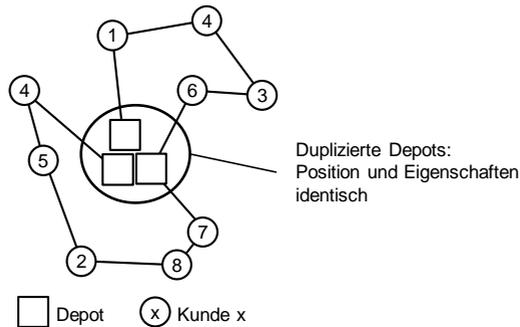


Abbildung 5.3: Modellierung des VRP mit duplizierten Depots

5.2 Entwicklung des Ameisenalgorithmus für Logistikprobleme

Wie die Analyse in Kapitel 4 zeigt, ist der Einsatz des Ameisenalgorithmus die erfolgversprechendste Option zur Verbesserung der Steuerungssysteme für Routenzüge. Im Folgenden wird daher zunächst im Detail auf die Herkunft und Funktionsweise des Ameisenalgorithmus eingegangen, sowie schrittweise Verbesserungen und Anpassungen für die Routenzuglogistik eingefügt.

5.2.1 Biologische Herkunft

Ameisen helfen sich durch das Auslegen von Duftstoffen (sog. Pheromonen) gegenseitig dabei, den schnellsten Weg zu einer Nahrungsquelle zu finden. Diese Kommunikation durch die Modifikation der Umgebung wird Stigmergie genannt

5 Dynamische Routenplanung

(GRASSÉ 1959). Dabei befolgen Ameisen nur einige wenige Regeln, die in Summe zur Herausbildung des kürzesten Weges führen.

Zu Beginn liegen den Ameisen keinerlei Informationen über die Lage der Nahrungsquellen vor, die Suche nach diesen verläuft daher mehr oder weniger zufällig. Sobald eine Ameise auf eine Nahrungsquelle stößt, macht sie sich beladen auf den Rückweg. Dabei wählt sie exakt denselben Weg wie auf dem Hinweg und legt Informationen über diesen Weg in Form von Pheromonen aus, welche von anderen Ameisen wahrgenommen werden können. Diese können nun wahlweise dieser Spur folgen oder neue Wege versuchen. Je stärker eine Pheromonspur mit der Zeit wird, desto mehr Ameisen folgen dieser Spur. Da Pheromone mit der Zeit verdunsten, verschwinden schlechte Wege nach und nach wieder, da sie zum einen seltener gewählt werden, zum anderen weil die Pheromone durch die längere Zeit zum Zurücklegen der Strecke bei der Rückkehr bereits ansatzweise wieder verdunstet sind. Zudem gibt es immer wieder Individuen, die vom etablierten Weg abweichen. Dies verhindert, dass ein initialer Fehler für alle Zeiten im System vorhanden bleibt. Dem Kollektiv an Ameisen gelingt so die Lösung eines Problems, welches für ein Einzelindividuum nicht lösbar wäre. Der Vorgang ist schematisch in Abbildung 5.4 dargestellt.⁶ (DOMSCHKE & SCHOLL 2010, S. 36; HÖLLDOBLER & WILSON 2011)

5.2.2 Funktionsweise des Ameisenalgorithmus

Die grundlegende Form des Ameisenalgorithmus ist das sog. Ant System (AS). Die übergeordnete Struktur des AS ist in Algorithmus 1 als Pseudocode⁷ gezeigt. Nach der Initialisierung des Graphen (siehe Abschnitt 2.3.2) wird analog zum Vorgehen der Ameisen immer jeweils eine Lösung konstruiert und anschließend die Pheromone auf den Kanten abgelegt. Zusätzlich können noch sog. *Daemon Actions* durchgeführt werden, worin durch nachgelagerte Verfahren die Lösungsqualität gesteigert werden kann. Die folgenden Absätze beschreiben dieses Vorgehen im Detail.

⁶ Unterschiedliche Ameisenspezies unterscheiden sich dabei in ihrem Vorgehen. So gibt es bspw. auch die Möglichkeit, bereits auf dem Hinweg Pheromone auszulegen. Auch bei den Techniken zum Markieren des Weges werden unterschiedliche Strategien verfolgt. Die hier vorgestellte Variante entspricht der Denkweise, die als Vorbild für den Ameisenalgorithmus herangezogen wurde.

⁷ Unter Pseudocode wird Code verstanden, der nicht zur maschinellen Interpretation, sondern zur Veranschaulichung gedacht ist. Die Syntax orientiert sich an höheren Programmiersprachen.

5.2 Entwicklung des Ameisenalgorithmus für Logistikprobleme

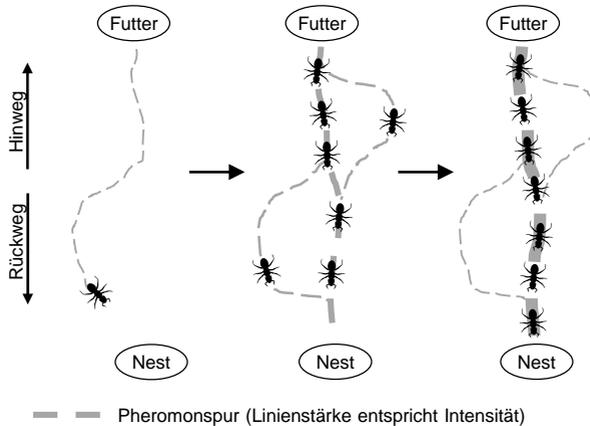


Abbildung 5.4: Vorgehen der Ameisen bei der Futersuche vom unbekanntem Terrain (links) zu voll ausgebildeten kurzen Wegen (rechts) (aufbauend auf GOSS ET AL. 1989, S. 579)

Der erste Schritt des Ameisenalgorithmus ist eine Formulierung des Problems als Graph wie in den Abschnitten 2.3.2 und 5.1 beschrieben (DORIGO ET AL. 1996). Die Knoten stellen dabei die Stationen dar, von denen jeweils zwei über eine Kante verbunden sind. Die Kanten wiederum entsprechen den Wegstrecken, auf denen durch Ablage von Pheromonen die Pheromonkonzentration erhöht werden kann und durch deren Verdunsten wieder gesenkt wird.

Die künstlichen Ameisen starten an einer beliebigen Station und suchen den kürzesten Weg durch alle Stationen. Für eine Ameise an Station i stehen alle Kanten

```
1 Initialisieren des Graphen, der Pheromonkonzentrationen und der Parameter
2
3 repeat
4   | Lösungskonstruktion
5   | Pheromonupdate
6   | Daemon Actions                               /* optional */
7 until Abbruchbedingung erfüllt
```

Algorithmus 1: Pseudocode des Ameisenalgorithmus in Anlehnung an DORIGO & STÜTZLE (2004, S. 38) und DANNINGER (2012, S. 63)

5 Dynamische Routenplanung

zu noch nicht bereisten Stationen zur Verfügung, dabei entscheidet sie sich mit der Wahrscheinlichkeit

$$p_{ij} = \begin{cases} \frac{(\tau_{ij})^\alpha \cdot (\eta_{ij})^\beta}{\sum_{j \in K} (\tau_{ij})^\alpha \cdot (\eta_{ij})^\beta} & \text{falls } j \in K \\ 0 & \text{sonst} \end{cases} \quad (5.8)$$

für die Kante zu Station j . τ_{ij} beschreibt die Pheromonkonzentration auf der Kante, wohingegen η_{ij} die heuristische Distanz zwischen zwei Stationen darstellt, welche im Falle des TSP mit $\eta_{ij} = \frac{1}{d_{ij}}$ und der Distanz d zwischen den beiden Knoten berechnet werden kann. Die heuristische Distanz (oder auch heuristische Funktion) gibt den Ameisen einen Hinweis darauf, welche Kanten durch Eigenschaften des Anwendungsfalls tendenziell zu bevorzugen sind.⁸ Mithilfe der beiden Parameter α und β kann die Gewichtung zwischen dem angelernten Wissen in Form von Pheromonen und dem inhärenten Wissen der heuristischen Funktion angepasst werden.⁹ Die Menge K ist die Menge der noch zu besuchenden Stationen. Gleichung 5.8 wird als *Übergangsregel* bezeichnet.

Anhand der Wahrscheinlichkeitsgewichtungen p_{ij} wird dann die nächste Station ausgewählt und der Lösung Ψ^k der Ameise k hinzugefügt. Danach wird ausgehend von dieser Station dasselbe Verfahren iterativ wiederholt, bis keine Stationen mehr in K vorhanden sind.

Nach der Lösungskonstruktion wird das Update der Pheromone durchgeführt. Dabei wird zunächst auf allen Kanten der Pheromonwert um einen konstanten Faktor gesenkt, anschließend platzieren die Ameisen neue Pheromone auf den Kanten ihrer Lösungen. Die Verdunstung der Pheromone wird mit

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} \quad (5.9)$$

erzielt. Der Verdunstungsfaktor ρ ($0 < \rho \leq 1$) verhindert dabei ein unendliches Ansteigen der Pheromone und ermöglicht dem Algorithmus, schlechte Lösungen

⁸ In der Regel sind dies Kanten mit besonders kurzen Wegstrecken, kurzen Wartezeiten oder anderen vorteilhaften Eigenschaften. Ein hoher heuristischer Wert heißt jedoch nicht automatisch, dass eine Kante automatisch ausgewählt wird, es steigert sich lediglich die Wahrscheinlichkeit für eine Auswahl.

⁹ Wird $\alpha = 1$ gesetzt und immer die Kante mit der höchsten Wahrscheinlichkeit ausgewählt, so entspricht das AS exakt einer Heuristik, wie sie in Abschnitt 4.2.1 vorgestellt wurden. Das AS kann daher abstrahiert auch als die Erweiterung einer Heuristik um Bestandteile zum künstlichen Lernen interpretiert werden.

5.2 Entwicklung des Ameisenalgorithmus für Logistikprobleme

wieder zu vergessen. Anschließend legen die Ameisen auf allen verwendeten Kanten durch

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^m \Delta \tau_{ij}^k \quad (5.10)$$

neue Pheromone aus, wobei $\Delta \tau_{ij}^k$ die Menge an Pheromonen beschreibt, die Ameise k auf den von ihr besuchten Kanten auslegt. Der Wert kann mit

$$\Delta \tau_{ij}^k = \begin{cases} \frac{1}{L_{\psi^k}} & \text{falls die Kante } ij \text{ in der Lösung von Ameise } k \text{ vorkommt} \\ 0 & \text{sonst} \end{cases} \quad (5.11)$$

berechnet werden. Dabei ist L_{ψ^k} die Länge der Lösung ψ^k der Ameise k (DORIGO ET AL. 2004, S. 72).

Dieses Vorgehen wird wie in Algorithmus 1 gezeigt iterativ wiederholt. Die *Daemon Actions* stellen eine zusätzliche Möglichkeit dar, die Lösungsgüte positiv zu beeinflussen. Hier können diverse nachgelagerte Schritte wie lokale Optimierungsverfahren zur Verbesserung der Lösung durch kleine Veränderungen im lokalen Lösungsumfeld oder alternative Pheromonupdatestrategien durchgeführt werden. Anschließend ist die Iteration abgeschlossen und solange das Abbruchkriterium nicht erfüllt ist, wird eine neue Iteration gestartet.

Je nach Implementierung ist auch ein mehrfaches Pheromonupdate möglich. Dann wird zusätzlich nach den *Daemon Actions* ein weiteres Pheromonupdate eingeführt, welches bereits die lokal optimierte Version der Lösung berücksichtigt.

5.2.3 Unterschiede zwischen künstlichen und realen Ameisen

Obleich der Ameisenalgorithmus dem natürlichen Verhalten von Ameisen nachgebildet ist, gibt es neben vielen Gemeinsamkeiten auch essenzielle Unterschiede zwischen realen und künstlichen Ameisen. Diese sind im Folgenden aufgeführt (DANINGER 2012, S. 61 f.; DORIGO ET AL. 1999, S. 141 f.; MERKLE 2002, S. 7 f.). Als Gemeinsamkeiten lassen sich feststellen:

- Beide Typen von Ameisen interagieren als Individuen, d. h. sie sind in der Lage, selbstständig Lösungen zu konstruieren. Gute Lösungen werden i. d. R. jedoch nur im Kollektiv erreicht.

5 Dynamische Routenplanung

- Die Kommunikation der Ameisen untereinander erfolgt ausschließlich über Pheromone.
- Beide Arten von Ameisen suchen kürzeste Wege und bewegen sich dazu in der direkten Umgebung ihres derzeitigen Standorts.
- Sie treffen beide wahrscheinlichsbasierte Entscheidungen unter Zuhilfenahme der auf dem Weg vorhandenen Informationen.

Es sind jedoch auch entscheidende Unterschiede festzustellen, welche die Leistungsfähigkeit des Algorithmus verbessern:

- Künstliche Ameisen suchen den kürzesten Weg in einer diskreten Umgebung, während reale Ameisen beliebige Strecken laufen können.
- Künstliche Ameisen haben ihrem natürlichen Pendant gegenüber ein eigenes Gedächtnis zur Speicherung verschiedener erweiterter Informationen. Dies wird verwendet, um:
 - zulässige Lösungen zu garantieren. Über das Verhalten der Ameise kann die Konstruktion ungültiger Lösungen von vornherein ausgeschlossen werden.
 - heuristische Informationen zu berücksichtigen. Dies ermöglicht der Ameise, neben der Information durch die Pheromone problemspezifisches Wissen in die Lösungsfindung einzubeziehen, um schneller gute Lösungen zu generieren.
 - die individuelle Lösung zu bewerten und die Intensität der Ablage an neuen Pheromonen damit zu beeinflussen.
 - den zurückgelegten Weg zu speichern und die Pheromonablage erst nach Abschluss der vollständigen Lösungskonstruktion durchzuführen.
- Den künstlichen Ameisen stehen Prozesse einer übergeordneten Ebene zur Verfügung. Damit können beispielsweise vor der Ablage von Pheromonen noch lokale Suchverfahren durchgeführt werden, um eine schnellere Lösungsfindung zu begünstigen.

Die Kenntnis über die Funktionsweise des Ant System hilft im weiteren Verlauf der Arbeit, die am Algorithmus vorgenommenen Weiterentwicklungen zu verstehen. Die Erweiterungen bilden die notwendige Grundlage für den Einsatz zur Steuerung der Routenzüge in der Logistik.

5.2.4 Abwandlungen des Ameisenalgorithmus

Die in Abschnitt 5.2.2 vorgestellte Grundform des Ant System zeigt zwar bereits, dass das Prinzip funktionsfähig ist, AS ist jedoch den besten konkurrierenden Algorithmen unterlegen und dient daher hauptsächlich als Ausgangsbasis für Abwandlungen, welche die Leistungsfähigkeit der Ameisenalgorithmen deutlich steigern. Eine Übersicht über Ameisenalgorithmen und deren Erweiterungen findet sich in DORIGO & BLUM (2005). Im Folgenden werden für die vorliegende Arbeit relevante Erweiterungen gezeigt.

Max-Min Ant System

Da beim AS sehr große Unterschiede in Bezug auf die Pheromonkonzentration auf verschiedenen Kanten entstehen können, werden beim Max-Min Ant System (MMAS) eine Ober- und eine Untergrenze der Pheromonkonzentration für die Kanten angegeben. Dies verhindert, dass der Algorithmus auf einer Lösung verharrt und den Lösungsraum nicht mehr weiter durchsucht. (STÜTZLE & HOOS 1997, S. 309 ff., 2000, S. 889 ff.)

Elitist Ant System

In der Grundform des Ameisenalgorithmus legt jede der virtuellen Ameisen während oder nach der Lösungskonstruktion Pheromone aus. Beim Elitist Ant System (EAS) nach DORIGO ET AL. (1991a,b, 1996) legt hingegen nur die beste Ameise Pheromone auf den Kanten ab. Je nach Implementierung kann dafür die beste Lösung der Iteration oder alternativ auch nur die global beste Lösung verwendet werden.

Ant Colony System

Das Ant Colony System (ACS) nach DORIGO & GAMBARDELLA (1997a,b) fasst mehrere der bereits genannten Modifikationen in einem Algorithmus zusammen und unterscheidet sich von Ant System in drei grundlegenden Punkten (DORIGO & STÜTZLE 2004, S. 77 ff.):

5 Dynamische Routenplanung

1. Es nutzt die bereits gewonnene „Sucherfahrung“ der Ameisen aus vergangenen Iterationen stärker als AS, indem eine aggressivere¹⁰ Übergangsregel verwendet wird. Diese stützt sich deutlich stärker auf die abgelegten Pheromone als Gleichung 5.8. Der Übergang zum nächsten Knoten wird hierbei mit der Formel

$$j = \begin{cases} \max_{j \in K} \{\tau_{ij}(\eta_{ij}^\beta)\}, & \text{falls } q \leq q_0 \\ J, & \text{sonst} \end{cases} \quad (5.12)$$

ermittelt, wobei q eine gleichmäßig verteilte Zufallszahl zwischen 0 und 1, q_0 ($0 \leq q_0 \leq 1$) ein Parameter für den Grenzwert und J eine zufällig nach der Wahrscheinlichkeitsverteilung nach Gleichung 5.8 ausgewählte Variable ist (mit $\alpha = 1$). Für die Fälle, in denen $q \leq q_0$ gilt, wird daher immer die Kante mit der höchsten Pheromonkonzentration ausgewählt, unabhängig vom heuristischen Wert.

2. Die Verdunstung bereits ausgelegter sowie die Ablage neuer Pheromone findet nur auf Kanten statt, welche Teil der global besten Lösung Ψ^{gb} sind.¹¹ ACS integriert an dieser Stelle die Vorgehensweise des EAS. Damit wird das Pheromonupdate durch die Formel

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} + \rho \Delta\tau_{ij}^{gb}, \quad \forall (i, j) \in \Psi^{gb} \quad (5.13)$$

mit $\Delta\tau_{ij}^{gb} = 1/L_{\Psi^{gb}}$ realisiert. Das Durchführen des Pheromoneupdates (sowohl Verdunstung als auch Ablegen neuer Pheromone) ausschließlich auf Kanten aus Ψ^{gb} verringert die Rechenkomplexität und ermöglicht eine schnellere Berechnung im Vergleich zum AS, bei dem alle Kanten des Systems aktualisiert werden.

3. Jedes Mal, wenn eine Ameise eine Kante nutzt, wird ein Teil der Pheromone dieser Kante entfernt, um für nachfolgende Ameisen die Nutzung anderer Kanten und somit die bessere Erkundung des Suchraums attraktiver zu machen.

¹⁰ Im Gegensatz zum AS wird deutlich häufiger die Kante mit der höchsten Pheromonkonzentration ausgewählt und nicht so häufig durch eine gewichtete Wahrscheinlichkeitsverteilung entschieden. Die Suche der Ameisen wird damit noch deutlich stärker durch das Wissen über gute Lösungen aus den vorangegangenen Iterationen beeinflusst.

¹¹ Ψ^{gb} beschreibt dabei die bis zu diesem Zeitpunkt beste gefundene Lösung.

Dies wird durch die Formel

$$\tau_{ij} \leftarrow (1 - \xi)\tau_{ij} + \xi\tau_0 \quad (5.14)$$

realisiert, wobei ξ ($0 < \xi < 1$) eine Verdunstungsrate analog zu ρ in Gleichung 5.13 darstellt und τ_0 der initiale Wert der Pheromonkonzentration auf den Kanten ist.

Obwohl im ACS im Gegensatz zum MMAS keine fixe Ober- bzw. Untergrenze definiert wird, sind diese dennoch implizit gesetzt. So kann gezeigt werden, dass die Pheromonkonzentration nie unter τ_0 sinken kann und andererseits die Pheromonwerte nie über $1/L_{\psi_{gb}}$ steigen.

ACS kombiniert die Vorteile aus AS, MMAS sowie EAS (vgl. DORIGO & BLUM 2005) und wird daher als Grundlage für die weitere Arbeit verwendet.

5.2.5 Berücksichtigung von Nebenbedingungen

Wie in Abschnitt 2.3 und Kapitel 3 beschrieben, müssen zur Modellierung des Problems zahlreiche Nebenbedingungen dargestellt werden, welche der Ameisenalgorithmus bei der Lösung ebendieses Problems berücksichtigen muss. Im Folgenden wird beschrieben, wie dies realisiert werden kann.

Harte Nebenbedingungen

Harte Nebenbedingungen müssen zwingend eingehalten werden (vgl. Abschnitt 2.3.3). Ein Nicht-Einhalten dieser Bedingungen würde eine Lösung zwangsläufig ungültig machen.¹² Eine ungeschickte Auswahl an harten Nebenbedingungen kann unter Umständen dazu führen, dass keine Lösung für das Problem gefunden werden kann.

¹² Beispiele für harte Nebenbedingungen in der innerbetrieblichen Logistik sind u. a. Kapazitätsgrenzen, bei denen auf der Ladefläche nur ein bestimmtes Volumen Platz findet, Zeitfenster, bei deren Ablauf schwerwiegende Konsequenzen auftreten oder Durchfahrtsbeschränkungen, bei denen eine bestimmte Größe Fahrzeug nicht durch alle Durchfahrten passt.

5 Dynamische Routenplanung

Um garantiert gültige Lösungen zu erlangen, muss der Ameisenalgorithmus den Lösungsraum für die Auswahl des nächsten Knotens einschränken, wie dies in TESCHEMACHER & REINHART (2016, S. 443 ff.) vorgestellt wird. Dazu stehen einem graphbasierten Algorithmus zwei Möglichkeiten zur Verfügung:

1. Der heuristische Wert η wird so modifiziert, dass eine Ameise keine Kanten bzw. Knoten wählt, welche eine harte Nebenbedingung verletzen würden. Die Wahrscheinlichkeit p für eine solche Kante wäre in Konsequenz 0. Problematisch ist die Tatsache, dass sich bei dynamischem η die Pheromonwerte aus Gleichung 5.8 durch das manuelle, nachträgliche Setzen einzelner Werte auf 0 in Relation zueinander verändern und somit der Lernprozess durch die Pheromone ineffizient werden kann.
2. Die Kandidatenliste für eine Ameise wird so reduziert, dass die ungültigen Knoten per se nicht zur Verfügung stehen. Dieses Vorgehen erfordert eine gewisse Vorausschau, welche zusätzlich zur eigentlichen Optimierung Rechenzeit erfordert.

In beiden Fällen ist das Resultat, dass eine Kante zu einer nicht gültigen Station nicht ausgewählt werden kann. Folglich ergibt sich für die Optimierung kein Unterschied, welche Variante gewählt wird. Da sich bei der zweiten Möglichkeit die Gefahr der Verschiebung der relativen Gewichte nicht ergibt, wurde diese für den vorliegenden Fall gewählt. Zu einer ähnlichen Empfehlung kommen B. MEYER & ERNST (2004, S. 169 f.) für allgemeine Anwendungsfälle mit dem Ameisenalgorithmus.

Weiche Nebenbedingungen

Im Gegensatz zu harten Nebenbedingungen dürfen weiche Nebenbedingungen verletzt werden, wodurch sich jedoch die Qualität der Lösung verschlechtert.¹³ Diese Art von Restriktionen muss daher in der Zielfunktion berücksichtigt werden, welche die Ameisen nach jeder Lösungskonstruktion zur Bewertung ihrer Lösung verwenden.

¹³ Weiche Nebenbedingungen können z. B. durch ergonomische Zielsetzungen, Arbeitszeiten der Fahrer oder deutlich zu frühe Anlieferungen entstehen. Weiche Nebenbedingungen enthalten aus Sicht des Entscheidungsträgers daher Aussagen bzgl. der Wichtigkeit einzelner Zielsetzungen.

Ob eine praktische Restriktion als harte oder weiche Nebenbedingung implementiert werden muss, lässt sich zum Teil auch durch organisatorische Änderungen beeinflussen. So können beispielsweise Zeitfenster hart ausgelegt werden (darf nicht überschritten werden) oder auch weich (eine solche Lösung ist zwar schlecht, aber durchaus gültig). Das Formulieren von eigentlich harten Nebenbedingungen als weiche Nebenbedingungen kann das Finden von Lösungen erleichtern bzw. überhaupt erst ermöglichen (vgl. Abschnitt 2.3.3).

5.3 Verwendung des Algorithmus für die Tourenplanung

Nach der grundsätzlichen Beschreibung des ACS wird im Folgenden betrachtet, wie der Algorithmus die Touren für ein innerbetriebliches Logistiksystem erstellen kann. Da die Erstellung der Touren innerhalb einer Fabrik wie ein VRP formuliert werden kann, ist der ACS durch entsprechende Anpassungen auf das Problem anwendbar.

Wie in Abschnitt 1.3 beschrieben, sind dabei für die betrachteten Szenarien der Tourenplanung zwei Zielgrößen relevant: Die Minimierung der Anzahl an Fahrzeugen sowie die Minimierung der insgesamt benötigten Zeit. Die Reduktion der notwendigen Anzahl an Fahrzeugen senkt bei langfristigen Reduktionen den Investitionsbedarf, kurzfristig schafft sie freie Kapazitäten, die in anderen Bereichen eingesetzt werden können. Den gegebenen Auftragsbestand mit so wenig Fahrzeugen wie möglich zu bewältigen hat deshalb Priorität über der Minimierung der Zeit¹⁴.

Die Umsetzung dieser Optimierungskriterien ist in Abbildung 5.5 dargestellt. Um einen Ausgangspunkt für die Optimierung zu haben, wird mit einer einfachen Heuristik eine initiale Lösung erzeugt. Je besser diese Lösung ist, desto schneller findet der Ameisenalgorithmus im Anschluss eine gute Lösung. Essenziell ist jedoch nur, dass es sich um eine gültige Lösung handelt, d. h. dass alle harten Nebenbedingungen erfüllt sind. Aus der initialen Lösung kann die Anzahl der benötigten Fahrzeuge n bestimmt werden. Im Anschluss wird die eigentliche Optimierung mittels Ameisenalgorithmus gestartet. Dazu laufen zwei Algorithmen parallel: Algorithmus 1 sucht nach einer

¹⁴ Die Minimierung der Zeit ist ein Ziel, welches prinzipiell dabei hilft, die Anzahl der Routenzüge zu minimieren. Als alleiniges Ziel ist es jedoch nicht geeignet, da das Hauptziel die Reduktion der notwendigen Kapazitäten ist, wodurch die Zeit zum Teil auch höher sein wird, als mit mehr Fahrzeugen.

5 Dynamische Routenplanung

Lösung mit $n - 1$ Fahrzeugen, während Algorithmus 2 mit n Fahrzeugen eine schnellere Route sucht. Wird eine Lösung mit weniger Fahrzeugen gefunden, so werden beide Algorithmen abgebrochen, der Wert n um 1 erniedrigt und anschließend beide Algorithmen neu gestartet. Sobald ein Abbruchkriterium erreicht ist (bspw. Anzahl Iterationen ohne Verbesserung, maximale Rechenzeit, etc.) wird das aktuell beste Ergebnis aus Algorithmus 2 zur Disposition der Logistikfahrzeuge freigegeben.

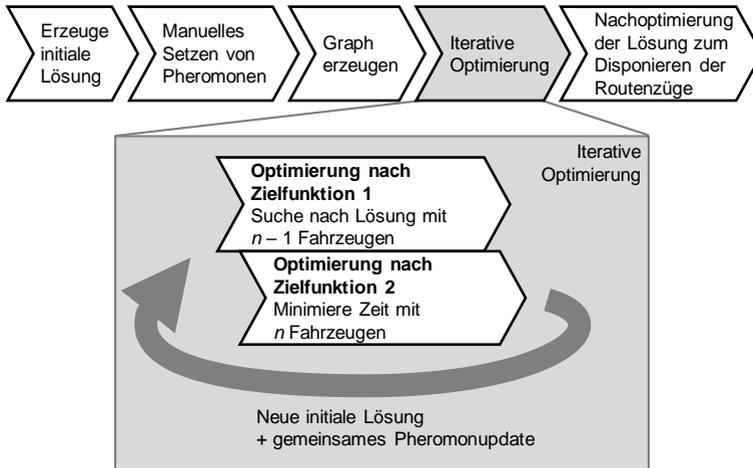


Abbildung 5.5: Zusammenspiel der parallel optimierenden Algorithmen zur Realisierung von zwei Zielfunktionen

Die Struktur der beiden parallel optimierenden Ameisenalgorithmen aus Abbildung 5.5 ist in Abbildung 5.6 gezeigt, welche beschreibt, wie der Ameisenalgorithmus aus Abschnitt 5.2.4 zur Generierung von Touren eingesetzt wird. Dabei ist das iterative Vorgehen zum Platzieren von Aufträgen auf dem Routenzug unter Berücksichtigung der Nebenbedingungen gezeigt. Die Ameise prüft zu Beginn jeder Iteration, ob noch zusätzliche Stationen einzuplanen sind. Dann wird entschieden, ob diese auf dem derzeit beplanten Routenzug Platz finden oder ob ein zusätzlicher eingepplant werden muss. Bei der Prüfung ist zudem sicherzustellen, dass sämtliche Nebenbedingungen erfüllt sind. Nach dem Hinzufügen zum Tourenplan wird die nächste Iteration gestartet. Zum Abschluss wird das Ergebnis zur weiteren Verarbeitung exportiert. Die nächsten Schritte umfassen die Entscheidung, welche Touren wann gestartet werden sollen sowie die Aufbereitung für die Routenzugfahrer.

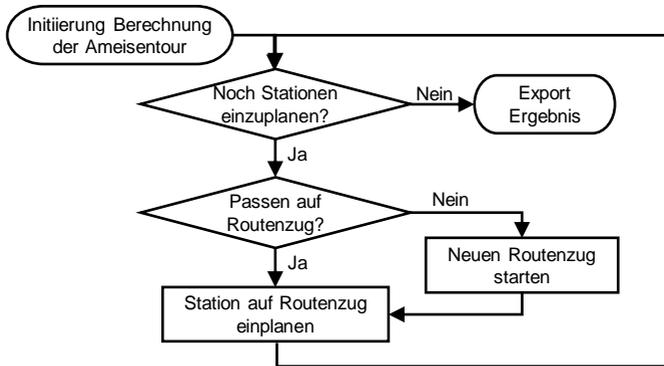


Abbildung 5.6: Zusammenstellung der Touren durch den Ameisenalgorithmus im Rahmen der Optimierungsläufe

5.4 Zyklische Wiederverwendung logistischer Ressourcen

Das grundlegende Tourenplanungsproblem verteilt die auszuliefernden Güter auf verschiedene Fahrzeuge, die alle zum selben Zeitpunkt starten und jeweils nur eine einzige Tour fahren. Im Falle eines Tourenplanungsproblems ohne Zeitfenster kann ohne eine Anpassung der Touren jedes Fahrzeug zu einem beliebigen Zeitpunkt starten, ohne dass die Lösung ungültig wird. Anstelle der Verteilung auf verschiedene Fahrzeuge könnte ein einziges Fahrzeug daher auch alle Touren nacheinander abfahren. Bei VRPs mit Zeitfenstern ist dies nicht möglich, da eine Verschiebung nach hinten im Regelfall zur Verletzung von Zeitfenstern führt. Die zyklische Wiederverwendung der Transportfahrzeuge muss daher explizit vom Algorithmus betrachtet werden.

Zur Verwendung in der Problemformulierung, wie sie in der innerbetrieblichen Logistik auftritt, müssen daher die jeweiligen Startzeiten der Transportmittel berücksichtigt werden. Bei großen oder weit in der Zukunft liegenden Zeitfenstern muss außerdem betrachtet werden, dass ein Fahrzeug gegebenenfalls auch eine oder mehrere nachfolgende Touren zusätzlich ausfahren kann. Dies kann in vielen Szenarien dazu führen, dass weniger Fahrzeuge als Touren benötigt werden.

Standardmäßig wird bei der Lösung von Tourenplanungsproblemen von einem Startzeitpunkt t_0 aus gerechnet, der für alle Fahrzeuge als identisch angenommen

5 Dynamische Routenplanung

wird. In der Intralogistik muss hingegen für ein Fahrzeug m für $t_{0,m}$ wahlweise die aktuelle Zeit oder, falls das Fahrzeug noch nicht am Depot ist, die erwartete Rückkehr als Startzeitpunkt verwendet werden. Diese Veränderung am Startzeitpunkt hat bei zeitpunktabhängigen Einflussgrößen (siehe dazu auch Abschnitt 5.5) auch Einfluss auf die absolute Dauer einer Tour.

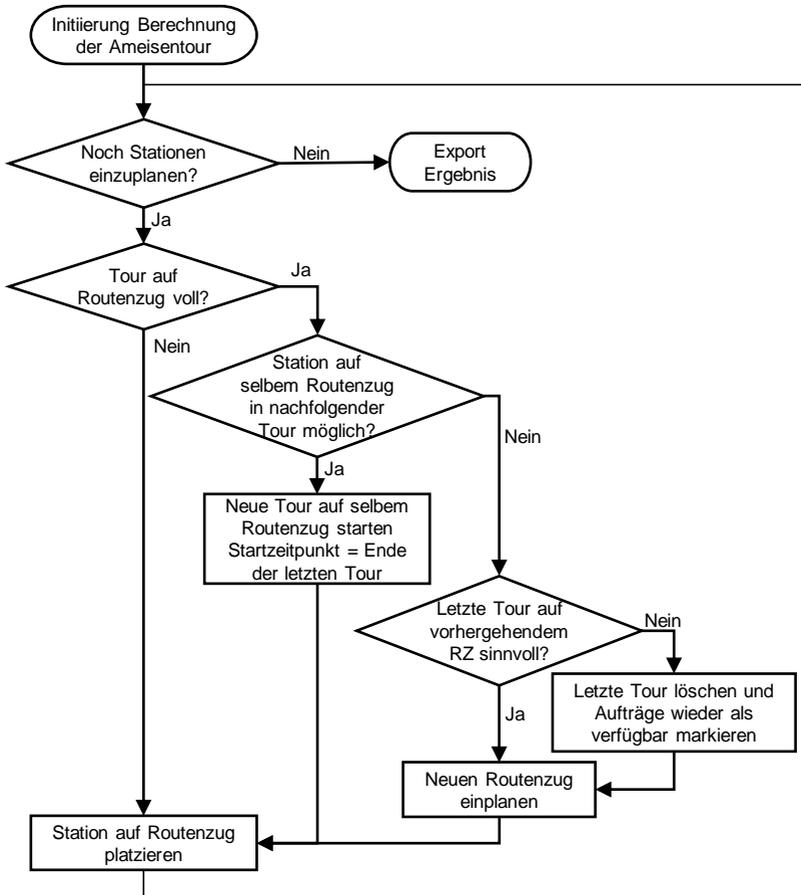


Abbildung 5.7: Erweiterter Ablauf der Tourenkonstruktion im Ameisenalgorithmus bei zyklischer Wiederverwendung von Transportfahrzeugen

Diese Anpassung ist mit der klassischen Implementierung des Ameisenalgorithmus nicht zu realisieren. In Abbildung 5.7 ist daher der dafür notwendige erweiterte

5.5 Berücksichtigung zeitlich veränderlicher Einflussgrößen

Ablauf dargestellt, wie die Ameisen bei der Zusammenstellung von Touren über die Auswahl von Stationen entscheiden. Zunächst wird wie im Standard-VRP geprüft, ob noch nicht eingeplante Aufträge übrig sind. Dann wird betrachtet, ob die Tour auf dem Routenzug bereits voll ist. Falls noch Plätze frei sind und sich Stationen unter Berücksichtigung aller Nebenbedingungen dort einfügen lassen, werden diese unter Verwendung des ACS in die noch offene Tour eingefügt. Für den Fall, dass die Tour bereits voll ist, wird nach Möglichkeit eine neue Tour auf demselben Fahrzeug geplant. Diese startet direkt nach Abschluss der vorherigen Tour und wird demnach bei zeitlich nahegelegenen Zeitfenstern entsprechend nur noch wenige potenzielle Aufträge enthalten. Lassen sich keine neuen Aufträge mehr auf dem Routenzug in dieser oder in weiteren Touren platzieren, so wird zunächst geprüft, ob die letzte eingeplante Tour sinnvoll¹⁵ im Sinne einer hohen Auslastung ist. Konnten nur sehr wenige Abrufe darauf platziert werden, wird die Tour verworfen und die Aufträge zur Kommissionierung auf den nächsten Routenzügen wieder freigegeben.

Das Vorgehen nach Abbildung 5.7 führt zu einer Belegung, bei der möglichst wenige Routenzüge eingeplant werden. Ein Spezialfall kann auftreten, wenn einer der ersten Routenzüge keinen einzigen Auftrag erhält, weil seine Rückkehrzeit zum Depot zu spät zur Einhaltung der Zeitfenster der Aufträge ist. Dieses Vorgehen ist erwünscht, da es zu einer Beplanung führt, welche mit einer minimalen Anzahl an Fahrzeugen auskommt, aber dennoch sicherstellt, dass alle Stationen rechtzeitig beliefert werden.

5.5 Berücksichtigung zeitlich veränderlicher Einflussgrößen

In Abschnitt 5.1.1 wurde das zugrundeliegende graphenbasierte Modell des VRP so erweitert, dass dort zeitlich veränderliche Einflussgrößen an den Stationen in Form von zeitpunktabhängigen Servicezeiten dargestellt werden können. Damit diese bereits bei der Lösungsfindung richtig berücksichtigt werden, muss auch der Optimierungsalgorithmus an mehreren Stellen angepasst werden. Der Ameisenalgorithmus bietet hierzu an drei Stellen die Möglichkeit zur Adaption. Dies ist die Zielfunktion, welche u. a. das Maß der Pheromonablage bestimmt, die heuristische

¹⁵ Konnte bspw. nur eine einzige Station auf einem Routenzug in einer späteren Tour eingeplant werden, so ist es zu bevorzugen, diese auf anderen Routenzügen zu vollständigen Touren einzuplanen.

5 Dynamische Routenplanung

Funktion, welche die Wahrscheinlichkeiten zur Auswahl einer Kante bzw. einer Station beeinflusst sowie die Gewichtung zwischen der Pheromonkonzentration und der heuristischen Information in der Übergangsregel. Alle drei Stellhebel werden im Folgenden betrachtet und angepasst.

5.5.1 Adaption der Zielfunktion

Wie in Abschnitt 2.3.6 erläutert, werden die meisten Tourenplanungsprobleme nach der Distanz optimiert. Da im innerbetrieblichen Umfeld jedoch nicht die zurückgelegte Strecke den entscheidenden Kostenfaktor darstellt, sondern die benötigte Zeit, bis der Routenzug wieder für neue Aufträge zur Verfügung steht, muss diese als Zielgröße genutzt werden. Es lässt sich daher allgemein für die Zielgröße schreiben:

$$\min t_{ges} \quad (5.15)$$

Die darin enthaltene Variable t_{ges} lässt sich wie in Gleichung 5.2 formelmäßig ermitteln. Die Formel setzt sich aus den Bestandteilen für die Fahrtzeit $t_{F,i-1 \rightarrow i}$ und den Umladezeiten bzw. Servicezeiten $t_{s,i}(t_i)$ zusammen, wobei die Umladezeiten vom Zeitpunkt t_i der Ankunft an der Station abhängig sind und sich durch Gleichung 5.1 bestimmen lassen.

Für die Optimierung lässt sich daher die Zielfunktion

$$\min t_{Komm.} + \sum_{i=1}^n [t_{F,i-1 \rightarrow i} + t_{s,i}(t_i)] \quad (5.16)$$

aufstellen.

Die Anpassung der Zielfunktion verändert die Entscheidung, welche Lösungen der Ameisen als gut erachtet werden. Dies beeinflusst auch die Ablage der Pheromone und damit die Lösungskonstruktion der neuen Iteration.

5.5.2 Berücksichtigung in der heuristischen Information

Einer der großen Vorteile der ACS-Meta-Heuristik ist die Möglichkeit, angelearnertes Wissen über das Problem durch Pheromone mit dem inhärent vorhandenen Wissen in

Form der Heuristik zu kombinieren. Zum schnelleren Finden einer guten Lösung nach Gleichung 5.16 ist daher die Adaption der Formel zur Ermittlung der heuristischen Information ebenfalls essenziell für die Leistungsfähigkeit des Algorithmus.

Grundlegende Ermittlung

Wie in Abschnitt 5.2.2 beschrieben, ist die heuristische Information eine der beiden Bestandteile der Übergangsregel (Gleichung 5.8). Gemeinsam mit dem Pheromonwert der entsprechenden Kante ergibt sie die relative Wahrscheinlichkeit für die Auswahl des mit ihr verknüpften Knotens und somit der Station. Dieses Vorgehen ermöglicht die Verknüpfung eines Erfahrungswertes in Form von Pheromonen mit lokal vorhandenem Wissen über die heuristische Funktion.

Im einfachsten Fall ist dies wie beim TSP der Kehrwert der Distanz, so dass der Zusammenhang $\eta_{ij} = \frac{1}{d_{ij}}$ gilt (vgl. Abschnitt 5.2.2). Je höher der Wert der heuristischen Funktion, desto wahrscheinlicher ist die Auswahl als nächster Knoten. In diesem Fall ist die Auswahl des nachfolgenden Knotens daher bei einer kurzen Strecke wahrscheinlicher als bei einer langen. Für komplexere Probleme muss die Heuristik entsprechend angepasst werden. Dies kann beispielsweise so erfolgen, dass eine zwar nahegelegene Station, die jedoch ein unpassendes Zeitfenster hat, einen niedrigen heuristischen Wert aufweist.

Eine gut gewählte heuristische Funktion hilft dem Ameisenalgorithmus, schnell zu guten Lösungen zu finden. Das Finden einer guten heuristischen Funktion ist aufgrund der Problemkomplexität nicht immer möglich, was zwar die finale Lösungsqualität nicht nachhaltig beeinflusst, jedoch längere Rechenzeiten mit sich bringt.

Erweiterung der Heuristik

Zur Einbeziehung der Zeitfenster und der variablen Servicezeiten wird die Heuristik zur besseren und schnelleren Lösungsfindung daher an die Problemcharakteristik angepasst. Folgende Axiome werden dabei als Grundlage für die Wahl der Nachfolgestation angenommen:

5 Dynamische Routenplanung

- Eine Station ist dann gut geeignet, wenn zum Erreichen nur wenig Fahrzeit t_F benötigt wird.
- Eine Station ist dann gut geeignet, wenn nicht an der Station das Erreichen der vorderen Grenze des Zeitfensters abgewartet werden muss oder zumindest diese Zeit t_W möglichst gering ist.
- Eine Station wird dann bevorzugt, wenn die Umladezeiten $t_{s,i}$ möglichst gering sind.

Diese drei Annahmen können im Einzelfall dieselbe Station als bevorzugt auswählen, im Allgemeinen werden jedoch unterschiedliche Stationen zur Auswahl stehen. Da sämtliche zur Auswahl herangezogenen Parameter Zeitkomponenten sind und in der Zielfunktion die Gesamtzeit als Optimierungsgröße definiert ist, kann die Summe dieser drei Komponenten als Berechnungsgrundlage für die heuristische Information genutzt werden. Dabei ergibt sich¹⁶:

$$\eta_{ij} = \frac{1}{t_F + t_W + t_{s,i}} \quad (5.17)$$

Zusammengefasst ergibt sich im Nenner die Zeit bis zum Abschluss der Lieferung an die Station. Abgesehen von der Fahrzeit t_F sind sämtliche Zeiten der heuristischen Funktion von den vorhergehenden Stationen des Routenzugs abhängig, welche die Ankunftszeit an der Station und somit sowohl die notwendige Wartezeit t_W als auch die notwendige Servicezeit $t_{s,i}$ zum Umladen beeinflusst.

5.5.3 Veränderung des Verhältnisses zwischen Pheromonkonzentration und heuristischem Wert

Durch die Veränderlichkeit von η_{ij} in Gleichung 5.17 wird die Gewichtung $\frac{\tau^\alpha \cdot \eta^\beta}{\sum \tau^\alpha \cdot \eta^\beta}$ der Übergangsregel gestört und somit der sich selbst verstärkende Lerneffekt durch die Pheromonablage potenziell ineffizient (vgl. hierzu u. a. B. MEYER & ERNST 2004,

¹⁶ Je größer der heuristische Wert, desto höher ist die Wahrscheinlichkeit für die Auswahl der Kante. Da eine höhere Summe der einzelnen Faktoren hier eine Verschlechterung darstellt, muss analog zu Abschnitt 5.2.2 der Kehrwert genutzt werden.

S. 170). Da die veränderlichen heuristischen Informationswerte für die Funktionsweise des Algorithmus jedoch zwingend erforderlich sind, muss durch möglichst gut passende Werte für α und β die korrekte Funktionsweise des Algorithmus gewährleistet werden (TESCHEMACHER & REINHART 2017, S. 766).

Eine analytische Vorgehensweise zur Ermittlung von α und β kann nicht gegeben werden. Werte in der Größenordnung von $\alpha = 1$ ermöglichen einen guten Kompromiss zwischen der Weiterverwendung alter Touren und der Möglichkeit zum Vergessen zu langer Touren (DORIGO & STÜTZLE 2004, S. 18 f.). Für den Parameter β empfehlen DORIGO & STÜTZLE (2004, S. 71) Werte zwischen 2 und 5, mit den in dieser Arbeit dargestellten Erweiterung erwiesen sich jedoch auch kleinere Werte von $1 \leq \beta \leq 2$ als gute Konfiguration.

5.6 Priorisierung kurzfristiger Abrufe

Die bisherigen Ausführungen beschreiben ein Vorgehen, bei dem alle Aufträge mit der selben Priorität eingeplant werden. Diese Annahme ist nicht für sämtliche Einsatzfälle zutreffend, weshalb im Folgenden mögliche Szenarien und ein Lösungsansatz dafür präsentiert werden.

5.6.1 Szenarien für kurzfristige Priorisierung

Unter der Bedingung, dass der geplante Logistikprozess fehlerfrei abläuft, sind keinerlei kurzfristige Aktionen notwendig – alle Materialabrufe können über der Standardprozess abgehandelt werden. Da allerdings selbst bei größter organisatorischer Sorgfalt Fehler auftreten können, müssen Alternativprozesse für Materialabrufe definiert werden, die im Standardvorgehen nicht abgearbeitet werden können.

Der in den vorhergehenden Kapiteln entwickelte Algorithmus ist in der Lage, Lösungen zu generieren, bei denen sämtliche Materialabrufe in der definierten Zeitspanne ausgeliefert werden können. Sind jedoch Behälter auszuliefern, deren hinteres Zeitfenster bereits so nah ist, dass eine pünktliche Lieferung bereits rein mathematisch nicht mehr möglich ist, so muss ein Vorgehen gewählt werden, welches die Verspätung der Belieferung zumindest so gering wie möglich hält.

5 Dynamische Routenplanung

Solche Szenarien treten zum Beispiel auf, wenn Material nicht rechtzeitig bestellt oder zu spät aus dem Lager bereitgestellt wurde. Auch wenn falsche Transporte durchgeführt wurden und demnach im Nachgang innerhalb von kurzer Zeit das richtige Material nachgeliefert werden muss entsteht die Notwendigkeit für priorisierte Transporte. Ein weiteres Szenario tritt im Fall von Mängeln auf, bspw. durch größere Mengen Ausschuss an Material, welche eine erneute Lieferung erfordern.

Sämtliche dieser Szenarien müssen von der Steuerungslogik gehandhabt werden können. Gegebenenfalls muss die Steuerung auch darauf adaptieren, dass Fahrer spontan besonders dringend erscheinende Materialien außerhalb jeglicher definierter Prozesse ausfahren. Derartiges Vorgehen sollte organisatorisch zwar unterbunden werden, da sonst die Vorteile durch die Optimierung nicht genutzt werden können, dennoch muss die Steuerung dieses denkbare Vorgehen handhaben können.

5.6.2 Berücksichtigung der Priorisierung

Der bereits beschriebene Standardablauf berücksichtigt sowohl die Rückkehrzeiten als auch die Länge der Touren, um eine optimale Tour ermitteln zu können. Zwar ist diese nach wie vor oberstes Ziel, dennoch überwiegt in den beschriebenen Szenarien die Notwendigkeit der priorisierten Auslieferung dringender Bedarfe.

Deshalb werden Eilaufträge nach dem in Abbildung 5.8 dargestellten Prozess vor der eigentlichen Beplanung auf die bereitstehenden Transportmittel eingeplant. Im Falle von mehreren Fahrzeugen werden die Eilaufträge gleichmäßig verteilt. Steht nur ein Fahrzeug zur Verfügung, werden diesem alle Eilaufträge zugeteilt. Ist hingegen kein Fahrzeug verfügbar, so wird das nächste eintreffende Fahrzeug genutzt. Sofern aus Kapazitätsgründen nicht zwingend erforderlich, wird nicht mit der rechtzeitigen Rückkehr anderer Fahrzeuge kalkuliert, da durch Unwägbarkeiten jederzeit zusätzliche Abweichungen auftreten können. Die verbleibenden Plätze auf den Fahrzeugen werden mit Standardlieferungen aufgefüllt.

Sind auf einem Fahrzeug zwei oder mehr Eilaufträge platziert, so wird deren Auslieferungsreihenfolge durch die Anwendung des Ameisenalgorithmus ermittelt. Die restlichen Plätze werden durch den global für alle Fahrzeuge gemeinsam rechnenden Algorithmus belegt und in die optimale Reihenfolge gebracht. Der Algorithmus berücksichtigt für die Fahrzeuge die Prämisse, dass die Eilaufträge alle zu Beginn

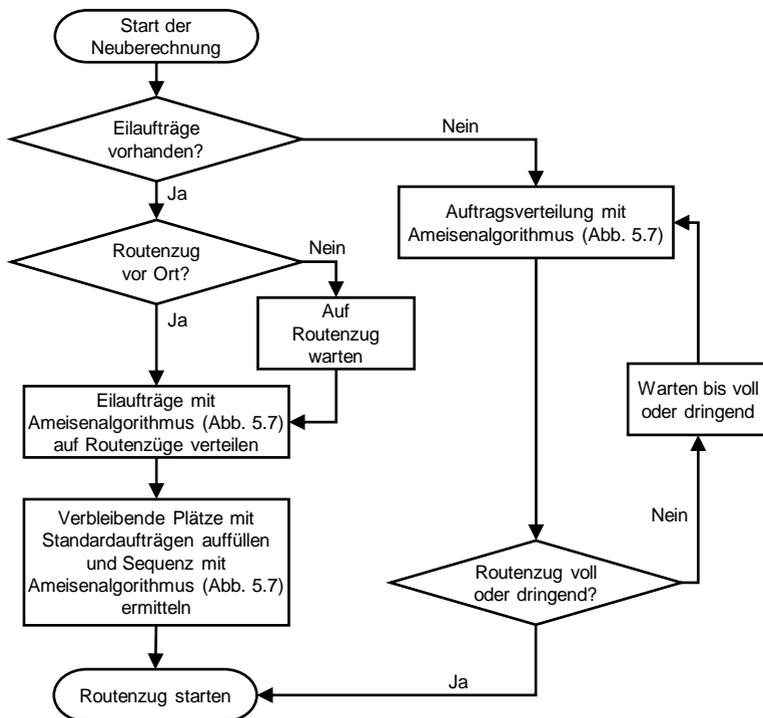


Abbildung 5.8: Integration der Eilaufträge in das standardmäßige Vorgehen

der Tour erledigt werden müssen, selbst wenn aus Sicht einer optimierten Tour eine andere Reihenfolge zu bevorzugen wäre. Das Verfahren stellt eine unter den gegebenen Umständen schnellstmögliche Auslieferung sicher.

Sind auf einem Routenzug Eilaufträge geladen, muss das Depot so schnell wie möglich verlassen werden. Das heißt konkret, dass selbst wenn auf einem Routenzug mit Eilaufträgen noch freie Plätze vorhanden sind, diese aber im Augenblick nicht mit anderen Aufträgen belegt werden können, die Tour sofort gestartet werden muss. Das Vorgehen ist in Abbildung 5.8 zusammengefasst.

Eine Besonderheit kann auftreten, wenn wenige Materialabrufe im Lager eingehen. Da es bei einem nicht vollständig ausgelasteten Fahrzeug ggf. sinnvoll ist, auf zusätzliche Aufträge zu warten, um die Auslastung zu erhöhen, kann ein normaler

5 Dynamische Routenplanung

Auftrag mit der Zeit zu einem dringlichen Auftrag werden. Dies ist vor allem bei der Implementierung zu berücksichtigen – das theoretische Vorgehen gleicht dem hier beschriebenen.

5.7 Lokale Suchverfahren zur Verbesserung der Lösung

Die bisher beschriebenen Ergänzungen führen zu einer funktionalen Erweiterung des Algorithmus. Im Folgenden sollen dazu ergänzend Methoden zur Verbesserung der Lösungsqualität aufgezeigt werden.

Wie im allgemeinen Ablauf des Ameisenalgorithmus (Algorithmus 1, Abschnitt 5.2.2) gezeigt, können nach der Lösungskonstruktion und dem Pheromonupdate noch sogenannte *Daemon Actions* durchgeführt werden. Im Rahmen dieses Schrittes können Verbesserungen an den bestehenden Lösungen erzielt werden, welche von einem einzelnen Individuum nicht realisiert werden könnten, da nicht-lokale Informationen notwendig sind. (vgl. DORIGO & STÜTZLE 2004, S. 62)

Im Folgenden werden zwei im entwickelten Ansatz eingesetzte Verbesserungsverfahren vorgestellt, welche die Lösungsqualität erhöhen können.¹⁷

5.7.1 Einfügeverfahren

Bei der Optimierung durch den Ameisenalgorithmus kann u. U. eine unvollständige Lösung entstehen, d. h. ein Tourenplan, der nicht alle zu beliefernden Stationen enthält. Unvollständige Lösungen entstehen in der Regel, weil Nebenbedingungen wie Zeitfenster vor allem bei der Suche nach Touren mit weniger Fahrzeugen für die noch übrigen Stationen nicht mehr eingehalten werden können. Diverse Quellen (z. B. GAMBARDILLA ET AL. 1999, S. 11; BALSEIRO ET AL. 2011; KINDERVATER & SAVELSBERGH 1997) schlagen als Lösung hierfür Einfügeheuristiken (siehe Abschnitt 4.2.1) vor, die versuchen, durch gezieltes Probieren nachträglich noch einen Platz für die verbleibenden Stationen zu finden. Dabei wird i. d. R. die Position mit der geringsten Verschlechterung der Route gewählt. Das Auswählen dieser Position kann bei

¹⁷ Verbesserungsverfahren können dabei nur bestehende Lösungen verbessern. Enthält eine Lösung kleinere Unstimmigkeiten, können diese verbessert werden. Komplett neue Lösungen sind jedoch nicht zu erwarten.

größeren Problemen mit vielen Nebenbedingungen erheblichen Rechenaufwand mit sich bringen, weshalb eine auf das gegebene Problem angepasste Einfügeheuristik entwickelt wurde und im Folgenden beschrieben wird.

Bei jedem Einfügen einer Station in bestehende Touren gilt es, die bestehenden Nebenbedingungen zu berücksichtigen und deren Einhaltung sicherzustellen. Zwei Überlegungen sind dabei entscheidend: Zum einen ist eine allgemeine Einschätzung, wie wahrscheinlich das Verletzen einer bestimmten Nebenbedingung in Relation zu den übrigen Nebenbedingungen ist, erforderlich. Zum anderen muss der Rechenaufwand für das Prüfen auf Einhaltung der Nebenbedingung abgeschätzt werden. Sind im vorliegenden Problem die Kapazitäten pro Transportfahrzeug niedrig, so sind Touren meistens sehr voll – das Platzieren eines zusätzlichen Auftrags würde die Nebenbedingung für die maximale Beladung daher häufig verletzen. Zusätzlich ist das Überprüfen, ob diese Nebenbedingung erfüllt ist, deutlich recheneffizienter als das Prüfen der eingehaltenen Zeitfenster. Da für eine mögliche Einfügeposition beide Nebenbedingungen erfüllt sein müssen, ist es aus Gründen der Recheneffizienz daher zu bevorzugen, die Nebenbedingung für die maximale Beladung zuerst zu prüfen und erst bei deren Erfüllung im Anschluss die Einhaltung der Zeitfenster abzu prüfen.

Mit dem Ziel, die Recheneffizienz beim Prüfen der Zeitfenster zu optimieren, wird für jede Station eine Pufferzeit ausgerechnet, die angibt, um welche Zeitspanne eine Anlieferung nach hinten verschoben werden kann. Dazu wird beginnend vom letzten Element der Tour der jeweils letztmögliche Ankunftszeitpunkt errechnet. Für das letzte Element ist dies das Ende des Zeitfensters $t_{e,i}$. Für die davor liegenden Elemente errechnet sich der letztmögliche Ankunftszeitpunkt jeweils durch

$$t_{\text{letztm\u00f6glich},i} = \min(t_{e,i}, t_{e,i+1} - t_{F,i \rightarrow i+1}). \quad (5.18)$$

Die Differenz aus geplanter Ankunftszeit und spätester Ankunftszeit ergibt die Pufferzeit. Durch das Ermitteln der letztmöglichen Ankunftszeit sind hier implizit auch die minimalen Umladezeiten bedingt durch die implementierte Zeitabhängigkeit berücksichtigt.

Die Pufferzeiten müssen für sämtliche einzufügenden Stationen nur einmal berechnet werden. Für das Einfügen muss nun ermittelt werden, wie viel Zeit für den Umweg über die neue Station benötigt wird. Ist dieser kleiner als die Pufferzeit, so kann diese

Station in die Tour integriert werden, sofern dabei die übrigen Nebenbedingungen nicht verletzt werden. Für die Stationen der Tour, die vor der neu eingefügten Station liegen, müssen im Anschluss die Pufferzeiten aktualisiert werden. Ist der Umweg größer als die errechnete Pufferzeit, so ist ein Einfügen schon theoretisch nicht mehr möglich.

5.7.2 Lokale Optimierungsmaßnahmen

Obwohl sich der Ameisenalgorithmus in vielen Tests als effiziente Möglichkeit zum Durchsuchen eines großen Lösungsraum beweist, können die erzielten Lösungen im Detail kleine Schwächen aufweisen. Dies kann beispielsweise eine verdrehte Reihenfolge zweier benachbarter Stationen sein, deren Vertauschung zu einer geringen Verbesserung der Lösung führt. Da es sich nur um eine kleine Änderung handelt, können im Rahmen der ersten Iterationen ungünstig platzierte Pheromone dazu führen, dass diese Fehler erst nach vielen Iterationen ausgeglichen werden. Aufgrund von Echtzeitanforderungen wird die Lösungsfindung im vorliegenden Anwendungsfall jedoch schon frühzeitig beendet, weshalb die Lösung mit lokalen Suchverfahren (siehe Abschnitt 4.2.1) nachträglich optimiert wird.

1. Einfachstes Verfahren ist das Verschieben einer einzelnen Station in eine andere Tour an verschiedene Stellen (vgl. Abbildung 5.9a). Verbessert sich durch das Verschieben einer Station die Gesamttourenplanung, so wird die verbesserte Lösung für die weitere Lösungsermittlung übernommen. Auch diese Optimierung kann anhand der errechneten Pufferzeiten (vgl. Abschnitt 5.7.1) effizient durchgeführt werden. Das Verschieben von Stationen kann vor allem bei der Suche nach Touren mit einer verringerten Anzahl an Fahrzeugen zu positiven Veränderungen in der Suche führen.
2. Zur Verbesserung der Touren können auch zwei Stationen, wahlweise aus der gleichen oder aus unterschiedlichen Touren, miteinander vertauscht werden (vgl. Abbildung 5.9b). Verkürzt sich die Gesamtlänge der Touren, wird diese Lösung für die weitere Tourenplanung übernommen.
3. Zwei Segmente einer Tour können wie einzelne Stationen gegeneinander vertauscht werden. Dabei kann nach Belieben auch die Reihenfolge eines Segments umgekehrt werden (vgl. Abbildung 5.9c). Die Segmente können,

5.7 Lokale Suchverfahren zur Verbesserung der Lösung

müssen jedoch nicht zwangsläufig, dieselbe Länge haben. Im Extremfall kann auch ein Segment keine Stationen enthalten oder das umgekehrte Segment an der gleichen Stelle wieder eingefügt werden. Die zu vertauschenden Segmente können innerhalb einer Tour oder in verschiedenen Touren liegen.

Die drei in der Literatur separat behandelten Verfahren lassen sich kompakt auf ein einziges Verfahren reduzieren, indem das dritte Verfahren verallgemeinert wird. Tauscht man zwei Segmente mit nur einem Element gegeneinander aus, so ergibt sich das zweite Verfahren, tauscht man ein Segment mit einem Element gegen ein leeres Element, so ergibt sich das erste Verfahren.

Lokale Optimierungsverfahren wie hier gezeigt erzeugen zusätzlichen Rechenaufwand, der stark ansteigt, wenn viele Vertauschungen getestet werden. Um die Rechengeschwindigkeiten hoch zu halten, sind daher nur geringe Optimierungsmaßnahmen realisierbar.

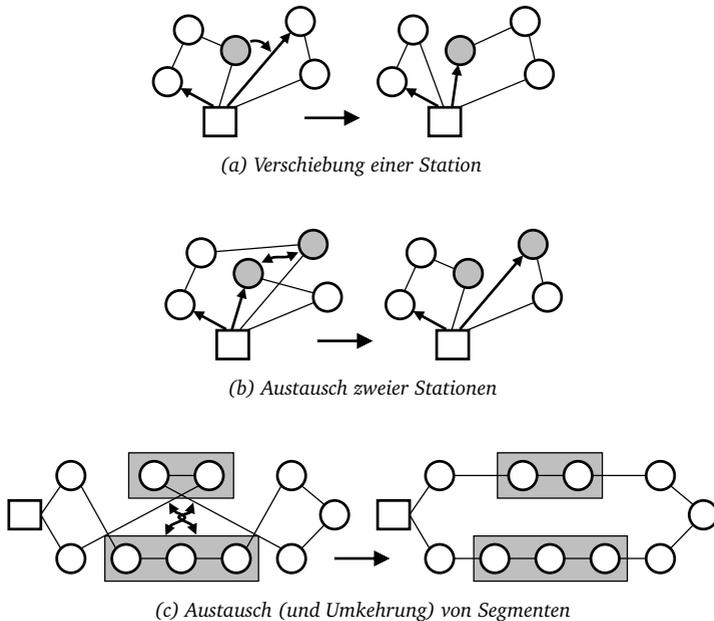


Abbildung 5.9: Lokale Optimierungsverfahren für den Einsatz in der fabrikinernen Logistik

5.8 Aufbau des Algorithmus zur Logistikoptimierung

In den vergangenen Abschnitten wurden einzelne Erweiterungen sowie deren theoretische Lösung aufgezeigt. Im Folgenden werden diese nun in einen gesamten Algorithmus inklusive dessen abstrahierter Implementierung in Form von Pseudocode¹⁸ zusammengeführt.

Die übergeordnete Funktion des Ameisenalgorithmus wird in Algorithmus 2 realisiert. Zu Beginn wird die initiale Lösung Ψ^0 generiert, welche durch eine Heuristik (z. B. die Methode des nächsten Nachbarn, siehe Abschnitt 4.2.1) erzeugt wird. Danach werden sowohl der Algorithmus *ACS_FAHRZEUGE* zur Optimierung der Fahrzeuganzahl als auch der Algorithmus *ACS_ZEIT* zur Optimierung der Fahrzeit parallel gestartet (vgl. Abbildung 5.5). Wird eine verbesserte Lösung gefunden, so wird diese in Ψ^{gb} gespeichert. Eine Lösung gilt dann als besser, wenn sie entweder weniger Fahrzeuge zur kompletten Abarbeitung oder zumindest mit der selben Anzahl an Fahrzeugen weniger Zeit benötigt. Dementsprechend muss nur eine Lösung gespeichert werden, obwohl zwei unterschiedliche Zielgrößen mit parallel laufenden Algorithmen verfolgt werden. Sobald das Abbruchkriterium erfüllt ist, wird Ψ^{gb} zurückgegeben. Sollte bis zu diesem Zeitpunkt noch keine optimierte Lösung gefunden worden sein, ist in Ψ^{gb} die initiale Lösung Ψ^0 gespeichert, weshalb in jedem Fall eine gültige Lösung zurückgegeben wird.

Als Abbruchkriterium kommen verschiedene Bedingungen infrage: Bei zeitkritischen Anwendungen ist es ratsam, die maximale Rechenzeit vorzugeben. Bei Berechnungen, deren Hauptziel möglichst gute Lösungen sind, kann zum Beispiel die Maximalanzahl an Iterationen ohne Verbesserung oder bei ungefähigem Wissen über die zu erreichende Lösungsgüte eine fest definierte Schranke als Abbruchkriterium dienen.

Algorithmus 3 zeigt den Aufbau zur Optimierung nach der Fahrzeit. Zunächst wird für jede Ameise k eine Lösung generiert, indem die Funktion *Ameisenlösung_konstruieren* (Algorithmus 5) aufgerufen wird. Falls eine der Lösungen gültig und besser als

¹⁸ Wie in Abschnitt 5.2.2 bereits angedeutet, dient Pseudocode nur zur abstrahierten Darstellung und Erläuterung von Abläufen in Algorithmen. Pseudocode kann nicht direkt von Computern interpretiert werden, der Algorithmus muss daher in im Detail in einer von Maschinen interpretierbaren Programmiersprache implementiert werden.

```

1 function ACS_TOURENOPTIMIERUNG()
  /*  $\Psi^{gb}$  speichert die bislang beste Lösung (d. h.
    niedrigste Anzahl an Fahrzeugen sowie kürzeste
    Fahrtzeit) */
2   $\Psi^{gb} \leftarrow \Psi^0$  /* Die bislang beste Lösung wird mit einer
    initialen Lösung befüllt, welche durch die Methode
    des nächsten Nachbarn ermittelt wird. */
3  repeat
4     $n \leftarrow$  Anzahl Fahrzeuge in  $\Psi^{gb}$ 
5    starte ACS_FAHRZEUGE( $n - 1$ )
6    starte ACS_ZEIT( $n$ )
7    while ACS_FAHRZEUGE und ACS_ZEIT aktiv do
8      warte auf eine verbesserte Lösung  $\Psi$  von ACS_FAHRZEUGE oder
        ACS_ZEIT
9       $\Psi^{gb} \leftarrow \Psi$ 
10     if Lösung  $\Psi$  mit geringerer Anzahl Fahrzeuge then
11       | beende ACS_FAHRZEUGE und ACS_ZEIT
12   until Abbruchkriterium erfüllt
13   return  $\Psi^{gb}$ 

```

Algorithmus 2: Struktur des Algorithmus zur Tourenoptimierung als Pseudocode

die bisher beste Lösung ist, wird diese in Ψ^i gespeichert. Im Anschluss wird das Pheromonupdate nach Gleichung 5.13 für alle Kanten der global besten Lösung durchgeführt. Dieses Vorgehen wird iterativ wiederholt, bis das zuvor definierte Abbruchkriterium erfüllt wurde.

Während die Funktion *ACS_ZEIT* (Algorithmus 3) versucht, mit der gegebenen Anzahl an Fahrzeugen die Fahrzeit zu verringern, wird in *ACS_FAHRZEUGE* (Algorithmus 4) nach einer Lösung mit weniger Fahrzeugen gesucht, auch wenn diese eine längere Fahrtzeit zur Folge hat. Auch hier wird im ersten Schritt die Initialisierung mithilfe einer Heuristik durchgeführt, wobei in diese initiale Lösung zwar so viele Stationen wie möglich integriert werden, jedoch im Regelfall nicht alle Stationen ohne die Verletzung von Zeitfensterrestriktionen in den Touren untergebracht werden können. Im nächsten Schritt wird für jede Ameise k eine neue Lösung nach der Funktion *Ameisenlösung_konstruieren* (Algorithmus 5) konstruiert. Für jede Station, die in dieser Tour nicht enthalten war, wird der Wert n_n dieser Station um 1 erhöht. Der Parameter n_n fungiert als Zähler, der zusätzlich in die heuristische Funktion

5 Dynamische Routenplanung

```
1 function ACS_ZEIT( $n$ )
2   Initialisiere Pheromone und Datenstrukturen auf Basis von  $n$ 
   /*  $\Psi^i$  speichert die beste Lösung des Algorithmus */
3   repeat
4     foreach Ameise  $k$  do
       /* Konstruiere eine Lösung  $\Psi^k$  */
5        $\Psi^k = \text{Ameisenlösung\_konstruieren}(k, \text{lokale Suche aktiv}, 0)$ 
       /* Falls eine der Lösungen der Ameisen besser als
          die bisher beste ist, wird diese aktualisiert */
6       if  $\exists k : \Psi^k$  gültig und  $L_{\Psi^k} < L_{\Psi^{gb}}$  then
7          $\Psi^i = \Psi^k$ 
       /* Globales Update nach Gleichung 5.13 */
8          $\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \rho / L_{\Psi^{gb}} \quad \forall (i, j) \in \Psi^{gb}$ 
9   until Abbruchkriterium erfüllt
10  return  $\Psi^i$ 
```

Algorithmus 3: Vorgehen bei der Optimierung nach der Zeit als Pseudocode

(Gleichung 5.17) integriert wird, um diese Station bei der nächsten Generierung von Touren relevanter zu gestalten. Ist die Anzahl der belieferten Stationen höher als in der bislang besten Lösung, so wird diese Lösung übernommen und sämtliche Werte von n_n zurückgesetzt. Falls die neu erzeugte Lösung gültig ist, d. h. sämtliche Stationen beliefert werden, wird diese Lösung als neue beste Lösung zurückgegeben. Zum Schluss der Methode findet ein Pheromonupdate statt, welches in diesem Falle sowohl mit der besten Lösung der Iteration durchgeführt wird, als auch mit der global besten Lösung.

In Funktion *Ameisenlösung_konstruieren* (Algorithmus 5) findet die eigentliche Konstruktion der Ameisentouren statt, auf welchen bereits aus *ACS_ZEIT* (Algorithmus 3) und *ACS_FAHRZEUGE* (Algorithmus 4) referenziert wurde. Zum Initialisieren wird die Ameise auf ein beliebig ausgewähltes dupliziertes Depot aus dem Graphen gesetzt und dieser als Startpunkt zur Lösung Ψ^k hinzugefügt. Im nächsten Schritt werden die dringendsten Stationen der Reihenfolge nach in die Lösung gebracht. Existieren mehrere überfällige Stationen, so können diese im Anschluss wie die restlichen Stationen durch den Algorithmus optimiert angeordnet werden, bleiben jedoch immer am Beginn einer Tour (vgl. Abschnitt 5.6).

```

1 function ACS_FAHRZEUGE (s)
2   Initialisiere Pheromone und Datenstrukturen auf Basis von s
3    $\Psi^{ACS-F} \leftarrow$  Initiale Lösung mit s Fahrzeugen, ermittelt durch Heuristik
      /*  $\Psi^{ACS-F}$  ist nicht zwingend vollständig */
4   foreach Ameise k do
5      $\Psi^k =$  Ameisenlösung_konstruieren( $k$ , Lokale Suche inaktiv,  $n_n$ )
6      $\forall$  Station  $j \notin \Psi^k : n_{n,j} \leftarrow n_{n,j} + 1$ 
7   if  $\exists k : \text{AnzahlKunden}(\Psi^k) > \text{AnzahlKunden}(\Psi^{ACS-F})$  then
8      $\Psi^{ACS-F} \leftarrow \Psi^k$ 
9      $\forall j : n_{n,j} \leftarrow 0$  /*  $n_{n,j}$  zurücksetzen */
10    if  $\Psi^{ACS-F}$  gültig then
11      return  $\Psi^{ACS-F}$ 
12  /* Globales Pheromonupdate für  $\Psi^{ACS-F}$  und  $\Psi^{gb}$  */
13   $\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \rho / L_{\Psi^{ACS-F}}$   $\forall (i, j) \in \Psi^{ACS-F}$ 
14   $\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \rho / L_{\Psi^{gb}}$   $\forall (i, j) \in \Psi^{gb}$ 
15  return  $\Psi^{ACS-F}$ 
    
```

Algorithmus 4: Vorgehen für die Minimierung der Anzahl an Fahrzeugen als Pseudocode

Anschließend werden die restlichen Touren eingeplant. Dazu wird zu jeder Station, welche vom aktuellen Routenzug unter Einhaltung sämtlicher Nebenbedingungen erreicht werden kann, die heuristische Distanz ermittelt und durch die Übergangsregel in Gleichung 5.8 und Gleichung 5.12 die nächste Station der Tour ausgewählt. Ist die nächste Station ein Depot, so bedeutet dies den Abschluss der aktuellen Tour. War die aktuelle Tour wirtschaftlich, d. h. gut ausgelastet, so wird versucht, auf demselben Routenzug eine weitere Tour zu generieren. War die aktuelle Tour bereits unwirtschaftlich, so ist das Einplanen neuer Touren auf diesem Routenzug nicht erfolversprechend (vgl. Abschnitt 5.6.2). Die Aufträge der unwirtschaftlichen Tour werden daher zurück in den Pool sämtlicher noch einzuplanender Aufträge N gespeichert und ein neuer Routenzug gestartet. Dazu wird der Startzeitpunkt auf die Verfügbarkeit des nächsten Routenzugs gesetzt sowie die kumulierte Beladung der aktuellen Tour auf 0 zurückgesetzt.

Sobald alle Stationen in die Touren eingeplant sind, wird durch die Einfügeheuristik versucht, ggf. übrig gebliebene Stationen nachträglich einzufügen und die Lösung

5 Dynamische Routenplanung

```

1 function new-active-ant (k, LokaleSuche,  $n_n$ )
   /* Initialisierung */
2   Setze die Ameise k auf ein beliebig ausgewähltes dupliziertes Depot i
3    $\Psi^k \leftarrow \langle i \rangle$ 
4    $t_{\text{aktuell}} \leftarrow 0, \text{Beladung}_k \leftarrow 0$ 
5
6   /* Ameise k erstellt ihre Tour, die in  $\Psi^k$  gespeichert wird. */
7   while N enthält dringende Stationen do
8      $\Psi^k \leftarrow$  dringendste Station aus N
9   while Nicht alle Stationen auf Touren eingeplant do
10     $N_{\text{möglich}} \leftarrow$  möglicheStationen(N) /* Alle Stationen, bei
11    denen die Nebenbedingungen (max. Beladung,
12    Zeitfenster, etc.) eingehalten sind */
13     $\forall j \in N$ , berechne  $\eta_{ij} = \frac{1}{t_F + t_W + t_U + n_{n,j}}$ 
14    Wähle nächsten Knoten j nach der Wahrscheinlichkeitsverteilung der
15    Gleichungen 5.8 und 5.12 aus
16     $\Psi^k \leftarrow \langle j \rangle$ 
17     $t_{\text{aktuell}} \leftarrow t_{\text{Ankunft},j}$ 
18     $\text{Beladung}_k \leftarrow \text{Beladung}_k + \text{Beladung}_j$ 
19    if j ist ein Depot then
20      if Aktuelle Tour wirtschaftlich then
21        | Starte neue Tour auf gleichem Routenzug
22      else
23        | Verschiebe Ziele von letztem Routenzug zurück in N
24        |  $t_{\text{aktuell}} \leftarrow t_{\text{Start},\text{nächsterRoutenzug}}, \text{Beladung}_k \leftarrow 0$ 
25       $\tau_{ij} \leftarrow (1 - \xi) \cdot \tau_{ij} + \xi \cdot \tau_0$  /* Lokales Pheromonupdate */
26       $i \leftarrow j$ 
27     $\Psi^k \leftarrow$  Einfügeverfahren( $\Psi^k$ )
28
29    if LokaleSuche = wahr und  $\Psi^k$  ist gültig then
30      |  $\Psi^k \leftarrow$  LokaleOptimierung( $\Psi^k$ )
31
32    return  $\Psi^k$ 

```

Algorithmus 5: Pseudocode des erweiterten Ameisenalgorithmus

5.9 Einbettung des Modells in das Produktionsumfeld

anschließend durch ein lokales Suchverfahren zu optimieren.¹⁹ Die beste Lösung Ψ^k wird dann als Rückgabewert an die aufrufende Funktion zurückgegeben.

Das hier beschriebene Verfahren berechnet zu jedem beliebigen Bestand an Aufträgen die zum aktuellen Augenblick optimale Verteilung auf den verfügbaren Bestand an Routenzügen. Dennoch kann es sinnvoll sein, nicht sämtliche Touren sofort zu starten, sondern eventuell auf zusätzliche Aufträge zu warten, um die Auslastung zu steigern. Die Einbettung in das Produktionsumfeld wird im folgenden Abschnitt beschrieben.

5.9 Einbettung des Modells in das Produktionsumfeld

Die vorhergehenden Abschnitte dienen zur Erläuterung der Erweiterungen zum Einsatz des Ameisenalgorithmus in der Intralogistik sowie zur Beschreibung der Zusammenführung all dieser Ergänzungen. Im Folgenden wird anhand einer allgemeinen Vorgehensweise dargestellt, wie die bisher entwickelten Bausteine zu einem praktisch einsatzfähigen Gesamtkonzept zusammengefügt werden. Der Gesamtzusammenhang ist in Abbildung 5.10 gezeigt. Die folgenden Prozessschritte fassen somit auch zusammen, wie die gewonnenen Erkenntnisse in eine Produktionsumgebung transferiert werden können.

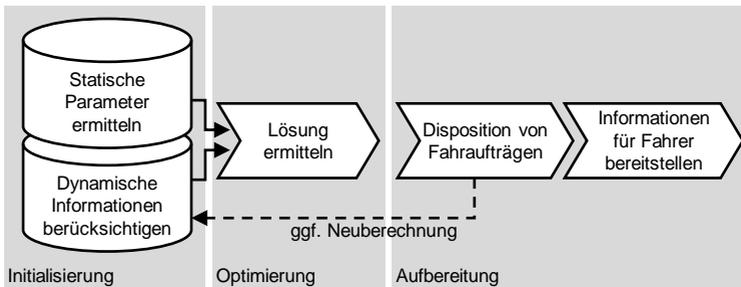


Abbildung 5.10: Schematische Darstellung der Einbettung des Optimierungsmodells in das Produktionsumfeld

¹⁹ Wie Algorithmus 5 zu entnehmen ist, kann die Funktion auch ungültige Lösungen zurückgeben. Dies ist so notwendig, da in Algorithmus 4 versucht wird, die verbliebenen Stationen in eine unvollständige Lösung zu integrieren, um eine neue Lösung mit weniger Fahrzeugen zu finden.

5.9.1 Schritt 1: Statische Parameter ermitteln

Zunächst müssen die statischen Parameter ermittelt und dem System zur Verfügung gestellt werden. Dazu gehören Informationen wie das Werkslayout inklusive sämtlicher befahrbarer Wege und den Stationen, an denen abgeliefert werden soll. Sofern eine feste Zuordnung zwischen bestimmten Materialnummern und Stationen existiert, ist diese ebenfalls zu erfassen. Schichtpläne, verfügbare Ressourcen, zeitlich befristete Änderungen am Layout oder bereits bekannte Aufträge müssen ebenfalls für das System bekannt gemacht werden.

Einen Sonderfall stellen Prozesszeiten dar. Da es sich im Regelfall um manuelle Tätigkeiten mit nicht eindeutig beschreibbaren Störeinflüssen handelt, ist methodisch keine klar definierte Zeit bestimmbar. Durch den Einsatz einer Regressionsanalyse (siehe Abschnitt 5.1.2) lassen sich stochastische Werte für diese Parameter ermitteln, welche sämtliche Störungen als Mittelwert berücksichtigen.

Statische Informationen ändern sich während des Betriebs nicht oder zumindest nicht spontan, sie müssen daher nicht automatisiert erfasst werden. Ändern sich statische Parameter, so ist eine Neuinitialisierung des Algorithmus durchzuführen.

5.9.2 Schritt 2: Dynamische Informationen berücksichtigen

Dynamische Informationen ändern sich während des Betriebs und sind dafür als elementare Grundlage permanent zu erfassen und aktuell zu halten. Derartige Informationen sind Materialabrufe von Stationen oder Prognosen zur erwarteten Rückkehr einzelner Transportmittel zum Depot. Diese Daten müssen automatisiert an das System übergeben werden, da andernfalls ein reibungsfreier, effizienter Betrieb nicht gewährleistet werden kann.

5.9.3 Schritt 3: Lösungen ermitteln

Der Schritt *Lösungen ermitteln* stellt den elementaren Baustein des Konzepts dar. Hier werden die durch Schritt 1 und 2 bereitgestellten Informationen verarbeitet und ein optimierter Tourenplan ermittelt. Dazu wird der in diesem Kapitel für

die dynamische Logistiksteuerung entwickelte Ameisenalgorithmus eingesetzt. Die Methodik in diesem Baustein ist in Abschnitt 5.8 detailliert dargestellt.

5.9.4 Schritt 4: Fahraufträge disponieren

Der entwickelte Algorithmus ermittelt zu jeder Datenlage eine möglichst optimale Lösung. Aus praktischer Sicht kann es jedoch vorteilhaft sein, noch weitere Bestelleingänge abzuwarten und somit die Transportmittel besser auszulasten. Die Entscheidung muss davon abhängig gemacht werden, ob Transportaufträge mit kritischen Zeitfenstern anstehen, oder ob noch genug Zeit zur Ablieferung bleibt. Diese Aufgabe fällt dem sogenannten Dispatcher zu, der die generierte Lösung und die aktuelle Lage des Werkes im Zusammenhang betrachtet und daraus die praktisch am besten geeignete Kombination für das Transportmittel generiert. Entscheidet der Dispatcher, mit dem Start der Tour noch zu warten, so wird beim Eingang der nächsten Bestellung oder spätestens sobald die bestehenden Aufträge dringend werden eine Neuberechnung initiiert.

5.9.5 Schritt 5: Informationen für die Fahrer bereitstellen

Die ermittelten Pläne müssen für den Fahrer dargestellt werden. Dazu können wahlweise Ausdrucke auf Papier dienen, aber auch die Anzeige auf Bildschirmen, Tablets oder Smartphones ist möglich. Dem Fahrer wird die Reihenfolge der zu beliefernden Stationen sowie deren Zeitfenster angezeigt. Kritische Zeitfenster werden dabei gesondert markiert. Besonderes Augenmerk liegt hierbei auf einer übersichtlichen Darstellung, für den Fahrer irrelevante Informationen werden daher nicht angezeigt. Ein exemplarischer Fahrplan für den Routenzugfahrer ist in Abbildung 5.11 gezeigt.

Für den Einsatz fahrerloser Transportsysteme können die Pläne digital übertragen werden. Je nach eingesetztem System kann dabei die detaillierte Wegplanung durch das Transportsystem selbst erfolgen oder muss durch das Programm vorgegeben und ebenfalls übermittelt werden.

Routenzug 2		
Station	Geplanter Ablieferzeitpunkt	Späteste Belieferung
Depot		
Station 34	09:28	09:30 
Station 2	09:29	09:42
Station 9	09:30	09:38
Station 22	09:32	09:41
Depot	9:35	

Abbildung 5.11: Exemplarische Darstellung der Tour-Informationen für den Fahrer

5.10 Zusammenfassung

Die in diesem Kapitel vorgestellten Konzepte befähigen den Ameisenalgorithmus dazu, flexibel und dynamisch im realen Produktionsumfeld agieren zu können. Dazu wurden notwendige Erweiterungen eingeführt sowie Maßnahmen zur Verbesserung der Lösungsqualität ergriffen. Um die wirtschaftlichen Einsparpotenziale tatsächlich heben zu können, ist zusätzlich eine Vorausschau notwendig, was Thema des folgenden Kapitels ist.

6 Prädiktion der notwendigen Kapazitäten

Wie in Kapitel 3 begründet, ist zur wirtschaftlichen Einsatzbarkeit einer dynamischen Logistikoptimierung eine Vorausschau über künftig notwendige Kapazitäten und daher über die zu transportierenden Güter erforderlich. Dazu wird anhand der vergangenen Materialabrufe eine Prognose für zu erwartende Abrufe erstellt und daraus die benötigte Kapazität für die nächsten Stunden abgeschätzt.¹

6.1 Methoden zur Vorhersage künftiger Kapazitätsbedarfe

Der Bedarf an Erzeugnissen wird untergliedert in Primärbedarf (der Bedarf des Marktes an Enderzeugnissen sowie eventuell verkaufbaren Ersatzteilen), Sekundärbedarf (der aus dem Primärbedarf abgeleitete Bedarf an Rohstoffen, Einzelteilen und Baugruppen zur Produktion des Primärbedarfs) sowie Tertiärbedarf (Hilfs-, Betriebsstoffe und Verschleißteile wie z. B. Werkzeug) (LENSING & SONNEMANN 1995, S. 61). Für die innerbetriebliche Logistik sind dabei hauptsächlich die Sekundärbedarfe von Relevanz, da diese zur Weiterverarbeitung aus dem Lager an die Produktionsstationen gebracht werden.

Zur Abschätzung der künftigen Bedarfe kommen mehrere Methoden in Frage, welche grundsätzlich verschiedene Vorgehensweisen verfolgen und demnach auch unterschiedliche Anforderungen für ihren Einsatz haben. Die folgende Übersicht gibt einen kurzen Überblick:

Deterministische Bedarfsermittlung: Bei der deterministischen Bedarfsermittlung wird der Sekundärbedarf vom Primärbedarf abgeleitet. Da die Steuerung der Logistik bedarfsorientiert erfolgt, ist eine äußerst genaue Erfassung von

¹ Je länger der Zeitraum ist, für den eine Prognose in die Zukunft zuverlässig gegeben werden kann, desto höher ist der wirtschaftliche Vorteil daraus. Bereits ab einigen Stunden lässt sich jedoch Personal für anderweitige Aufgaben einplanen und sich somit die Kosten optimieren.

6 Prädiktion der notwendigen Kapazitäten

Stücklisten, Kundenbestellungen und dem geplanten Fertigungsprogramm erforderlich (ARNOLDS ET AL. 2017, S. 41). Die deterministische Bedarfsermittlung ist nur anwendbar, wenn der Primärbedarf exakt bestimmt ist und klar definierte Materialmengen an Einzelteilen in den Ladungsträgern in die Produktion geliefert werden.

Stochastische Bedarfsermittlung: Die stochastische Bedarfsermittlung kann angewendet werden, wenn deterministische Methoden nicht oder nur unwirtschaftlich anwendbar sind bzw. wenn mit unvorhergesehenem Ausschuss zu rechnen ist. Ihr liegt die sogenannte *Zeitstabilitätshypothese* zugrunde, d. h. es wird angenommen, dass die Ursachen für den Verbrauch der Vergangenheit auch für die zukünftige Periode gelten, selbst wenn diese nicht im Detail bekannt sind (BLOHM ET AL. 2016, S. 311). Zur Ermittlung der künftigen Bedarfe wird demnach versucht, den bisherigen Verbrauch in die Zukunft zu extrapolieren. Dafür können verschiedene stochastische Methoden zum Einsatz kommen, im einfachsten Fall ein konstanter Verbrauch (WANNENWETSCH 2014, S. 92). Der Vorteil dieser Form der Bedarfsermittlung ist, dass weder die Pflege der Stücklisten, noch Informationen zur exakten Teileanzahl in den Behältern notwendig sind. Prinzipbedingt erreichen stochastische Verfahren nicht die Vorhersagequalität deterministischer Verfahren, dafür ist ihr potenzielles Einsatzgebiet deutlich größer. Nachteilig ist die Anfälligkeit für Fehlprognosen bei Verletzung der Zeitstabilitätshypothese (z. B. bei Änderungen am Produktionssystem), wodurch sowohl hohe Bestände als auch die Knappheit von Bauteilen entstehen können.

Subjektive Bedarfsschätzung: Die subjektive Bedarfsschätzung wird angewandt, wenn keine ausreichende Datenbasis aus der Vergangenheit vorhanden ist (z. B. neues Produkt) oder ein vollkommen unregelmäßiger Bedarfsverlauf vorliegt (z. B. Einzel- und Sonderanfertigungen). Der Bedarf wird durch einen Disponenten subjektiv abgeschätzt. Hierbei kann zwischen zwei verschiedenen Methoden gewählt werden: Zum einen die Analogieschätzung, bei welcher der Bedarf anhand vergleichbarer Materialien bzw. Produkte abgeschätzt wird sowie zum anderen die Intuitivschätzung, bei der Expertenmeinungen zusammengetragen und konsolidiert werden. Die Fehleinschätzung ist bei diesem Vorgehen häufig sehr groß, der Einsatz sollte daher nur erfolgen, wenn kein anderes Prognoseverfahren möglich ist. (WANNENWETSCH 2014, S. 97 f.)

6.2 Auswahl der Methode zur Bedarfsprognose

Damit das Konzept nicht durch die entstehende Schnittstellenkomplexität unwirtschaftlich wird, ist Grundanforderung an die Prädiktion im Rahmen der fabrikinernen Logistik, dass nur so wenige Eingangsdaten wie möglich genutzt werden sollen. Zwar ist es prinzipiell denkbar, über jedem Behälter eine Messeinrichtung zur Ermittlung des aktuellen Bestands einzurichten, allerdings wäre eine derartige Erweiterung zum einen mit erheblichen Investitionen verbunden, zum anderen wäre sie eine Behinderung bei künftigen Erweiterungen oder Umbauten im Werk.

Das Produktionsprogramm des Werks steht zwar i. d. R. zur Verfügung, allerdings sind zum einen Stücklisten in vielen Werken nicht in der notwendigen Granularität vorhanden², zum anderen haben Behälter gleichen Materials unter Umständen unterschiedliche Füllgrade. Diese sind nicht vorhersehbar und ergeben sich durch zufällige Entnahmen für die Qualitätssicherung oder ähnliche Prozesse. Eine *deterministische Bedarfsermittlung* ist daher nicht möglich (vgl. Abschnitt 6.1).

Bei der *subjektiven Bedarfsschätzung*, bei welcher durch Experten der künftige Bedarf abgeschätzt wird, ist der Aufwand für den geplanten Einsatzzweck zu hoch. Gleichzeitig haben Experten durch die weit verteilten Bestände innerhalb der Fabrik keinen Überblick über den aktuellen Stand und können so ggf. häufig keine Aussage treffen.

Da der Verbrauch in Großserienwerken relativ konstant ist, sind die Anforderungen für eine *stochastische Bedarfsermittlung* praktisch gegeben. Diese sind dann erfüllt, wenn ein Zusammenhang zwischen dem Verbrauch der Vergangenheit und dem Bedarf in der Zukunft besteht. Abweichend vom allgemeinen Fall kann auch für einzelne große Bauteile eine deterministische Prognose und damit eine exakte Vorausplanung möglich sein. Stehen derart präzise Daten zur Verfügung, so sind diese Daten zu bevorzugen und entsprechend im weiteren Verlauf für die Prognose zu verwenden.

² Dies gilt vor allem für Schüttgut, welches in vielen Konstruktionszeichnungen nicht detailliert erfasst wird.

6.3 Vorhersage künftiger Abrufe

Abbildung 6.1 zeigt das grundsätzliche Vorgehen. Die zu produzierenden Varianten müssen dabei nicht im Detail bekannt sein. Die Information darüber, mit welcher Ausbringungsmenge im Werk produziert wird, ist für die meisten Fälle ausreichend, um die Logistikbedarfe abzuschätzen.

Aus den Vergangenheitsdaten lassen sich sowohl die gemittelten zeitlichen Abstände zwischen zwei Bestellungen (Erwartungswert), als auch der zu erwartende Zeitkorridor für die Bestellung (Standardabweichung), ermitteln (Abbildung 6.1 oben rechts). Da in den meisten Werken ein möglichst gleichmäßiger Produktmix angestrebt wird (vgl. ROTHER & SHOOK 2000, S. 50), sind zwar Schwankungen in der Bestellzeit am Lager zu erwarten – bei Teilen, die in hoher Stückzahl verbaut werden, sind diese jedoch verhältnismäßig gering. Auf Basis dieser Information lässt sich der Zeitpunkt der nächsten Bestellung mit hoher Wahrscheinlichkeit ermitteln. Daraus lässt sich wiederum auch der übernächste Zeitpunkt errechnen, wobei sich die zu erwartenden Abweichungen addieren und zu einem ungenaueren Ergebnis führen als bei dem ersten prognostizierten Bestellzeitpunkt. Dieser Mechanismus führt dazu, dass für die nahe Zukunft eine gute Prognose erzielt werden kann, für weiter entfernte Bestellungen diese allerdings immer ungenauer wird.

Daraus lässt sich ein Gesamtbedarf prognostizieren, welcher unschärfer wird, je weiter in die Zukunft geblickt wird (Abbildung 6.1 unten links). Aus diesem zu erwartenden Bedarf lässt sich wiederum abschätzen, welche Anzahl an Fahrzeugen zur Bewältigung des Transportaufkommens notwendig ist.

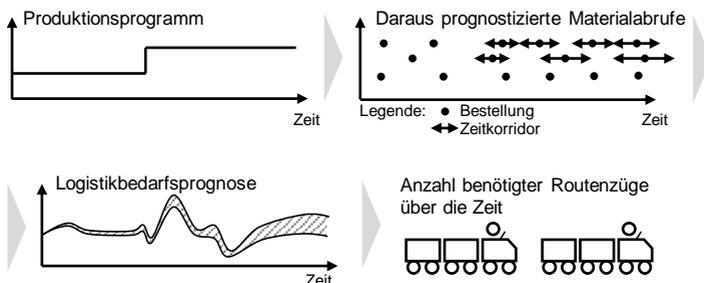


Abbildung 6.1: Konzept zur Prädiktion von Materialabrufen

Wie oben erläutert, kann der genaue Zeitpunkt der nächsten Bestellung nicht präzise vorhergesehen werden, es ist nur möglich, den richtigen Zeitpunkt mit einer gewissen Wahrscheinlichkeit zu prognostizieren. Da für jede Bestellung somit viele verschiedene Bestellzeitpunkte realistisch sind, ist es nicht möglich, sämtliche Kombinationen aus verfrühter oder verspäteter Bestellung für alle zu erwartenden Bestellungen zu simulieren, da die Rechenzeit hierfür ins Unermessliche steigen würde. Ohnehin wäre eine derart präzise Simulation von unscharfen Daten zu diesem Zeitpunkt noch nicht zielführend, da lediglich eine ungefähre Prognose gefordert ist. Im Folgenden wird daher ein Ansatz gewählt, welcher die durchschnittlichen Kapazitätsbedarfe annähert.

6.4 Wahrscheinlichkeitstheoretische Betrachtung

Der zeitliche Abstand zwischen zwei Bestellungen lässt sich, wie in Abschnitt 6.3 beschrieben, im Regelfall als Normalverteilung $\mathcal{N}(\Delta t, \sigma^2)$ mit dem Erwartungswert Δt für den Abstand zwischen zwei Bestellungen und der Standardabweichung σ für die Verteilung der Bestellung um den Erwartungswert darstellen. Ob die Bestellungen im Einzelfall normalverteilt sind muss zunächst nachgewiesen werden³. Dafür kann entweder eine optische Bewertung der geplotteten Abweichungen von der erwarteten Bestellzeit erfolgen oder alternativ mathematisch-statistische Tests zur Überprüfung der Verteilungsfunktion wie beispielsweise der Kolmogorow-Smirnow-Test⁴ (RÜSCHENDORF 2014, S. 13).

Der Erwartungswert errechnet sich allgemein mit

$$\Delta t = \frac{\sum_{i=1}^{i=n-1} t_{B,i+1} - t_{B,i}}{n - 1} \quad (6.1)$$

³ Bei aus einer realen Produktion aufgenommenen Vergangenheitsdaten ist zusätzlich ein manuelles Prüfen auf Fehler im Datensatz durchzuführen. So können in fehlerhaften Daten beispielsweise einzelne Bestellungen komplett fehlen, was eine Verdopplung der Zeit zwischen zwei Bestellungen bedeutet. Andererseits können durch ungeschickte Absprachen Bestellungen mehrfach im Lager eingehen. Diese werden dann im Regelfall nur einmal ausgeliefert, in der Software sind die Bestellungen jedoch vermerkt und können bei der späteren Prüfung auf Normalverteilung zu Problemen führen.

⁴ Der Kolmogorow-Smirnow-Test stellt dabei nur eine Möglichkeit von vielen dar. Er ist jedoch für viele Anwendungsfälle zu bevorzugen, da er auch für kleine Stichproben geeignet ist. Alternative Verfahren, wie beispielsweise der für hinreichend große Stichproben geeignete χ^2 -Test, können bei Sicherstellung der Eignung ebenso gewählt werden.

6 Prädiktion der notwendigen Kapazitäten

und die Standardabweichung der Grundgesamtheit bei Stichproben mit

$$\sigma = \sqrt{\frac{1}{1-n} \sum_{i=1}^n (t_{B,i} - \Delta t)^2} \quad (6.2)$$

mit

- σ Standardabweichung
- n Stichprobenumfang
- $t_{B,i}$ Bestellzeitpunkt des i -ten Elements
- Δt Erwartungswert für den zeitlichen Abstand zweier Bestellungen

Die n Bestellungen sind die aus der Vergangenheit, demnach sind für sie die Bestellzeitpunkte bekannt. Der erste Bestellzeitpunkt $t_{B,n+1}$ der Prognose lässt sich dann durch den Zusammenhang

$$t_{B,n+1} = t_{B,n} + \mathcal{N}(\Delta t, \sigma^2) \quad (6.3)$$

beschreiben. Das Quadrat der Standardabweichung σ^2 wird als *Varianz* bezeichnet. Das Ergebnis ist eine um den Erwartungswert gestreute Wahrscheinlichkeitsverteilung für den Bestellzeitpunkt. Diese Verteilung wird zur Ermittlung des übernächsten Zeitpunkts eingesetzt:

$$\begin{aligned} t_{B,n+2} &= t_{B,n+1} + \mathcal{N}(\Delta t, \sigma^2) \\ &= t_{B,n} + \mathcal{N}(\Delta t, \sigma^2) + \mathcal{N}(\Delta t, \sigma^2) \\ &= t_{B,n} + \mathcal{N}(2\Delta t, 2\sigma^2) \end{aligned} \quad (6.4)$$

Der Zusammenhang lässt sich für die Prognose $t_{B,p}$ (mit $p > n$) für die p -te Bestellung verallgemeinern zu

$$t_{B,p} = t_{B,n} + \mathcal{N}(p\Delta t, p\sigma^2). \quad (6.5)$$

Das Vorgehen ist in Abbildung 6.2 für die ersten beiden Bestellungen nach dem Startzeitpunkt t_0 der Prognose skizziert.

Anhand der Formel lassen sich prinzipiell beliebig weit in der Zukunft liegende Bestellungen prognostizieren. Allerdings wird durch die wachsende Standardabweichung die Streuung für die einzelnen Materialabrufe so groß, dass auf dieser Basis keine sinnvoll belastbare Aussage mehr getroffen werden kann.

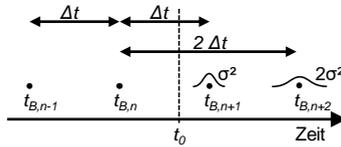


Abbildung 6.2: Visualisierung der Fortpflanzung der Wahrscheinlichkeiten

Für jede auszuliefernde Bestellung wird nun eine durchschnittliche Zeit \bar{d}_a zur Auslieferung ermittelt, welche die Rechteckfunktion in Abbildung 6.3 darstellt. Ebenfalls in diesem Faktor enthalten ist der Wirkungsgrad, welcher berücksichtigt, dass ein Routenzug ggf. hin und wieder kurz warten muss. Für den Zeitraum \bar{d}_a ist damit die Kapazität für einen Ladungsträger belegt (siehe Funktion $h(t)$ in Abbildung 6.3). Die Formel für die Rechteckfunktion zur Beschreibung des Kapazitätsbedarfs $h(t)$ wird durch

$$h(t) = \begin{cases} 0 & t < 0 \\ 1 & 0 \leq t \leq \bar{d}_a \\ 0 & t > \bar{d}_a \end{cases} \quad (6.6)$$

beschrieben.

Die gesamthaft vorzuhaltende Kapazität für die Auslieferung hängt damit sowohl vom Zeitpunkt der Bestellung mit der normalverteilten Streuung $f(t)$ ab, als auch von der benötigten Zeit, welche wie oben beschrieben mit der Rechteckfunktion $h(t)$ dargestellt werden kann. Die Verknüpfung dieser beiden Verteilungen wird wie in Abbildung 6.3 gezeigt über eine Faltung realisiert. Mit der Standardnormalverteilung $\Phi(\zeta)$ ergibt sich somit folgender Zusammenhang:

$$(f(t) * g(t))(\zeta) = \Phi\left(\frac{\zeta - \mu}{\sigma}\right) - \Phi\left(\frac{\zeta - \bar{d}_a - \mu}{\sigma}\right) \quad (6.7)$$

Für den Erwartungswert μ wird darin der zu erwartende Bestellzeitpunkt eingesetzt, für die Standardabweichung σ die errechnete Streuung.

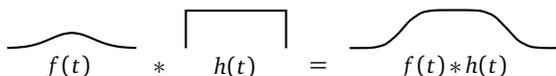


Abbildung 6.3: Faltung der Dichtefunktion $f(t)$ mit der Rechteckfunktion $h(t)$

6 Prädiktion der notwendigen Kapazitäten

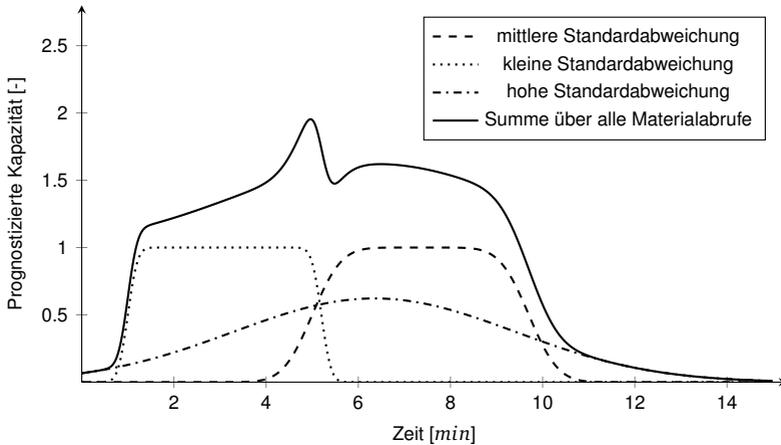


Abbildung 6.4: Vorgehen beim Aufsummieren der einzelnen erwarteten Kapazitäten

Wird die Standardabweichung der Normalverteilung in Relation zur Breite des Rechtecks größer, so wird die annähernd ebene Fläche der gefalteten Funktion in Abbildung 6.3 immer kürzer, bis eine Funktion übrig bleibt, welche optisch kaum noch von einer Normalverteilung zu unterscheiden ist.

Diese gefaltete Funktion wird nun mit den entsprechenden Parametern der jeweiligen prognostizierten Materialabrufe mit der zugehörigen Standardabweichung und dem Erwartungswert geplottet. Die Summe all dieser Funktionswerte ergibt die kumulierte Bedarfsprognose. Das Vorgehen ist exemplarisch für drei Bestellungen mit unterschiedlichen Standardabweichungen in Abbildung 6.4 dargestellt.

6.5 Ableitung von Maßnahmen aus der Kapazitätsprognose

Aus der mathematischen Analyse der Vergangenheitsdaten und dem Aufbau des Modells zur Prognose für die nähere Zukunft werden in den folgenden Abschnitten die Logiken für das Ableiten der tatsächlichen Bedarfe an Routenzügen hergeleitet.

Abbildung 6.5 zeigt dabei zum einen das bisher erläuterte Vorgehen, als auch die Möglichkeiten zur Reaktion auf die Prognoseergebnisse. Zum einen kann aus der kumulierten Bedarfsprognose aus Abbildung 6.4 der Bedarf an Fahrzeugen abgelesen

6.5 Ableitung von Maßnahmen aus der Kapazitätsprognose

und bereitgestellt werden. Dieses Vorgehen ist in Abschnitt 6.5.1 beschrieben. Zum anderen kann Kapazitätsspitzen auch gezielt entgegengewirkt werden und damit bereits in der Simulation zur Erstellung der Tourenpläne auf den Kapazitätsbedarf Einfluss genommen werden, was in Abschnitt 6.5.2 behandelt wird.

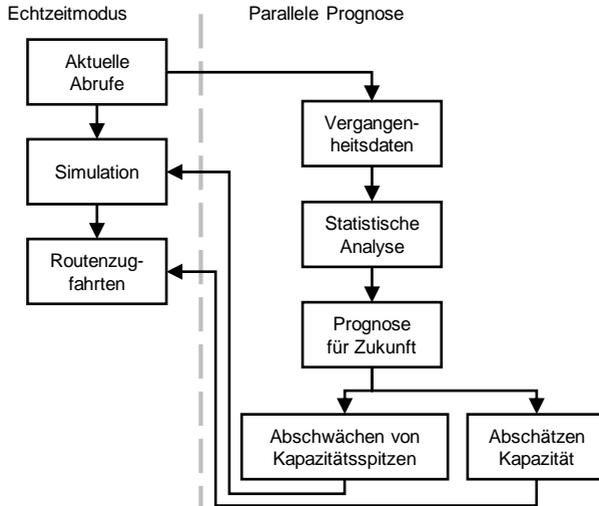


Abbildung 6.5: Vorgehen und Integration der Prognose in den Standardprozess

6.5.1 Ableiten der zu erwartenden Kapazitätsbedarfe

Aus der Information, wie viele Behälter auf einem Routenzug transportiert werden können, lässt sich aus dem erstellten Graph in Abschnitt 6.4 durch Zusammenfassen zu vollen Routenzügen die benötigte Kapazität abschätzen, wie dies in Abbildung 6.6 durchgeführt wurde. Zuzüglich eines gewählten Sicherheitsfaktors kann nun die benötigte Kapazität abgelesen werden. Dazu werden die notwendigen Kapazitäten jeweils auf voll beladene Routenzüge aufgerundet. Zusätzlich werden kleine Abstände, in denen ein Routenzug warten muss, überbrückt, da ein Routenzug für kurze Zeiträume (wie bspw. 5 Minuten) nicht anderweitig eingesetzt werden kann.⁵

⁵ In Abbildung 6.6 finden sich derartige Konstellationen beispielsweise mehrfach im Bereich zwischen Stunde 2 und 3. Dies führt dazu, dass in diesem Zeitraum die Kapazität von drei Routenzügen eingeplant wird, obwohl für kurze Zeiträume nur zwei erforderlich wären.

6 Prädiktion der notwendigen Kapazitäten

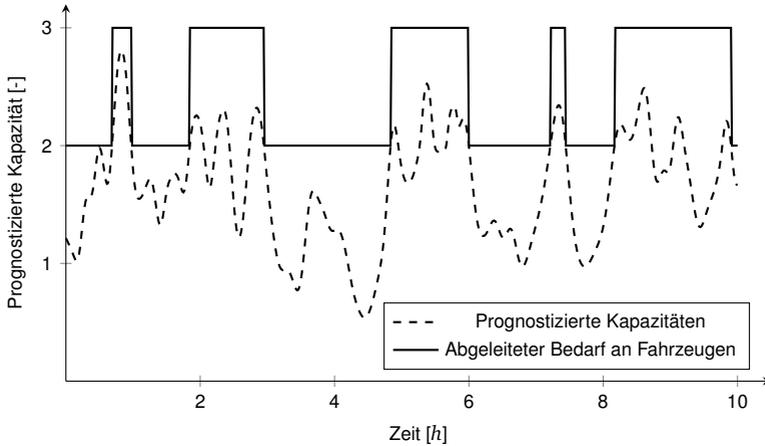


Abbildung 6.6: Ableitung der Bedarfe an Routenzügen aus der Kapazitätsprognose

Mithilfe des Ergebnisses lassen sich die ungenutzten Kapazitäten bereits vorab erkennen und darauf aufbauend anderweitig einplanen. Wird ein Routenzug über einen längeren Zeitraum nicht genutzt, so können Fahrer und Fahrzeug anderweitig eingesetzt werden und somit anstelle von ungenützten Wartezeiten einen sinnvollen Beitrag zur Wertschöpfung leisten. Die Prognose ist somit in der Lage, einen Beitrag zur wirtschaftlichen Einsetzbarkeit des Optimierungsalgorithmus zu leisten.

6.5.2 Abmindern von Kapazitätsspitzen

Ist nur über kurze Zeiträume ein zusätzlicher Routenzug erforderlich und sind im sonstigen Zeitraum die anderen Routenzüge nicht vollständig ausgelastet, so besteht eine effiziente Alternative zur Einplanung eines zusätzlichen Fahrzeugs, um sämtliche Materialien ohne zusätzliche Ressourcen rechtzeitig auszuliefern.

Während der Kapazitätsspitze werden von besonders vielen Stationen gleichzeitig neue Materialien angefragt. Einige davon werden mit hoher Wahrscheinlichkeit in diesem Zeitraum anfallen (kurze Bestellzyklen, kleine Standardabweichung), bei anderen ist dieser Zeitraum nicht so genau vorhersehbar (lange Bestellzyklen, große Standardabweichung). Zusätzlich ist bei langen Bestellzyklen der verbleibende Bestand im Behälter bereits nahezu aufgebraucht, wodurch eine zu frühe Anlieferung

zu keiner übermäßig verlängerten Umladezeit führt. Es kann daher versucht werden, diese Bestellungen vorzuverlegen und damit auch deren Auslieferung noch vor dem eigentlichen Bestellzeitpunkt durchzuführen. Dies mildert die Kapazitätsspitze ab und ermöglicht somit die Bereitstellung sämtlicher Materialien ohne den Einsatz eines zusätzlichen Fahrzeugs. Das Vorgehen ist schematisch in Abbildung 6.7 dargestellt.

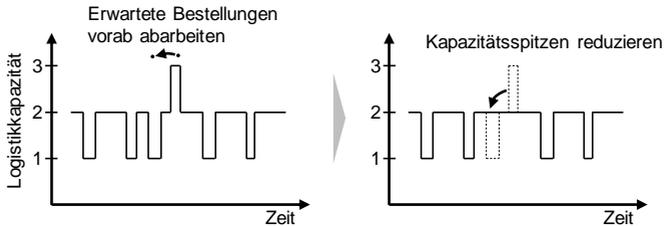


Abbildung 6.7: Prinzip der Vorverlagerung von Bestellungen zur Abminderung von Kapazitätsspitzen

Die Abminderung der Kapazitätsspitzen führt zu einer gleichmäßigeren Auslastung der Routenzüge über die Zeit. Im Idealfall lassen sich Routenzüge über längere Zeiträume vollständig einsparen, was ein signifikantes Einsparpotenzial darstellt.

6.6 Zusammenfassung

Das in diesem Kapitel vorgestellte Vorgehen ermöglicht eine Vorschau auf in der Zukunft liegende, erforderliche Kapazitäten. Damit wird es möglich, proaktiv mit Kapazitätsengpässen zu agieren und somit die wirtschaftlichen Einsparpotenziale bei Kapazitätsüberschüssen zu heben. Das Gesamtkonzept wird in Abschnitt 7.3 anhand realer Daten evaluiert.

7 Verifikation und praktische Validierung

Im folgenden Kapitel wird der Nachweis der Leistungsfähigkeit des entwickelten Ansatzes geführt. Zunächst wird im Rahmen der Verifikation sichergestellt, dass das Konzept und die mathematische Modellierung die aus den praktischen Erfordernissen abgeleiteten theoretischen Anforderungen (siehe Kapitel 3) erfüllen. Anschließend wird anhand der Umsetzung validiert, ob die ursprünglichen Anforderungen des Anwendungsfalls (vgl. Kapitel 1) durch das gewählte Vorgehen hinreichend gut umgesetzt werden. Das strukturelle Vorgehen der Absicherung ist in Abbildung 7.1 dargestellt.

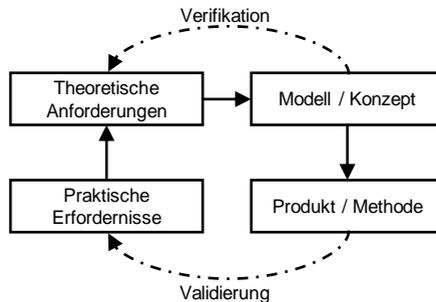


Abbildung 7.1: Absicherung der Korrektheit durch Verifikation und Validierung

Zur Verifikation und Validierung in Produktion und Logistik können diverse Techniken eingesetzt werden. Im Rahmen dieser Arbeit wurden dazu die folgenden Verfahren nach RABE (2008, S. 93 ff.) ausgewählt:

Test von Teilmodellen: Systematisches Überprüfen einzelner Bestandteile des Konzepts (RABE 2008, S. 106).

Monitoring: Grafische Darstellung der Ergebnisse und Prüfen auf Konsistenz (RABE 2008, S. 101 f.).

Ereignisvaliditätstest: Bewertung des Auftretens von Ereignissen, welche aufgrund der Eingangsdaten zu erwarten sind (RABE 2008, S. 98 f.).

Vergleich mit aufgezeichneten Daten: Überprüfung auf Korrektheit der Berechnungen durch Vergleich mit real aufgezeichneten Daten (RABE 2008, S. 111).

Validierung im Dialog: Durchsprechen des Konzepts und kritische Überprüfung der Simulationsergebnisse mit Experten (RABE 2008, S. 109).

Begutachtung: Qualitative Prüfung des Erfüllungsgrads durch den Anwender (RABE 2008, S. 97).

In Abschnitt 7.1 wird die Entwicklung des Softwaretools beschrieben, welches für viele der Absicherungsschritte verwendet wird. Das weitere Vorgehen wird gemäß dem *Test von Teilmodellen* in Einzelschritten an realistischen Daten durchgeführt. In Abschnitt 7.2 wird dazu mithilfe dynamischer Benchmarks das in den vorhergehenden Kapiteln entwickelte Modell evaluiert. Hierbei wurden die Verfahren *Monitoring*, der *Ereignisvaliditätstest*, die *Begutachtung* sowie die *Validierung im Dialog* durchgeführt. Des Weiteren werden in Abschnitt 7.3 die Prognosefähigkeiten der in Kapitel 6 dargestellten Methode untersucht.

Die praktische Validierung der in dieser Arbeit entwickelten Bausteine anhand einer praktischen Implementierung und der Pilotierung im Werk erfolgt in Abschnitt 7.4. Hierbei wurde auf Basis der theoretischen Validierung zusätzlich der *Vergleich mit aufgezeichneten Daten* durchgeführt. Die realen Daten wurden aus Gründen der Geheimhaltung mit einem Skalierungsfaktor versehen. Dabei wurde darauf geachtet, dass keine Verzerrung der auftretenden Effekte auftritt.

7.1 Entwicklung des Softwaretools

Zur Erprobung des entwickelten Konzepts wurde ein Softwaretool in C#¹ entwickelt, welches sowohl für die theoretischen Experimente als auch für die praktische Pilotierung verwendet wurde.

¹ Die Programmiersprache C# basiert auf dem .NET-Framework der Firma Microsoft. Die Wahl der Programmiersprache fiel aufgrund projektspezifischer Rahmenbedingungen (u. a. Schnittstellen, Kompatibilität, Möglichkeit zur Ausführung in administrierten IT-Umgebungen). Sämtliche im Rahmen dieser Arbeit erarbeiteten Erkenntnisse können auch in anderen Programmiersprachen realisiert werden.

7.1.1 Funktionalität des Programms

Das im Rahmen dieser Arbeit entwickelte Programm simuliert und prognostiziert die Abläufe in der innerbetrieblichen Logistik. Dazu sind sämtliche Forschungsergebnisse aus den vorangegangenen Kapiteln in die Software integriert. Für den Einsatz stehen darin verschiedene Modi zur Verfügung:

Statischer Modus: Im statischen Modus können Benchmarkdateien geladen und simuliert werden. Charakteristisches Merkmal dieses Modus ist, dass bereits zu Beginn der Berechnungen sämtliche Abrufe bekannt sind und dafür die optimale Lösung gesucht wird. Dieser Modus ist vor allem zum Vergleich der mathematischen Lösungsqualität hilfreich. Mithilfe dieses Modus können in der Wissenschaft verbreitete Probleminstanzen (vgl. z. B. SOLOMON 1987) simuliert werden. Dies ist einerseits notwendig, um die prinzipielle Funktionsfähigkeit des Algorithmus zu zeigen, vor allem aber, um Parameterstudien als Grundlage für die praktische Anwendung durchführen zu können.

Echtzeitmodus: Im Echtzeitmodus können einzelne Materialabrufe inklusive deren Zeitfenster live in die Software eingegeben werden, so wie diese an den Stationen auftreten. Das Programm ermittelt automatisch die zur aktuellen Datenlage optimalen Routen. Für den Fahrer wird ein Ausdruck erstellt, der zum Beladen und zur Navigation während der Tour dient. Dieser Programmmodus eignet sich damit zur Pilotierung des Steuerungskonzepts im Werk.

Simulationsmodus: Im Simulationsmodus können die Informationen über Materialabrufe aus einer vorab erstellten Datei geladen werden. Somit können Vergangenheitsdaten verwendet werden, um verschiedene Szenarien oder Algorithmusvarianten zu erproben und zu vergleichen. Zusätzlich läuft die Simulation schneller als in Realität, um die Wartezeiten gering zu halten. Der Simulationsmodus eignet sich ferner auch zur Analyse der Auswirkungen bei der Neuordnung von Stationen.

Der Programmausschnitt in Abbildung 7.2 zeigt das Layout des simulierten Werks. Mit Kreisen markiert sind sämtliche Stationen mit potenziellen Materialabrufen, mit den Fähnchen werden die derzeit für die Simulation berücksichtigten Materialabrufe gekennzeichnet. Im aktuellen Zustand sind zwei Routenzüge unterwegs, die mit unterschiedlich farbigen Linien markiert sind. Die Verbindungslinien sind der Übersicht



Abbildung 7.2: Ausschnitt aus dem Simulationsprogramm mit dem in Abschnitt 7.4 beschriebenen Anwendungsbeispiel

halber diagonal eingezeichnet, gerechnet wird jedoch mit der tatsächlichen Distanz, welche sich an den rechtwinklig verlaufenden Wegen des Werks orientiert.

Zur Auswertung ist die Auslastung der Routenzüge über die Zeit in einem Gantt-Diagramm aufgezeichnet (siehe Abbildung 7.3). Somit können der Zeitanteil der Einsätze einzelner Routenzüge abgeschätzt sowie Aussagen über die zeitliche Gleichmäßigkeit der Auslastung getroffen werden. Zusätzlich bietet das Programm die Möglichkeit, Detailinformationen zu jeder Tour und jedem Materialabruf zu anzuzeigen und so die Touren auf deren Konsistenz zu prüfen.

In der Software können außerdem sämtliche Parameter für die Optimierung und die für den Anwendungsfall relevanten Zeitanteile individuell eingestellt werden. Somit kann eine allgemein einsatzfähige Software zur Validierung verwendet werden, welche auch für andere künftige potenzielle Szenarien ohne Modifikation einsatzfähig ist. Durch die Anpassbarkeit sind auch Studien möglich, wie sich Veränderungen (z. B. andere Routenzüge oder die örtliche Verschiebung des Lagers) auf das vorliegende Logistiksystem auswirken.

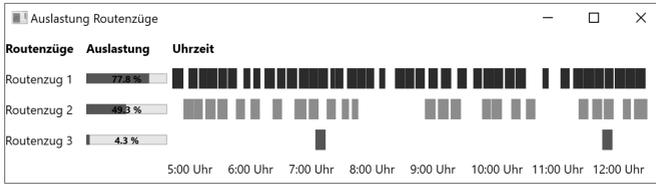


Abbildung 7.3: Gantt-Chart zur Bewertung der Auslastung der einzelnen Routenzüge

7.1.2 Methodisches Vorgehen bei der Entwicklung

Zur effizienten Entwicklung des Programms wurde nach dem Vorgehensmodell *Scrum* gearbeitet, welches einen Ansatz für agiles Projektmanagement darstellt (SUTHERLAND 2015; RUBIN 2014, S. 33 f.). Scrum unterstützt durch die iterative Vorgehensweise den methodischen Rahmen aus Abschnitt 1.6 und erlaubt mithilfe der lauffähigen Zwischenstände ein regelmäßiges Überprüfen der Projektergebnisse. Einen Überblick über den Ablauf eines mit Scrum gesteuerten Projekts zeigt Abbildung 7.4.

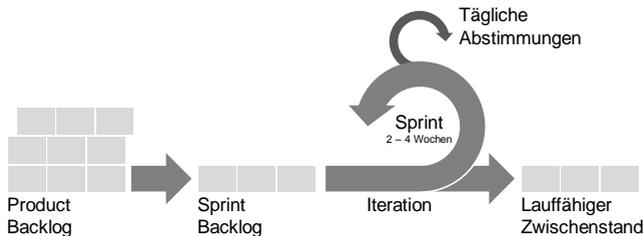


Abbildung 7.4: Ablauf eines Scrum-Projekts (in Anlehnung an OLTERSDORFF 2016, S. 62)

Dazu wurde zunächst das *Product Backlog*² definiert, welches eine priorisierte Liste der Funktionalitäten sowie der erforderlichen Fähigkeiten des erfolgreichen Produkts enthält. Die zu erreichenden Ziele wurden dabei gemeinsam im Team erarbeitet, um die Produktvision so gut wie möglich realisieren zu können.

² Als *Product Backlog* wird die Anforderungsliste an ein Produkt definiert. Die einzelnen *Product Backlog Items* enthalten dabei auch eine Schätzung des Aufwands, werden im Gegensatz zum klassischen Projektmanagement jedoch noch nicht zeitlich eingeordnet. (GLOGER 2011, S. 11)

Die Umsetzung der Ziele aus dem Product Backlog erfolgte in *Sprints*³, wobei jeder einzelne Sprint mit der Planung begann, welche der Aufgaben aus dem Product Backlog in den *Sprint Backlog* übernommen werden sollen und bis zum Abschluss dieses Sprints abgearbeitet sein müssen. Ziel eines jeden Sprints war ein lauffähiges Programm. Zum Abschluss des Sprints wurden der Fortschritt kontrolliert sowie ggf. Änderungen am Product Backlog vorgenommen, soweit dies durch neue Erkenntnisse aufgrund des Projektfortschritts als angebracht erschien. Die Ergebnisse eines jeden Sprints wurden mit dem Industriepartner diskutiert und evaluiert (siehe Abschnitt 7.4).

7.2 Simulation dynamischer Testfälle

Da der Fokus des entwickelten Ansatzes auf dynamischen Umgebungen liegt, ist die Aussagekraft statischer Benchmarks nicht ausreichend (vgl. Abschnitt 4.3). Die Leistungsfähigkeit in realitätsnahen Szenarien muss daher mit dynamischen, theoretischen Testfällen gezeigt werden, welche nicht vorgefertigt existieren und daher anwendungsfallspezifisch erstellt werden müssen. Im Folgenden wird daher beschrieben, wie diese Benchmarks erstellt und damit die dynamische Funktionalität nachgewiesen werden kann.

7.2.1 Entwicklung dynamischer Benchmarks

Dynamische Testfälle für das vorliegende Szenario müssen sämtliche Besonderheiten der Anforderungen aus Abschnitt 3.2 erfüllen. Konkret sind dies beim Algorithmus die Berücksichtigung variabler Umladezeiten, die zyklische Wiederverwendung von Fahrzeugen sowie die Behandlung dringender Bestellungen. Damit wird der konkrete Anwendungsfall der fabrikinternen Logistik abgedeckt.

Dazu wurden mitprotokollierte Daten aus einem praktischen Anwendungsfall verwendet und allen Auswertungen zugrundegelegt. Die Daten enthalten den Bestellzeitpunkt, die Station, die Bestellmenge sowie die Positionen sämtlicher Stationen.

³ Als *Sprint* werden bei Scrum die zwei- bis vierwöchigen Arbeitsintervalle bezeichnet, in denen ein zuvor definierter Umfang aus dem Product Backlog abgearbeitet wird. Ziel eines Sprints ist immer, zum Abschluss ein in sich stimmiges und soweit möglich lauffähiges Ergebnis zu haben. (VERHEYEN 2017, S. 59)

Ebenso wurden die Entfernungen, die Fahrt- und die Servicezeiten einheitlich festgelegt. Der Benchmark entspricht somit exakt dem Einsatz in der realen Produktionsumgebung und ist somit zum Testen sämtlicher Besonderheiten geeignet.

Zur Simulation dieser Szenarien wurde in das Softwaretool (siehe Abschnitt 7.1) ein Simulationsmodus integriert, in welchem reale Szenarien hinterlegt und simuliert werden können. Dabei werden Bestellungen im Lager geordert, die innerhalb einer definierten Vorlaufzeit ausgeliefert sein müssen.

7.2.2 Beurteilung der Leistungsfähigkeit

Für die in Abschnitt 7.2.1 entwickelten Simulationsläufe existieren keine Referenzwerte zum Vergleich. Zur Bewertung wurde daher auf die ausgewählten Vorgehensweisen *Begutachtung*, *Validierung im Dialog*, *Monitoring* sowie den *Ereignisvaliditätstest* aus Kapitel 7 zurückgegriffen. Der *Vergleich mit aufgezeichneten Daten* erfährt in Abschnitt 7.4 gesondert Beachtung.

Im Rahmen der *Begutachtung* (vgl. RABE 2008, S. 97) wurde das Konzept in regelmäßigen Abständen mit potenziellen Anwendern diskutiert⁴. Hierdurch wurde sichergestellt, dass die Modellierung in Übereinstimmung mit den Forschungszielen und den zu berücksichtigenden Nebenbedingungen erfolgt. Als Kriterien wurden hierbei die Eignung der Ziel- und Systembeschreibungen, die Angemessenheit der getroffenen Annahmen, die Angemessenheit und Effizienz der verwendeten Modellierungstechniken, der Grad der Übereinstimmung des Modells mit der Realität, die Nachvollziehbarkeit des Modells sowie dessen Vollständigkeit geprüft. Sämtliche Punkte konnten durch das iterative Vorgehen erfolgreich evaluiert werden.

Die *Validierung im Dialog* erfolgte durch Expertengespräche im Rahmen der Wissenschaftsgemeinde sowie mit Experten aus den Bereichen Logistik- und Fertigungsplanung. Durch die Einbeziehung der unterschiedlichen Bereiche konnten die verschiedenen Aspekte des Systems hinterfragt und auf dessen Stimmigkeit geprüft werden. Dabei identifizierte Schwachstellen wurden sowohl im Konzept als auch in

⁴ Analog erfolgte auch die schrittweise Weiterentwicklung des Algorithmus und der Simulationssoftware. Details zum Vorgehen finden sich in Abschnitt 7.1.2.

7 Verifikation und praktische Validierung

der Implementierung eliminiert, sodass das Gesamtkonzept unter Fachexperten als in sich stimmig befunden wurde.

Im Rahmen des *Monitoring* wurden die Ergebnisse des Modells sichtbar gemacht, um die Konsistenz der Ergebnisse zu prüfen. Zum einen wurden Darstellungsformen wie das Visualisieren der Routen und die Auswertung als Gantt-Chart implementiert (Abbildung 7.3). Zum anderen wurden zusätzliche Überprüfungsmöglichkeiten für die Nebenbedingungen integriert und damit die Lösungsqualität unter deren Berücksichtigung kontrolliert. Das Plotten der durchgeführten Touren, deren Gesamtzeiten sowie sämtliche Einzelzeitanteile ermöglichten die Sicherstellung der praktischen Richtigkeit.

Für den *Ereignisvaliditätstest* wurden zum Test der Notfallstrategien manuell Aufträge eingebracht, welche zu kurzfristig für eine Auslieferung im Standardprozess sind. Als Hypothese wurde formuliert, dass derartige Eilaufträge im Simulationsergebnis bei den generierten Touren in der nächsten Tour auf der ersten Position stehen müssen, wie dies in Abbildung 7.5 gezeigt ist. Selbiges Verhalten wurde bei der zweiten Hypothese erwartet, wenn ein Auftrag aufgrund zu später Ankunft von Routenzügen im Nachhinein als dringend eingestuft wurde. Die Simulation zeigte in allen durchgeführten Simulationen für beide Fälle das erwartete Verhalten, womit die Verifikation des Notfallmodus in der Simulation erbracht werden konnte. Dasselbe Vorgehen wurde gewählt, um durch systematisches Prüfen der Reihenfolgen die Optimalität sicherzustellen. Dazu wurden zum einen Rechenbeispiele gewählt, die zum Vergleich vollständig von Hand gelöst werden konnten, zum anderen wurden Lösungen zu komplexen Aufgabenstellungen auf deren Plausibilität geprüft. Auch diese Tests konnten erfolgreich abgeschlossen werden.

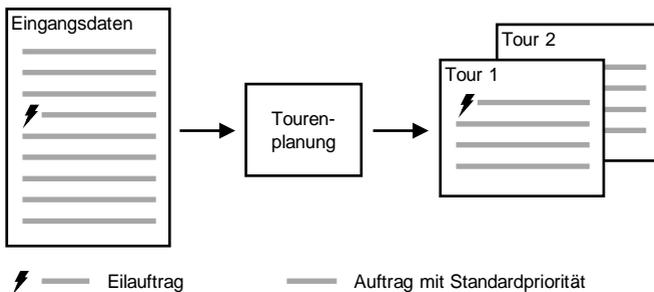


Abbildung 7.5: Erwartetes Verhalten für Aufträge mit Eilaufträgen

Abschließend lässt sich zusammenfassen, dass die zur Beurteilung der dynamischen Leistungsfähigkeit durchgeführten qualitativen Tests allesamt erfolgreich abgeschlossen werden konnten. Wie bereits beschrieben, können diese Tests nichts über die qualitative Optimalität der Lösung aussagen, da die zum Vergleich notwendige optimale Lösung für die Probleme nicht bekannt ist. Es lässt sich jedoch eindeutig feststellen, dass die Ergebnisse plausibel sind und manuell keine effizienteren Lösungen gefunden werden konnten.

7.3 Evaluierung der Prognosefähigkeit

Aufbauend auf die kurzfristige Planung ist die Vorausplanung über mehrere Stunden, wie sie in Kapitel 6 beschrieben wurde, notwendig. Deren Funktionsweise sowie Leistungsfähigkeit werden im Folgenden nachgewiesen.⁵

Wie in Abschnitt 7.2 sind auch hier reale Daten zur Evaluation notwendig.⁶ Dazu wurde ein Zeitraum aus einem Datensatz der Vergangenheit verwendet (Abschnitt 7.2.1). Die Bestellungen bis zum gewählten Startzeitpunkt wurden zur Generierung der Parameter für die Prognose genutzt, für 60 Stunden ab dem diesem Zeitpunkt wurden die Kapazitäten prognostiziert. Da ebenfalls bekannt war, welche Bestellungen in diesem Zeitraum tatsächlich anfielen, konnte mit den realen Daten wie in Abschnitt 7.2 eine Simulation durchgeführt werden. Aus der Differenz wurde die Abweichung zwischen Prognose und Realität im betrachteten Zeitraum zu jedem Zeitpunkt ermittelt. Ein beispielhafter Ausschnitt aus einem solchen Vergleich ist in Abbildung 7.6 zu sehen. Darin ist der tatsächliche Bedarf mit einer durchgezogenen Linie, die Prognose mit einer gestrichpunkteten Linie gezeichnet. An den Stellen, an denen die Prognose über dem tatsächlichen Bedarf liegt, ergibt sich eine Überkapazität, welche die Effizienz verschlechtert. Liegt die Linie des tatsächlichen Bedarfs unter der Prognose, so kann es zu Versorgungsengpässen kommen. Dieser Zustand ist daher möglichst zu vermeiden.

⁵ Zur Ableitung der Kapazitätsbedarfe sowie zur Bewertung der Prognose müssen diverse Rahmenbedingungen definiert sein. Die Beschreibung und Begründung dieser Annahmen findet sich in Abschnitt A.3.

⁶ Würden für den Nachweis künstlich erzeugte Testdatensätze verwendet, bestünde das Risiko, dass der Prognosealgorithmus nur die zuvor definierten Parameter der Normalverteilung wieder identifiziert, da die zu beweisende Grundannahme der Normalverteilung als Ausgangslage für die Generierung der Daten genutzt werden müsste.

7 Verifikation und praktische Validierung

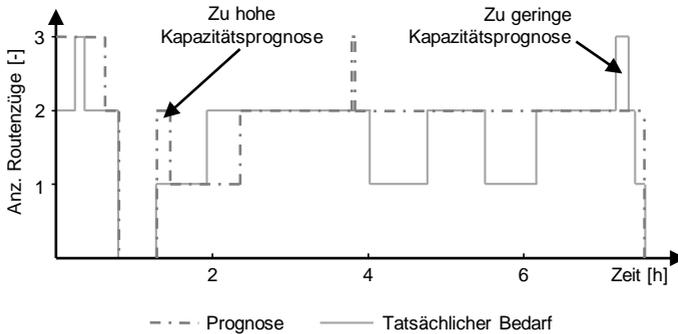


Abbildung 7.6: Vergleich zwischen Prognose und realen Abrufen zu einem beliebigen Zeitpunkt (Startzeitpunkt: 08.08.2015 10:13)

Diese Analyse wurde für alle Startpunkte über einen Zeitraum von 20 Stunden im Minutentakt durchgeführt. Die Vielzahl an Messungen konnte zur Auswertung in Abbildung 7.7 akkumuliert werden. Dabei wurde zu jedem Zeitpunkt ab Prognosestart der Mittelwert über alle Prognosen zu diesem Zeitpunkt ermittelt. Aus dem Graph lässt sich beispielhaft ablesen, dass 25 Stunden nach dem Prognosestart bei 2,8% der Prognosedurchläufe zu wenig Kapazität prognostiziert wurde, bei 13,8% der Durchläufe hingegen mehr Kapazität erwartet wurde, als tatsächlich notwendig gewesen wäre.

Die Betrachtung der Auswertung in Abbildung 7.7 lässt Schlüsse über die Qualität der Prognose über die Zeit zu. So ist die Prognosequalität zu Beginn des Zeitintervalls sehr präzise, je weiter in die Zukunft prognostiziert wird, desto ungenauer wird die Vorhersage. Da die hier verwendeten Daten aus der Anlaufphase eines Produktionsbereichs stammen, ist die geforderte Zeitstabilitätshypothese nicht exakt erfüllt. In der vorliegenden Auswertung wird die Prognose dahingehend schlecht, dass zu wenig Kapazität prognostiziert wird. Bei der Detailbetrachtung der zugrundeliegenden Datensätze zeigt sich, dass im Betrachtungsbereich eine Steigerung des Produktionsvolumens vorlag, die zu wenig prognostizierten Routenzüge daher zu erwarten waren.

Auf den ersten Blick überraschend ist die vergleichsweise schlechte Prognose in den ersten Minuten der Vorhersage. Hier ist zu bedenken, dass das hier vorgestellte Modell ausschließlich kommende Bestellungen prognostiziert, die Berücksichtigung

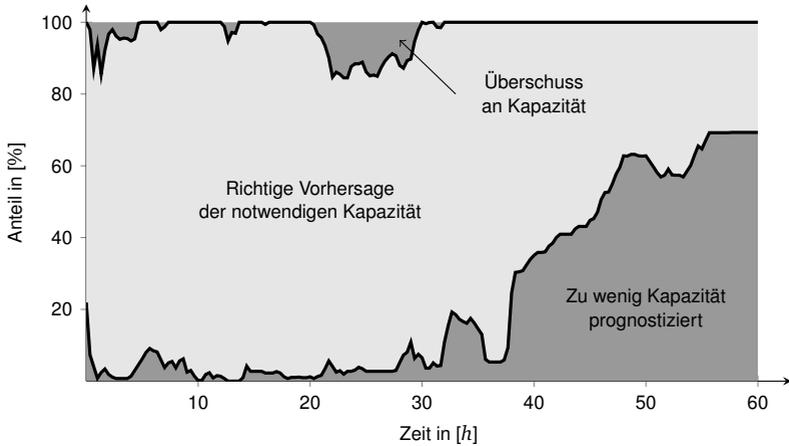


Abbildung 7.7: Verlauf der Prognosequalität über einen Zeitraum von 60 Stunden

bereits eingegangener, aber noch nicht abgearbeiteter Bestellungen wird nicht mehr vorgenommen. Die ersten Minuten der Prognose sind daher nicht repräsentativ. Dies stellt jedoch keine Einschränkung für den Einsatz des Modells dar, da dieser Zeitraum ohnehin durch den Algorithmus aus Kapitel 5 geplant wird und für die Prognose irrelevant ist.⁷

Die Alternativenbetrachtung zeigt, dass im Falle von permanent vorhandenen Kapazitäten diese im betrachteten Zeitraum nur zu 62,3% ausgelastet wären. Ohne Prognose bleibt jedoch keine andere Möglichkeit, als die volle Kapazität permanent vorzuhalten. Tabelle 7.1 zeigt die numerische Auswertung der Verläufe in Abbildung 7.7. So weist die Prognose eine hohe Trefferquote innerhalb der ersten 20 Stunden des Prognosezeitraums auf. Nur 2,8% der Zeit wird eine zu niedrige Kapazität prognostiziert, ein Kapazitätsüberschuss ist nur 1,5% der Zeit vorhanden.⁸

Die Auswertung zeigt, dass in den ersten 20 Stunden ab Prognosebeginn der Kapazitätsüberschuss bis auf 1,5% vollständig eliminiert werden kann. Selbst nach Abzug

⁷ Die ungenaue Vorhersagequalität bezieht sich dabei auf ungefähr die ersten zehn Minuten der Prognose. Der Algorithmus zur Tourenplanung berechnet die Touren je nach Auslastung für die nächsten 20 bis 45 Minuten, sodass hier eine ausreichende Überlappung in der Vorausschau vorliegt.

⁸ Für die Bewertung der Prognose müssen diverse Grenzwerte und Annahmen definiert werden, die in Abschnitt A.3 dargelegt sind. Diese gelten sowohl für die Berechnung als auch für die Auswertung.

7 Verifikation und praktische Validierung

Tabelle 7.1: Abweichung der Prognose vom realen Verlauf in Abhängigkeit der Zeit

Zeitbereich	Anteil zu hohe Prognose	Anteil zu niedrige Prognose
0 h – 20 h	1,5 %	2,8 %
20 h – 40 h	5,1 %	9,2 %
40 h – 60 h	0,0 %	56,6 %

der 2,8 % Zeitanteile, in denen zu wenig Kapazität prognostiziert wurde und somit kurzfristig Personal geholt werden musste, zeigt sich die positive Wirksamkeit der Prognose. Speziell in großen Fertigungsbereichen ist dieses kurzfristige Einspringen zusätzlicher Mitarbeiter i. d. R. möglich, nicht zeitkritische Tätigkeiten müssen hierfür zurückgestellt werden.

Wie sich zeigt, bietet die Prognose bereits in der hier vorgestellten Form wirtschaftliche Vorteile. Dennoch besteht noch Potenzial für Verbesserungen: Besonders die Zeiten, in denen eine Unterversorgung entsteht, gilt es durch weitere Maßnahmen in künftigen Forschungsarbeiten zu eliminieren.

7.4 Praktische Anwendung

7.4.1 Ausgangssituation des Anwenders

Das im Rahmen dieser Arbeit entwickelte Konzept zur Steuerung der Routenzüge konnte im Werk eines Automobilherstellers evaluiert werden. Das dort eingesetzte Einbehältermanagement erfordert einen 1:1-Tausch der Behälter, da an den meisten Stellen im Werk kein Platz für einen zusätzlichen Behälter an der Station ist. Wird ein neuer Behälter deutlich vor dem vollständigen Verbrauch der Teile des alten Behälters angeliefert, so wird das Umladen dieser Teile in den neuen Behälter erforderlich. Kann das Umladen aufgrund ungünstiger Konstellationen (zu hoher Füllgrad, Zeitknappheit, etc.) nicht durchgeführt werden, so wird der neue Behälter auf dem Weg abgestellt und in einer der nächsten Touren der alte mitgenommen. Dies führt zu einem zu versperrten Wegen innerhalb des Werks, zum anderen kann auch ein Engpass der Ladungsträger bzw. der Rollenwagen zum Transport der Ladungsträger auftreten und stellt daher eine ungünstige Vorgehensweise dar.

Die Versorgungssicherheit im Werk konnte bereits ohne die neue Steuerung sichergestellt werden, dennoch waren die genannten Problemfelder im Werk in manchen Bereichen sichtbar. Eine Optimierung der Prozesse schien daher lohnenswert und bot einen passenden Anwendungsfall für die entwickelte Routenzugsteuerung.

7.4.2 Ablauf der Logistikprozesse

Der Abruf neuer Materialien für eine Station des Werks erfolgt bestandsgesteuert und wird durch das Unterschreiten eines Grenzwerts des Materialbestands an der Station durch den Werker ausgelöst. Die Bestellung geht im Logistiklager ein, das bestellte Material wird aus dem Lager geholt und für die Routenzüge zur Auslieferung bereitgestellt. Das Vorgehen ist in Abbildung 7.8 schematisch gezeigt.

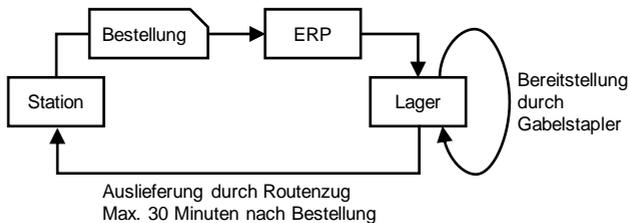


Abbildung 7.8: Schematischer Ablauf der Abarbeitung einer Bestellung

Ab dem Zeitpunkt der Bestellung bis zur spätest möglichen Lieferung dürfen maximal 30 Minuten vergehen. In dieser Zeitspanne muss die Auslagerung zur Bereitstellung für die Routenzüge ebenfalls stattfinden. Die Anlieferung ist möglichst spät im Zeitfenster durchzuführen, um die oben genannten Problemfälle zu vermeiden.

Es existieren vordefinierte starre Routen zu Auslieferung, in der Praxis werden die Touren für die Routenzüge jedoch durch die Routenzugfahrer manuell zusammengestellt. Dabei wird der aktuelle Auftragsbestand der ausgelagerten Teile in Betracht gezogen, i. d. R. erfolgt die Zuteilung ausschließlich nach der spätesten Ankunftszeit.

7.4.3 Integration der Steuerung

Zur Erprobung konnte die Steuerung aufgrund mangelnder Exportfunktionalitäten des eingesetzten Enterprise Resource Planning (ERP)-Systems nicht direkt mit der entwickelten Software (Abschnitt 7.1) gekoppelt werden. Deshalb wurden sämtliche im Lager zur Abfahrt bereitstehenden Behälter manuell eingetragen, hierbei waren jeweils nur der Ablieferort sowie die späteste Ablieferzeit relevant. Diese Daten werden für die Gabelstaplerfahrer aus dem ERP ausgedruckt, die Datenverfügbarkeit ist daher prinzipiell gegeben. Die Allgemeingültigkeit ist durch diesen manuellen Zwischenschritt nicht eingeschränkt.

Die Software zur Routenzugoptimierung erzeugt für jede Tour einen Fahrauftrag, der dem Routenzugfahrer in gedruckter Form oder als dynamische Liste auf dem Smartphone bzw. Tablet zur Durchführung übergeben wird. Dabei sind jeweils die Station sowie die geplante und die spätestmögliche Ankunftszeit angegeben. Die Information über die spätestmögliche Ankunft ermöglicht dem Routenzugfahrer bei Störungen (z. B. Strecke nicht passierbar, Station in Wartung, etc.) eine Einschätzung, inwiefern er von der generierten Route abweichen kann. Materialabrufe, die bereits dringlich sind, müssen dabei zwingend zu Beginn der Tour ausgefahren werden und sind daher auf dem Ausdruck auffällig markiert (siehe Abbildung 5.11).

7.4.4 Parametrierung auf die physischen Gegebenheiten des Anwenders

Die Parameter für das Tourenplanungsproblem (siehe Abschnitt 5.1.1) konnten aufgrund organisatorischer Restriktionen nicht direkt erfasst werden und wurden daher durch eine Regressionsanalyse aufgenommen (siehe Abschnitt 5.1.2). Vorteil dieses Verfahren ist es, dass sämtliche Sondereffekte (z. B. Umwege, kurze Stopps wegen anderer Fahrzeuge, etc.) implizit in diesen durchschnittlichen Werten mit berücksichtigt werden (vgl. Abschnitt 5.1.2 und Abschnitt 5.9.1).

Es wurde bei ca. 30 Touren die jeweilige Gesamtdauer gemessen und daraus die Parameter der durchschnittlichen Routenzuggeschwindigkeit v , die konstante Umladezeit t_c sowie der Faktor für die variable Umladezeit f_c ermittelt. In weiteren Tests stimmten die errechneten Zeiten gut mit den tatsächlich gefahrenen Zeiten überein, die Abweichungen lagen im Rahmen der Messungenauigkeit.

7.4.5 Anwendung der Steuerung

Der Probelauf fand an einem zufällig ausgewählten Vormittag über einen Zeitraum von eineinhalb Stunden von 9:30 – 11:00 statt, die Anzahl der Materialabrufe lag in diesem Zeitraum in einer üblichen Größenordnung. Im Betrachtungszeitraum waren weder Pausen, noch traten auffälligen Störungen im Werk auf, so dass der Eingang von Materialabrufen keine Aussetzer hatte.

Wie in Abschnitt 7.4.3 beschrieben, wurden sämtliche Materialabrufe nach Bestelleingang im Lager manuell in die Software zur Routenzugoptimierung übertragen und die berechneten Touren in Papierform an die Fahrer übergeben. Dieser Prozess erwies sich als gangbare Lösung für die Validierung, es offenbarten sich durch diese Form der Schnittstelle jedoch viele Fehlerquellen, die leichte Verzögerungen in der Abfahrt von Routenzügen verursachten.

7.4.6 Analyse der Simulationsergebnisse

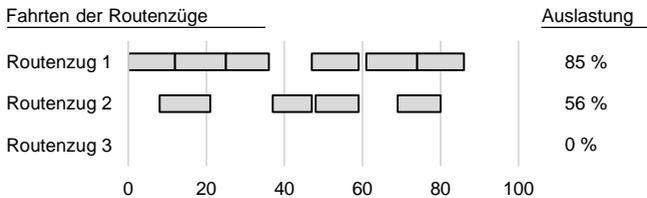


Abbildung 7.9: Ergebnisse der praktischen Validierung im Werk

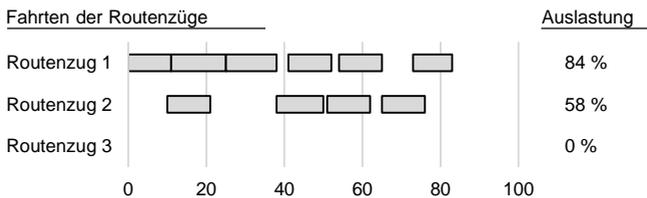


Abbildung 7.10: Ergebnisse der Simulation mit den Eingangsdaten der praktischen Validierung

Für die Fahrten der Routenzüge wurden jeweils die Abfahrts- und Ankunftszeiten protokolliert. Abbildung 7.9 zeigt diese Fahrten. Wie sichtbar wird, wurde Routen-

zug 1 in diesem Zeitraum mit 85 % relativ gut ausgelastet, der zweite Routenzug ist mit 56 % auch noch rentabel eingesetzt. Routenzug 3 wurde im Beobachtungszeitraum nicht benötigt und hätte bei einer vorausschauenden Planung eingespart werden können (vgl. Kapitel 6 und Abschnitt 7.3).

Nachdem sämtliche Materialabrufe im ERP-System protokolliert werden, war es möglich, die exakt gleichen Materialabrufe im Nachgang in der Simulation zu evaluieren. Die Ergebnisse der Simulation sind in Abbildung 7.10 zum Vergleich gezeigt. Es zeigte sich hierbei eine sehr gute Übereinstimmung der Auslastung (85 % vs. 84 % bzw. 56 % vs. 58 %), was ein deutliches Indiz dafür ist, dass die Modellierung die Realität gut abbildet. Im Experiment mussten einige der Touren im Vergleich zur Simulation verzögert gestartet werden. Dies konnte auf Prozessinstabilitäten durch die manuelle Dateneingabe sowie die daraus folgenden notwendigen Korrekturen bei Fehleingaben zurückgeführt werden und resultierte in Abbildung 7.9 zur Verschiebung einiger Zeitblöcke nach hinten.

Zusammenfassend lässt sich feststellen, dass sich die Ergebnisse aus der Praxis mit denen aus der Simulation decken und demnach die simulierten Einsparpotenziale auch im praktischen Einsatz gültig sind. Welche Rahmenbedingungen erfüllt sein müssen, damit durch die entwickelte Steuerung Einsparpotenziale entstehen, ist Inhalt des folgenden Abschnitts.

7.5 Identifikation wirtschaftlich relevanter Einsatzgebiete

Die im Rahmen dieser Arbeit entwickelte Methode zur Steuerung und Optimierung der fabrikinternen Logistik wurde im Hinblick auf eine möglichst breite Einsetzbarkeit konzipiert. Dennoch kann das volle Einsparpotenzial nur unter bestimmten Voraussetzungen realisiert werden, welche im Folgenden diskutiert werden. Dafür wurden im Rahmen einer Literaturanalyse relevante Aspekte identifiziert sowie durch Expertengespräche validiert und konsolidiert, welche sich in Tabelle A.1 finden. Daraus wurden für das in dieser Arbeit vorgestellte Konzept die Gegebenheiten für potenzielle Anwendungsfälle abgeleitet und spezifiziert, bei deren Erfüllung der maximale wirtschaftliche Vorteil erzielt werden kann. Diese finden sich in Tabelle 7.2. Die Nicht-Erfüllung eines einzelnen Aspekts ist dabei nicht zwangsläufig

7.5 Identifikation wirtschaftlich relevanter Einsatzgebiete

ein Ausschlusskriterium. Es ist jedoch sorgfältig zu prüfen, wie sich dies auf die Wirtschaftlichkeit auswirkt.

Tabelle 7.2: Erforderliche Gegebenheiten für den erfolgreichen Einsatz des entwickelten Konzepts zur Routenoptimierung

Kategorie	Anforderung
Physisch	Existierende oder geplante interne Routenzuglogistik vorhanden Hohe Wegkomplexität gegeben Hohe Anzahl an Stationen Dynamisches und kurzfristiges Bestellverhalten Zeitfenster für die Belieferung Mehrfachverwendung von Fahrzeugen erforderlich
Modellierung	Routenanpassung einziger kurzfristiger Freiheitsgrad Rückführung leerer Behälter nur durch 1:1-Tausch Kontinuierlich fortlaufender Planungshorizont möglich
Organisatorisch	Finanzielle Mittel gegeben Personalkapazität vorhanden Hinreichend Zeit zur Einführung Datenverfügbarkeit Technische Machbarkeit Wirtschaftlichkeit im Einzelfall gegeben

Die physischen Anforderungen beschreiben die notwendigen Begebenheiten im Werk vor Ort. Dazu gehört zum einen, dass in der internen Logistik die Möglichkeit zur gebündelten Auslieferung gegeben sein muss, klassischerweise wird dies durch den Einsatz von Routenzügen ermöglicht. Zum anderen muss sowohl durch die Situation an Wegen vor Ort sowie die Anzahl an Stationen eine Komplexität gegeben sein, die manuell nicht mehr überblickt werden kann. Dynamische Systeme sowie enge Zeitfenster sorgen ebenso für einen Vorteil dynamischer algorithmusbasierter Systeme. Zu guter Letzt ist die mehrfache Verwendung von Fahrzeugen ein in der Praxis sehr wichtiger Schritt, welcher von vielen Standardlösungsverfahren jedoch nicht gelöst werden kann.

Das vorgestellte Konzept löst die logistische Aufgabe allein durch die Anpassung der Routen. Das Verschieben von Stationen oder Lagern sowie die Anpassung von Kapazitäten kann zwar durch die integrierten Simulationsmöglichkeiten unterstützt werden, ist jedoch nicht elementarer Bestandteil des Konzepts. Ebenso wenig werden gesonderte Rückführungen von Leerbehältern betrachtet, da so optimiert wird, dass

diese immer nach der Auslieferung zurückgebracht werden.⁹ Zudem ist es im Rahmen des Konzepts am gewinnbringendsten, mit einem kontinuierlich fortlaufenden, jedoch zeitlich limitierten Planungshorizont zu rechnen.

Von organisatorischer Seite muss der entsprechende finanzielle Rahmen zur Einführung und zum Betrieb gewährleistet sein. Ebenso muss die notwendige Personalkapazität und ein ausreichender Zeitrahmen zur Implementierung zur Verfügung stehen. Essenziell ist auch das Vorhandensein von Daten, sowohl statischer (wie bspw. das Werkslayout), als auch dynamischer Daten (wie Materialabrufe), die automatisiert über Schnittstellen aus der Produktionsplanung an die Logistiksteuerung übermittelt werden müssen. Die technische Machbarkeit im Einzelfall muss gegeben sein, außerdem sollte vor Implementierung eine Betrachtung der Wirtschaftlichkeit durchgeführt werden.

Werden sämtliche Aspekte vom betrachteten potenziellen Anwendungsfall in hohem Maß erfüllt, so sind große Einsparungen durch die Implementierung des Konzepts zu erwarten. Mit schlechterer Erfüllung sinken diese. Die hier aufgeführten Aspekte können nur einen ersten Anhaltspunkt für gute Einsatzgebiete geben, eine detaillierte Wirtschaftlichkeitsbewertung können sie nicht ersetzen.

7.6 Kritische Bewertung der Forschungsergebnisse

Die Ergebnisse werden anhand der übergeordneten Anforderungen aus Abschnitt 3.1 sowie daraus abgeleiteten problemspezifischen Anforderungen aus Abschnitt 3.2 bewertet und somit der Zielerreichungsgrad der Ergebnisse ermittelt.

Die übergeordnete Anforderung **Richtigkeit der Ergebnisse** kann als erfüllt angesehen werden, da das Modell aufgrund der *berücksichtigten variablen Zeitanteile* beim Umladen genauer ist als bestehende Modelle. Differenzen zwischen Modell und Realität sind daher kleiner bzw. seltener als bei bereits existierenden Ansätzen. Zusätzlich konnte in Tests nachgewiesen werden, dass das System eine *pünktliche Belieferung* realisieren kann und dabei trotzdem *dynamisch reaktionsfähig* bleibt. Die

⁹ Rücktransporte sind im Modell trotzdem darstellbar. Sie werden mit einem leeren Hintransport dargestellt, sodass auf dem Rückweg auf dem Leerplatz der entsprechende Behälter mitgenommen werden kann. Dieses Vorgehen verschlechtert die Auslastung und sollte daher nur im Ausnahmefall zur Anwendung kommen.

Lösungsqualität erwies sich in theoretischen Untersuchungen sowie im Praxiseinsatz zudem als ausreichend, wenngleich die optimale Lösung aufgrund des Einsatzes von Meta-Heuristiken nicht garantiert werden kann. Die praktische Evaluierung zeigte zudem, dass auch bei schlechter bzw. nicht aktueller Datenlage¹⁰ gute Ergebnisse erzielt werden können und dementsprechend die erforderliche *Fehlertoleranz* gegeben ist.

Die **Anwendbarkeit im industriellen Kontext** konnte durch die prototypische Implementierung (siehe Abschnitt 7.4) nachgewiesen werden. Dabei wurde darauf Wert gelegt, keine anwendungsfallspezifischen Annahmen oder Voraussetzungen zugrunde zu legen, um die Anwendbarkeit auch bei anderen Unternehmen sicherstellen zu können.

Da für den Betrieb nur die aktuellen Materialabrufe sowie statische Daten wie bspw. das Layout der Halle benötigt werden, ist das System in nahezu alle denkbaren Produktionsszenarien *integrierbar* (vgl. Abschnitt 7.5). Erweiterte *Mitarbeitererfahrung* ist nur bei der Einrichtung notwendig, anschließend können auch Mitarbeiter in der Logistik ohne spezifische Systemkenntnis optimale Routen erzeugen. Die *Erweiterbarkeit* hinsichtlich neuer Stationen ist ebenfalls ohne Anpassung des Optimierungsmodells möglich. Eine zusätzlich identifizierte Anforderung ist die *Korrektheit* der Ergebnisse, d. h. die errechneten Touren müssen sich praktisch in der vorhergesagten Zeit durchführen lassen. Dies konnte im Rahmen der prototypischen Implementierung erfolgreich nachgewiesen werden. Die *Rechenzeit* war für die meisten Fälle ausreichend und erwies sich im Anwendungsbeispiel als unproblematisch.¹¹ Für deutlich größere Szenarien ist denkbar, den Auslieferungsbereich zu unterteilen, um die Problemgröße zu reduzieren und somit die Rechengeschwindigkeit nochmals zu erhöhen.¹² Die *Prognosefähigkeit* der Bedarfe ermöglicht die monetäre Optimierung der fabrikinternen Materialbereitstellung. Die erzielten Ergebnisse erlauben eine

¹⁰ Die vorhandenen Daten waren von guter Qualität und in Echtzeit vorhanden, durch das manuelle Eintragen passierten jedoch Fehler. Dass sich diese nach der Korrektur nicht nachhaltig auf das Ergebnis auswirkten, zeigt die Fehlertoleranz des Systems.

¹¹ Für die praktische Evaluierung wurde die maximale Rechenzeit im Algorithmus auf zwei Sekunden begrenzt. Dies war für hinreichend viele Iterationen des Ameisenalgorithmus ausreichend, so dass eine gute Ergebnisqualität erzielt werden konnte.

¹² Experimente mit 100 gleichzeitigen Bestellungen zeigen, dass nur noch wenige Iterationen innerhalb der gewählten Rechenzeit von zwei Sekunden möglich sind und somit eine gute Lösung nicht garantiert werden kann. In derart großen Szenarien ist jedoch im Regelfall eine Unterteilung in Segmente möglich, sodass die Rechenzeit ohne signifikante Einschränkungen bei der Lösungsqualität erheblich verbessert werden kann.

solide Abschätzung, wobei eine längere Vorhersage die finanziellen Einsparungen nochmals steigern kann.

Die **Wirtschaftlichkeitsbetrachtung** erfolgt durch die Sicherstellung des Aufwand-Nutzen-Verhältnisses: An zusätzlichem Aufwand sind die Kosten für die Lizenzierung der Optimierungssoftware¹³, ein handelsüblicher Rechner sowie ein Drucker für die Ausgabe der Informationen an die Routenzugfahrer anzusetzen. An internen Kosten ist der Implementierungsaufwand für die Schnittstelle zum ERP-System für den Import der Lieferaufträge an die Optimierungssoftware anzuführen. Genaue Kosten hierfür unterliegen sehr stark firmeninternen Kostensätzen bzw. der verfügbaren Personalkapazität bei der Einführung. Das System ist wartungsarm konzipiert, im laufenden Betrieb sind daher keine Änderungen notwendig. Auf der Nutzenseite sind u. a. die *hohe Auslastung* sowie der damit verbundene *effiziente Ressourceneinsatz*, die zeitweise Möglichkeit zur anderweitigen Nutzung von Fahrzeugen und Personal sowie die Fähigkeit zur *Prognose* von notwendigen Kapazitäten gegeben. Zusätzlich kann die bisher erforderliche, personalintensive Neuplanung der statischen Routen bei neuen Materialnummern oder neuen Produktionsbereichen nahezu vollständig eingespart werden, da ohnehin für jeder Tour eine automatisierte komplette Neuplanung durchgeführt wird. Durch seinen Aufbau ist das System auch für große Bereiche einsetzbar und kann somit mit der Fabrik mitwachsen. Die *Skalierbarkeit hinsichtlich der Problemgröße* kann daher als erfüllt angesehen werden. Dies führt in Konsequenz zu einer verbesserten *Fehlertoleranz*, einem niedrigeren Flächenverbrauch in der Produktion sowie geringeren Kosten für Anpassungen am Produktionssystem. In Summe führen die Verbesserungen zur *Vermeidung von Verschwendung* und stellen somit einem wichtigen Baustein im Hinblick auf schlanke und effiziente Produktionssysteme dar. Tabelle 7.3 zeigt eine fiktive Kostenrechnung, wobei die Kosten zur Einrichtung des Systems tendenziell zu hoch angenommen wurden, die Kostensätze für Mitarbeiter und Routenzug tendenziell zu niedrig. Dennoch ergibt sich bereits ab dem 86. Tag ein positiver Anwendungsfall, wenn durch den Einsatz des Konzepts dauerhaft ein Routenzug eingespart werden kann. In großen Logistikbereichen ist dies eine durchaus realistische Annahme.

¹³ Das Softwaretool wird derzeit nicht kommerziell vertrieben, daher können für diesen Posten keine realistischen Kostensätze angenommen werden.

7.6 Kritische Bewertung der Forschungsergebnisse

Tabelle 7.3: Wirtschaftlichkeitsbetrachtung mit fiktiven Kostensätzen

Einmalige Kosten des Systems zur Einrichtung	
Lizenzkosten	10.000€
Hardware	2.000€
Personalkosten zur Einrichtung	2.000€
Entwicklung der Schnittstelle zum ERP	10.000€
Summe	24.000€
Einsparung pro Tag	
1 Routenzug	80€
1 Mitarbeiter	200€
Summe	280€
Gewinnschwelle [Tage]	
Kosten / Ersparnisse	86 Tage

Bei der Modellierung wurden keine anwenderspezifischen Einschränkungen getroffen. Die Annahmen sind für jedes Großserienwerk gültig. Die geforderte **Allgemeingültigkeit** ist dadurch gegeben, dass zum einen allgemeingültige Annahmen und Voraussetzungen für die Konzeptionierung gewählt wurden, zum anderen die Optimierung so gestaltet ist, dass sowohl die Zielfunktion, als auch die Nebenbedingungen schnell erweitert bzw. geändert werden können. Das System kann somit leicht auf andere Szenarien adaptiert werden. Die in Tabelle 3.1 aufgeführten Anforderungen *Skalierbarkeit hinsichtlich der Problemgröße, Integrierbarkeit, Erweiterbarkeit, Korrektheit* und *Berücksichtigung variabler Zeitanteile* wurden in den vorigen Abschnitten bereits betrachtet.

Zusammengefasst erfüllt das entwickelte Konzept die in Kapitel 3 definierten Anforderungen zum Großteil vollkommen, bei den restlichen Anforderungen wurde ein ausreichender Erfüllungsgrad erreicht. Tabelle 7.4 fasst die Erfüllung der einzelnen Anforderungen zusammen.

7 Verifikation und praktische Validierung

Tabelle 7.4: Bewertung der Erfüllung der einzelnen Anforderungen

	Grad der Erfüllung
Pünktlichkeit	●
Dynamische Reaktionsfähigkeit	●
Fehlertoleranz	●
Vermeidung von Verschwendung	●
Geringer Ressourceneinsatz	●
Hohe Auslastung	●
Prognosefähigkeit	◐
Skalierbarkeit hins. Problemgröße	◐
Integrierbarkeit	●
Erweiterbarkeit	●
Korrektheit	●
Unabhängigkeit von Mitarbeitererfahrung	●
Rechenzeit	◐
Lösungsqualität	◐
Berücksichtigung variabler Zeitanteile	●

Legende:

- Anforderung voll erfüllt
- ◐ Anforderung zum Großteil für den praktischen Einsatz hinreichend erfüllt

8 Zusammenfassung und Ausblick

8.1 Zusammenfassung

Aufgrund gesättigter Märkte und einer in den meisten Märkten großen Anzahl an Substitutionsgütern verschärft sich für viele Unternehmen der Zwang zu hoher Variantenflexibilität. Dies fordert sowohl vom Produktionssystem als auch von der daran angrenzenden fabrikinternen Materialbereitstellung ein Höchstmaß an Flexibilität und Reaktionsfähigkeit. In Konsequenz muss die Logistik sowohl schnell auf Störungen oder Planänderungen reagieren können als auch aufwandsarm an strukturelle Änderungen anpassbar sein. Das entwickelte Konzept ermöglicht dabei unter anderem die schnelle Reaktion auf neue Bestellungen oder Störungen, stellt eine echtzeitfähige Routenberechnung zur Verfügung und zeigt eine hohe Fehlertoleranz auf.

Die Analyse zum derzeitigen Stand der Wissenschaft in Kapitel 4 offenbart dabei im Hinblick auf die definierten Anforderungen (vgl. Kapitel 3), dass existierende Ansätze diese Herausforderungen nur bedingt lösen können. So können bspw. nur die wenigsten Algorithmen mit variablen Zeitanteilen umgehen, zusätzlich sind sie in der Regel auf statische Benchmarkinstanzen ausgelegt und somit nicht auf dynamische Reaktionsfähigkeit optimiert. Die Vorhersage von benötigten Kapazitäten ist bei derartigen Konzepten i. d. R. nicht vorgesehen, was die finanziellen Einsparpotenziale beschränkt.

Auf diesen Erkenntnissen wurde im Rahmen der vorliegenden Arbeit der Ameisenalgorithmus für den Einsatz in der Intralogistik weiterentwickelt, um als Kernelement der dynamischen Routenzugsteuerung den Einsatz von Routenzügen in der fabrikinternen Materialbereitstellung zu optimieren. Dazu wurden zahlreiche zusätzliche Restriktionen wie die zyklische Wiederverwendung von Routenzügen oder die Berücksichtigung zeitlich veränderlicher Einflussgrößen berücksichtigt. Zusätzlich wurde das Modell in die Regelabläufe der Logistik integriert.

Darauf aufbauend wurde die Problematik der schwierigen Vorhersehbarkeit von Kapazitätsbedarfen in Kapitel 6 adressiert. Dazu wurde auf Basis einer statistischen Analyse der vergangenen Bestellungen ein Prognoseverfahren entwickelt, welches hinreichend genaue Werte für die nähere Zukunft liefert.

Die Ergebnisse der Validierung und Verifikation (vgl. Kapitel 7) zeigen, dass der Einsatz des Algorithmus in der Produktion sowohl fachlich richtige Lösungen generiert, als auch in der Praxis eine deutliche Effizienzsteigerung ermöglicht. Die Prognose erlaubt eine sinnvolle Vorhersage der erforderlichen Kapazitäten über die nächsten Stunden, für eine garantierte Verfügbarkeit der richtigen Kapazität über mehrere Tage im Voraus ist jedoch eine präzisere Prognose erforderlich. Die Bewertung erfolgte anhand der in Kapitel 3 definierten Anforderungen, wobei sich herausstellte, dass die vier übergeordneten Anforderungen *Richtigkeit der Ergebnisse*, *Anwendbarkeit im industriellen Kontext*, *Wirtschaftlichkeit* sowie *Allgemeingültigkeit* in einem hohen Maße erfüllt werden.

8.2 Ausblick

Die Ergebnisse der vorliegenden Arbeit bilden die Grundlage für weitere Forschungsaktivitäten zur Effizienzsteigerung der fabrikinternen Materialbereitstellung. Die im Folgenden aufgeführten Themengebiete können die Grundlage für weitere Forschungsvorhaben bilden:

Optimierung der Rechenzeit: Die Rechenzeit kann durch diverse Optimierungsmaßnahmen weiter verbessert werden. Beispielsweise kann durch gezieltes Unterteilen des Lösungsraums die Anzahl möglicher Lösungen reduziert werden, ohne dabei die Lösungsqualität erheblich zu verschlechtern. Außerdem können bei größeren Optimierungsproblemen neue Materialabrufe in die laufende Rechnung integriert werden, ohne die bisherige Rechnung zu unterbrechen. Zuletzt ist es denkbar, die geltenden Nebenbedingungen zunächst zu vernachlässigen und erst nach und nach in der Optimierung zu berücksichtigen. Dieses Vorgehen, welches unter dem Begriff *Relaxierung* in diversen Algorithmen zu finden ist, ist auch für Ameisenalgorithmen mit vielen Nebenbedingungen ein aussichtsreiches Forschungsgebiet. Sämtliche dieser

Verbesserungen kommen neben einer besseren Rechenzeit auch dem Ziel der *Skalierbarkeit hinsichtlich der Problemgröße* zugute.

Verbesserung der Prognosefähigkeit: Ein weiteres Feld für Verbesserungen ergibt sich auf dem Gebiet der Prognose künftiger Kapazitätsbedarfe. Das derzeitige Vorgehen hat Vorteile im Hinblick auf die einfache Integrierbarkeit, kann jedoch keine garantierte Prognose abgeben. Daher ist zu überlegen, ob zusätzliche Prognosedaten durch die Beobachtung von Mustern (wie bspw. der Materialverbrauch in angrenzenden Bereichen) erfasst werden können. Dies würde eine Verbesserung der Prognosequalität ermöglichen, ohne die Schnittstellenkomplexität zu erhöhen und damit die Anwendbarkeit im industriellen Kontext zu gefährden. Aktuelle Fortschritte auf dem Gebiet des maschinellen Lernens können dabei von erheblichem Nutzen sein. Zusätzlich ist eine engere Verzahnung zwischen aktuellen Fahrten und der Prognose erstrebenswert, um die Prognoseergebnisse bereits vom ersten Augenblick an verwertbar zu machen.

Validierung mit vollautomatisierter Logistik: Die in dieser Arbeit vorgestellten Konzepte wurden allesamt mit der Option zur vollkommenen Automatisierung der fabrikinternen Logistik entwickelt. Allerdings sind für den vollautonomen Betrieb neben neuen Konzepten für Fahrzeuge (vgl. WEHNING & POPP 2015, S. 400 f.) noch einige zusätzliche Bestandteile notwendig, wie beispielsweise eine detaillierte Ausleitung der zu fahrenden Wege. Zusätzlich treten bei derartigen Szenarien Effekte auf, die durch die bisherigen Validierungsschritte nicht abgeprüft werden konnten. Das Ziel künftiger Forschungsarbeiten sollte daher die Erprobung auf automatischen Logistiksystemen sowie die Ableitung sinnvoller Verbesserungen und Erweiterungen aus den daraus gewonnenen Erkenntnissen sein.

Die hier vorgeschlagenen weiteren Forschungsansätze sind nicht als exklusive Aufzählung zu verstehen, vielmehr stellen sie einen aus heutiger Sicht sinnvollen weiteren Weg zur Verbesserung der Intralogistik dar. Aufgrund der in Kapitel 1 beschriebenen starken Veränderungen im Unternehmensumfeld können sich Anforderungen und die dafür benötigten Lösungen schnell ändern und erfordern somit möglicherweise bereits in naher Zukunft gänzlich andere Erweiterungen zum hier vorgestellten Konzept.

Literatur

ABELE & REINHART 2011

Abele, E.; Reinhart, G.: *Zukunft der Produktion. Herausforderungen, Forschungsfelder, Chancen*. München: Hanser. 2011.

AI & KACHITVICHYANUKUL 2007

Ai, J.; Kachitvichyanukul, V.: A Particle Swarm Optimization for the Capacitated Vehicle Routing Problem. In: *International Journal of Logistics and SCM Systems*. Band 1. 2. 2007, S. 50–55.

ARMAS & MELIÁN-BATISTA 2015

Armas, J. de; Melián-Batista, B.: Variable Neighborhood Search for a Dynamic Rich Vehicle Routing Problem with time windows. *Computers & Industrial Engineering* Band 85 (2015), S. 120–131.

ARNDT 2015

Arndt, H.: *Logistikmanagement*. Wiesbaden: Springer Fachmedien. 2015.

ARNOLD 2008

Arnold, D.: *Handbuch Logistik*. 3. Auflage. VDI-Buch. Berlin: Springer. 2008.

ARNOLDS ET AL. 2017

Arnolds, H.; Heege, F.; Röh, C.; Tussing, W.: *Materialwirtschaft und Einkauf. Grundlagen - Spezialthemen - Übungen*. 13. Auflage. Wiesbaden: Springer Fachmedien. 2017.

AUDI 2016

Audi, Hrsg. (2016): *Die Modulare Montage. Fertigungsinseln statt Fließband*. Audi AG. URL: <https://www.audi-mediacenter.com/de/audi-techday-smart-factory-7076/die-modulare-montage-7078> (besucht am 14.01.2017).

AUSTIN & STEYERBERG 2015

Austin, P. C.; Steyerberg, E. W.: The number of subjects per variable required in linear regression analyses. *Journal of clinical epidemiology* Band 68 (2015) (6), S. 627–636.

BACKHAUS ET AL. 2016

Backhaus, K.; Erichson, B.; Plinke, W.; Weiber, R.: *Multivariate Analysemethoden. Eine anwendungsorientierte Einführung*. 14. Auflage. Berlin, Heidelberg: Springer Gabler. 2016.

BAHILL & GISSING 1998

Bahill, A. T.; Gissing, B.: Re-evaluating systems engineering concepts using systems thinking. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)* Band 28 (1998) (4), S. 516–527.

BALDACCIO ET AL. 2012

Baldacci, R.; Mingozzi, A.; Roberti, R.: Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. *European Journal of Operational Research* Band 218 (2012) (1), S. 1–6.

BALSEIRO ET AL. 2011

Balseiro, S. R.; Loiseau, I.; Ramonet, J.: An Ant Colony algorithm hybridized with insertion heuristics for the Time Dependent Vehicle Routing Problem with Time Windows. *Computers & Operations Research* Band 38 (2011) (6), S. 954–966.

BATTARRA ET AL. 2009

Battarra, M.; Monaci, M.; Vigo, D.: An adaptive guidance approach for the heuristic solution of a minimum multiple trip vehicle routing problem. *Computers & Operations Research* Band 36 (2009) (11), S. 3041–3050.

BEASLEY 1983

Beasley, J. E.: Route first—Cluster second methods for vehicle routing. *Omega* Band 11 (1983) (4), S. 403–408.

BEKTAŞ ET AL. 2014

Bektaş, T.; Repoussis, P. P.; Tarantilis, C. D.: Dynamic Vehicle Routing Problems. In: *Vehicle Routing. Problems, Methods, and Applications*. Hrsg. von Toth, P; Vigo, D. 2. Auflage. MOS-SIAM series on optimization. Philadelphia: Society for Industrial and Applied Mathematics. 2014, S. 298–347.

BELFIORE & YOSHIZAKI 2009

Belfiore, P.; Yoshizaki, H. T. Y.: Scatter search for a real-life heterogeneous fleet vehicle routing problem with time windows and split deliveries in Brazil. *European Journal of Operational Research* Band 199 (2009) (3), S. 750–758.

BELHAIZA ET AL. 2014

Belhaiza, S.; Hansen, P.; Laporte, G.: A hybrid variable neighborhood tabu search heuristic for the vehicle routing problem with multiple time windows. *Computers & Operations Research* Band 52 (2014), S. 269–281.

BERGER & BARKAOUI 2004

Berger, J.; Barkaoui, M.: A parallel hybrid genetic algorithm for the vehicle routing problem with time windows. *Computers & Operations Research* Band 31 (2004) (12), S. 2037–2053.

BETTINELLI ET AL. 2011

Bettinelli, A.; Ceselli, A.; Righini, G.: A branch-and-cut-and-price algorithm for the multi-depot heterogeneous vehicle routing problem with time windows. *Transportation Research Part C: Emerging Technologies* Band 19 (2011) (5), S. 723–740.

BLOHM ET AL. 2016

Blohm, H.; Beer, T.; Seidenberg, U.; Silber, H.: *Produktionswirtschaft. Potenziale, Prozesse, Produkte*. 5. Auflage. Herne: NWB. 2016.

BOUSONVILLE 2017

Bousonville, T.: *Logistik 4.0. Die digitale Transformation der Wertschöpfungskette*. Wiesbaden: Springer Gabler. 2017.

BRAEKERS ET AL. 2016

Braekers, K.; Ramaekers, K.; van Nieuwenhuysse, I.: The vehicle routing problem. State of the art classification and review. *Computers & Industrial Engineering* Band 99 (2016), S. 300–313.

BRÄYSY 2003

Bräysy, O.: A Reactive Variable Neighborhood Search for the Vehicle-Routing Problem with Time Windows. *INFORMS Journal on Computing* Band 15 (2003) (4), S. 347–368.

Literatur

BRÄYSY & GENDREAU 2005

Bräysy, O.; Gendreau, M.: Vehicle Routing Problem with Time Windows, Part I. Route Construction and Local Search Algorithms. *Transportation Science* Band 39 (2005) (1), S. 104–118.

BRONŠTEJN ET AL. 2013

Bronštejn, I. N.; Hackbusch, W.; Zeidler, E.: *Springer-Taschenbuch der Mathematik*. 3. Auflage. Wiesbaden: Springer Spektrum. 2013.

BRUNGS 2012

Brungs, F.: Der Milkrun in der Produktionslogistik. Dissertation. Darmstadt: Technische Universität Darmstadt. 2012.

BUCHHOLZ & CLAUSEN 2009

Buchholz, P.; Clausen, U.: *Große Netze der Logistik. Die Ergebnisse des Sonderforschungsbereichs 559*. Berlin, Heidelberg: Springer. 2009.

BUSSE 2013

Busse, T. D.: Modellbasierte Bewertung der Belastungsstreuung auf das logistische Systemverhalten. Institut für Fabrikanlagen und Logistik. Dissertation. Garbsen: Leibniz Universität Hannover. 2013.

CACERES-CRUZ ET AL. 2014

Caceres-Cruz, J.; Arias, P.; Guimaranas, D.; Riera, D.; Juan, A.: Rich Vehicle Routing Problem. Survey. *ACM Computing Surveys* Band 47 (2014) (2), S. 1–28.

CARIĆ ET AL. 2008

Carić, T.; Galić, A.; Fosin, J.; Gold, H.; Reinholz, A.: A Modelling and Optimization Framework for Real-World Vehicle Routing Problems. In: *Vehicle routing problem*. Hrsg. von Carić, T.; Gold, H. Rijeka: InTech. 2008, S. 15–34.

CATTARUZZA ET AL. 2014

Cattaruzza, D.; Absi, N.; Feillet, D.; Vidal, T.: A memetic algorithm for the Multi Trip Vehicle Routing Problem. *European Journal of Operational Research* Band 236 (2014) (3), S. 833–848.

CHANG & YEN 2012

Chang, T.-S.; Yen, H.-M.: City-courier routing and scheduling problems. *European Journal of Operational Research* Band 223 (2012) (2), S. 489–498.

M.-C. CHEN ET AL. 2016

Chen, M.-C.; Hsiao, Y.-H.; Reddy, R. H.; Tiwari, M. K.: The Self-Learning Particle Swarm Optimization approach for routing pickup and delivery of multiple products with material handling in multiple cross-docks. *Transportation Research Part E: Logistics and Transportation Review* Band 91 (2016), S. 208–226.

A.-L. CHEN ET AL. 2006

Chen, A.-L.; Yang, G.-K.; Wu, Z.-M.: Hybrid discrete particle swarm optimization algorithm for capacitated vehicle routing problem. *Journal of Zhejiang University-SCIENCE A* Band 7 (2006) (4), S. 607–614.

CHIANG & RUSSELL 1996

Chiang, W.-C.; Russell, R. A.: Simulated annealing metaheuristics for the vehicle routing problem with time windows. *Annals of Operations Research* Band 63 (1996) (1), S. 3–27.

CLARKE & WRIGHT 1964

Clarke, G.; Wright, J. W.: Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research: The Journal of the Operations Research Society of America* Band 12 (1964) (4), S. 568–581.

COLORNI ET AL. 1992

Colorni, A.; Dorigo, M.; Maniezzo, V.: Distributed Optimization by Ant Colonies. In: *Proceedings of ECAL91 - European Conference on Artificial Life*. Paris: Elsevier. 1992, S. 134–142.

CORMEN 2009

Cormen, T. H.: *Introduction to algorithms*. 3. Auflage. Cambridge: MIT Press. 2009.

CRÉPUT ET AL. 2007

Créput, J.-C.; Koukam, A.; Hajjam, A.: Self-Organizing Maps in Evolutionary Approach for the Vehicle Routing Problem with Time Windows. *International Journal of Computer Science and Network Security* Band 7 (2007) (1), S. 103–110.

Literatur

DALI & BOUAMAMA 2015

Dali, N.; Bouamama, S.: GPU-PSO. Parallel Particle Swarm Optimization Approaches on Graphical Processing Unit for Constraint Reasoning: Case of Max-CSPs. *Procedia Computer Science* Band 60 (2015), S. 1070–1080.

DANNINGER 2012

Danninger, M.: Ablaufplanung bei Werkstattfertigung. Ameisenalgorithmen zur Minimierung der mittleren Durchlaufzeit. Lehrstuhl für Betriebswirtschaftslehre. Dissertation. Passau: Universität Passau. 2012.

DANTZIG & RAMSER 1959

Dantzig, G. B.; Ramser, J. H.: The Truck Dispatching Problem. *Management Science* Band 6 (1959) (1), S. 80–91.

DESAULNIERS ET AL. 2014

Desaulniers, G.; Madsen, O.; Ropke, S.: The Vehicle Routing Problem with Time Windows. In: *Vehicle Routing. Problems, Methods, and Applications*. Hrsg. von Toth, P.; Vigo, D. 2. Auflage. MOS-SIAM series on optimization. Philadelphia: Society for Industrial and Applied Mathematics. 2014, S. 117–160.

DITTES 2015

Dittes, F.-M.: *Optimierung. Wie man aus allem das Beste macht*. Technik im Fokus. Berlin: Springer. 2015.

DOMBROWSKI & MIELKE 2015

Dombrowski, U.; Mielke, T., Hrsg. (2015): *Ganzheitliche Produktionssysteme. Aktueller Stand und zukünftige Entwicklungen*. VDI-Buch. Berlin: Springer. 2015.

DOMSCHKE & SCHOLL 2010

Domschke, W.; Scholl, A.: *Logistik. Rundreisen und Touren*. 5. Auflage. München: Oldenbourg. 2010.

DOMSCHKE ET AL. 2015

Domschke, W.; Drexl, A.; Klein, R.; Scholl, A.: *Einführung in Operations Research*. 9. Auflage. Berlin, Heidelberg: Springer Gabler. 2015.

DONATI ET AL. 2008

Donati, A. V.; Montemanni, R.; Casagrande, N.; Rizzoli, A. E.; Gambardella, L. M.:

Time dependent vehicle routing problem with a multi ant colony system. *European Journal of Operational Research* Band 185 (2008) (3), S. 1174–1191.

DORIGO & BLUM 2005

Dorigo, M.; Blum, C.: Ant colony optimization theory. A survey. *Theoretical Computer Science* Band 344 (2005) (2-3), S. 243–278.

DORIGO & GAMBARDELLA 1997a

Dorigo, M.; Gambardella, L. M.: Ant colonies for the travelling salesman problem. *Biosystems* Band 43 (1997) (2), S. 73–81.

DORIGO & GAMBARDELLA 1997b

Dorigo, M.; Gambardella, L. M.: Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. *IEEE Transactions on Evolutionary Computation* Band 1 (1997), S. 53–66.

DORIGO & STÜTZLE 2004

Dorigo, M.; Stützle, T.: *Ant colony optimization*. Cambridge: MIT Press. 2004.

DORIGO ET AL. 1991a

Dorigo, M.; Maniezzo, V.; Colorni, A.: Ant System: An Autocatalytic Optimizing Process. In: *Technical Report*. Band 91-016. Milano. 1991.

DORIGO ET AL. 1991b

Dorigo, M.; Maniezzo, V.; Colorni, A.: Positive Feedback as a Search Strategy. In: *Technical Report*. Band 91-016. Milano. 1991.

DORIGO ET AL. 1996

Dorigo, M.; Maniezzo, V.; Colorni, A.: Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics* Band 26 (1996) (1), S. 29–41.

DORIGO ET AL. 1999

Dorigo, M.; Di Caro, G.; Gambardella, L. M.: Ant Algorithms for Discrete Optimization. *Artificial Life* Band 5 (1999) (2), S. 137–172.

DORIGO ET AL. 2004

Dorigo, M.; Birattari, M.; Blum, C.; Gambardella, L. M.; Mondada, F.; Stützle, T.,

Literatur

Hrsg. (2004): *Ant Colony Optimization and Swarm Intelligence*. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer. 2004.

DU & SWAMY 2016

Du, K.-L.; Swamy, M. N. S.: *Search and Optimization by Metaheuristics. Techniques and Algorithms Inspired by Nature*. Basel: Springer. 2016.

DURCHHOLZ 2014

Durchholz, J.: Vorgehen zur Planung eines schlanken Logistikprozesses. Wertstromdesign für die Logistik. Lehrstuhl für Fördertechnik Materialfluss Logistik. Dissertation. München: Technische Universität München. 2014.

EBERHART & KENNEDY 1995

Eberhart, R.; Kennedy, J.: A New Optimizer Using Particle Swarm Theory. In: *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*. Nagoya. 1995, S. 39–43.

EGLESE 1990

Eglese, R. W.: Simulated annealing. A tool for operational research. *European Journal of Operational Research* Band 46 (1990) (3), S. 271–281.

EHM 2017

Ehm, J.: Integrierte Planung von Produktions- und Transportprozessen mittels gemischt-ganzzahliger Optimierung. BIBA. Dissertation. Universität Bremen. 2017.

ELLINGER ET AL. 2003

Ellinger, T.; Beuermann, G.; Leisten, R.: *Operations Research. Eine Einführung*. 6. Auflage. Berlin, Heidelberg: Springer. 2003.

ERTEL 2016

Ertel, W.: *Grundkurs Künstliche Intelligenz. Eine praxisorientierte Einführung*. 4. Auflage. Computational Intelligence. Wiesbaden: Springer Vieweg. 2016.

FEO & RESENDE 1989

Feo, T. A.; Resende, M. G.: A probabilistic heuristic for a computationally difficult set covering problem. *Operations Research Letters* Band 8 (1989) (2), S. 67–71.

FERRER ET AL. 2016

Ferrer, J. M.; Ortuño, M. T.; Tirado, G.: A GRASP metaheuristic for humanitarian aid distribution. *Journal of Heuristics* Band 22 (2016) (1), S. 55–87.

FESTA 2014

Festa, P.: A brief introduction to exact, approximation, and heuristic algorithms for solving hard combinatorial optimization problems. In: *16th International Conference on Transparent Optical Networks (ICTON)*. Graz. 2014, S. 1–20.

FINK 2017

Fink, A.: Simulated Annealing. In: *Enzyklopädie der Wirtschaftsinformatik*. Hrsg. von Gronau, N.; Becker, J.; Sinz, E.; Suhl, L.; Leimeister, J. M. URL: <http://www.enzyklopaedie-der-wirtschaftsinformatik.de/lexikon/technologien-methoden/Operations-Research/Metaheuristik/Simulated-Annealing> (besucht am 25.01.2017).

GAJPAL & ABAD 2009

Gajpal, Y.; Abad, P.L.: Multi-ant colony system (MACS) for a vehicle routing problem with backhauls. *European Journal of Operational Research* Band 196 (2009) (1), S. 102–117.

GAMBARDELLA ET AL. 1999

Gambardella, L. M.; Taillard, É.; Agazzi, G.: MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows. In: *New Ideas in Optimization*. Hrsg. von Corne, D.; Dorigo, M.; Glover, F. London: McGraw-Hill. 1999, S. 63–76.

GERDES ET AL. 2004

Gerdes, I.; Klawonn, F.; Kruse, R.: *Evolutionäre Algorithmen. Genetische Algorithmen – Strategien und Optimierungsverfahren – Beispielanwendungen*. Wiesbaden: Vieweg. 2004.

GERDTS & LEMPIO 2011

Gerdts, M.; Lempio, F.: *Mathematische Optimierungsverfahren des Operations Research*. Berlin, Boston: De Gruyter. 2011.

GHAZIRI & OSMAN 2006

Ghaziri, H.; Osman, I. H.: Self-organizing feature maps for the vehicle routing problem with backhauls. *Journal of Scheduling* Band 9 (2006) (2), S. 97–114.

GLOGER 2011

Gloger, B.: *Scrum. Produkte zuverlässig und schnell entwickeln*. 3. Auflage. München: Hanser. 2011.

GLOVER 1986

Glover, F.: Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research* Band 13 (1986) (5), S. 533–549.

GLOVER 1990

Glover, F.: Tabu Search. A Tutorial. *Interfaces* Band 20 (1990) (4), S. 74–94.

GLOVER ET AL. 2000

Glover, F.; Laguna, M.; Martí, R.: Fundamentals of scatter search and path relinking. *Control and Cybernetics* Band 29 (2000) (3), S. 653–684.

GONG ET AL. 2012

Gong, Y.-J.; Zhang, J.; Liu, O.; Huang, R.-Z.; Chung, H. S.-H.; Shi, Y.-H.: Optimizing the Vehicle Routing Problem With Time Windows. A Discrete Particle Swarm Optimization Approach. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)* Band 42 (2012) (2), S. 254–267.

GOSS ET AL. 1989

Goss, S.; Aron, S.; Deneubourg, J. L.; Pasteels, J. M.: Self-organized shortcuts in the Argentine ant. *Naturwissenschaften* Band 76 (1989) (12), S. 579–581.

GRASSÉ 1959

Grassé, P.-P.: La Reconstruction du Nid et les Coordinations Interindividuelles chez *Bellicositermes Natalensis* et *Cubitermes Sp.* La Théorie de la Stigmergie: Essai d'Interprétation du Comportement des Termites Constructeurs. (Deutsch: Nestbau und Koordination der Individuen bei *Bellicositermes Natalensis* und *Cubitermes Sp.* Die Theorie der Stigmergie: Interpretationsversuch des Verhaltens von Nestbauenden Termiten). *Insectes Sociaux* Band 6 (1959) (1), S. 41–80.

GREEN 1991

Green, S. B.: How Many Subjects Does It Take to Do a Regression Analysis. *Multivariate behavioral research* Band 26 (1991) (3), S. 499–510.

GUDEHUS 2012

Gudehus, T.: *Logistik 1. Grundlagen, Verfahren und Strategien*. 4. Auflage. VDI-Buch. Berlin u.a.: Springer. 2012.

GÜNTHER 2006

Günther, P.: Eine Branche entdeckt ihre Potentiale. In: *Intralogistik. Potentiale, Perspektiven, Prognosen*. Hrsg. von Arnold, D. VDI. Berlin, Heidelberg, New York: Springer. 2006, S. 5–51.

GÜNTHNER 2007

Günthner, W. A.: *Neue Wege in der Automobillogistik. Die Vision der Supra-Adaptivität*. VDI-Buch. Berlin, Heidelberg: Springer. 2007.

GÜNTHNER & BOPPERS 2013

Günthner, W. A.; Boppert, J.: *Lean Logistics. Methodisches Vorgehen und praktische Anwendung in der Automobilindustrie*. Berlin, Heidelberg: Springer Vieweg. 2013.

GÜNTHNER ET AL. 2002

Günthner, W. A.; Klenk, E.; Galka, S.; Dewitz, M.: Stand und Entwicklung von Routenzugsystemen für den innerbetrieblichen Materialtransport. Ergebnisse einer Studie. München: Technische Universität München. 2002.

GYULAI ET AL. 2013

Gyulai, D.; Pfeiffer, A.; Sobottka, T.; Váncza, J.: Milkrun Vehicle Routing Approach for Shop-floor Logistics. *Procedia CIRP* Band 7 (2013), S. 127–132.

HAGHANI & JUNG 2005

Haghani, A.; Jung, S.: A dynamic vehicle routing problem with time-dependent travel times. *Computers & Operations Research* Band 32 (2005) (11), S. 2959–2986.

HAGHIGHI ET AL. 2009

Haghighi, M. S.; Hadi Zahedi, M.; Mojtaba Rouhani, S.: A multi level priority clustering NN based approach for solving heterogeneous vehicle routing problem. In: *2009*

Literatur

IEEE Instrumentation and Measurement Technology Conference (I2MTC). Singapur. 2009, S. 224–229.

HANSEN & MLADENović 2005

Hansen, P; Mladenović, N.: Variable Neighborhood Search. In: *Search Methodologies*. Hrsg. von Burke, E. K.; Kendall, G. Boston: Springer. 2005, S. 211–238.

HANSEN & MLADENović 2009

Hansen, P; Mladenović, N.: Variable Neighborhood Search Methods. In: *Encyclopedia of Optimization*. Hrsg. von Floudas, C. A.; Pardalos, P. M. Boston: Springer. 2009, S. 3975–3989.

HARTL 2014

Hartl, R.: *Variable Neighborhood Search*. Hrsg. von Gronau, N.; Becker, J.; Sinz, E.; Suhl, L.; Leimeister, J. M. URL: <http://www.enzyklopaedie-der-wirtschaftsinformatik.de/lexikon/technologien-methoden/Operations-Research/Metaheuristik/Variable-Neighborhood-Search> (besucht am 24.01.2017).

HEPPNER & GRENANDER 1990

Hepppner, F; Grenander, U.: A stochastic nonlinear model for coordinated bird flocks. *The ubiquity of chaos* (1990), S. 233–238.

HO & SZETO 2014

Ho, S. C.; Szeto, W. Y.: Solving a static repositioning problem in bike-sharing systems using iterated tabu search. *Transportation Research Part E: Logistics and Transportation Review* Band 69 (2014), S. 180–198.

HÖLLDOBLER & WILSON 2011

Hölldobler, B.; Wilson, E. O.: *Blattschneiderameisen – der perfekte Superorganismus*. Berlin, Heidelberg u.a.: Springer. 2011.

HOPFIELD 1982

Hopfield, J. J.: Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America* Band 79 (1982) (8), S. 2554–2558.

INCOSE o.D.

INCOSE, Hrsg. (o.D.): *What is Systems Engineering?* URL: <http://www.incose.org/AboutSE/WhatIsSE> (besucht am 14.09.2017).

IRNICH ET AL. 2014

Irnich, S.; Toth, P.; Vigo, D.: The Family of Vehicle Routing Problems. In: *Vehicle Routing. Problems, Methods, and Applications*. Hrsg. von Toth, P.; Vigo, D. 2. Auflage. MOS-SIAM series on optimization. Philadelphia: Society for Industrial and Applied Mathematics. 2014, S. 1–33.

JAEGERE ET AL. 2014

Jaegere, N. de; Defraeye, M.; van Nieuwenhuysse, I.: The vehicle routing problem: state of the art classification and review. Faculty of Economics and Business. Diskussionspapier. Löwen: Ku Leuven. 2014.

KALAYCI & KAYA 2016

Kalayci, C. B.; Kaya, C.: An ant colony system empowered variable neighborhood search algorithm for the vehicle routing problem with simultaneous pickup and delivery. *Expert Systems with Applications* Band 66 (2016), S. 163–175.

KALLRATH 2013

Kallrath, J.: *Gemischt-ganzzahlige Optimierung: Modellierung in der Praxis. Mit Fallstudien aus Chemie, Energiewirtschaft, Metallgewerbe, Produktion und Logistik*. 2. Auflage. Wiesbaden: Springer Spektrum. 2013.

KARA ET AL. 2008

Kara, İ.; Yetiş Kara, B.; Kadri Yetiş, M.: Cumulative Vehicle Routing Problems. In: *Vehicle routing problem*. Hrsg. von Carić, T.; Gold, H. Rijeka: InTech. 2008, S. 85–98.

KEPAPTSOGLOU ET AL. 2015

Keaptsoglou, K.; Fountas, G.; Karlaftis, M. G.: Weather impact on containership routing in closed seas. A chance-constraint optimization approach. *Transportation Research Part C: Emerging Technologies* Band 55 (2015), S. 139–155.

KINDERVATER & SAVELSBERGH 1997

Kindervater, G.; Savelsbergh, M.: Vehicle Routing. Handling Edge Exchanges. In: *Local Search in Combinatorial Optimization*. Hrsg. von Aarts, E.; Lenstra, J. Wiley, Chichester. 1997, S. 337–360.

Literatur

KISTNER 2003

Kistner, K.-P.: *Optimierungsmethoden. Einführung in die Unternehmensforschung für Wirtschaftswissenschaftler*. 3. Auflage. Heidelberg: Springer. 2003.

KLENK & GALKA 2013

Klenk, E.; Galka, S.: Das Hauptziel: Dynamische und flexible Systeme. *Hebezeuge Fördermittel* Band 53 (2013) (7-8), S. 386–388.

KLUG 2018

Klug, F.: *Logistikmanagement in der Automobilindustrie*. Berlin, Heidelberg: Springer. 2018.

KONTORAVDIS & BARD 1995

Kontoravdis, G.; Bard, J. F.: A GRASP for the Vehicle Routing Problem with Time Windows. *ORSA Journal on Computing* Band 7 (1995) (1), S. 10–23.

KORTE & VYGEN 2012

Korte, B.; Vygen, J.: *Kombinatorische Optimierung. Theorie und Algorithmen*. 2. Auflage. Berlin: Springer. 2012.

LABADIE ET AL. 2012

Labadie, N.; Mansini, R.; Melechovský, J.; Wolfler Calvo, R.: The Team Orienteering Problem with Time Windows. An LP-based Granular Variable Neighborhood Search. *European Journal of Operational Research* Band 220 (2012) (1), S. 15–27.

LABADIE ET AL. 2016

Labadie, N.; Prins, C.; Prodhon, C.: *Metaheuristics for vehicle routing problems*. Band 3. Metaheuristics Set. London, Hoboken: ISTE & Wiley. 2016.

LACKNER 2004

Lackner, A.: Dynamische Tourenplanung mit ausgewählten Metaheuristiken. Eine Untersuchung am Beispiel des kapazitätsrestriktiven dynamischen Tourenplanungsproblems mit Zeitfenstern. Institut für Wirtschaftsinformatik. Dissertation. Göttingen: Georg-August-Universität. 2004.

LAHYANI ET AL. 2015

Lahyani, R.; Khemakhem, M.; Semet, F.: Rich vehicle routing problems. From a

taxonomy to a definition. *European Journal of Operational Research* (2015) (241), S. 1–14.

LAU ET AL. 2003

Lau, H. C.; Sim, M.; Teo, K. M.: Vehicle routing problem with time windows and a limited number of vehicles. *European Journal of Operational Research* Band 148 (2003) (3), S. 559–569.

LENSING & SONNEMANN 1995

Lensing, M.; Sonnemann, K.: *Materialwirtschaft und Einkauf*. Praxis der Unternehmensführung. Wiesbaden: Gabler. 1995.

LENSTRA & KAN 1981

Lenstra, J.; Kan, A. H. G. R.: Complexity of vehicle routing and scheduling problems. *Networks* Band 11 (1981) (2), S. 221–227.

LEUNG ET AL. 2011

Leung, S. C.; Zhou, X.; Zhang, D.; Zheng, J.: Extended guided tabu search and a new packing algorithm for the two-dimensional loading vehicle routing problem. *Computers & Operations Research* Band 38 (2011) (1), S. 205–215.

LIEB ET AL. 2017

Lieb, C.; Klenk, E.; Galka, S.; Keuntje, C.: Einsatz von Routenzugsystemen zur Produktionsversorgung. Studie zu Planung, Steuerung und Betrieb. Lehrstuhl für Fördertechnik Materialfluss Logistik. München. 2017.

LIEBEROTH-LEDEN ET AL. 2017

Lieberoth-Leden, C.; Röschinger, M.; Lechner, J.; Günthner, W. A.: Logistik 4.0. In: *Handbuch Industrie 4.0. Geschäftsmodelle, Prozesse, Technik*. Hrsg. von Reinhart, G. München: Hanser. 2017, S. 451–512.

C. LIN ET AL. 2014

Lin, C.; Choy, K.; Ho, G.; Lam, H.; Pang, G.; Chin, K.: A decision support system for optimizing dynamic courier routing operations. *Expert Systems with Applications* Band 41 (2014) (15), S. 6917–6933.

S. LIN & KERNIGHAN 1973

Lin, S.; Kernighan, B. W.: An effective heuristic algorithm for the traveling-salesman

problem. *Operations Research: The Journal of the Operations Research Society of America* Band 21 (1973) (2), S. 498–516.

S. LIN ET AL. 2011

Lin, S.; Yu, V.; Lu, C.-C.: A simulated annealing heuristic for the truck and trailer routing problem with time windows. *Expert Systems with Applications* Band 38 (2011) (12), S. 15244–15252.

LÜBBECKE ET AL. 2014

Lübbecke, M.; Weiler, A.; Werners, B.: *Zukunftsperspektiven des Operations Research. Erfolgreicher Einsatz und Potenziale*. Wiesbaden: Springer Gabler. 2014.

MARTÍ ET AL. 2005

Martí, R.; Laguna, M.; Campos, V.: Scatter Search vs. Genetic Algorithms. In: *Meta-heuristic Optimization via Memory and Evolution*. Hrsg. von Sharda, R.; Voß, S.; Rego, C.; Alidaee, B. Operations Research/Computer Science Interfaces Series. Boston: Kluwer Academic. 2005, S. 263–282.

MARTÍ ET AL. 2006

Martí, R.; Laguna, M.; Glover, F.: Principles of scatter search. *European Journal of Operational Research* Band 169 (2006) (2), S. 359–372.

MARTIN 2014

Martin, H.: *Transport- und Lagerlogistik. Planung, Struktur, Steuerung und Kosten von Systemen der Intralogistik*. 9. Auflage. Wiesbaden: Springer Vieweg. 2014.

MATTFELD & VAHRENKAMP 2014

Mattfeld, D.; Vahrenkamp, R.: *Logistiknetzwerke. Modelle für Standortwahl und Tourenplanung*. 2. Auflage. Wiesbaden: Springer Gabler. 2014.

MCKINSEY 2015

McKinsey: Industry 4.0. How to navigate digitization of the manufacturing sector. URL: <https://www.mckinsey.com/business-functions/operations/our-insights/industry-four-point-o-how-to-navigae-the-digitization-of-the-manufacturing-sector>.

MERKLE 2002

Merkle, D.: Ameisenalgorithmen. Optimierung und Modellierung. Dissertation. Karlsruhe: Universität Fridericiana zu Karlsruhe. 2002.

A. MEYER 2017

Meyer, A.: Milk Run Design. Definitions, Concepts and Solution Approaches. Institut für Fördertechnik und Logistiksysteme. Dissertation. Karlsruher Institut für Technologie. 2017.

B. MEYER & ERNST 2004

Meyer, B.; Ernst, A.: Integrating ACO and Constraint Propagation. In: *Ant Colony Optimization and Swarm Intelligence*. Hrsg. von Dorigo, M.; Birattari, M.; Blum, C.; Gambardella, L. M.; Mondada, F.; Stützle, T. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer. 2004, S. 166–177.

MICHALEWICZ 1996

Michalewicz, Z.: *Genetic Algorithms + Data Structures = Evolution Programs*. 3. Auflage. Berlin, New York: Springer. 1996.

MITCHELL 1998

Mitchell, M.: L.D. Davis, handbook of genetic algorithms. *Artificial Intelligence* Band 100 (1998) (1-2), S. 325–330.

MLADENović & HANSEN 1997

Mladenović, N.; Hansen, P.: Variable neighborhood search. *Computers & Operations Research* Band 24 (1997) (11), S. 1097–1100.

MONTEMANNI ET AL. 2005

Montemanni, R.; Gambardella, L. M.; Rizzoli, A. E.; Donati, A. V.: Ant Colony System for a Dynamic Vehicle Routing Problem. *Journal of Combinatorial Optimization* Band 10 (2005) (4), S. 327–343.

MURATA & ITAI 2008

Murata, T.; Itai, R.: Enhancing Solution Similarity in Multi-Objective Vehicle Routing Problems with Different Demand Periods. In: *Vehicle routing problem*. Hrsg. von Carić, T.; Gold, H. Rijeka: InTech. 2008, S. 99–112.

Literatur

NIEHUES 2017

Niehues, M.: Adaptive Produktionssteuerung für Werkstattfertigungssysteme durch fertigungsbegleitende Reihenfolgebildung. Institut für Werkzeugmaschinen und Betriebswissenschaften. Dissertation. München: Technische Universität München. 2017.

NOROUZI ET AL. 2015

Norouzi, N.; Sadegh-Amalnick, M.; Alinaghiyan, M.: Evaluating of the particle swarm optimization in a periodic vehicle routing problem. *Measurement* Band 62 (2015), S. 162–169.

NOVAES ET AL. 2015

Novaes, A. G.; Bez, E. T.; Burin, P. J.; Aragão, D. P.: Dynamic milk-run OEM operations in over-congested traffic conditions. *Computers & Industrial Engineering* Band 88 (2015), S. 326–340.

OLTERSDORFF 2016

Oltersdorff, H.: Ein gutes Doppel. User Experience Design und Scrum. *Wirtschaftsinformatik & Management* Band 8 (2016) (6), S. 60–65.

ÕNO 2013

Õno, T.: *Das Toyota-Produktionssystem*. 3. Auflage. Frankfurt am Main, New York: Campus. 2013.

OSMAN 1993

Osman, I. H.: Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research* Band 41 (1993) (4), S. 421–451.

PAPAGEORGIU ET AL. 2015

Papageorgiou, M.; Leibold, M.; Buss, M.: *Optimierung. Statische, dynamische, stochastische Verfahren für die Anwendung*. 4. Auflage. Berlin, Heidelberg: Springer. 2015.

PARDALOS & MAVRIDOU 2009

Pardalos, P. M.; Mavridou, T. D.: Simulated Annealing. In: *Encyclopedia of Optimization*. Hrsg. von Floudas, C. A.; Pardalos, P. M. Boston: Springer. 2009, S. 3591–3593.

PFOHL 2004

Pfohl, H.-C.: *Logistikmanagement. Konzeption und Funktionen*. 2. Auflage. Berlin u.a.: Springer. 2004.

PINO ET AL. 2012

Pino, R.; Martínez, C.; Villanueva, V.; Priore, P.; Fernández, I.: Application of GRASP methodology to Vehicle Routing Problem (VRP). In: *Proceedings on the International Conference on Artificial Intelligence (ICAI)*. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp). 2012.

PIRIM ET AL. 2008

Pirim, H.; Bayraktar, E.; Eksioğlu, B.: Tabu Search: A Comparative Study. In: *Tabu Search*. Hrsg. von Jaziri, W. I-Tech. 2008, S. 1–28.

POPPER 2017

Popper, K.: Die Logik der Sozialwissenschaften. *Kölner Zeitschrift für Soziologie und Sozialpsychologie* Band 69 (2017) (S1), S. 215–228.

PRAIS ET AL. 1998

Prais, M.; Ribeiro, C. C.; Celso; Ribeiro, C.: Reactive GRASP: An Application To A Matrix Decomposition Problem In TDMA Traffic Assignment. *INFORMS Journal on Computing* Band 12 (1998), S. 164–176.

PRINS 2004

Prins, C.: A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research* Band 31 (2004) (12), S. 1985–2002.

RABE 2008

Rabe, M.: *Verifikation und Validierung für die Simulation in Produktion und Logistik. Vorgehensmodelle und Techniken*. VDI-Buch. Berlin, Heidelberg: Springer. 2008.

REGO & LEÃO 2005

Rego, C.; Leão, P.: A Scatter Search Tutorial for Graph-Based Permutation Problems. In: *Metaheuristic Optimization via Memory and Evolution*. Hrsg. von Sharda, R.; Voß, S.; Rego, C.; Alidaee, B. Band 30. Operations Research/Computer Science Interfaces Series. Boston: Kluwer Academic. 2005, S. 1–24.

Literatur

REYNOLDS 1987

Reynolds, C. W.: Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH computer graphics* Band 21 (1987) (4), S. 25–34.

RICHTER 2005

Richter, A.: Dynamische Tourenplanung. Modifikation von klassischen Heuristiken für das Dynamische Rundreiseproblem (DTSP) und das Dynamische Tourenplanungsproblem (DVRP) mit der Möglichkeit der Änderung des aktuellen Fahrzeugzuges. Dissertation. Dresden: Technische Universität Dresden. 2005.

RIMSCHA 2014

Rimscha, M.: *Algorithmen kompakt und verständlich. Lösungsstrategien am Computer*. Wiesbaden: Springer Vieweg. 2014.

ROTHER & SHOOK 2000

Rother, M.; Shook, J.: *Sehen lernen. Mit Wertstromdesign die Wertschöpfung erhöhen und Verschwendung beseitigen*. Stuttgart: LOG_X. 2000.

RUBIN 2014

Rubin, K. S.: *Essential Scrum. Umfassendes Scrum-Wissen aus der Praxis*. Heidelberg u.a.: mitp. 2014.

RÜSCHENDORF 2014

Rüschendorf, L.: *Mathematische Statistik*. Berlin, Heidelberg: Springer Spektrum. 2014.

RUSSELL & CHIANG 2006

Russell, R. A.; Chiang, W.-C.: Scatter search for the vehicle routing problem with time windows. *European Journal of Operational Research* Band 169 (2006) (2), S. 606–622.

SAGER 2019

Sager, B.: Konfiguration globaler Produktionsnetzwerke. Institut für Werkzeugmaschinen und Betriebswissenschaften. Dissertation. München: Technische Universität München. 2019.

SAVELSBERGH 1985

Savelsbergh, M.: Local search in routing problems with time windows. *Annals of Operations Research* Band 4 (1985) (1), S. 285–305.

SCHMIDT ET AL. 2016

Schmidt, T.; Meinhardt, I.; Schulze, F.: New design guidelines for Inplant milk-run systems. In: *Proceedings of the 14th International Material Handling Research Colloquium*. Karlsruhe. 2016.

SCHOLZ 2010

Scholz, D.: Innerbetriebliche Standortplanung. Das Konzept der Slicing Trees bei der Optimierung von Layoutstrukturen. Fachgebiet für Operations Research. Dissertation. Wiesbaden: Technische Universität Darmstadt. 2010.

SCHUH 2012

Schuh, G., Hrsg. (2012): *Innovationsmanagement*. 2. Auflage. Band 3. Handbuch Produktion und Management. Berlin, Heidelberg: Springer. 2012.

SCHYNS 2015

Schyns, M.: An ant colony system for responsive dynamic vehicle routing. *European Journal of Operational Research* Band 245 (2015) (3), S. 704–718.

SEECK 2010

Seeck, S.: *Erfolgsfaktor Logistik. Klassische Fehler erkennen und vermeiden*. Wiesbaden: Gabler. 2010.

SEITZ 2016

Seitz, K.-F.: Modellierung von Transportprozessen in Lieferketten. Institut für Fabrikanlagen und Logistik. Dissertation. Leibniz Universität Hannover. 2016.

SILBERHOLZ & GOLDEN 2010

Silberholz, J.; Golden, B.: Comparison of Metaheuristics. In: *Handbook of Metaheuristics*. Hrsg. von Gendreau, M.; Potvin, J.-Y. 2. Auflage. International Series in Operations Research & Management Science 146. Boston: Springer. 2010, S. 625–640.

J. SINGH & H. SINGH 2015

Singh, J.; Singh, H.: Continuous improvement philosophy – literature review and directions. *Benchmarking: An International Journal* Band 22 (2015) (1), S. 75–119.

S. SINGH & SHARMA 2006

Singh, S.; Sharma, R.: A review of different approaches to the facility layout problems. *The International Journal of Advanced Manufacturing Technology* Band 30 (2006) (5-6), S. 425–433.

SOLOMON 1987

Solomon, M.: Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. *Operations Research* Band 35 (1987) (2), S. 254–265.

SPOUN 1998

Spoun, S.: *Universität und Praxis. Tendenzen und Perspektiven wissenschaftlicher Verantwortung für Wirtschaft und Gesellschaft*. Der Universität St. Gallen zum 100-Jahr-Jubiläum. Zürich: NZZ. 1998.

STÄHLER 2001

Stähler, P.: Geschäftsmodelle in der digitalen Ökonomie. Merkmale, Strategien und Auswirkungen. Dissertation. St. Gallen. 2001.

STEINHAUS 2015

Steinhaus, M.: The Application of the Self Organizing Map to the Vehicle Routing Problem. Dissertation. University of Rhode Island. 2015.

STEPANENKO 2008

Stepanenko, S.: Global optimization methods based on Tabu search. GTS, GOTS, TSPA, application for conformation searches. Institut für Organische Chemie. Würzburg: Julius-Maximilians-Universität Würzburg. 2008.

STÜTZLE & HOOS 1997

Stützle, T.; Hoos, H.: MAX-MIN Ant System and Local Search for the Traveling Salesman Problem. In: *IEEE International Conference on Evolutionary Computation (ICEC '97)*. 1997, S. 309–314.

STÜTZLE & HOOS 2000

Stützle, T.; Hoos, H.: MAX-MIN Ant System. *Future Generation Computer Systems* Band 16 (2000) (8), S. 889–914.

SUTHERLAND 2015

Sutherland, J.: *Die Scrum-Revolution. Management mit der bahnbrechenden Methode der erfolgreichsten Unternehmen*. Frankfurt: Campus. 2015.

SZYMON & DOMINIK 2013

Szymon, J.; Dominik, Ż.: Solving Multi-criteria Vehicle Routing Problem by Parallel Tabu Search on GPU. *Procedia Computer Science* Band 18 (2013), S. 2529–2532.

TAILLARD ET AL. 1997

Taillard, É.; Badeau, P.; Gendreau, M.; Guertin, F.; Potvin, J.-Y.: A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows. *Transportation Science* Band 31 (1997) (2), S. 170–186.

TALBI 2009

Talbi, E.-G.: *Metaheuristics. From design to implementation*. Hoboken: Wiley. 2009.

TANG ET AL. 2010

Tang, J.; Zhang, J.; Pan, Z.: A scatter search algorithm for solving vehicle routing problem with loading cost. *Expert Systems with Applications* Band 37 (2010) (6), S. 4073–4083.

TESCHEMACHER & REINHART 2016

Teschemacher, U.; Reinhart, G.: Enhancing Constraint Propagation in ACO-based Schedulers for Solving the Job Shop Scheduling Problem. *Procedia CIRP* Band 41 (2016), S. 443–447.

TESCHEMACHER & REINHART 2017

Teschemacher, U.; Reinhart, G.: Ant Colony Optimization Algorithms to Enable Dynamic Milkrun Logistics. *Procedia CIRP* Band 63 (2017), S. 762–767.

TURAN ET AL. 2017

Turan, B.; Minner, S.; Hartl, R.: A VNS approach to multi-location inventory redistribution with vehicle routing. *Computers & Operations Research* Band 78 (2017), S. 526–536.

Literatur

TURAU & WEYER 2015

Turau, V.; Weyer, C.: *Algorithmische Graphentheorie*. 4. Auflage. Berlin: De Gruyter. 2015.

ULRICH & HILL 1976

Ulrich, P.; Hill, W.: Wissenschaftstheoretische Grundlagen der Betriebswirtschaftslehre. *Wirtschaftswissenschaftliches Studium: Zeitschrift für Ausbildung und Hochschulkontakt* Band 5 (1976) (7+8), S. 304–309.

URBAN & MAYERL 2011

Urban, D.; Mayerl, J.: *Regressionsanalyse. Theorie, Technik und Anwendung*. 4. Auflage. Wiesbaden: Springer Fachmedien. 2011.

VDI-RICHTLINIE 5586-1 2016

VDI-Richtlinie 5586-1: *Routenzugsysteme - Grundlagen, Gestaltung und Praxisbeispiele. Blatt 1. Entwurf*. Band ICS 53.060. Berlin: Beuth. 2016.

VDI-RICHTLINIE 5586-2 2016

VDI-Richtlinie 5586-2: *Routenzugsysteme - Grundlagen, Gestaltung und Praxisbeispiele. Blatt 2. Entwurf*. Band ICS 53.060. Berlin: Beuth. 2016.

VERHEYEN 2017

Verheyen, G.: *Scrum Taschenbuch*. Zaltbommel: Van Haren. 2017.

VIDAL ET AL. 2013

Vidal, T.; Crainic, T. G.; Gendreau, M.; Prins, C.: Heuristics for multi-attribute vehicle routing problems. A survey and synthesis. *European Journal of Operational Research* Band 231 (2013) (1), S. 1–21.

VOGEL-HEUSER ET AL. 2017

Vogel-Heuser, B.; Bauernhansl, T.; Hompel, M. ten, Hrsg. (2017): *Handbuch Industrie 4.0 Bd. 2. Automatisierung*. 2. Auflage. Berlin: Springer Vieweg. 2017.

VOSS 2009

Voß, S.: Metaheuristics. In: *Encyclopedia of Optimization*. Hrsg. von Floudas, C. A.; Pardalos, P. M. Boston: Springer. 2009, S. 2061–2075.

WANG ET AL. 2015

Wang, C.; Mu, D.; Zhao, F.; Sutherland, J. W.: A parallel simulated annealing method

for the vehicle routing problem with simultaneous pickup–delivery and time windows. *Computers & Industrial Engineering* Band 83 (2015), S. 111–122.

WANNENWETSCH 2014

Wannenwetsch, H.: *Integrierte Materialwirtschaft, Logistik und Beschaffung*. 5. Auflage. Berlin: Springer Vieweg. 2014.

WATANABE & SAKAKIBARA 2008

Watanabe, S.; Sakakibara, K.: A Multiobjectivization Approach for Vehicle Routing Problems. In: *Vehicle routing problem*. Hrsg. von Carić, T.; Gold, H. Rijeka: InTech. 2008, S. 113–124.

WEHKING & POPP 2015

Wehking, K.-H.; Popp, J.: Changes in production logistics for automobile manufacturing. In: *15. Internationales Stuttgarter Symposium. Automobil- und Motorentechnik*. Hrsg. von Bargende, M.; Reuss, H.-C.; Wiedemann, J. Wiesbaden: Springer Vieweg. 2015, S. 395–402.

WEICKER 2015

Weicker, K.: *Evolutionäre Algorithmen*. 3. Auflage. Wiesbaden: Springer Fachmedien. 2015.

WEISE ET AL. 2009

Weise, T.; Podlich, A.; Gorltd, C.: Solving Real-World Vehicle Routing Problems with Evolutionary Algorithms. In: *Natural Intelligence for Scheduling, Planning and Packing Problems*. Hrsg. von Chiong, R.; Dhakal, S. Berlin, Heidelberg: Springer. 2009, S. 29–53.

WENDT 1995

Wendt, O.: Tourenplanung durch Einsatz naturanaloger Verfahren. Integration von Genetischen Algorithmen und Simulated Annealing. Fachbereich Wirtschaftswissenschaften. Dissertation. Frankfurt: Universität Frankfurt am Main. 1995.

WENGER 2010

Wenger, W.: *Multikriterielle Tourenplanung*. Wiesbaden: Gabler. 2010.

Literatur

WERNERS 2013

Werners, B.: *Grundlagen des Operations Research*. 3. Auflage. Berlin, Heidelberg: Springer Gabler. 2013.

WESTKÄMPER & LÖFFLER 2016

Westkämper, E.; Löffler, C.: *Strategien der Produktion. Technologien, Konzepte und Wege in die Praxis*. Heidelberg: Springer Vieweg. 2016.

WIENDAHL 2014

Wiendahl, H.-P.: *Betriebsorganisation für Ingenieure*. 8. Auflage. München: Hanser. 2014.

WINSTON & GOLDBERG 2004

Winston, W. L.; Goldberg, J. B.: *Operations research. Applications and algorithms*. 4. Auflage. Belmont: Thomson/Brooks/Cole. 2004.

WINZER 2016

Winzer, P.: *Generic systems engineering. Ein methodischer Ansatz zur Komplexitätsbewältigung*. 2. Auflage. Berlin, Heidelberg: Springer Vieweg. 2016.

WOLPERT & MACREADY 1997

Wolpert, D. H.; Macready, W. G.: No Free Lunch Theorems for Optimization. *IEEE Transactions on Evolutionary Computation* Band 1 (1997) (1), S. 67–82.

YU ET AL. 2011

Yu, S.; Ding, C.; Zhu, K.: A hybrid GA-TS algorithm for open vehicle routing optimization of coal mines material. *Expert Systems with Applications* Band 38 (2011) (8), S. 10568–10573.

ZHANG ET AL. 2012

Zhang, T.; Chaovalitwongse, W. A.; Zhang, Y.: Scatter search for the stochastic travel-time vehicle routing problem with simultaneous pick-ups and deliveries. *Computers & Operations Research* Band 39 (2012) (10), S. 2277–2290.

ZIMMERMANN & GUTSCHE 1991

Zimmermann, H.-J.; Gutsche, L.: *Multi-Criteria Analyse. Einführung in die Theorie der Entscheidungen bei Mehrfachzielsetzungen*. Berlin, Heidelberg: Springer. 1991.

Abbildungsverzeichnis

1.1	Das logistische Zieldreieck (ARNDT 2015, S. 81)	4
1.2	Wissenschaftssystematik nach ULRICH & HILL (1976, S. 305)	10
1.3	Vorgehensmodell SIMILAR-Process (in Anlehnung an BAHILL & GIS- SING 1998, S. 518)	13
1.4	Aufbau der Arbeit	15
2.1	Abgrenzung der innerbetrieblichen Logistik (in Anlehnung an GUDE- HUS 2012, S. 5)	18
2.2	Die sieben Arten der Verschwendung in der Logistik (nach GÜNTNER & BOPPERT 2013)	19
2.3	Modellierung des VRP als Graph	29
2.4	Ableitung der Adjazenzmatrix aus dem gewichteten Graphen	30
4.1	Klassifizierung der für die Tourenplanung relevanten Optimierungs- methoden (in Anlehnung an CACERES-CRUZ ET AL. 2014, S. 32:7)	44
4.2	Schematisches Vorgehen der Savings-Heuristik (in Anlehnung an RICHTER 2005, S. 46)	48
4.3	Moves für lokale Suchverfahren	50
4.4	Unterschiedliche Anordnungen der Stationen im Raum bei den Pro- blemklassen R, RC und C	64
5.1	Zeitliche Bestandteile einer Routenzugfahrt (aufbauend auf BUCH- HOLZ & CLAUSEN 2009, S. 162)	70
5.2	Arbeitsschritte zur Durchführung einer Regressionsanalyse (in An- lehnung an URBAN & MAYERL 2011, S. 22 f.)	73
5.3	Modellierung des VRP mit duplizierten Depots	75
5.4	Vorgehen der Ameisen bei der Futtersuche vom unbekanntem Terrain (links) zu voll ausgebildeten kurzen Wegen (rechts) (aufbauend auf GOSS ET AL. 1989, S. 579)	77

5.5	Zusammenspiel der parallel optimierenden Algorithmen zur Realisierung von zwei Zielfunktionen	86
5.6	Zusammenstellung der Touren durch den Ameisenalgorithmus im Rahmen der Optimierungsläufe	87
5.7	Erweiterter Ablauf der Tourenkonstruktion im Ameisenalgorithmus bei zyklischer Wiederverwendung von Transportfahrzeugen	88
5.8	Integration der Eilaufträge in das standardmäßige Vorgehen	95
5.9	Lokale Optimierungsverfahren für den Einsatz in der fabrikinternen Logistik	99
5.10	Schematische Darstellung der Einbettung des Optimierungsmodells in das Produktionsumfeld	105
5.11	Exemplarische Darstellung der Tour-Informationen für den Fahrer	108
6.1	Konzept zur Prädiktion von Materialabrufen	112
6.2	Visualisierung der Fortpflanzung der Wahrscheinlichkeiten	115
6.3	Faltung der Dichtefunktion $f(t)$ mit der Rechteckfunktion $h(t)$	115
6.4	Vorgehen beim Aufsummieren der einzelnen erwarteten Kapazitäten	116
6.5	Vorgehen und Integration der Prognose in den Standardprozess	117
6.6	Ableitung der Bedarfe an Routenzügen aus der Kapazitätsprognose	118
6.7	Prinzip der Vorverlagerung von Bestellungen zur Abminderung von Kapazitätsspitzen	119
7.1	Absicherung der Korrektheit durch Verifikation und Validierung	121
7.2	Ausschnitt aus dem Simulationsprogramm mit dem in Abschnitt 7.4 beschriebenen Anwendungsbeispiel	124
7.3	Gantt-Chart zur Bewertung der Auslastung der einzelnen Routenzüge	125
7.4	Ablauf eines Scrum-Projekts (in Anlehnung an OLTERS DORFF 2016, S. 62)	125
7.5	Erwartetes Verhalten für Aufträge mit Eilaufträgen	128
7.6	Vergleich zwischen Prognose und realen Abrufen zu einem beliebigen ausgewählten Zeitpunkt (Startzeitpunkt: 08.08.2015 10:13)	130
7.7	Verlauf der Prognosequalität über einen Zeitraum von 60 Stunden	131
7.8	Schematischer Ablauf der Abarbeitung einer Bestellung	133
7.9	Ergebnisse der praktischen Validierung im Werk	135
7.10	Ergebnisse der Simulation mit den Eingangsdaten der praktischen Validierung	135

A.1 Klassifizierung von Charakteristiken zur Modellierung eines erweiterten VRP 182

Tabellenverzeichnis

1.1	Tabellarischer Vergleich von Grundlagen- und Anwendungsforschung (STÄHLER 2001, S. 14; SPOUN 1998, S. 163)	11
3.1	Zuordnung der Anforderungen zu den übergeordneten Anforderungen	41
4.1	Gegenüberstellung der Lösungsverfahren anhand der Solomon- Benchmarks	65
4.2	Bewertung des Erfüllungsgrads der für die Algorithmen relevanten Anforderungen	68
7.1	Abweichung der Prognose vom realen Verlauf in Abhängigkeit der Zeit	132
7.2	Erforderliche Gegebenheiten für den erfolgreichen Einsatz des entwi- ckelten Konzepts zur Routenoptimierung	137
7.3	Wirtschaftlichkeitsbetrachtung mit fiktiven Kostensätzen	141
7.4	Bewertung der Erfüllung der einzelnen Anforderungen	142
A.1	Einflüsse auf den erfolgreichen Einsatz von algorithmusbasierten Logistikroutenoptimierungen	181

Algorithmenverzeichnis

1	Pseudocode des Ameisenalgorithmus in Anlehnung an DORIGO & STÜTZLE (2004, S. 38) und DANNINGER (2012, S. 63)	77
2	Struktur des Algorithmus zur Tourenoptimierung als Pseudocode . . .	101
3	Vorgehen bei der Optimierung nach der Zeit als Pseudocode	102
4	Vorgehen für die Minimierung der Anzahl an Fahrzeugen als Pseudocode	103
5	Pseudocode des erweiterten Ameisenalgorithmus	104

Anhang

A.1 Einflüsse auf den wirtschaftlichen Betrieb von dynamischen Logistiksystemen

Die folgende Tabelle zeigt die Eigenschaften eines Anwendungsfalls, welche einen Einfluss auf die Wirtschaftlichkeit des dynamischen Logistiksystems haben. In Abschnitt 7.5 werden daraus die für das Szenario der vorliegenden Arbeit relevanten Gegebenheiten abgeleitet.

Tabelle A.1: Einflüsse auf den erfolgreichen Einsatz von algorithmusbasierten Logistikroutenoptimierungen

Kategorie	Gegebenheit
Physisch	Existierende oder geplante interne Routenzugloistik
	Internes oder externes System
	Fahrzeugtyp
	Dynamik
	Zeitfenster
	Systemgrenzen
	Schnittstelle zur Produktionsplanung
	Leerbehältermanagement
	Mehrfachverwendung von Fahrzeugen
	Aufsplitten von Ladungen
Anzahl Depots	
Homogenität des Fuhrparks	
Modellierung	Entscheidungsvariablen
	Zielfunktion(en)
	Planungshorizont
	Nebenbedingungen
	Variabilität von Service- bzw. Wegezeiten
Organisatorisch	Finanzielle Aspekte
	Personalkapazität
	Datenverfügbarkeit
	Notwendige Zeit
	Technische Machbarkeit
	Wirtschaftlichkeit

A.2 Klassifizierung von Tourenplanungsproblemen

- | | |
|--|---|
| 1. Szenariospezifische Charakteristiken | 2. Problemspezifische Charakteristiken |
| strategisch | operativ |
| 1.1. Komponenten des Entscheidungsmanagement | 2.1. Fahrzeuge |
| 1.1.1. Tourenplanung | 2.1.1. Typ |
| 1.1.2. Bestands- und Tourenplanung | 2.1.1.1. Homogene Fahrzeuge |
| 1.1.3. Standortplanung | 2.1.1.2. Heterogene Fahrzeuge |
| 1.1.4. Touren- und Fahrereinsatzplanung | 2.1.2. Anzahl |
| 1.1.5. Produktions- und Distributionsplanung | 2.1.2.1. Beschränkt |
| 1.2. Anzahl der Depots | 2.1.2.2. Unbeschränkt |
| 1.2.1. Ein Depot | 2.1.3. Laderaumstruktur |
| 1.2.2. Mehrere Depots | 2.1.3.1. Unterteilt |
| taktisch | 2.1.3.2. Nicht unterteilt |
| 1.3. Belieferungsart | 2.1.4. Kapazitätsbeschränkung |
| 1.3.1. Pickup or Delivery | 2.1.5. Beladungspolitik |
| 1.3.2. Pickup and Delivery | 2.1.5.1. Chronologisch |
| 1.3.3. Mit Rücktransport | 2.1.5.2. Keine Beladungspolitik |
| 1.3.4. Separate Transportanfragen | 2.1.6. Restriktionen Fahrer |
| 1.4. Teillieferungen | 2.2. Zeitrestriktionen |
| 1.4.1. Erlaubt | 2.2.1. Kunde |
| 1.4.2. Nicht erlaubt | 2.2.2. Transportwege |
| 1.5. Planungsperiode | 2.2.3. Depot |
| 1.5.1. Einfache Periode | 2.2.4. Servicezeit |
| 1.5.2. mehrere Perioden | 2.2.5. Wartezeit |
| 1.6. Fahrzeugnutzung | 2.3. Zeitfensterarten |
| 1.6.1. Einmalnutzung | 2.3.1. Ein Zeitfenster |
| 1.6.2. Mehrfache Nutzung | 2.3.2. Mehrere Zeitfenster |
| 1.7. Eingangsdaten | 2.4. Inkompatibilitäten |
| 1.7.1. Statisch | 2.5. Spezifische Beschränkungen |
| 1.7.2. Dynamisch | 2.6. Zielfunktion |
| 1.7.3. Deterministisch | 2.6.1. Ein Ziel |
| 1.7.4. Stochastisch | 2.6.2. Mehrere Ziele |

Abbildung A.1: Klassifizierung von Charakteristiken zur Modellierung eines erweiterten VRP – Taxonomie nach LAHYANI ET AL. (2015)

A.3 Getroffene Annahmen für die Kapazitätsprognose

Für die Kapazitätsprognose mussten diverse Annahmen getroffen werden, unter deren Berücksichtigung sowohl die Berechnung als auch die Auswertung erfolgte. Diese sind im Folgenden aufgeführt:

1. Als durchschnittliche gebundene Kapazität pro auszulieferndem Behälter wurden 4,6 Minuten angenommen. Dieser Wert setzt sich zusammen aus der gleichmäßigen Verteilung sämtlicher Zeiten der Routen und den nicht zu vermeidenden Stillstandszeiten selbst bei optimaler Belegung (Beladung, Warten auf neue Bestelleingänge, etc.). Der Faktor muss bei einer Adaption auf eine anderes System angepasst werden.
2. Zur Kalibrierung wurde ein Sicherheitsfaktor von 1,4 definiert. Über den Sicherheitsfaktor lässt sich steuern, ob die Prognose tendenziell eher auf eine Unterversorgung oder eine Überversorgung hinarbeitet. Mit dem Faktor 1,4 konnte für das vorliegende System ein guter Kompromiss gefunden werden. Der Faktor muss bei einer Adaption auf ein anderes System angepasst werden.
3. Zur Erhöhung der praktischen Prozessstabilität wurde festgelegt, dass ein Routenzug erst bei Kapazitätslücken von mehr als 30 Minuten aus dem Pool der verfügbaren Ressourcen entfernt werden soll. Diese Randbedingung wurde sowohl für die Prognose als auch für die Bewertung ebendieser berücksichtigt.
4. Bei der Prognose ist es kein Problem, wenn die Kapazitätsprognose nicht minutengenau mit dem tatsächlichen Verlauf übereinstimmt, solange sämtliche Behälter termingerecht ausgeliefert werden können. Es ist daher möglich, einzelne Bestellungen in beschränktem Umfang nach hinten zu terminieren, eine Verschiebung nach vorne ist hingegen nicht möglich.
5. Bei der Betrachtung der maximalen Obergrenze wurden die aus dem praktischen Einsatz maximal erreichbare Obergrenzen (bedingt durch nicht vermeidbare Wartezeiten, Zeiten ohne Bestelleingang, etc.) als Referenzwert angenommen.

A.4 Betreute Studienarbeiten

Im Rahmen dieser Dissertation entstanden am Institut für Werkzeugmaschinen und Betriebswissenschaften (*iwb*) der Technischen Universität München (TUM) unter wesentlicher wissenschaftlicher, fachlicher und inhaltlicher Anleitung des Autors dieser Arbeit die im Folgenden aufgeführten studentischen Arbeiten, in welchen verschiedene Fragestellungen rund um die Thematik der dynamischen Logistik und Optimierungsverfahren im industriellen Kontext im Allgemeinen untersucht wurden. Deren Ergebnisse sind zum Teil in die vorliegende Arbeit eingeflossen. Der Autor dankt an dieser Stelle allen Studierenden herzlichst für ihr Engagement bei der Unterstützung dieser wissenschaftlichen Arbeit. Im Folgenden sind die Studienarbeiten chronologisch aufgelistet:

LARISSA TEBBE

Ameisenalgorithmen in der Maschinenbelegungsplanung – Analyse und Bewertung der Einsetzbarkeit von Ant Colony Optimization zur Lösung des Job Shop Scheduling Problems. Semesterarbeit. 2014.

Eingeflossen in Abschnitt 4.3

MICHAEL VON STADEN

Entwicklung und Implementierung eines Ameisenalgorithmus zur Maschinenbelegungsplanung bei alternativen Prozessketten. Masterarbeit. 2016.

Eingeflossen in die Abschnitte 5.3 bis 5.6

SOPHIE BAWIN

Praktische Implementierung von Nebenbedingungen in ACO-basierten Produktionssteuerungen. Semesterarbeit. 2016.

JULIANE KARSCHNICK

Entwicklung einer Methodik zur Abschätzung des Datenerhebungsaufwands für Ablaufsimulationen. Masterarbeit. 2016.

DANIEL KÜNSTNER

Effiziente werksinterne Logistik durch optimierte Tourenplanung. Masterarbeit. 2016.
Eingeflossen in die Abschnitte 2.3 und 5.1

RICHARD DOBLER

Methoden zur fabrikinternen Logistikoptimierung. Semesterarbeit. 2017.

Eingeflossen in Kapitel 2 und 4

JOHANNES DECKERT

Anforderungen an ein PPS-System mit Fokus auf Produktionsprogrammplanung und Ablaufplanung am Beispiel einer dynamischen Fertigung bei der Lufthansa Technik AG. Masterarbeit. 2017.

JOSÉ MIGUEL ESCOBAR COTO

Analyse und Implementierung dynamischer routenzugbasierter Logistiksysteme in einem praktischen industriellen Fall. Masterarbeit. 2017.

Eingeflossen in die Abschnitte 2.1, 2.3, 7.5 und A.1.

