

Thomas Lienert

Methodik zur simulationsbasierten Durchsatzanalyse FTF-basierter Kommissioniersysteme



TECHNISCHE UNIVERSITÄT MÜNCHEN

Fakultät für Maschinenwesen

Lehrstuhl für Fördertechnik Materialfluss Logistik

Methodik zur simulationsbasierten Durchsatzanalyse FTF-basierter Kommissioniersysteme

Thomas Lienert

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. Markus Lienkamp

Prüfer der Dissertation:

1. Prof. Dr.-Ing. Johannes Fottner
2. Prof. Dr.-Ing. Jochen Kreuzfeldt
3. Prof. Dr.-Ing. Willibald A. Günthner

Die Dissertation wurde am 17.11.2020 bei der Technischen Universität München eingereicht und durch die Fakultät für Maschinenwesen am 14.04.2021 angenommen.

Herausgegeben von:

Prof. Dr.-Ing. Johannes Fottner

fml – Lehrstuhl für Fördertechnik Materialfluss Logistik

Technische Universität München

Zugleich: Dissertation, München, Technische Universität München, 2021

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, der Entnahme von Abbildungen, der Wiedergabe auf fotomechanischem oder ähnlichem Wege und der Speicherung in Datenverarbeitungsanlagen bleiben – auch bei nur auszugsweiser Verwendung – dem Autor vorbehalten.

Layout und Satz: Thomas Lienert

Copyright © Thomas Lienert, 2021

ISBN: 978-3-948514-12-9

Printed in Germany, 2021

Vorwort

Auf welchem Weg wird das nächste Ziel erreicht? Diese Frage war zentraler Bestandteil der Steuerungsstrategien für die Fahrzeugflotte, welche im Rahmen der vorliegenden Arbeit entwickelt wurden.

Für mein persönliches Ziel der Promotion habe ich den ein oder anderen Umweg genommen (auch auf dem Rad). Meine Forschungsarbeit am Lehrstuhl fml begann mit einem ZIM Projekt, in welchem ein Simulationsmodell für ein sogenanntes 3D Shuttle-System entwickelt wurde. Es folgten mehrere Industrieprojekte in ganz verschiedenen Bereichen, bei deren Bearbeitung ich jedoch immer wieder auf Arbeiten aus dem Forschungsprojekt zurückgreifen und diese auch weiterentwickeln konnte. So formte sich im Laufe der Zeit das Thema der vorliegenden Arbeit: eine generische Methodik zur Simulation FTF-basierter Kommissioniersysteme. Mein Dank gilt all denjenigen, die mich bei meinem Promotionsvorhaben auf ganz unterschiedliche Weise unterstützt haben.

Für die Betreuung meiner Forschungsaktivitäten bedanke ich mich bei Prof. Günthner sowie seinem Nachfolger Prof. Fottner. Bei Prof. Kreuzfeldt bedanke ich mich für das Koreferat; bei Prof. Lienkamp für die Übernahmen des Prüfungsvorsitzes.

Für die stets inspirierende Zusammenarbeit bedanke mich bei all meinen ehemaligen Kolleginnen und Kollegen am Lehrstuhl fml. Insbesondere jedoch bei meinen langjährigen Bürokollegen Markus und Christian für die fachlichen Diskussionen; bei Sandra und Marcus für Anmerkungen zu dieser sowie den vorangegangenen Arbeiten, sowie bei Julius, für den interdisziplinären Austausch während unzähliger Stunden auf dem Rad.

Nicht zuletzt gilt mein Dank meinem Mentor Hanno, der mir vorgelebt hat, wie sich Wissenschaft und Radsport kombinieren lassen, sowie ganz besonders meinen Eltern Ingrid und Klaus und meiner Freundin Katharina, die mich auf ihre Art und Weise während der gesamten Zeit unterstützt haben.

München, im Juli 2021

Thomas Lienert

Kurzzusammenfassung

Methodik zur simulationsbasierten Durchsatzanalyse FTF-basierter Kommissioniersysteme

Kommissioniersysteme, in welchen eine Flotte Fahrerloser Transportfahrzeuge (FTF) die Ein- und Auslageraufträge ausführen, bieten verschiedene Vorteile gegenüber herkömmlichen, zumeist regelbediengerätbasierten Automatisierungslösungen. Sie sind hinsichtlich des Durchsatzes leicht skalierbar, können flexibel gestaltet und erweitert werden, weisen eine erhöhte Redundanz auf und ermöglichen eine Sequenzierung bereits innerhalb des Lagersystems. Auf der anderen Seite erfordern FTF-basierte Kommissioniersysteme komplexe Steuerungsstrategien, um einen effizienten und robusten Betrieb zu ermöglichen. Insbesondere der Verkehr innerhalb des Lagersystems muss derart gesteuert werden, dass es weder zu Kollisionen noch zu Blockaden zwischen den Fahrzeugen kommt.

Eine Herausforderung bei der Planung und Auslegung FTF-basierter Kommissioniersysteme besteht in der Bestimmung der Anzahl der Fahrzeuge, welche notwendig ist, um den geforderten Durchsatz zu erzielen. Analytische Methoden ermöglichen aufgrund der dynamischen Prozesse lediglich eine Näherungslösung. Die Absicherung der Planung erfordert die Durchführung einer Simulation, welche mit einem hohen Aufwand verbunden ist.

In der vorliegenden Arbeit wird eine Vorgehensweise zur Simulation von FTF-basierten Kommissioniersystemen vorgestellt, welche zum einen die Modellierung der Systeme beinhaltet, um diese in einer Simulationsumgebung abzubilden. Zum anderen umfasst die Vorgehensweise Steuerungsstrategien, welche notwendig sind, um die Ablauffähigkeit der Simulation sicherzustellen. Der Aufbau des Modells wird allgemeingültig strukturiert, sodass sich die Methodik in unterschiedlichen Simulationsumgebungen umsetzen lässt.

Mithilfe der vorgestellten Methodik lassen sich unterschiedliche FTF-basierte Kommissioniersysteme hinsichtlich des erzielbaren Durchsatzes analysieren. Eine individuelle Modellerstellung für jeden Anwendungsfall ist nicht mehr notwendig, wodurch sich der Aufwand bei der Durchführung der Simulation reduzieren lässt.

Abstract

Methodology for simulation-based throughput analysis of AGV-based order picking systems

Order picking systems operated by a fleet of automated guided vehicles (AGV) offer various advantages over conventional, mostly stacker-crane-based automation solutions. They are easily scalable in terms of throughput, can be flexibly designed and expanded, have increased redundancy and enable sequencing within the warehouse system. On the other hand, AGV-based picking systems require complex control strategies to enable efficient and robust operation. In particular, traffic within the warehouse system must be controlled in such a way that there are no collisions or blockages between the vehicles.

A challenge in the planning and design of AGV-based picking systems is the determination of the number of vehicles necessary to achieve the required throughput. Due to the dynamic processes, analytical methods only allow an approximate solution. The validation of the planning requires the execution of a simulation, which is connected with a high expenditure.

A procedure for the simulation of AGV-based order picking systems is presented, which on the one hand includes the modelling of the systems in order to reproduce them in a simulation environment. On the other hand, the approach includes control strategies, which are necessary to ensure the executability of the simulation. The structure of the model is generally structured so that the methodology can be implemented in different simulation environments.

With the help of the methodology presented, a wide variety of AGV-based picking systems can be analyzed with regard to the achievable throughput. It is no longer necessary to create an individual model for each use case, which reduces the effort involved in carrying out the simulation.

Inhaltsverzeichnis

Abkürzungsverzeichnis	XI
Symbolverzeichnis	XIII
1 Einleitung	1
1.1 Ausgangssituation	1
1.2 Problemstellung und Zielsetzung	1
1.3 Vorgehensweise und Struktur der Arbeit	2
2 Grundlagen	5
2.1 Fahrerlose Transportsysteme	5
2.2 Kommissioniersysteme	7
2.3 FTF-basierte Kommissioniersysteme	9
2.3.1 Einsatz von FTF in Kommissioniersystemen	9
2.3.2 Eingrenzung des Betrachtungsgegenstandes	11
2.4 Ereignisdiskrete Simulation	14
2.4.1 Einsatz von Simulation in der Logistik	14
2.4.2 Funktionsweise der ereignisdiskreten Simulation	15
3 Stand der Technik und Forschung	19
3.1 Aspekte bei der Auslegung und Steuerung von FTS	19
3.1.1 Fahrweglayout	19
3.1.2 Auftragsvergabe	21
3.1.3 Routing und Deadlock-Handling	23
3.1.4 Strategie für Fahrzeuge ohne zugewiesene Aufträge	28
3.1.5 Störungsmanagement	29
3.1.6 Batteriemanagement	29
3.1.7 Flottengröße	30
3.2 Einsatz von Simulation bei der Planung von FTS	33
3.2.1 Simulationsbasierte Durchsatzanalyse von FTS	34
3.2.2 Simulationsbasierte Durchsatzanalyse von FTF-basierten Kommissioniersystemen	35
3.3 Detaillierung der Forschungslücke	37

4	Konzept	41
4.1	Erforderliche Steuerungsstrategien	41
4.2	Aufbau der Methodik	43
5	Modellierung	47
5.1	Modellierung des Layouts	47
5.2	Modellierung der Bewegungen der Fahrzeuge	51
5.2.1	Wechsel der Ausrichtung	51
5.2.2	Fahrzeiten	52
5.2.3	Zeitstempel und Zeitdauern	61
6	Steuerungsstrategien	63
6.1	Zeitfensterbasiertes Routing	63
6.1.1	Definition der Zeitfenster	63
6.1.2	Eingaben in das zeitfensterbasierte Routing	64
6.1.3	Ablauf des zeitfensterbasierten Routings	67
6.1.4	Erreichbarkeit freier Zeitfenster	75
6.1.5	Exkurs: Erreichbarkeit freier Zeitfenster bei weiteren Reservierungsformen	87
6.2	Routingbasierte Sequenzierung	90
6.3	Ausführung einer berechneten Route	94
6.3.1	Ausführung eines Segmentes	94
6.3.2	Beanspruchung von Knoten	98
6.3.3	Zugriffe auf die Datenstruktur der Zeitfenster	99
6.4	Strategien für den Umgang mit Störungen	100
6.4.1	Störungen ignorieren	101
6.4.2	System pausieren	102
6.4.3	System neustarten	102
6.4.4	Betroffene Fahrzeuge umleiten	107
7	Umsetzung im Simulationsmodell	111
7.1	Module des Simulationsmodells	111
7.1.1	Layoutmodul	112
7.1.2	Parametermodul	112
7.1.3	Auftragsmodul	113
7.1.4	Routingmodul	115
7.1.5	Auswertungsmodul	115

7.1.6	Lagersystem	116
7.2	Implementierung	116
7.2.1	Verwendete Simulationsumgebung	116
7.2.2	Implementierung der Module	117
7.2.3	Implementierung der Knoten	123
7.2.4	Implementierung der Fahrzeuge	123
7.2.5	Implementierung des Lagerbereichs	124
7.2.6	Modellerstellung	125
7.2.7	Visualisierung der Steuerungsalgorithmen	126
8	Evaluierung	131
8.1	Durchsatzanalyse eines automatisierten Kommissioniersystems mit Unterfahr-FTF	131
8.1.1	Systembeschreibung	132
8.1.2	Modellierung	134
8.1.3	Simulationsstudie	135
8.2	Durchsatzanalyse eines Shuttle-Systems mit gassen- und ebenenwechselnden Shuttle-Fahrzeugen	142
8.2.1	Systembeschreibung	142
8.2.2	Modellierung	144
8.2.3	Simulationsstudie	145
9	Zielerfüllung der Methodik zur simulationsbasierten Durchsatzanalyse	149
9.1	Überprüfung der Anforderungen	149
9.2	Kritische Diskussion der Forschungsergebnisse	151
10	Zusammenfassung und Ausblick	155
10.1	Zusammenfassung	155
10.2	Anwendung der Methodik auf weitere Systemtypen	156
10.3	Anwendung der Methodik in fachfremden Bereichen	157
10.4	Erweiterung der Methodik	158
	Literaturverzeichnis	161
	Abbildungsverzeichnis	171
	Tabellenverzeichnis	177

Anhang A

A-1

A.1 Spielzeitberechnung der Lifte

A-1

Abkürzungsverzeichnis

Abkürzung	Bedeutung
AGV	Automated Guided Vehicle
CARP	Context Aware Route Planning
E/A-Ebene	Ein- und Auslagerebene
E/A-Punkt	Ein- und Auslagerpunkt
fml	Lehrstuhl für Fördertechnik Materialfluss Logistik
FTF	Fahrerloses Transportfahrzeug
FTS	Fahrerloses Transportsystem
MTTR	Mean Time To Repair
RMFS	Robotic Mobile Fulfillment System
SCC	Sample Container Carrier
VDI	Verein Deutscher Ingenieure
WHCA*	Windowed Hierarchical Cooperative A*
ZE	Zeiteinheiten
ZF	Zeitfenster

Symbolverzeichnis

Zeichen	Einheit	Bedeutung
---------	---------	-----------

Lateinische Buchstaben

A	$[-]$	Ausrichtung.
A_i	$[-]$	Menge der Ausrichtungen, welche angibt, entlang welcher Achsen der Knoten i befahren werden kann.
$A_{f_{i,l}}$	$[-]$	Ausrichtung, in welcher die Ankunft in dem Zeitfenster $f_{i,l}$ erfolgt.
A_n	$[-]$	Ausrichtung, in welcher das n -te Ziel erreicht werden muss.
A_S	$[-]$	Ausrichtung des Segmentes S .
a^A	$[m/s^2]$	Beschleunigung und Verzögerung bei Fahrt in Ausrichtung A .
Adj_i	$[-]$	Menge der Nachfolger des Knotens i .
$B_{f_{i,l}}$	$[-]$	Zeiger, welcher auf das Zeitfenster zeigt, aus welchem das freie Zeitfenster $f_{i,l}$ erreicht wird.
$D_{f_{i,l}}$	$[s]$	Schätzer für die kürzeste Gesamtfahrzeit vom Startknoten zum finalen Ziel Z_z bei einer Fahrt, bei der der Knoten i im Zeitfenster $f_{i,l}$ angefahren wird.
E	$[-]$	Lagerebene.
F_i	$[-]$	Menge aller freien Zeitfenster auf dem Knoten i .
$f_{i,l}$	$[-]$	l -tes freies Zeitfenster auf dem Knoten i .
$h_{f_{i,l}}$	$[s]$	Schätzer für die verbleibende Gesamtfahrzeit bis zur Ankunft am finalen Ziel Z_z , ausgehend von der Ankunft im freien Zeitfenster $f_{i,l}$.
$ID_{f_{i,l}}$	$[-]$	ID des Zeitfensters $f_{i,l}$.
$K_{X,Y}$	$[-]$	Knoten mit den Koordinaten X und Y .
l	$[m]$	Gesamtstrecke.
l'	$[m]$	Verbleibende, verkürzte Fahrstrecke.
L_F^A	$[m]$	Länge des Fahrzeuges bei Fahrt in Ausrichtung A .
L_i^A	$[m]$	Ausdehnung des Knotens i in Ausrichtung A .

m	[−]	Anzahl der zu überwindenden Lagerebenen.
n	[−]	Anzahl der Lagerebenen.
$n_{f_{i,l}}$	[−]	Zähler, welcher angibt, zu welchem Ziel aus der Liste der übergebenen Ziele aus dem freien Zeitfenster $f_{i,l}$ geroutet wird.
R_i	[−]	Quadrupel mit Booleschen Variablen, welches die Informationen enthält, in welcher Fahrtrichtung der Knoten i verlassen werden kann.
$s(t, l)$	[m]	Teilstrecke, welche nach der Zeit t zurückgelegt ist, wenn die Gesamtstrecke l gefahren wird.
s_b	[m]	Beschleunigungs-/ Verzögerungsstrecke.
s_c	[m]	Strecke, welche mit konstanter Geschwindigkeit zurückgelegt wird.
$s_{Ex,i}^A$	[m]	Austrittsdistanz, welche von einem Fahrzeug zurückgelegt werden muss, bis es ausgehend von der Ankunftsposition in Ausrichtung A den Knoten i vollständig verlassen hat.
$s_{i,j}$	[m]	Euklidischer Abstand zwischen den Knoten i und j .
s_{vb}	[m]	Verbleibende Strecke, welche während der Beschleunigungsphase zurückgelegt wird, bis das Fahrzeug seine maximale Geschwindigkeit v_{max} erreicht.
s_{zb}	[m]	Strecke, welche das Fahrzeug während der Beschleunigungsphase bereits zurückgelegt hat.
$S_{f_{i,l}}$	[−]	Segment, welches alle Zeitfenster umfasst, die bei der Fahrt in das Zeitfenster $f_{i,l}$ durchfahren werden.
S_n	[−]	Sequenznummer, welche angibt, an welcher Position im Sequenzstrom das n -te Ziel zu erreichen ist.
$t(l)$	[s]	Fahrzeit für die Strecke l .
$t(s, l)$	[s]	Zeitdauer, welche notwendig ist, die Teilstrecke s zurückzulegen, wenn die Gesamtstrecke l gefahren wird.
$t(s_{i,j})$	[s]	Fahrzeit zwischen den Knoten i und j .
t_b	[s]	Beschleunigungs-/ Verzögerungszeit.
t_c	[s]	Zeitdauer für Fahrt mit konstanter Geschwindigkeit.
$t_{Ex,i}^A$	[s]	Kürzest mögliche Austrittsdauer, welche ein Fahrzeug benötigt, um einen Knoten i , ausgehend von der Ankunftsposition in Ausrichtung A , vollständig zu verlassen.

$t_{Lagerspiel}$	[s]	Spielzeit eines Lastwechsels.
t_{Lift}	[s]	Mittlere Spielzeit eines Fahrzeugliftes.
$t_{Lift}(m)$	[s]	Fahrzeit, die ein Lift benötigt, um m Lagerebenen zu überwinden.
t_P	[s]	Zeitlicher Puffer zwischen zwei reservierten Zeitfenstern.
t_{vb}	[s]	Verbleibende Zeitdauer während der Beschleunigungsphase, bis das Fahrzeug seine maximale Geschwindigkeit v_{max} erreicht.
$t_{V,n}$	[s]	Vorgegebene Verweildauer am n -ten Ziel.
t_{VZ}	[s]	Zeitdauer, nach welcher die Verzögerung eingeleitet wird.
t_{zb}	[s]	Zeitdauer, welche das Fahrzeug bereits beschleunigt hat.
t_{RW}	[s]	Ausrichtungswechselzeit.
$T_{Ab,i}$	[s]	Abfahrtszeitpunkt eines Fahrzeuges von dem Knoten i .
$T_{Ab,f_{i,l}}$	[s]	Abfahrtszeitpunkt eines Fahrzeuges aus dem Zeitfenster $f_{i,l}$.
$T_{AE,i}$	[s]	Ende des Austritts eines Fahrzeuges aus dem Knoten i .
$T_{AE,f_{i,l}}$	[s]	Ende des Austritts aus dem Zeitfenster $f_{i,l}$.
$T_{An,i}$	[s]	Ankunftszeitpunkt eines Fahrzeuges auf dem Knoten i .
$T_{An,f_{i,l}}$	[s]	Ankunftszeitpunkt in dem Zeitfenster $f_{i,l}$.
$T_{An,k}^{S_n}$	[s]	Ankunftszeitpunkt der Sequenznummer S_n am Sequenzpunkt k .
$T_{Ende,f_{i,l}}$	[s]	Ende des l -ten freien Zeitfensters auf dem Knoten i .
$T_{ES,i}$	[s]	Beginn des Eintritts eines Fahrzeuges auf dem Knoten i .
$T_{ES,f_{i,l}}$	[s]	Beginn des Eintritts eines Fahrzeuges in das Zeitfenster $f_{i,l}$.
$T_{notwStart,f_{k,p}}$	[s]	Zeitpunkt, zu welchem das freie Zeitfenster $f_{k,p}$ spätestens beginnen muss, damit dieses bei gegebener Abfahrt $T_{Ab,f_{i,l}}$ angefahren werden kann.
$T_{RE,f_{i,l}}$	[s]	Ende der Reservierung des Zeitfensters $f_{i,l}$.
$T_{RS,f_{i,l}}$	[s]	Beginn der Reservierung des Zeitfensters $f_{i,l}$.
T_{Start}	[s]	Früheste mögliche Abfahrtszeit des zu routenden Fahrzeuges.
$T_{Start,f_{i,l}}$	[s]	Beginn des l -ten freien Zeitfensters auf dem Knoten i .
V	[–]	Menge der Knoten im zugrundeliegenden Layoutgraphen.

v_{max}^A	[m/s]	Maximale Geschwindigkeit bei Fahrt in Ausrichtung A .
v_{akt}	[m/s]	Aktuelle Geschwindigkeit.
$v(t, l)$	[m/s]	Geschwindigkeit, mit welcher das Fahrzeug nach der Zeit t fährt, wenn die Gesamtstrecke l gefahren wird.
Z_n	[–]	n -tes Ziel in der Liste der übergebenen Ziele.
ZKM_n	[–]	Zielknotenmenge des Ziels Z_n .

Griechische Buchstaben

λ	[1/h]	Durchsatz.
ω_m	[–]	Wahrscheinlichkeit für einen Ebenenwechsel über m Ebenen.

1 Einleitung

In dieser Einleitung wird zunächst die Ausgangssituation erläutert, bevor anschließend die zugrundeliegende Problemstellung und Zielsetzung skizziert werden sowie der Aufbau der vorliegenden Arbeit vorgestellt wird.

1.1 Ausgangssituation

Die Kommissionierung ist ein Kernelement der Intralogistik und bildet die Basis des wirtschaftlichen Erfolges logistischer und produzierender Unternehmen. Dies ist auf zwei Aspekte zurückzuführen: Zum einen stellt die Kommissionierung den personalintensivsten Bereich innerhalb eines Lager- und Distributionssystems dar, auf welchen – je nach Branche – mit bis zu 55 - 70% ein Großteil der Logistikkosten entfallen [De - 2007], [Nav-2009, S. 17]. Zum anderen beeinflussen die Kommissionierabläufe die Liefertreue gegenüber externen und internen Kunden. Eine effiziente und fehlerfreie Kommissionierung ist somit die Grundlage für die Wettbewerbsfähigkeit und Voraussetzung für die Erfüllung der hohen Anforderungen des Marktes [Hom-2010, S. 3].

Ein Beitrag zur Effizienz- und Qualitätssteigerung sowie zur Kostensenkung lässt sich durch Automatisierung im Kommissionierprozess erzielen. Dafür stehen unterschiedliche Konzepte zur Verfügung. Ein Konzept beruht auf dem Einsatz einer Flotte von Fahrerlosen Transportfahrzeugen – in diesem Kontext auch Kommissionierroboter genannt – welche sich im Lagersystem bewegen, um die Ein- wie Auslageraufträge durchzuführen.

Diese FTF-basierten Kommissioniersysteme weisen eine Reihe unterschiedlicher Vorteile gegenüber herkömmlichen, zumeist regalbediengerätbasierten Automatisierungslösungen auf, erfordern aber zugleich aufwändige Steuerungsstrategien, um einen effizienten und robusten Betrieb zu ermöglichen.

1.2 Problemstellung und Zielsetzung

Eine zentrale Aufgabe bei der Planung und Auslegung von FTF-basierten Kommissioniersystemen ist die Bestimmung der Anzahl der Fahrzeuge, welche notwendig ist, um den geforderten Durchsatz zu erzielen. Der erzielbare Durchsatz einer FTF-Flotte hängt von unterschiedlichen Faktoren, wie beispielsweise dem zugrundeliegenden Layout, der Steuerung der Flotte sowie den Betriebsbedingungen ab. Aufgrund von

dynamischen Prozessen lässt sich der Durchsatz durch den Einsatz analytischer Methoden nur näherungsweise bestimmen. Dies ist insbesondere auf die Quantifizierung der sich ergebenden Leerfahrten der Fahrzeuge zwischen zwei Aufträgen sowie der Blockierungseffekte zwischen den Fahrzeugen – dem Stau – zurückzuführen. Darüber hinaus erschweren wechselseitige Abhängigkeiten wie beispielsweise eine Auslagerung in einer vorgegebenen Auftragssequenz die analytische Erfassung.

Ein Lösungsansatz für die Durchsatzbestimmung besteht in der Durchführung einer Simulation. Diese erlaubt die Absicherung und Optimierung der Planung und ermöglicht eine tiefgehende Systemanalyse. Jedoch ist die Durchführung einer Simulation aufwändig und erfordert Fachkenntnisse. Um ein ablauffähiges Simulationsmodell zu erstellen, müssen bereits detaillierte Steuerungsstrategien, insbesondere für die Steuerung des Fahrzeugverkehrs (Routing und Deadlock-Handling), bekannt sein und implementiert werden.

Das Ziel der vorliegenden Arbeit besteht daher in der Entwicklung einer generischen

**Methodik zur simulationsbasierten Durchsatzanalyse FTF-basierter
Kommissioniersysteme,**

welche Steuerungsstrategien enthält, die die Durchführung einer Simulation ermöglichen und welche somit während der Planungsphase zur Bestimmung der notwendigen Fahrzeuganzahl in FTF-basierten Kommissioniersystemen herangezogen werden kann.

1.3 Vorgehensweise und Struktur der Arbeit

Um dieses Ziel zu erreichen, ist die vorliegende Arbeit wie in Abbildung 1-1 dargestellt aufgebaut.

Im Kapitel 2 werden die Grundlagen zu Fahrerlosen Transportsystemen (FTS), zu Kommissioniersystemen sowie zu der verwendeten Methode der ereignisdiskreten Simulation erläutert. Darüber hinaus wird in diesem Kapitel der Betrachtungsgegenstand dieser Arbeit festgelegt. Das Kapitel 3 gibt den Stand der Technik und Forschung wieder und präzisiert die Forschungslücke.

Im Kapitel 4 wird das Konzept beschrieben, um das Forschungsziel zu erreichen.

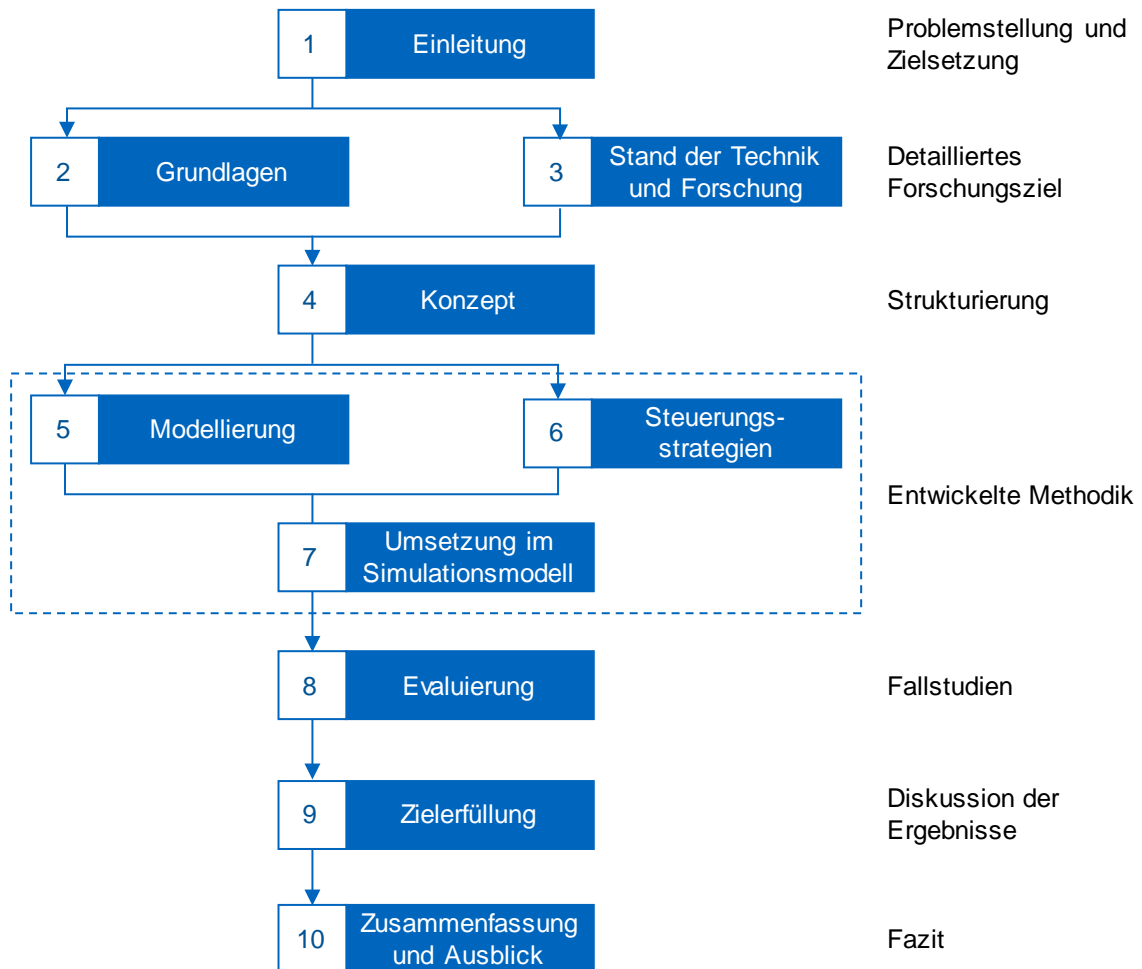


Abbildung 1-1: Aufbau der Arbeit.

Die Kapitel 5 bis 7 beschreiben die Methodik, welche die Modellierung (Kapitel 5), die hinterlegten Steuerungsstrategien (Kapitel 6) sowie die Umsetzung in einem Simulationsmodell (Kapitel 7) umfasst.

Im Kapitel 8 erfolgt die Evaluierung der entwickelten Methodik anhand zweier Fallstudien, in welchen ein Kommissioniersystem mit Unterfahr-FTF sowie ein Shuttle-System mit gassen- und ebenenwechselnden Shuttle-Fahrzeugen betrachtet werden.

Das Kapitel 9 setzt sich mit der Zielerfüllung der vorliegenden Arbeit auseinander und beinhaltet eine kritische Diskussion der Forschungsergebnisse.

Die Arbeit schließt mit einer Zusammenfassung sowie einem Ausblick im Kapitel 10.

2 Grundlagen

In diesem Kapitel werden zunächst die Grundlagen zu Fahrerlosen Transportsystemen (Unterkapitel 2.1) sowie zu Kommissioniersystemen (Unterkapitel 2.2) erläutert. Anschließend wird auf den Betrachtungsgegenstand dieser Arbeit eingegangen: Kommissioniersysteme, welche auf Basis einer FTF-Flotte betrieben werden (Unterkapitel 2.3). Abschließend wird die Methode der ereignisdiskreten Simulation vorgestellt, auf welche in dieser Arbeit zurückgegriffen wird (Unterkapitel 2.4).

2.1 Fahrerlose Transportsysteme

Fahrerlose Transportsysteme (engl. automated guided vehicle systems) sind innerbetriebliche, flurgebundene, gleislose Fördersysteme mit automatisch gesteuerten Fahrerlosen Transportfahrzeugen, deren primäre Aufgabe der Materialtransport, nicht aber der Personentransport ist [VDI-2510].

FTS werden in unterschiedlichen Branchen zur Produktionsversorgung, Lagerbedien- und Kommissionierung sowohl im Indoor-Bereich, als auch im Outdoor-Bereich eingesetzt [Ull-2014, S. 17ff.]. Abbildung 2-1 zeigt die Einordnung von FTS in die Systematik der Fördertechnik.

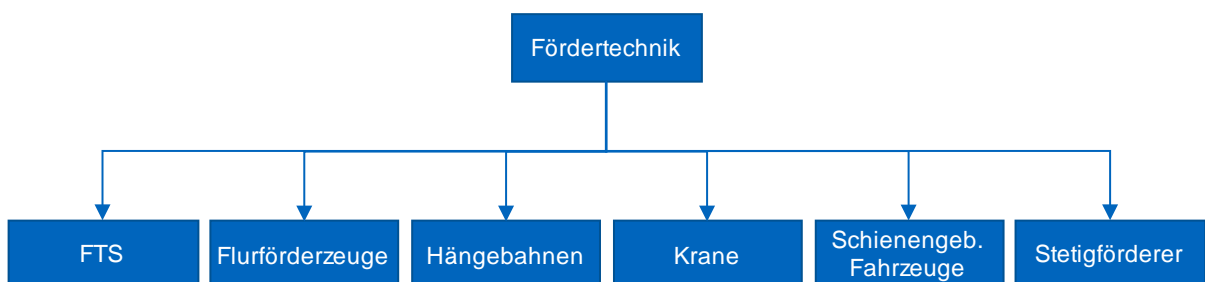


Abbildung 2-1: Einordnung von FTS in die Systematik der Fördertechnik [VDI-2510].

Wesentliche Bestandteile eines FTS sind

- ein oder mehrere FTF,
- eine Leitsteuerung,
- ein Navigationssystem,
- sowie Einrichtungen zur Datenübertragung [VDI-2510].

Ein FTF ist ein flurgebundenen Fördermittel mit eigenem Fahrtrieb, welches automatisch gesteuert und berührungslos geführt werden kann. Es existieren unterschiedliche Bauformen von FTF, wobei sich die FTF hinsichtlich der Gestaltung der Lastaufnahme grundsätzlich in lastziehende sowie lasttragende Fahrzeuge einteilen lassen. [VDI-2510]

Die Aufgabe der Leitsteuerung ist die Koordination der Fahrzeuge sowie die Integration des FTS in die innerbetrieblichen Abläufe. Hauptbestandteil der Leitsteuerung ist die Transportauftragsabwicklung, welche die Verwaltung der Transportaufträge, die Fahrzeugdisposition sowie die Abwicklung der Fahraufträge beinhaltet [VDI-4451]. Weitere Komponenten der Leitsteuerung sind eine Benutzerschnittstelle sowie Service-Funktionen, welche die Anlage visualisieren, Statistiken über die Fahrzeugzustände erfassen und Diagnosen bei Störungen erstellen [Ull-2014, S. 131ff.].

Die Aufgabe der Navigation ist das Führen eines Fahrzeuges von einem Startort zu einem Zielort auf einem bestimmten Weg. Dazu gehören die erforderlichen Mess- und Rechenvorgänge zur Bestimmung des augenblicklichen Standortes sowie des Kurses. Über den Fahrbereich der Fahrzeuge wird ein ortsfestes Koordinatensystem gelegt, innerhalb dessen die Position der Fahrzeuge durch die beiden translatorischen Koordinaten der Grundfläche sowie durch eine rotatorische Koordinate beschrieben wird. Die Bestimmung der Position erfolgt durch Kopplung und Peilung: Die Kopplung – auch Odometrie genannt – bezeichnet die Positionsbestimmung mittels interner Sensoren, wohingegen die Peilung die Position des Fahrzeuges unter Zuhilfenahme von stationären Peilmarken oder Sendern bestimmt. [Ull-2014, S. 107ff.]

Der Informationsaustausch zwischen der Leitsteuerung, den FTF sowie sonstigen stationären Einrichtungen – wie beispielsweise Aufzüge und Förderer – erfolgt über Datenübertragungssysteme. Zu übermitteln sind beispielsweise Fahraufträge, Fahrzeugpositionen und Fahrzeugzustände. Dafür werden vorzugsweise berührungslose Übertragungstechniken – heutzutage zumeist WLAN – eingesetzt. [VDI-2510]

Als Argumente für den Einsatz von FTS werden, sofern die Wirtschaftlichkeit gegeben ist, unter anderem eine hohe Verfügbarkeit und Zuverlässigkeit aufgrund von Redundanzen bei der Fahrzeugflotte, geringe Infrastrukturmaßnahmen bei der Implementierung oder bei Layoutänderungen, eine hohe Transparenz des Fördergeschehens sowie positive Innen- und Außenwirkungen auf die Belegschaft und Kunden aufgeführt [Ull-2014, S. 36f.].

2.2 Kommissioniersysteme

Kommissionieren bezeichnet das Zusammenstellen von Lagerartikeln zu Aufträgen, die bestimmte Mengen unterschiedlicher Artikel verlangen. Bei diesen Aufträgen kann es sich zum einen um Kundenaufträge handeln, für deren Erfüllung die entsprechenden Artikel versendet werden. Zum anderen kann es sich um Materialbedarfe für Produktionsstellen handeln. [Arn-2002, S. A1-7]

Die Durchführung des Kommissionierens erfolgt in einem Kommissioniersystem. Ein solches System besteht grundsätzlich aus

- einem Informationssystem,
- einem Materialflusssystem sowie
- einem Organisationssystem [VDI-3590].

Das zentrale Element des Informationssystems ist der Auftrag. Das Eintreffen eines Auftrags stellt den Anstoß eines Kommissioniervorgangs dar. Er enthält die Informationen zur Artikelidentifikation sowie die jeweiligen Bestellmengen. Ein oder mehrere Aufträge lassen sich, unter Berücksichtigung spezifischer Daten des Kommissioniersystems, in einen oder mehrere Kommissionieraufträge überführen. Diese setzen sich aus sogenannten Positionen zusammen. Jede Position enthält alle für den Kommissioniervorgang eines Artikels notwendigen Informationen wie beispielsweise den Entnahmeort und die Entnahmemenge. [VDI-3590]

Innerhalb des Materialflusssystems erfolgt der physische Transport der Artikel. Der Kommissionierprozess lässt sich in folgende Materialfluss-Grundfunktionen unterteilen [VDI-3590]:

- Transport der Artikel zur Bereitstellung
- Bereitstellung der Artikel
- Fortbewegung des Kommissionierers zur Bereitstellung
- Entnahme der geforderten Warenmenge aus den Bereitstelleinheiten
- Transport der Artikel zum Abgabeort
- Abgabe der entnommenen Warenmenge
- Rücktransport der Artikel, sofern nach erfolgter Entnahme eine Restmenge in der Bereitstelleinheit verbleibt

Für den Transport sowie die Lagerung der Artikel werden innerhalb des Materialflusssystems verschiedene Materialflusseinheiten gebildet: Die Artikel, welche in einem Kommissioniersystem bevorratet sind, werden in *Lagereinheiten* gelagert. Die Ver-

und Entsorgung des Bereitstellensystems erfolgt in *Transporteinheiten*. Die *Beschickungseinheit* bezeichnet diejenige Einheit, mit der der jeweilige Bereitstellort eines Artikels nachgefüllt wird. Diejenige Menge eines Artikels, welche dem Kommissionierer für die Entnahme angeboten wird, wird als *Bereitstelleinheit* bezeichnet. Je nach technischer Umsetzung des Kommissioniersystems kann die *Transporteinheit* der *Lagereinheit*, der *Beschickungseinheit* wie auch der *Bereitstelleinheit* entsprechen. Die *Entnahmeeinheit* bezeichnet die Menge des zu kommissionierenden Artikels, welcher durch einen Zugriff der *Bereitstelleinheit* entnommen wird. Mehrere *Entnahmeeinheiten* bilden eine *Sammeleinheit*. Diese kann sowohl aus identischen, als auch aus verschiedenen Artikeln bestehen. Eine oder mehrere *Sammeleinheiten* bilden abschließend eine *Versandeinheit*. [VDI-3590]

Das Organisationssystem – der dritte Bestandteil eines Kommissioniersystems – teilt sich wiederum in die Teilsysteme Aufbau-, Ablauf- und Betriebsorganisation auf. Die Aufgabe der Ablauforganisation besteht in der geeigneten Strukturierung des Kommissioniersystems. Dabei werden Artikel mit ähnlichen Eigenschaften zu Gruppen in Kommissionierzonen zusammengefasst, in welchen unterschiedliche Lagerbedingungen vorherrschen und verschiedene Kommissioniertechniken zum Einsatz kommen können. Die Ablauforganisation definiert die operativen Verfahrensweisen bei der Bearbeitung von Kommissionieraufträgen. Die Aufgabe der Betriebsorganisation ist die zeitliche Einlastung der Aufträge in das Kommissioniersystem mit dem Ziel, das Gesamtsystem hinsichtlich verschiedener Zielgrößen zu optimieren. [Hom-2010, S. 33ff.]

Für die Ausgestaltung der einzelnen Komponenten innerhalb der drei Teilbereiche Informationssystem, Materialflusssystem und Organisationssystem stehen verschiedene Alternativen zur Verfügung, welche in der VDI-Richtlinie 3590 in Form von Morphologischen Kästen zusammengefasst werden [VDI-3590]. Durch die Kombination der unterschiedlichen Ausprägungen entsteht eine Vielzahl von möglichen Systemlösungen.

Grundlegend lassen sich Kommissioniersysteme jedoch nach zwei Prinzipien klassifizieren: die Kommissionierung nach dem Prinzip *Person-zur-Ware* sowie die Kommissionierung nach dem Prinzip *Ware-zur-Person*. Merkmal des ersten Prinzips ist die Fortbewegung des Kommissionierers hin zur Bereitstelleinheit, welche statisch in entsprechenden Lagermitteln bereitgehalten wird. Für gewöhnlich bewegt sich der Kommissionierer mit einem Kommissionierwagen durch die Regalanlage. An den jeweiligen Entnahmeorten entnimmt er die angeforderten Artikel aus den Bereitstelleinheiten in den Regalen und legt diese in einen Sammelbehälter. Nach Fertigstellung des Kommissionierauftrags bewegt sich der Kommissionierer zu einer zentralen Basis, an der

er den Sammelbehälter übergibt. In Kommissioniersystemen, welche nach dem *Prinzip Ware-zur-Person* betrieben werden, werden die Waren hingegen zu einem vorgegebenen Platz transportiert, an welchem der Kommissionierer die angeforderten Artikel entnimmt. In diesem Fall findet keine Bewegung des Kommissionierers statt, die Ware wird zur Person bewegt. Eine Kombination aus diesen beiden Prinzipien ist möglich. [Hom-2010, S. 41f.]

2.3 FTF-basierte Kommissioniersysteme

In diesem Unterkapitel wird zunächst erläutert, wie FTF in Kommissioniersystemen eingesetzt werden und welche Vorteile dies mit sich bringt (Abschnitt 2.3.1). Anschließend wird im Abschnitt 2.3.2 der Betrachtungsgegenstand der Arbeit spezifiziert.

2.3.1 Einsatz von FTF in Kommissioniersystemen

Kommissioniersysteme können durch den Einsatz von FTF (teil-)automatisiert werden. In Kommissioniersystemen, welche nach dem Prinzip *Person-zur-Ware* betrieben werden, erfolgt dies durch FTF, welche in diesem Kontext auch als Pick- oder Kommissionierroboter bezeichnet werden. Diese verfügen über entsprechende Vorrichtungen für die Durchführung des Pickvorgangs. Die zu entnehmenden Artikel müssen identifiziert und anschließend der Bereitstellereinheit entnommen werden. In solchen Systemen wird das Kommissionieren auch als *Pick-by-Robot* bezeichnet. Abbildung 2-2 zeigt eine beispielhafte Umsetzung.



Abbildung 2-2: Kommissionierroboter TORU der Firma Magazino [Quelle: Magazino GmbH].

In Kommissioniersystemen, welche nach dem Prinzip *Ware-zur-Person* betrieben werden, lagern die FTF die Lagereinheiten aus und transportieren diese für den Pickvorgang zu den Kommissionierern. Anschließend sind die Lagereinheiten wieder einzula-

gern. Diese Vorgänge können beispielsweise durch Unterfahr-FTF (Abbildung 2-3) realisiert werden. In der Literatur wird ein solches System auch als Robotic Mobile Fulfillment System (RMFS) bezeichnet.



Abbildung 2-3: Unterfahr-FTF L600 der Firma Grenzebach [Quelle: Grenzebach Maschinenbau GmbH].

FTF-basierte Kommissioniersysteme bieten unterschiedliche Vorteile gegenüber herkömmlichen, regalbediengerätbasierten Automatisierungslösungen. FTF-basierte Kommissioniersysteme ermöglichen

- eine leichte Skalierbarkeit hinsichtlich des Durchsatzes [Mal-2002], [Kar-2012], [Aza-2019],
- eine flexible Layoutgestaltung und –umgestaltung, [Lam-2017], [Kri-2018], [Zou-2018],
- eine erhöhte Redundanz [Wur-2008], [Hua-2015], [Tap-2017] sowie
- eine Sequenzierung innerhalb des Lagersystems.

Leichte Skalierbarkeit

Der erzielbare Durchsatz kann sowohl kurz- als auch mittelfristig über die Anzahl der eingesetzten Fahrzeuge im System angepasst werden. Ein Kommissioniersystem kann mit einer bestimmten Anzahl an Fahrzeugen in Betrieb genommen werden. Wenn der erforderliche Durchsatz mittelfristig steigt, können weitere Fahrzeuge angeschafft und eingesetzt werden. Ebenso kann auf kurzfristige Schwankungen reagiert werden, indem nicht benötigte Fahrzeuge zwischenzeitlich geparkt werden.

Flexible Layoutgestaltung und –umgestaltung

Das Layout eines Systems kann aufwandsarm umgestaltet und erweitert werden. In manchen Systemausprägungen ist nicht einmal ein fest installiertes Regalsystem notwendig. Die Artikel werden in kleinen Regalen gelagert, welche durch die Unterfahr-FTF ein- und ausgelagert werden. Diese Regale können auf dem Boden abgestellt werden. Regalreihen können dementsprechend einfach verlängert werden, ebenso

können Quergassen in das Lagersystem eingezogen werden. Darüber hinaus kann die Anzahl sowie die Anordnung der Kommissionierstationen leicht verändert werden.

Erhöhte Redundanz

Fällt in einem Automatischen Kleinteillager ein Regalbediengerät aus, können keinerlei Artikel aus der entsprechenden Gasse ausgelagert werden. Fällt hingegen ein einzelnes Fahrzeug aus, kann die verbleibende Flotte – bei Verwendung entsprechender Strategien – trotzdem weiterarbeiten und der Durchsatz ist nur leicht betroffen. Somit weisen FTF-basierte Kommissioniersysteme eine höhere Gesamtverfügbarkeit auf.

Sequenzierung

Es ist möglich, die Reihenfolge, in welcher die Bereitstellereinheiten an der Kommissionierstation abgeliefert werden, innerhalb des Lagersystems zu manipulieren. Die Fahrzeuge können, da es in der Regel stets mehrere alternative Wege zum Ziel gibt, sich gegenseitig überholen, oder aber auch Ausweichmanöver durchführen, um eine vorgegebene Reihenfolge, die sogenannte Sequenz, herzustellen. Eine Sequenzierung in der Lagervorzone, welche zusätzliche Fördertechnik sowie zusätzlichen Raum benötigt, ist daher in diesen Systemen nicht notwendig.

FTF-basierte Kommissioniersysteme bieten unter anderem die zuvor genannten Vorteile. Jedoch bedarf es für die Umsetzung eines erhöhten Planungsaufwands. Neben der Ausgestaltung des Informations- sowie des Organisationssystems, muss im Rahmen der Planung für das Materialflusssystem speziell auf das FTS eingegangen werden. Dabei muss insbesondere die Anzahl der Fahrzeuge bestimmt werden, welche notwendig sind, um die vorgegebene Kommissionierleistung zu erbringen. Die Flottengröße ist der wesentliche Einflussfaktor auf die Investitionen und somit auf die Wirtschaftlichkeit des Systems. Deren Ermittlung ist für den Vergleich unterschiedlicher Automatisierungslösungen eine Voraussetzung. Auf die Bestimmung der notwendigen Anzahl an Fahrzeugen sowie die weiteren, allgemeinen Aspekte, welche bei der Auslegung eines FTS zu berücksichtigen sind, wird im Unterkapitel 3.1 eingegangen.

2.3.2 Eingrenzung des Betrachtungsgegenstandes

Im Rahmen der vorliegenden Arbeit werden FTF-basierte Kommissioniersysteme betrachtet, in welchen sich die Fahrzeuge auf einem rechtwinkligen, vorgegebenen Wegenetz bewegen. In Lager- und Kommissioniersystemen ist eine rechtwinklige Anordnung der Lager- und Quergassen üblich. Eine solche Einschränkung vereinfacht die später zu entwickelnden Strategien für die Organisation der Fahrzeugflotte, da sich die

Fahrzeuge lediglich entlang der beiden Hauptachsen in der Ebene fortbewegen können. Eine Verallgemeinerung bleibt jedoch durch die entsprechenden Erweiterungen der Strategien möglich.

Darüber hinaus wird eingeschränkt, dass die Fahrzeuge vollständig frei verfahren können. Um sich durch das Regalsystem zu bewegen, folgen die Fahrzeuge vorgegebenen Fahrspuren, welche jedoch auch rein virtuell definiert sein können. Diese Einschränkung, welche abermals die zu entwickelnden Strategien vereinfacht, entspricht weitestgehend der Realität. Durch die eingesetzte Navigationstechnik ist oftmals ein Wegenetz fest definiert, welches die Fahrzeuge lediglich zum Ausweichen von Personen oder Hindernissen innerhalb von definierten Grenzen verlassen können.

Eine verbreitete Umsetzung von Ware-zur-Person-Kommissioniersystemen sind Kommissioniersysteme mit Shuttle-Fahrzeugen. Shuttle-Fahrzeuge sind Fahrzeuge, welche zur Ein- und Auslagerung sowie zum Transport von leichten Stückgütern eingesetzt werden [VDI-2696]. Gemäß der Systematik der Fördertechnik (Abbildung 2-1) fallen Shuttle-Fahrzeuge nicht in die Kategorie der Fahrerlosen Transportfahrzeuge, da Shuttle-Fahrzeuge stets schienenengebunden verfahren. Da für die betrachteten Systeme jedoch ein spurgebundenes Verfahren vorausgesetzt wird, werden Shuttle-Systeme in den Betrachtungsgegenstand dieser Arbeit aufgenommen.

Grundsätzlich lassen sich zwei unterschiedliche Systemvarianten von Shuttle-Systemen unterscheiden: Shuttle-Systeme mit Behälterliften und Shuttle-Systeme mit Fahrzeugliften. In Systemen mit Behälterliften sind die Fahrzeuge einer Ebene fest zugeordnet. Lagereinheiten werden an die Lifte übergeben, welche den vertikalen Transport zwischen den Lagerebenen und der Ein- und Auslagerebene (E/A-Ebene) übernehmen. In Systemen mit Fahrzeugliften hingegen können die Fahrzeuge die Ebene wechseln und somit Lagerplätze auf unterschiedlichen Ebenen anfahren. Wird diese Klassifizierung um die Möglichkeit eines Gassenwechsels auf einer Ebene erweitert, ergeben sich, wie in Abbildung 2-4 dargestellt, vier unterschiedliche Varianten, welche sich hinsichtlich der Bewegungsachsen der Fahrzeuge unterscheiden [Lie-2016]. Dabei entspricht die X-Achse den Lagergassen, die Y-Achse möglichen Quergassen sowie die Z-Achse den Liften.

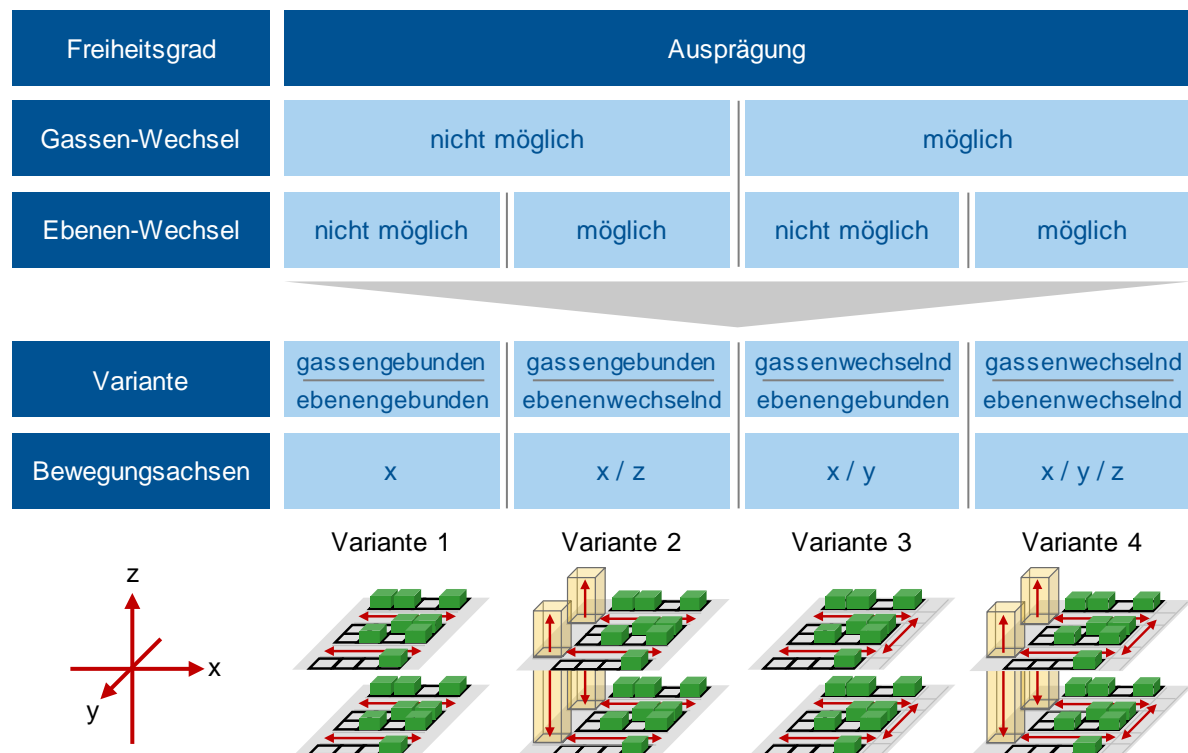


Abbildung 2-4: Klassifizierung von Shuttle-Systemen nach den Bewegungsachsen der Fahrzeuge.

Da in Shuttle-Systemen der Varianten 3 und 4 die Shuttle-Fahrzeuge auf den einzelnen Ebenen auf einem vorgegebenen, rechtwinkligen Wegenetz verfahren, werden diese Varianten in den Betrachtungsgegenstand der Arbeit aufgenommen. Abbildung 2-5 zeigt ein solches gassenwechselndes Shuttle-Fahrzeug. Shuttle-Systeme mit gassengebundenen Shuttle-Fahrzeugen – sogenannte captive-Varianten – werden hingegen nicht berücksichtigt.

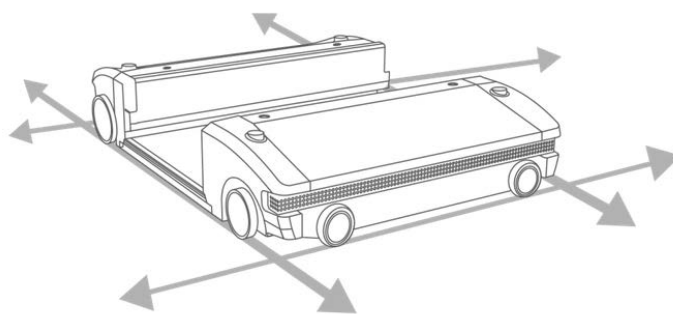


Abbildung 2-5: Gassenwechselndes Shuttle-Fahrzeug OSR Shuttle™ Evo der Firma Knapp [Quelle: KNAPP AG].

Die im Rahmen dieser Arbeit entwickelte Methodik fokussiert sich auf Kommissioniersysteme, welche nach dem Prinzip *Ware-zur-Person* betrieben werden. Jedoch lässt sich das Vorgehen ohne Weiteres auch auf Kommissioniersysteme anwenden, welche nach dem Prinzip *Person-zur-Ware* betrieben werden.

2.4 Ereignisdiskrete Simulation

In diesem Unterkapitel werden zunächst die Anwendungsfelder der Simulation beschrieben (Abschnitt 2.4.1), bevor die Funktionsweise der ereignisdiskreten Simulation erläutert wird (Abschnitt 2.4.2).

2.4.1 Einsatz von Simulation in der Logistik

Simulation ist ein Werkzeug, welches bei Planung, Realisierung und Betrieb von technischen Systemen eingesetzt wird. Der Begriff Simulation bezeichnet die Durchführung von Experimenten an einem Modell, welches ein geplantes oder existierendes System vereinfacht nachbildet. Dabei wird das Ziel verfolgt, Erkenntnisse zu erlangen, welche auf die Wirklichkeit übertragbar sind. [VDI-3633]

In der Planungsphase wird Simulation dafür genutzt, die Auslegung von Systemen und Prozessen abzusichern und zu verbessern. Der Einsatz von Simulation in der Realisierungsphase ermöglicht beispielsweise die virtuelle Inbetriebnahme einer Anlage. Bei der sogenannten Emulation wird die reale Steuerungssoftware mit einem Simulationsmodell verbunden, um die Steuerungssoftware zu testen und zu optimieren. In der Betriebsphase erlaubt Simulation den Vergleich kurzfristiger und situationsabhängiger Ablaufvarianten und kann darüber hinaus zur Ausbildung neuer Mitarbeiter eingesetzt werden.

Der Einsatz von Simulation ermöglicht die Untersuchung

- geplanter, noch nicht existierender Systeme,
- existierender Systeme ohne direkten Betriebseingriff,
- mehrerer Gestaltungsvarianten bei geringem Arbeitsaufwand,
- des Systemverhaltens über lange Zeiträume hinweg sowie
- von Anlaufvorgängen und Übergängen zwischen definierten Betriebszuständen [VDI-3633].

Die Durchführung einer Simulation ist aufwändig und erfordert Fachkenntnisse. Daher ist der Einsatz von Simulation in jedem Anwendungsfall abzuwägen. Eine Aufgabenstellung ist simulationswürdig, wenn dynamische Prozesse oder stochastische Einflüsse eine nicht zu vernachlässigende Rolle in dem zu untersuchenden System spielen und die Lösung der Aufgabenstellung nur durch den Einsatz von Simulation möglich ist. Eine Aufgabenstellung ist aber ebenso simulationswürdig, falls die Lösung zwar durch analytische Methoden bestimmt werden kann, ein Simulationsmodell die

Lösung jedoch erheblich vereinfacht. Der Einsatz von Simulation ist auch dann zu vertreten, falls eine Aufgabenstellung mit anderen Mitteln einfacher und schneller zu lösen ist, an die Visualisierung der Ergebnisse jedoch besondere Anforderungen gestellt werden, welche durch die Simulation vereinfacht erfüllt werden können. [Wen-2008, S. 15]

Simulationsmodelle lassen sich hinsichtlich der Dynamik, des Zufallseinflusses sowie der Art und Weise, in welcher sich Systemzustände verändern, klassifizieren. Ein statisches Simulationsmodell repräsentiert entweder ein System zu einem bestimmten Zeitpunkt, oder aber ein System, in welchem die Zeit keine Rolle spielt. Ein dynamisches Simulationsmodell repräsentiert hingegen ein System, welches sich über die Zeit verändert. Ein Simulationsmodell, welches keinerlei stochastischen Einflüssen unterliegt, wird als deterministisch bezeichnet. Das Ergebnis der Simulation steht fest, sobald die Eingabeparameter definiert sind, auch wenn für die Ermittlung dieses Ergebnisses ein erheblicher Zeitbedarf notwendig ist. In einem stochastischen Simulationsmodell wird das Systemverhalten hingegen durch zufällige Ereignisse beeinflusst. In einem kontinuierlichen Simulationsmodell ändern sich die Systemzustände kontinuierlich. In einem diskreten Simulationsmodell ändern sich die Systemzustände hingegen zu diskreten Zeitpunkten. [Law-2015, S. 5f.]

2.4.2 Funktionsweise der ereignisdiskreten Simulation

Im Bereich der Produktion und Logistik werden zumeist Anlagen und Abläufe durch den Einsatz von Simulationssoftware betrachtet, welche das dynamische Systemverhalten unter der Verwendung stochastischer Komponenten mit Zustandsänderungen zu diskreten Zeitpunkten abbilden. Dabei wird das Systemverhalten nachgebildet, indem Zustandsänderungen beim Eintritt von Ereignissen beschrieben werden. Diese Art der Simulation wird als ereignisdiskrete Simulation bezeichnet. [Ros-2011, S. 14]

Bei der Betrachtung eines Lagersystems stellt eine Einlagerung ein Ereignis dar. Der Systemzustand – in diesem Fall der Lagerbestand – verändert sich. Bis zum Eintreffen des nächsten Ereignisses – beispielsweise einer Auslagerung – bleibt der Systemzustand unverändert und die Zeit zwischen den beiden Ereignissen wird übersprungen.

Die Software zur Erstellung eines Simulationsmodells sowie zur Durchführung von Simulationsstudien wird als Simulationsumgebung bezeichnet. Für die Durchführung einer Simulation existieren verschiedene Simulationsumgebungen, welche sich hinsichtlich der Programmiersprache, des Funktionsumfangs sowie der Benutzeroberfläche unterscheiden. Jede Simulationsumgebung zur Durchführung von ereignisdiskreten

Simulationen weist jedoch einige Kernkomponenten auf, welche nachfolgend beschrieben werden [Law-2015, S. 9], [Ros-2011, S. 15]:

- Systemzustand: Menge von Variablen, die das System zu einem bestimmten Zeitpunkt beschreiben.
- Simulationsuhr: Variable, welche die aktuelle Simulationszeit wiedergibt.
- Ereigniskalender: Liste, welche für jeden Ereignistyp den Zeitpunkt des nächsten Eintritts enthält.
- Statistische Zähler: Variablen zur Speicherung von Informationen über das Systemverhalten.
- Initialisierungsroutine: Unterprogramm, welches zu Beginn der Simulation ausgeführt wird, um das Simulationsmodell zu initialisieren.
- Zeitführungsroutine: Unterprogramm zur Bestimmung des Ereignisses, welches in der Simulation als nächstes eintritt, sowie zum Vorstellen der Simulationsuhr auf den entsprechenden Ereigniszeitpunkt.
- Ereignisroutinen: Unterprogramme, welche den Systemzustand beim Eintreten eines Ereignisses ändern.
- Bibliotheksroutinen: Unterprogramme zur Generierung von Zufallszahlen gemäß unterschiedlichen Verteilungen.
- Ergebnisroutine: Unterprogramm, welches nach Simulationsende ausgeführt wird, um aus den statistischen Zählern Schätzwerte für definierte Kennzahlen zu berechnen und diese dem Anwender bereitstellt.
- Hauptprogramm: Unterprogramm, welches den Ablauf der gesamten Simulation durch die Verwendung der unterschiedlichen Routinen steuert.

Das Diagramm in Abbildung 2-6 zeigt den Ablauf des Hauptprogrammes und gibt die Zusammenhänge zwischen den einzelnen Komponenten wieder.

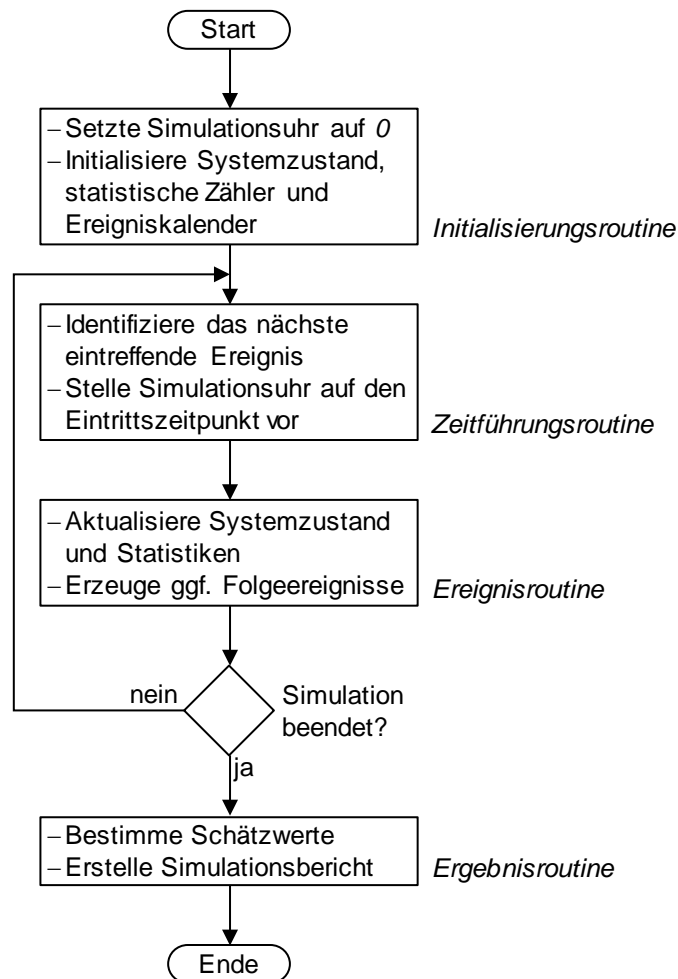


Abbildung 2-6: Ablauf der ereignisorientierten Simulation.

Zu Beginn eines Simulationslaufes ruft das *Hauptprogramm* die *Initialisierungsroutine* auf: Die *Simulationsuhr* wird auf 0 gesetzt, der *Systemzustand*, die *statistischen Zähler* sowie der *Ereigniskalender* werden initialisiert. Der nun wiederkehrende Zyklus besteht aus dem Aufrufen der *Zeitführungsroutine*, um das nächste eintreffende Ereignis zu bestimmen und die *Simulationsuhr* auf den *Eintrittszeitpunkt* dieses Ereignisses vorzustellen. Anschließend wird die entsprechende *Ereignisroutine* ausgeführt, in welcher der *Systemzustand* sowie die *statistischen Zähler* aktualisiert werden. Je nach Ereignistyp werden durch die Ausführung Folgeereignisse erzeugt, welche in den *Ereigniskalender* eingetragen werden. Für die Bestimmung der Eintrittszeitpunkte dieser Folgeereignisse wird oftmals auf Wahrscheinlichkeitsverteilungen zurückgegriffen, welche in den *Bibliotheksroutinen* bereitgestellt werden. In einem nächsten Schritt wird geprüft, ob definierte Abbruchkriterien erfüllt sind und die Simulation beendet ist. In diesem Fall wird für die Auswertung der Simulation und die Aufbereitung der Ergebnisse die *Ergebnisroutine* ausgeführt. Anderenfalls beginnt der Zyklus mit der Bestimmung des nächsten Ereignisses erneut. [Law-2015, S. 10f.]

Die einmalige Ausführung des Hauptprogramms wird als Simulationslauf bezeichnet. Unterliegt ein Simulationsmodell stochastischen Einflüssen, sind mit den unveränderten Eingangsparametern mehrere Simulationsläufe zu wiederholen, um eine hinreichend hohe statistische Signifikanz bezüglich der durch die Simulation ermittelten Kennzahlen zu erlangen. Dabei werden für jeden Simulationslauf unterschiedliche Startwerte für die Generierung der Zufallszahlen verwendet. Diese Wiederholungen werden als Replikationen bezeichnet. [Law-2015, S. 191]

Für die Bestimmung der Länge eines Simulationslaufes existieren keine eindeutigen Richtlinien, jedoch sollte die Länge derart gewählt werden, dass alle für das betrachtete System relevanten Ereignisse mehrmals stattfinden. Die Anzahl der durchzuführenden Replikationen kann beispielsweise über die Bestimmung von Konfidenzintervallen abgeschätzt werden. Diese geben für eine gegebene Stichprobe (Werte aus den Simulationsläufen) sowie für eine definierte Vertrauenswahrscheinlichkeit an, mit welcher Wahrscheinlichkeit die wahre statistische Kennzahl – beispielsweise die mittlere Auslagerleistung je Stunde – in diesem Intervall liegt. Um eine hohe statistische Signifikanz zu erhalten, sind schmale Konfidenzintervalle erstrebenswert. Für den Zusammenhang zwischen der Anzahl der Replikationen und der Breite des Konfidenzintervalls gilt: Je größer die Anzahl der Strichproben ist, desto schmaler ist das Konfidenzintervall. [Wen-2008, S. 144], [VDI-3633]

3 Stand der Technik und Forschung

Im Unterkapitel 3.1 wird auf die Aspekte bei der Auslegung und Steuerung von FTS eingegangen. Im Unterkapitel 3.2 werden Ansätze zum Einsatz von Simulation bei der Planung von FTS erläutert. Abschließend wird im Unterkapitel 3.3 der Forschungsbedarf erörtert.

3.1 Aspekte bei der Auslegung und Steuerung von FTS

In der Literatur werden folgende Aspekte aufgeführt, welche es bei der Auslegung und Steuerung von FTS zu berücksichtigen gilt [Gan-1998], [Le--2006], [Vis-2006]:

- die Gestaltung des Fahrweglayouts,
- die Auftragsvergabe,
- das Routing und Deadlock-Handling,
- das Verhalten von Fahrzeugen ohne zugewiesene Aufträge,
- das Störungsmanagement,
- das Batteriemangement sowie
- die Bestimmung der Flottengröße.

Diese Aspekte werden in den folgenden Abschnitten erläutert, wobei stets auf den Kontext der FTF-basierten Kommissioniersysteme eingegangen wird.

3.1.1 Fahrweglayout

Die Fahrzeuge bewegen sich auf einem vorgegebenen Wegenetz, welches die verschiedenen Quellen und Senken im System verbindet. Das Layout wird gewöhnlich durch einen Graphen abgebildet, bei welchem Quellen, Senken sowie Kreuzungen durch die Knoten repräsentiert werden. Die Kanten entsprechen den Verbindungen, entlang welcher die Fahrzeuge verfahren können und werden für gewöhnlich mit der Fahrzeit oder aber der Distanz zwischen den Knoten gewichtet. Die Gestaltung dieses Wegenetzes – auch Fahrkurstopologie genannt – hat einen direkten Einfluss auf die Fahrzeiten der Fahrzeuge, die notwendige Anzahl an Fahrzeugen im System sowie das Verkehrsaufkommen [Vis-2006].

In der Literatur wird zwischen konventionellen Fahrkurstopologien sowie alternativen Fahrkurstopologien unterschieden [Sin-1994]. In konventionellen Fahrkurstopologien können die Fahrzeuge jede beliebige Verbindung zwischen zwei Übergabestationen

befahren. Charakteristisch für diese Art von Wegenetzen sind eine Vielzahl von Verzweigungen und Zusammenführungen, wodurch den Fahrzeugen für das Erreichen eines bestimmten Ziels meist mehrere Alternativen zur Verfügung stehen [Sch-2000, S. 24]. Die alternativen Fahrkurstopologien zeichnen sich durch eine verringerte Komplexität aus und lassen sich in sogenannte Single-Loop-Topologien sowie Tandem-Topologien unterteilen. In einer Single-Loop-Topologie enthält das Wegenetz weder Verzweigungen noch Zusammenführungen. Somit existieren für die Fahrzeuge keine Alternativen bei der Wahl der Route, denn es wird immerzu derselbe Loop zurückgelegt. Eine Tandem-Topologie setzt sich aus mehreren, sich nicht überlappenden Single-Loops zusammen. Um den durchgängigen Materialfluss zu ermöglichen, sind Transferstationen zwischen den einzelnen Loops notwendig (Abbildung 3-1).

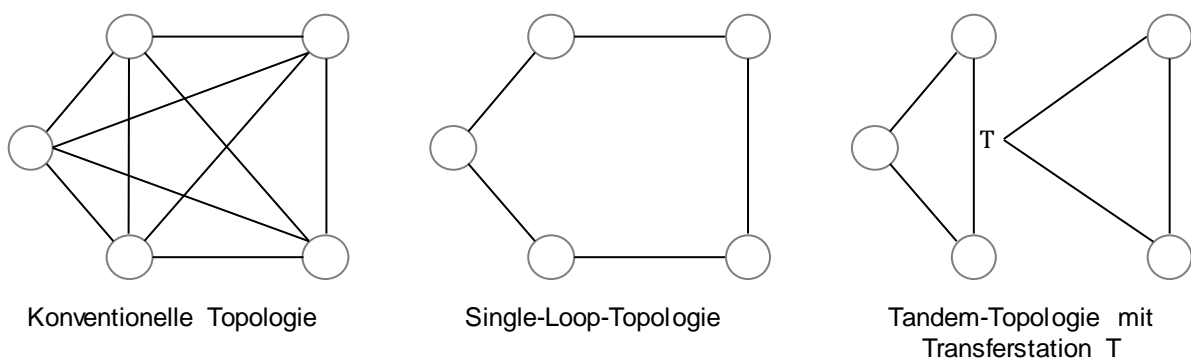


Abbildung 3-1: Unterschiedliche Fahrkurstopologien.

In Lager- und Distributionszentren werden überwiegend konventionelle Topologien eingesetzt, während Single-Loop-Topologien beispielsweise in Cross-Docks verwendet werden. Tandem-Topologien hingegen eignen sich für bestimmte Produktionsumgebungen. [Le--2006].

Neben der Wahl der Topologie ist zu entscheiden, in welchen Fahrrichtungen sich die Fahrzeuge entlang einzelner Layoutabschnitte bewegen dürfen. Bei unidirektionalem Verkehr können Layoutabschnitte lediglich entlang einer Fahrtrichtung zurückgelegt werden. Somit entsteht ein Einbahnstraßensystem. Bei bidirektionalem Verkehr hingegen können Layoutabschnitte in entgegengesetzten Fahrrichtungen befahren werden. Dadurch lassen sich Fahrstrecken gegenüber dem Einbahnstraßensystem verkürzen, jedoch bedarf es einer aufwändigeren Koordination des Verkehrs, da sich zwei Fahrzeuge auf einem Layoutabschnitt in entgegengesetzter Fahrtrichtung begegnen können und sich somit gegenseitig blockieren. Eine dritte Möglichkeit besteht darin, zwei unidirektionale Spuren je Layoutabschnitt zu verwenden. Somit können die Layoutabschnitte in beide Richtungen befahren werden, ohne dass sich zwei entgegenkommende Fahrzeuge blockieren. Jedoch erfordert dies einen erhöhten Platzbedarf. [Vis-2006]

Um zu entscheiden, welche Fahrwegorientierung für ein System geeignet ist, sollte auf das Werkzeug der Simulation zurückgegriffen werden [Egb-1986].

Im Kontext der FTF-basierten Kommissioniersysteme ist das Wegenetz im Wesentlichen durch die Anordnung des Regalsystems vorgegeben und entspricht den Lager- und Quergassen. Die Fahrtrichtung sowie die Anzahl der Spuren bleiben dennoch festzulegen.

3.1.2 Auftragsvergabe

Die Auftragsvergabe (engl. dispatching) ist die Zuordnung von Aufträgen zu Fahrzeugen. Unter einem Auftrag wird ein Transportauftrag verstanden. Ein Transportauftrag ist durch einen Startort, an welchem die Transporteinheit aufzunehmen ist, einen Zielort, an welchem die Transporteinheit abzugeben ist, sowie dem Zeitpunkt, zu welchem der Auftrag durchzuführen ist, charakterisiert [Hom-2008]. Die Ausführung eines Auftrags setzt sich aus den in Abbildung 3-2 aufgeführten Prozessschritten zusammen.

Stimmt die aktuelle Position des Fahrzeuges nicht mit dem Startort des Auftrags überein, erfolgt eine Leerfahrt des Fahrzeuges zum Startort. An diesem wird die Transporteinheit aufgenommen und anschließend zum Zielort transportiert. Falls nach Abgabe der Transporteinheit am Zielort dem Fahrzeug kein weiterer Auftrag zugewiesen wird, verbleibt es entweder an seiner aktuellen Position oder aber nimmt eine zugewiesene Parkposition ein (siehe Leerfahrtstrategien, Abschnitt 3.1.4).

Grundsätzlich gibt es zwei Sichtweisen auf das Entscheidungsproblem der Auftragsvergabe. Erstens, ein Auftrag ist freigegeben und muss einem Fahrzeug zugewiesen werden. Zweitens, ein Fahrzeug beendet einen Auftrag – in Abbildung 3-2 nach der Abgabe der Transporteinheit oder dem Erreichen der Parkposition – und kann einen neuen Auftrag übernehmen. Entsprechend lassen sich Strategien für die Auftragsvergabe in Auftrags-initiierte Strategien sowie Fahrzeug-initiierte Strategien unterteilen [Egb-1984].

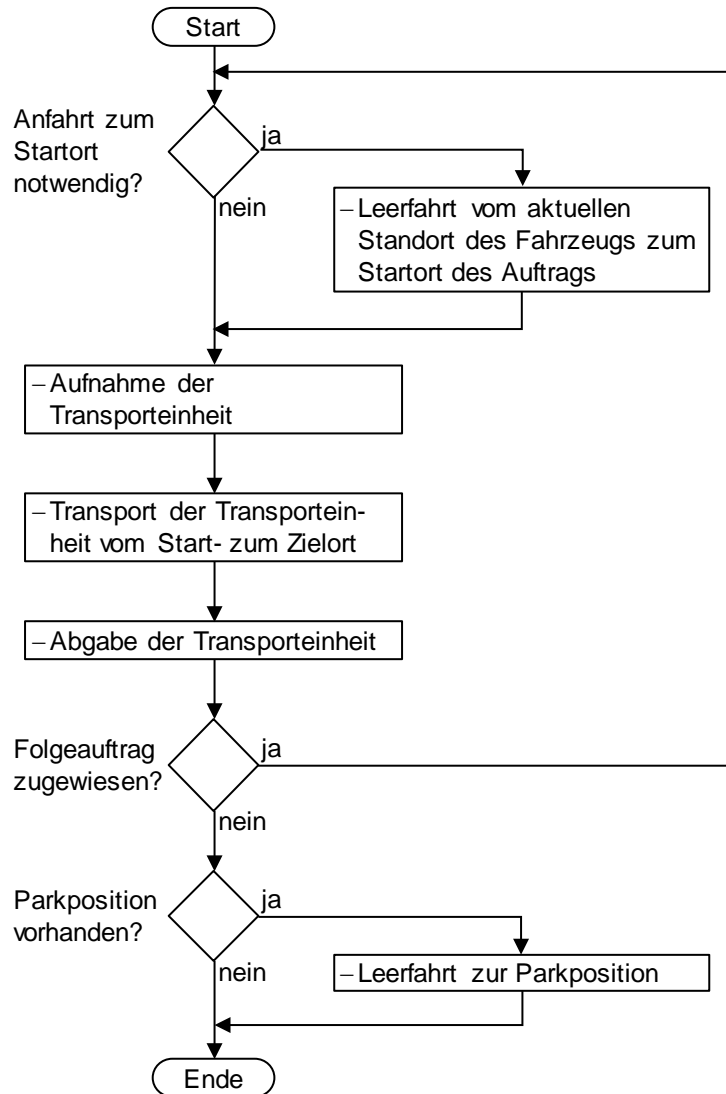


Abbildung 3-2: Ausführung eines Transportauftrags.

Bei einer Auftrags-initiierten Strategie wird für einen bestimmten Auftrag das Fahrzeug gewählt, welches den Auftrag ausführt. Mögliche Strategien bestehen in der Wahl des Fahrzeuges,

- welches sich in der geringsten Entfernung zum Startort des Auftrags befindet,
- welches am längsten untätig war,
- welches am wenigsten ausgelastet ist,
- welches zufällig ausgewählt wird [Egb-1984].

Für die Auswahl des Fahrzeuges können auch mehrere Kriterien zum Einsatz kommen. Durch eine multikriterielle Vergabe lässt sich verhindern, dass einzelne Fahrzeuge über einen längeren Zeitraum nicht berücksichtigt werden, da sie das einzige angelegte Kriterium nie am besten erfüllen. Bei der *simple additive weighting method*

wird ein Fahrzeug für die Ausführung eines Auftrags über die normierte Gewichtung unterschiedlicher Kriterien ausgewählt. [Kle-1996]

Bei einer Fahrzeug-initiierten Strategie hingegen wird für ein bestimmtes Fahrzeug der als nächstes auszuführende Auftrag ausgewählt. Mögliche Strategien bestehen in der Wahl des Auftrags,

- dessen Startort sich in der geringsten Entfernung zur aktuellen Position des Fahrzeuges befindet,
- welcher am längsten auf die Zuweisung zu einem Fahrzeug wartet,
- welcher zufällig gewählt wird [Egb-1984].

Ein Beispiel für eine multikriterielle Fahrzeug-initiierte Vergabe ist die Strategie *C100FCFS*. Bei diesem Vergabeverfahren wählt das Fahrzeug den Auftrag, dessen Startort sich am nächsten an der aktuellen Position des Fahrzeuges befindet. Jedoch wird dabei nur ein vorgegebener Radius berücksichtigt. Ist kein Auftrag verfügbar, dessen Startort innerhalb dieses Radius liegt, wird der Auftrag gewählt, welcher bereits am längsten auf die Zuweisung zu einem Fahrzeug wartet. [De -2004]

Im Bereich der FTF-basierten Kommissioniersysteme führen die Fahrzeuge Einlager-, Auslager- und gegebenenfalls Umlageraufträge aus. Dabei können unterschiedliche Strategien verfolgt werden: Bei der Einzelspielstrategie werden entweder nur Einlager- oder nur Auslageraufträge durchgeführt. Bei der Doppelspielstrategie werden jeweils ein Einlager- sowie ein Auslagerauftrag kombiniert. Wird der Einlagerplatz in der Nähe des Auslagerplatzes gewählt, lassen sich die notwendigen Leerfahrten verkürzen und der Durchsatz erhöhen. [Gud-2010, S. 600f.]

Im Rahmen der Auftragsvergabe wird somit festgelegt, welches Fahrzeug welchen Ein- und/ oder Auslagerauftrag ausführt.

3.1.3 Routing und Deadlock-Handling

Wurde einem Fahrzeug ein Auftrag zugewiesen, muss entschieden werden, auf welchem Weg das Fahrzeug fährt, um den Auftrag auszuführen. Dieses Problem wird als Routing bezeichnet. Die Aufgabe des Deadlock-Handling besteht in der Ausführung der berechneten Routen derart, dass mit wechselseitigen Blockaden zwischen zwei oder mehreren Fahrzeugen – sogenannten Deadlocks – umgegangen werden kann.

Routing und Deadlock-Handling können sowohl getrennt voneinander als auch gemeinsam betrachtet werden. Im ersten Fall wird eine Route bestimmt und während der

Ausführung dieser Route werden Strategien für den Umgang mit Deadlocks eingesetzt. Im zweiten Fall werden mögliche Deadlocks bereits bei der Planung der Route berücksichtigt. Routing und Deadlock-Handling sind daher eng miteinander verknüpft und werden in diesem Abschnitt gemeinsam behandelt.

Routingalgorithmen lassen sich in statische sowie dynamische Algorithmen unterteilen [Vis-2006]. Beim statischen Routing wird für eine bestimmte Start-Ziel-Relation stets der gleiche Weg gewählt, welcher zumeist auf der kürzesten Entfernung zwischen dem Start- und dem Zielort beruht. Zur Bestimmung dieses Weges stehen aus dem Bereich der Graphentheorie unterschiedliche Algorithmen wie der Dijkstra-Algorithmus [Dij-1959], der A*-Algorithmus [Har-1968], der Triple-Algorithmus – auch Floyd-Warschall-Algorithmus genannt [Flo-1962] – und der Algorithmus von Bellman und Ford [Bel-1958], [For-1956] zur Verfügung. Der A*-Algorithmus bestimmt den kürzesten Weg von einem vorgegebenen Startknoten zu einem vorgegebenen Zielknoten. Der Dijkstra-Algorithmus sowie der Algorithmus von Bellman und Ford bestimmen hingegen von einem vorgegebenen Startknoten die kürzesten Wege zu allen anderen Knoten im Layout, wobei der Algorithmus von Bellman und Ford auch mit negativen Kantengewichtungen umgehen kann. Der Triple-Algorithmus bestimmt die kürzesten Wege zwischen allen Knotenpaaren im Layout.

Im Gegensatz zu statischen Routingalgorithmen bestimmen dynamische Routingalgorithmen den Weg zwischen einem Start- und einem Zielort auf Basis von Echtzeitinformationen wie beispielsweise dem Verkehrsaufkommen entlang einzelner Layoutabschnitte. Somit können für eine Start-Ziel-Relation je nach Zeitpunkt und Systemzustand unterschiedliche Wege gewählt werden. [Vis-2006]

Dynamische Routingalgorithmen lassen sich wiederum in Algorithmen unterteilen, welche eine Route vollständig oder abschnittsweise berechnen (Abbildung 3-3) [Tag-1995]. Wird eine Route vor Beginn der Fahrt vollständig bestimmt, besteht die Möglichkeit, dass während der Fahrt die Ausführung der Route durch das Auftreten unvorhergesehener Ereignisse erschwert oder sogar unmöglich wird. Durch die abschnittsweise Berechnung einer Route kann auf solche Ereignisse reagiert werden, jedoch kann eine Optimierung der Gesamtroute nur eingeschränkt erfolgen.

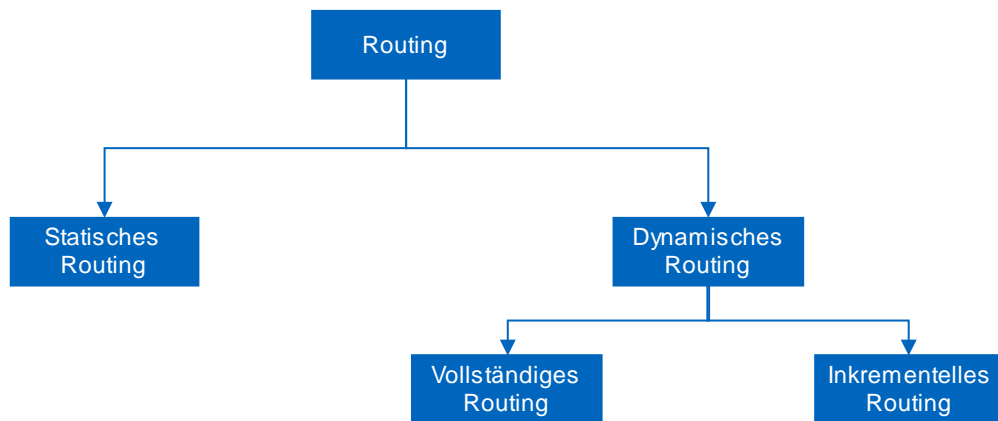


Abbildung 3-3: Klassifizierung von Routing-Algorithmen.

Das Auftreten von Deadlocks ist ein logisches Problem, welches in unterschiedlichen Bereichen auftreten kann. Formal lässt sich ein Deadlock als eine Situation beschreiben, in welcher ein oder mehrere gleichzeitig stattfindende Prozesse auf ewig blockiert sind, da die Ressourcenanforderungen der Prozesse nie erfüllt werden können [Kim-1997].

Ein Deadlock entsteht genau dann, wenn die nachfolgenden vier Bedingungen gleichzeitig erfüllt sind [Cof-1971]:

1. *mutual exclusion*: Keine Ressource kann von mehr als einem Prozess zeitgleich in Anspruch genommen werden.
2. *wait for*: Prozesse blockieren einmal zugewiesene Ressourcen, während sie auf die Zuweisung weiterer Ressourcen warten.
3. *no preemption*: Prozesse geben einmal zugewiesene Ressourcen erst dann frei, wenn diese nicht mehr benötigt werden.
4. *circular wait*: Es liegt ein Kreisschluss von Prozessen vor, welche jeweils auf Ressourcen warten, die von anderen Prozessen innerhalb dieses Kreises blockiert werden.

Im Kontext des FTF-Verkehrs entsprechen einzelne Layoutabschnitte den Ressourcen sowie die Fahrzeuge den Prozessen (Abbildung 3-4). Auf jedem Layoutabschnitt – vorausgesetzt diese werden als hinreichend klein definiert – darf sich zu jedem Zeitpunkt nur ein Fahrzeug befinden. Während ein Fahrzeug darauf wartet, dass der nächste Layoutabschnitt frei wird, blockiert es den Layoutabschnitt, auf welchem es sich aktuell befindet. Dieser wird erst dann wieder freigegeben, wenn der nächste Layoutabschnitt frei geworden ist, das Fahrzeug seine Fahrt fortsetzen konnte und den vorherigen Layoutabschnitt vollständig verlassen hat.

Die ersten drei Bedingungen sind somit stets erfüllt und das Deadlock-Handling im Bereich der FTS setzt an der vierten Bedingung an.

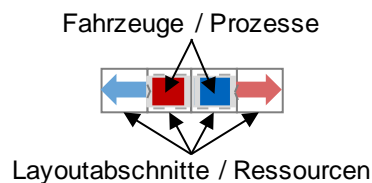


Abbildung 3-4: Deadlock zwischen zwei Fahrzeugen.

In der Literatur werden hinsichtlich des Umgangs mit Deadlocks drei generische Ansätze unterschieden [Cof-1971], [Liu-2001]:

- Deadlock-Prävention,
- Deadlock-Vermeidung sowie
- Erkennen und Auflösen von Deadlocks.

Lehmann veranschaulicht diese unterschiedlichen Ansätze anhand einspuriger Brücken, welche über einen Fluss führen. Begegnen sich zwei Fahrzeuge auf einer dieser Brücken, entsteht ein Deadlock. Die Deadlock-Prävention ist ein statischer Offline-Ansatz. Auf Basis generischer Regeln wird verhindert, dass ein Deadlock eintritt. Werden den Fahrspuren auf den Brücken eindeutige Fahrtrichtungen zugewiesen, können sich die Fahrzeuge auf einer Brücke nicht begegnen und es kommt nie zu einem Deadlock. Bei der Deadlock-Vermeidung werden die Ressourcen hingegen auf Basis des aktuellen Systemzustandes dynamisch derart zugewiesen, dass es nicht zu einem Deadlock kommen kann. Durch den Einsatz einer Ampelanlage können den Fahrspuren auf den Brücken temporär und je nach Bedarf eindeutige Fahrtrichtungen zugewiesen werden, wodurch sich Deadlocks vermeiden lassen. Das Ziel der dritten Strategie ist es nicht, Deadlocks zu verhindern. Sobald jedoch ein Deadlock vorliegt, muss dieser erkannt und durch entsprechende Maßnahmen aufgelöst werden. In dem Brückenbeispiel muss dafür eines der beiden Fahrzeuge die Brücke rückwärts verlassen. [Leh-2006, S. 104f.]

Im Bereich der FTF-basierten Kommissioniersysteme ist der einfache Ansatz Deadlocks zuzulassen, diese zu erkennen und anschließend aufzulösen, aufgrund der hohen Fahrzeugdichte nicht praktikabel. Bezogen auf das vorangegangene Beispiel stößt diese Strategie an ihre Grenze, wenn hinter den beiden sich begegnenden Fahrzeugen bereits weitere Fahrzeuge warten.

Die Deadlock-Prävention ist ein restriktiver Ansatz, welcher die Auslastung der Ressourcen einschränkt. Zudem können in FTF-basierten Kommissioniersystemen Deadlocks auch durch die Vermeidung von bidirektionalem Verkehr nicht ausgeschlossen werden, wie Abbildung 3-5 veranschaulicht. Durch die Vorgabe von Fahrtrichtungen entlang der einzelnen Layoutabschnitte werden Deadlocks ausgeschlossen, bei welchen sich zwei Fahrzeuge gegenüberstehen. Dennoch können sich Kreisschlüsse von vier oder mehr Fahrzeugen ergeben, bei denen keines der Fahrzeuge seine ursprünglich geplante Route fortsetzen kann. Ein Fahrzeug muss seine ursprünglich geplante Route ändern und den Deadlock auflösen.

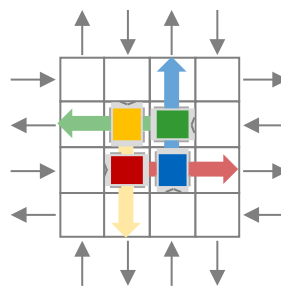


Abbildung 3-5: Deadlock durch einen Kreisschluss von vier Fahrzeugen.

Die Vermeidung von Deadlocks erlaubt eine hohe Ressourcenauslastung, erfordert jedoch komplexere Strategien. Erfolgt die Deadlock-Vermeidung auf einer festen, im Voraus bestimmten Route, so lässt sich das Problem der verklemmungsfreien Ausführung in ein Job-Shop-Scheduling-Problem überführen [Hat-2001]. Der sogenannte Banker's Algorithmus steuert Fahrzeuge entlang einer berechneten Route derart, dass keine Deadlocks entstehen [Kal-2011].

Ein Ansatz, bei welchem mögliche Deadlocks bereits während der Planung der Route berücksichtigt werden, ist das zeitfensterbasierte Routing. Bei dieser Methode werden für die Fahrzeuge vor dem Beginn einer Fahrt auf jedem Knoten entlang einer Route Zeitintervalle – die sogenannten Zeitfenster – reserviert, zu welchen die Knoten durch die Fahrzeuge belegt sind. Muss ein neues Fahrzeug geroutet werden, wird nach einer Route durch die freien Zeitfenster gesucht. Das Konzept wurde von *Kim und Tanchoco* eingeführt [Kim-1991] und hinsichtlich der Laufzeit von *ter Mors et al.* verbessert [Mor-2007].

Für das zeitfensterbasierte Routing wird der sogenannte Zeitfenstergraph erstellt: Jedes freie Zeitfenster auf einem Knoten im Layoutgraphen wird durch einen Knoten im Zeitfenstergraphen repräsentiert. Eine gerichtete Kante zwischen zwei Zeitfensterknoten existiert genau dann, wenn das eine freie Zeitfenster aus dem anderen freien Zeitfenster heraus erreichbar ist. Die Voraussetzung dafür ist, dass die entsprechenden

Knoten auch im Layoutgraphen durch eine Kante verbunden sind sowie dass sich die beiden Zeitfenster hinreichend überlappen. Das Routing erfolgt durch die Suche nach dem schnellsten Weg auf dem Zeitfenstergraphen, wobei der A*-Algorithmus verwendet wird.

Mit dem zeitfensterbasierten Ansatz, welcher auch als *Context Aware Route Planning* (CARP) bezeichnet wird, wurden bessere Resultate als mit Scheduling Ansätzen auf statischen Routen erzielt [Mor-2010]. Dieses Ergebnis konnte bei der Simulation eines Shuttle-Lagers bestätigt werden [Pen-2014]. Das zeitfensterbasierte Routing wurde in unterschiedlichen Anwendungsbereichen eingesetzt, wie beispielsweise dem Routing von:

- Flugzeugen zur Optimierung des Flughafen-Boden Rollverkehrs [Bus-2004],
- FTF für den Containertransport im Hafenbetrieb [Ste-2008],
- FTF zur Produktionsversorgung in einer flexiblen Fertigungsumgebung [Ble-2017],
- Shuttle-Fahrzeugen in einem Shuttle-Lager [Pen-2014],
- Unterfahr-FTF in einem Kommissioniersystem [Hvě-2018],
- Agenten, die sich ganz allgemein über eine Infrastruktur bewegen [Mor-2009].

Mit dem zeitfensterbasierten Routing ist es möglich, für ein Fahrzeug die schnellste Route zum Ziel zu identifizieren, welche unter der Berücksichtigung der bestehenden Reservierungen bereits gerouteter Fahrzeuge möglich ist. Die Reihenfolge, in welcher das Routing für mehrere Fahrzeuge erfolgt, hat einen Einfluss auf das Gesamtergebnis. Denn während das erste zu routende Fahrzeug auf keine Reservierungen Rücksicht nehmen muss, sind bei allen weiteren zu routenden Fahrzeugen die benötigten Zeitfenster der bereits gerouteten Fahrzeuge reserviert. Das Routing erfolgt strikt sequentiell.

3.1.4 Strategie für Fahrzeuge ohne zugewiesene Aufträge

Beendet ein Fahrzeug einen Auftrag und bekommt es nicht unmittelbar einen Folgeauftrag zugewiesen, muss entschieden werden, wo das Fahrzeug auf den nächsten Auftrag wartet (engl. *idle vehicle management*). Dabei können unterschiedliche Zielsetzungen verfolgt werden, wie beispielsweise

- die Minimierung der maximalen Anfahrdauer eines Fahrzeuges,
- die Minimierung der mittleren Anfahrdauer aller Fahrzeuge sowie
- eine gleichmäßige Verteilung der Fahrzeuge über das Layout [Egb-1993].

Ein Ansatz zur Erreichung dieser Zielsetzungen besteht darin, Fahrzeuge ohne Aufträge auf vorgesehenen Parkplätzen zu puffern. Dabei sind die Anzahl sowie die Anordnung der Parkplätze im Layout zu bestimmen. Eine einfache Strategie, welche ohne die Verwendung zusätzlicher Parkplätze im Layout auskommt, besteht darin, Fahrzeuge an dem Ort zu puffern, an welchem sie den letzten Auftrag beendet haben. Ein Nachteil dieser Strategie ist, dass möglicherweise andere Fahrzeuge durch parkende Fahrzeuge blockiert werden und sogenannte Verdrängungsfahrten ausgeführt werden müssen.

Um den im Abschnitt 2.3.1 erwähnten Vorteil der leichten Skalierbarkeit hinsichtlich des Durchsatzes von FTF-basierten Kommissioniersystemen zu nutzen, kommt den eingesetzten Strategien eine wichtige Bedeutung zu. Vorübergehend nicht benötigte Fahrzeuge müssen in Betriebszeiten mit niedrigerer Auftragslast entsprechend gepuffert werden. Für die Bestimmung der notwendigen Anzahl an Fahrzeugen (Abschnitt 3.1.7) ist dies hingegen nicht relevant. Durch einen Überhang von Aufträgen sind sämtliche Fahrzeuge im System stets beschäftigt.

3.1.5 Störungsmanagement

Strategien für den Umgang mit Störungen kommen dann zum Einsatz, wenn ein Fahrzeug aufgrund einer Störung ausfällt. In der Literatur werden Verfügbarkeiten der Fahrzeuge vernachlässigt und bei der Planung und Auslegung von FTS für gewöhnlich von einem Idealzustand ohne Störungen ausgegangen. Bei einer geringen Fahrzeugdichte hat der Ausfall eines Fahrzeuges eine beschränkte Wirkung auf andere Fahrzeuge. Bei einer hohen Fahrzeugdichte hingegen können Störungen zu Staus und Blockaden führen und den Durchsatz erheblich beeinflussen. [Ebb-2001, S. 183], [Vis-2006]

FTF-basierte Kommissioniersysteme, welche für gewöhnlich eine hohe Fahrzeugdichte aufweisen, bieten – wie bereits im Abschnitt 2.3.1 aufgezeigt – eine erhöhte Redundanz gegenüber regelbediengerätbasierten Automatisierungslösungen. Voraussetzung hierfür sind jedoch geeignete Strategien für den Umgang mit Störungen, um einen reibungslosen Betrieb der verbleibenden Fahrzeugflotte im Störfall zu ermöglichen. Darüber hinaus sind für die Bestimmung der notwendigen Anzahl an Fahrzeugen (Abschnitt 3.1.7) die Verfügbarkeiten der einzelnen Fahrzeuge zu berücksichtigen, da anderenfalls die notwendige Flottengröße zu gering berechnet wird.

3.1.6 Batteriemangement

Für gewöhnlich werden die Fahrzeuge über einen elektrischen Motor angetrieben. Die Energie hierfür wird über Batterien oder über sogenannte Superkondensatoren (Supercaps) bereitgestellt. Für das Nachladen stehen mehrere technische Alternativen

zur Verfügung: Die Batterien können getauscht, wie Supercaps an stationären Punkten oder aber während der Fahrt über Ladeschienen geladen werden. Für den Batterietausch sind Vorrichtungen sowie eine entsprechende Anzahl an Batterien notwendig. Beim Nachladen an stationären Punkten im Layout, stehen die Fahrzeuge in Abhängigkeit der eingesetzten Technologie für einen bestimmten Zeitraum nicht zur Verfügung. Der Einsatz von Ladeschienen erfordert zusätzliche Infrastruktureinrichtungen, beeinflusst den operativen Ablauf jedoch nicht. [Ebb-2001, S. 170f.]

Der Aspekt des Batteriemangements kann unter verschiedenen Bedingungen bei der Planung von FTS vernachlässigt werden. Werden Systeme betrachtet, welche schichtweise betrieben werden und ein Batteriezyklus genügend Energie für eine gesamte Schicht bereitstellt, kann der Wechsel oder das Laden der Batterie zwischen den Schichten erfolgen. In Systemen, in welchen die Auslastung der Fahrzeuge entsprechend gering ist, kann die Wartezeit zwischen zwei Aufträgen zum Wechseln oder Laden der Energiespeicher genutzt werden. In Systemen, welche über Ladeschienen verfügen, können bei entsprechender Anzahl und Anordnung dieser Schienen die Batterien im laufenden Betrieb geladen werden. [McH-1995]

Im Kontext der FTF-basierten Kommissioniersysteme ist ein Schichtbetrieb üblich [Le-2006]. Darüber hinaus entstehen im Kommissionierbetrieb an bestimmten Punkten immer wieder Wartezeiten, wie beispielsweise während der Bereitstellung am Kommissionierplatz, welche für Ladevorgänge genutzt werden können. Das Batteriemangement wird daher im Rahmen dieser Arbeit nicht weiter berücksichtigt.

3.1.7 Flottengröße

Bei der Planung eines FTS muss die Anzahl der Fahrzeuge bestimmt werden, welche notwendig ist, um alle anstehenden Transportaufträge fristgerecht zu erfüllen. Dafür existieren in der Literatur Ansätze, welche sich hinsichtlich des Aufwands sowie der Genauigkeit unterscheiden. Mit der Qualität der Planung wird das Investitionsvolumen, das maßgeblich über die Fahrzeuganzahl bestimmt wird, unmittelbar beeinflusst. Fehler bei der Fahrzeugbedarfsermittlung führen zu erheblichen nachträglichen Kosten für zusätzliche Fahrzeuge, falls die Fahrzeuganzahl zu gering berechnet wurde. Eine zu hoch angesetzte Anzahl an Fahrzeugen hat einen negativen Einfluss auf die Wirtschaftlichkeit und kann nach einem Vergleich mit anderen Automatisierungslösungen die Realisierung eines FTS-Projektes verhindern. [Sch-2000, S. 2f.]

Als wesentliche Einflussfaktoren auf die notwendige Fahrzeuganzahl werden in der Literatur neben dem Transportaufkommen das zugrundeliegende Layout, die Auftragsvergabestrategien, das Verkehrsaufkommen einschließlich der gegenseitigen

Blockierungen, Verfügbarkeiten sowie die Kapazität und Geschwindigkeit der Fahrzeuge (die FTF-Parameter) genannt [Mül-1983, S. 122], [Sch-1989, S. 8f.], [Sch-2000, S. 50], [Le-2006], [Vis-2006].

Das Verkehrsaufkommen sowie die wechselseitigen Blockierungen zwischen den Fahrzeugen werden neben dem zugrundeliegenden Layout sowie dem Transportaufkommen maßgeblich durch das Routing und Deadlock-Handling bestimmt. Darüber hinaus können bestimmte Betriebsbedingungen, wie beispielsweise eine Bereitstellung der Transporteinheiten in einer vordefinierten Sequenz, einen Einfluss auf den Durchsatz und somit auf die notwendige Flottengröße haben. Somit ergeben sich – zusammenfassend und wie in den vorhergehenden Abschnitten angedeutet – die in Abbildung 3-6 dargestellten Haupteinflussfaktoren auf die notwendige Flottengröße.

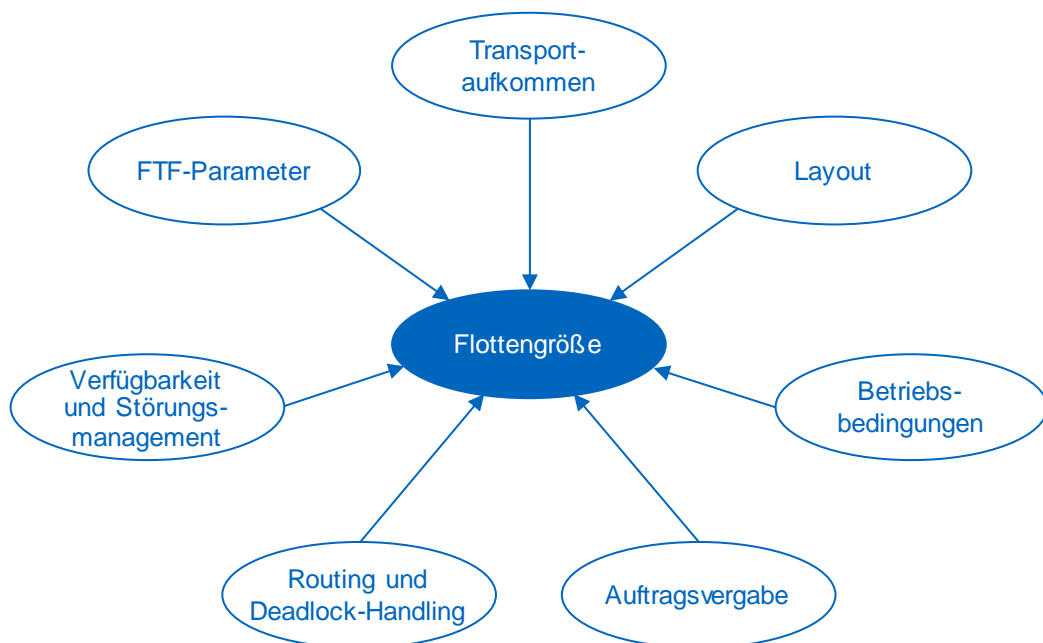


Abbildung 3-6: Einflussfaktoren auf die notwendige Flottengröße.

Großeschallau formuliert ein einfaches, analytisches Vorgehen zur Abschätzung der Flottengröße, welches darauf basiert, die gesamte Nutzungsdauer der Fahrzeuge pro Periode durch die Länge des Betrachtungszeitraums zu dividieren [Gro-1984, S. 158ff.]. Die Nutzungsdauer der Fahrzeuge setzt sich aus den Zeitdauern für Transportfahrten, Leerfahrten sowie Lastaufnahme- und -abgabevorgänge zusammen. Zunächst ist eine kürzeste Fahrzeitenmatrix zwischen allen relevanten Start-Ziel-Relationen zu ermitteln. Über die Transportmatrix, welche die Anzahl der Transportaufträge zwischen den verschiedenen Knoten im Betrachtungszeitraum enthält, lässt sich die Dauer der Transportfahrten bestimmen. Da sich für gewöhnlich nicht an jeden abgeschlossenen Transportauftrag ein weiterer Auftrag anschließt, dessen Startort dem Zielort des vorhergehenden Auftrags entspricht, entstehen zwangsläufig Leerfahrten.

Enden an einem Knoten mehr Transportaufträge als an diesem Knoten beginnen, so entsteht innerhalb des Betrachtungszeitraums ein Fahrzeugangebot an diesem Knoten. Hingegen entsteht ein Fahrzeugbedarf, falls an dem Knoten mehr Transportaufträge beginnen als enden. Über die Formulierung als klassisches Transportproblem [Dom-2015, S. 87ff.] lässt sich eine zeitoptimale Verteilung der Leerfahrten zum Ausgleich zwischen Angebot und Bedarf ermitteln. Abermals unter der Berücksichtigung der minimalen Fahrzeitenmatrix lässt sich somit nun die Dauer der Leerfahrten bestimmen. Zu beachten ist, dass bei diesem Vorgehen vorausgesetzt wird, dass an den Knoten keine Wartezeiten entstehen und Transportaufträge stets ausführbar sind, sobald die Fahrzeuge am Startort eintreffen.

Die Herausforderung bei analytischen Ansätzen besteht aufgrund der stochastischen Einflussfaktoren in der exakten Bestimmung der Leerfahrten zwischen zwei Transportaufträgen sowie der Quantifizierung der Blockierungseffekte zwischen den Fahrzeugen [Le-2006]. In der Literatur besteht daher der Konsens, dass die durch analytische Berechnungen bestimmte Flottengröße in jedem Fall durch die Durchführung einer Simulationsstudie überprüft werden muss [Mül-1983, S. 122], [Sch-1989, S. 2], [Gan-1998], [Le-2006], [Vis-2006] [Ull-2014, S. 223], [VDI-2710]. Auf den Einsatz der Simulation in der Planungsphase wird im anschließenden Unterkapitel 3.2 eingegangen.

Eine andere Sichtweise auf den Aspekt der Flottengröße ist die Transportleistung, welche mit einer gegebenen Anzahl an Fahrzeugen erzielt werden kann.

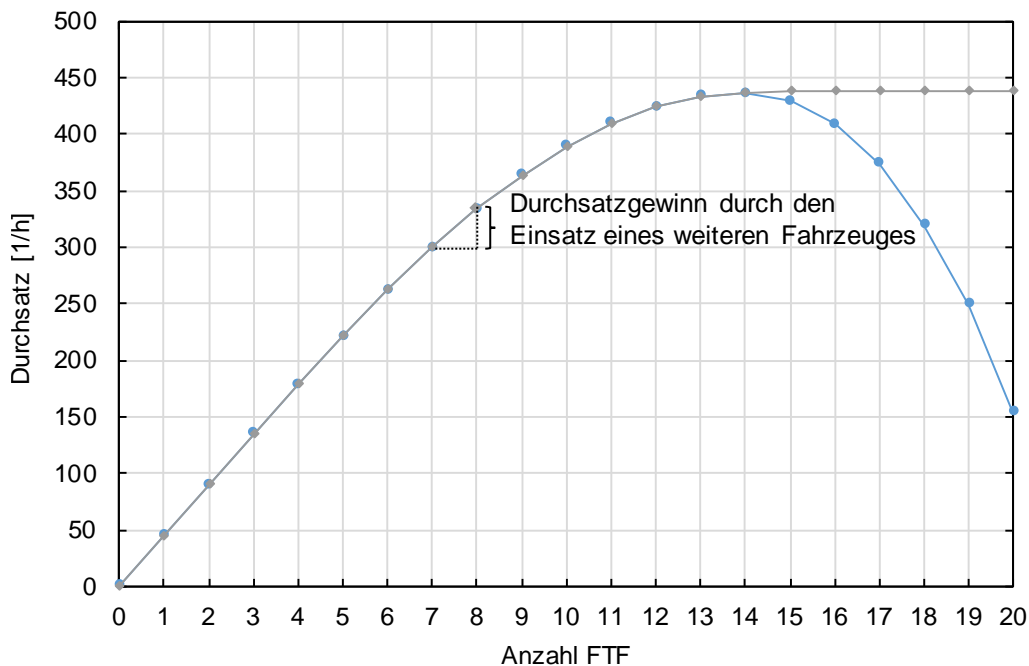


Abbildung 3-7: Beispielhafte Durchsatzverläufe: Der Durchsatz steigt zunächst nahezu linear an, flacht dann ab und stagniert (graue Kurve) oder sinkt dramatisch (blaue Kurve).

Im Kontext der FTF-basierten Kommissioniersysteme ist dies die Anzahl der Ein- und Auslagerungen, welche während eines bestimmten Zeitraums durchgeführt werden können. Es ist naheliegend, dass der Durchsatz mit zunehmender Anzahl an Fahrzeugen steigt. Jedoch nehmen auch das Verkehrsaufkommen und insbesondere die Blockierungseffekte zwischen den Fahrzeugen zu, je mehr Fahrzeuge eingesetzt werden. Aufgrund der hohen Fahrzeugdichte ist dieser Effekt stärker als beispielsweise bei einer herkömmlichen Produktionsversorgung. Somit sinkt die zusätzlich gewonnene Leistung je zusätzlich eingesetztem Fahrzeug, bis schließlich durch den Einsatz weiterer Fahrzeuge überhaupt kein Durchsatzgewinn mehr erzielt werden kann und die Grenzleistung des Gesamtsystems erreicht wird (Abbildung 3-7, graue Kurve). Anzumerken ist, dass diese Grenzleistung oftmals durch periphere Systeme gegeben ist, wie beispielsweise durch die Anzahl und Leistung der Lifte in einem Shuttle-System. Denkbar ist auch, dass der Durchsatz ab einer bestimmten Fahrzeuganzahl wieder abnimmt, der zusätzliche Durchsatzgewinn je zusätzlich eingesetztem Fahrzeug somit negativ wird (Abbildung 3-7, blaue Kurve). Die Ermittlung des Durchsatzgewinns je zusätzlich eingesetztem Fahrzeug ist insbesondere hinsichtlich der Skalierung von FTF-basierten Kommissioniersystemen relevant, um etwaige Durchsatzsteigerungen und die damit verbundenen Investitionen abschätzen zu können.

Unter dem Begriff Durchsatzanalyse wird in dieser Arbeit die Ermittlung der charakteristischen Durchsatzverläufe verstanden, welche zunächst linear verlaufen, dann abflachen und möglicherweise wieder fallen.

3.2 Einsatz von Simulation bei der Planung von FTS

Wie im Abschnitt 3.1.7 dargelegt, ist der Einsatz von Simulation für die Bestimmung der Flottengröße unabdingbar. Der Aufwand für die Durchführung einer Simulationsstudie ist erheblich, jedoch eröffnet ein einmal erstelltes Simulationsmodell neben der Bestimmung der Flottengröße weitere Möglichkeiten. Die Simulation erlaubt grundsätzlich die Überprüfung des logistischen Konzeptes hinsichtlich der Leistungsfähigkeit (siehe Abschnitt 2.4.1). Darüber hinaus ermöglicht die Simulation

- die Analyse des Systemverhaltens unter verschiedenen Betriebsbedingungen, einschließlich Ausfällen von Fahrzeugen und Notfallstrategien,
- die Untersuchung unterschiedlicher Layoutvarianten,
- die Untersuchung unterschiedlicher Steuerungsstrategien hinsichtlich der Auftragsvergabe, der Leerfahrten und des Routings,
- die Analyse der Ressourcennutzung, beispielsweise von einzelnen Layoutabschnitten und Pufferplätzen,

- die Analyse des Startverhaltens sowie
- die Identifikation von System- und Strategiefehlern [Mül-1983, S. 122f.], [VDI-2710].

Diesen Potentialen gegenüber steht der hohe Aufwand, welcher durch die Erstellung eines Simulationsmodells sowie der Durchführung einer Simulationsstudie entsteht [Mül-1983, S. 123], [Sch-2000, S. 49f.], [Vis-2006], [VDI-2710]. In der Literatur wurden für unterschiedliche Anwendungsfälle entsprechende Simulationsmodelle erstellt und Simulationsstudien mit verschiedenen Zielsetzungen durchgeführt. Im Folgenden wird zunächst auf relevante Arbeiten zur simulationsbasierten Durchsatzanalyse von FTS im Allgemeinen (Abschnitt 3.2.1) eingegangen, bevor Ansätze vorgestellt werden, welche sich auf die Simulation von FTF-basierten Kommissioniersystemen fokussieren (Abschnitt 3.2.2).

3.2.1 Simulationsbasierte Durchsatzanalyse von FTS

Raabe erweitert ein auf Funktionsbausteinen (Fertigen, Montieren, Prüfen, Fördern und Lagern) basierendes Simulationssystem um die Modellierung des Layouts durch einen Graphen, um Transportzeiten detaillierter zu berücksichtigen. Diese Erweiterung gegenüber den genannten Funktionsbausteinen ermöglicht die Simulation eines FTS. Bei der Modellierung der Fahrzeuge werden Beschleunigungs- und Verzögerungsvorgänge berücksichtigt. Für die Auswahl der Routen werden die Kanten entsprechend ihrer mittleren Belegung gewichtet. Deadlocks werden vermieden, indem während der Ausführung der Routen sichergestellt wird, dass einzelne Knoten oder Regionen (Zusammenschlüsse mehrerer Knoten) nur jeweils von einem Fahrzeug zur gleichen Zeit belegt werden. Ein regionenübergreifendes Deadlock-Handling wird hingegen nicht beschrieben. Im Rahmen einer Fallstudie wird ein FTS für den Transport von Kleinteilebehältern zur Produktionsversorgung betrachtet. Dabei werden die Auswirkungen unterschiedlicher Fahrzeuganzahlen auf den Durchsatz, die Durchlaufzeiten sowie die Behälterliegezeiten analysiert. [Rab-2002]

Schrecker erstellt ein Simulationsmodell zur Evaluation unterschiedlicher Strategien für die Optimierung der Fahrkursorientierung in unidirektionalen Topologien. Ein zweiter Schwerpunkt liegt auf dem Vergleich von Auftragsvergabestrategien. Dabei werden Fahrzeuge betrachtet, welche nicht nur eine Transporteinheit, sondern bis zu vier Transporteinheiten gleichzeitig transportieren können. In unidirektionalen Fahrkursen erfolgt das Routing auf Basis der kürzesten Wege. In bidirektionalen Fahrkursen wird hingegen auf das zeitfensterbasierte Routing zurückgegriffen, wobei Beschleunigungs- und Verzögerungsvorgänge nicht berücksichtigt werden. In unterschiedlichen

Simulationsstudien werden die verschiedenen Ansätze unter variierender Fahrzeuganzahl verglichen. [Sch-2000]

Maza und Castagna gehen der Frage nach, wie geplante, zeitfensterbasierte Routen auch dann robust ausgeführt werden können, wenn sich einzelne Fahrzeuge verspäten und nicht die reservierten Zeitfenster treffen. Sie zeigen, dass die Abwesenheit von Deadlocks garantiert werden kann, sofern die Reihenfolge auf den Knoten eingehalten wird, welche sich durch die Reservierungen der Zeitfenster ergibt. Unter bestimmten Voraussetzungen können Fahrzeuge in der Reihenfolge sogar vorgezogen werden, damit diese nicht auf verspätete Fahrzeuge warten müssen. In einer Simulationsstudie wird ein Layout mit 45 Knoten betrachtet, auf welchem bis zu acht Fahrzeuge Transportaufträge durchführen. Die Fahrzeuge werden künstlich verzögert und die Auswirkungen auf den Durchsatz analysiert. [Maz-2005]

3.2.2 Simulationsbasierte Durchsatzanalyse von FTF-basierten Kommissioniersystemen

Penners betrachtet eine einzelne Ebene eines Shuttle-Systems. Die Modellierung der Lagerebene erfolgt stark vereinfacht, denn jeder Abschnitt einer Lagergasse zwischen zwei Quergassen oder einer Quergasse und dem Gassenende wird als eine Resource abgebildet, auf welcher sich nur ein Fahrzeug zur gleichen Zeit befinden kann. Das Modell ermöglicht die Berücksichtigung von Beschleunigungs- und Verzögerungsvorgängen, sofern die Modellierung des Layouts derart erfolgt, dass die einzelnen Layoutabschnitte, welche durch die Knoten eines Graphen repräsentiert werden, in jeder Fahrtrichtung die gleiche Länge aufweisen. In einer Simulationsstudie werden zwei unterschiedliche Routingstrategien verglichen. Durch die Anwendung des zeitfensterbasierten Routings lässt sich ein höherer Durchsatz erzielen als mit dem Ansatz, Deadlocks auf dem kürzesten Weg durch ein entsprechendes Scheduling der Fahrzeuge zu vermeiden. Darüber hinaus werden unterschiedliche Layoutkonfigurationen (Anzahl und Fahrtrichtung der Quergassen) unter variierender Fahrzeuganzahl untersucht. [Pen-2014]

Roy et al. erstellen ein Simulationsmodell einer Lagerebene eines Shuttle-Lagers mit einer Quergasse mit dem Ziel, Blockierungseffekte zwischen den Shuttle-Fahrzeugen zu quantifizieren. Die Umsetzung des Deadlock-Handlings besteht in der Umsetzung folgender dreier Maßnahmen. Die Quergasse kann zu jedem Zeitpunkt lediglich entlang einer Fahrtrichtung befahren werden. Begegnen sich zwei Shuttle-Fahrzeuge innerhalb einer Lagergasse, so verfährt das hintere Fahrzeug bis zum Gassenende und wartet, bis das vordere Fahrzeug das Lagerspiel vollzogen hat. Diese Ausweichfahrten können sich mehrmals direkt nacheinander wiederholen. Um Deadlocks auf Kreuzungen zu vermeiden, warten Shuttle-Fahrzeuge auf Pufferplätzen zwischen Lager-

Quergasse, bis sie die Quergasse in der gewünschten Richtung befahren können. Durch die Anwendung dieser Strategien lassen sich jedoch nicht alle Deadlocks vermeiden oder auflösen. Muss das vordere Fahrzeug in einer Lagergasse etwa den hintersten Lagerplatz anfahren, stoßen diese einfachen Strategien an ihre Grenzen. Im Rahmen einer Simulationsstudie werden die Blockierungseffekte auf unterschiedlichen Layouts unter variierender Fahrzeuganzahl analysiert. [Roy-2016]

Schloz et al. simulieren ebenfalls eine Ebene eines Shuttle-Systems. Das Ziel der Arbeit besteht in der Optimierung der Reihenfolge, in welcher Auslageraufträge durchgeführt werden. Dafür wird auf die Methode des Maschinellen Lernens zurückgegriffen. Deadlocks zwischen zwei Fahrzeugen werden zugelassen und durch das Anwenden verschiedener Strategien aufgelöst. Dafür wird, wie in [Roy-2016], unterschieden, ob ein Deadlock in einer Quergasse, in einer Lagergasse oder aber auf einer Kreuzung erfolgt. Deadlock-Handling Strategien für mehr als zwei Fahrzeuge werden nicht beschrieben. [Sch-2019]

Kumar et al. betrachten ein gesamtes Shuttle-System mit mehreren Ebenen, bei dem die Besonderheit darin besteht, dass die Lifte entweder leere Shuttle-Fahrzeuge oder aber Lagereinheiten vertikal fördern, nie aber beladene Shuttle-Fahrzeuge. Wechselwirkungen zwischen den Shuttle-Fahrzeugen auf einer Ebene werden nicht modelliert. In einer Simulationsstudie werden zwei verschiedene Anordnungen der Quergassen auf den Lagerebenen verglichen. [Kum-2014]

Ekren et al. simulieren ebenfalls ein gesamtes Shuttle-System. In einer Simulationsstudie wird zunächst die Anzahl der Shuttle-Fahrzeuge sowie die Anzahl der Lifte variiert. Dabei sind die Shuttle-Fahrzeuge stets einem Lift fest zugeordnet. Anschließend werden unterschiedliche Leerfahrtstrategien und Auftragsvergabestrategien verglichen sowie alternative Anordnungen der Ein- und Auslagerebene untersucht. Es ist nicht ersichtlich, wie Routing und Deadlock-Handling der Shuttle-Fahrzeuge erfolgen. [Ekr-2010]

In einer Reihe weiterer Arbeiten (beispielsweise [Ekr-2014], [Roy-2015] und [Ler-2018]) erfolgt die Durchsatzanalyse von Shuttle-Systemen mit analytischen Modellen, vorzugsweise aus dem Bereich der Warteschlangentheorie. Diese Modelle werden durch Simulationsmodelle überprüft. Auf das Routing und Deadlock-Handling wird in diesen Arbeiten nicht eingegangen.

Merschformann et al. beschreiben den Aufbau eines agentenbasierten Simulationsmodells zur Analyse von RMFS [Mer-2018]. Dieses Modell berücksichtigt die Be-

schleunigung und Verzögerungsvorgänge der einzelnen Fahrzeuge. In einer Simulationsstudie werden unterschiedliche Routingalgorithmen miteinander verglichen [Mer-2017]. Den höchsten Durchsatz erzielt der Algorithmus *Windowed Hierarchical Cooperative A** (WHCA*) [Sil-2005]. Bei diesem Ansatz wird die Routenberechnung für mehrere Fahrzeuge parallelisiert: Die zeitfensterbasierten Routen werden nicht vollständig, sondern lediglich abschnittsweise für ein vorgegebenes Zeitintervall bestimmt. Ist dieses verstrichen, werden die nächsten Routenabschnitte für das nächste Zeitintervall berechnet. Dabei wird die Reihenfolge, in welcher die Routen für die einzelnen Fahrzeuge berechnet werden, variiert. Zur Bestimmung dieser Reihenfolge wird eine vergebene Priorität, der Beladungszustand der Fahrzeuge sowie die Entfernung zum Ziel herangezogen.

Hvězda et al. betrachten ebenfalls das Problem, mehrere Fahrzeuge in einem RMFS simultan zu routen. Wie im Abschnitt 3.1.3 hat die Reihenfolge, in welcher das zeitfensterbasierte Routing der einzelnen Fahrzeuge erfolgt, einen Einfluss auf das Gesamtergebnis. Das optimale Gesamtergebnis kann beispielsweise über eine vollständige Enumeration bestimmt werden. Allerdings existieren bereits bei zehn Fahrzeugen über drei Millionen verschiedene Kombinationen, in welcher das Routing erfolgen kann. In einer Simulationsstudie werden Heuristiken verglichen, welche unterschiedliche Strategien bei der Reihenfolgenbildung verfolgen. [Hvě-2018]

3.3 Detaillierung der Forschungslücke

In der Literatur existieren – wie im vorhergehenden Unterkapitel 3.2 dargelegt – verschiedene Ansätze für die simulationsbasierte Durchsatzanalyse von FTF-basierten Kommissioniersystemen. Zusammenfassend lässt sich festhalten:

- Die Ansätze beziehen sich stets auf ein bestimmtes System und lassen sich nicht unmittelbar auf andere Systeme oder Systemtypen (Shuttle-System oder RMFS) übertragen.
- Das Deadlock-Handling wird insbesondere im Bereich der Shuttle-Systeme überhaupt nicht oder nur unzureichend betrachtet.
- Beschleunigungs- und Verzögerungsvorgänge der Fahrzeuge werden überwiegend vernachlässigt.
- Der Detaillierungsgrad bei der Modellierung des Layouts ist überwiegend hoch.
- Verfügbarkeiten der Fahrzeuge sowie Strategien für den Umgang mit Störungen werden im Bereich der FTF-basierten Kommissioniersysteme nicht betrachtet.
- Bereitstellungen in einer vordefinierten Auftragssequenz werden ebenfalls nicht berücksichtigt.

Im Bereich der FTF-basierten Kommissioniersysteme setzen sich bestehende Ansätze entweder mit Shuttle-Systemen oder mit RMFS auseinander. Diese beiden Systemtypen unterscheiden sich grundlegend hinsichtlich der Art und Weise, wie die Lastwechsel vollzogen werden. Während in RMFS die Fahrzeuge die Transporteinheiten unterfahren und anheben, positionieren sich Shuttle-Fahrzeuge vor dem Lagerplatz, um die Transporteinheit über ein Lastaufnahmemittel aufzunehmen. Da die Shuttle-Fahrzeuge für gewöhnlich länger als die Lagerplätze breit sind, ist es in Shuttle-Systemen im Gegensatz zu RMFS nicht möglich, dass zwei nebeneinander gelagerte Transporteinheiten zeitgleich durch zwei unterschiedliche Fahrzeuge aufgenommen werden. Darüber hinaus existieren auch innerhalb der beiden Systemtypen Unterschiede, beispielsweise hinsichtlich der Art und Weise, wie Fahrzeuge einen Wechsel der Fahrtrichtung um 90 Grad ausführen. In manchen Systemen vollziehen die Fahrzeuge dafür eine 90 Grad Rotation um die eigene Achse, in anderen Systemen verfügen die Fahrzeuge wiederum über einen Mechanismus zum seitlichen Verfahren. Diese Unterschiede lassen eine unmittelbare Übertragbarkeit der existierenden Ansätze auf andere Systeme und Systemtypen nicht zu. Wünschenswert ist ein generischer Ansatz, welcher die unterschiedlichen Systemeigenschaften berücksichtigt und sich bei der Betrachtung möglichst vieler Systeme verwenden lässt.

In der Mehrzahl der Arbeiten, welche ein zuverlässiges Deadlock-Handling berücksichtigen, wird auf das zeitfensterbasierte Routing nach ter Mors et al. [Mor-2007] zurückgegriffen. Da dieser Algorithmus keine Beschleunigungs- und Verzögerungsvorgänge berücksichtigt, werden die Beschleunigungs- und Verzögerungsvorgänge in den entsprechenden Simulationsmodellen ebenfalls gar nicht oder – wie in [Pen-2014] – nur eingeschränkt berücksichtigt. Da die Beschleunigungs- und Verzögerungsvorgänge in Systemen mit einer hohen Fahrzeugdichte jedoch einen nicht zu vernachlässigenden Einfluss auf die Durchlaufzeiten haben, sollten diese im Rahmen einer Durchsatzanalyse berücksichtigt werden.

Grundlage für die Anwendung des zeitfensterbasierten Routings ist die Modellierung des Layouts durch einen Graphen. Dafür ist das Layout derart in einzelne Knoten zu zerlegen, dass jeder Knoten ein Fahrzeug vollständig aufnehmen kann. Im Bereich der RMFS ist diese Zerlegung einfach umsetzbar und durch die Lagerplätze vorgegeben. Im Bereich der Shuttle-Systeme hingegen stellt die Zerlegung eine bislang ungelöste Herausforderung dar. Shuttle-Fahrzeuge sind – wie zuvor erwähnt – für gewöhnlich länger als die Lagerplätze breit sind, somit kann nicht jedem Lagerplatz ein Knoten zugeordnet werden und die Umsetzung des zeitfensterbasierten Routings ist anderweitig zu lösen.

Im Bereich der FTF-basierten Kommissioniersysteme werden Verfügbarkeiten und die damit einhergehenden Ausfälle von Fahrzeugen nicht berücksichtigt. Selbst die Ansätze von Maza und Castagna [Maz-2005] lassen sich nicht unmittelbar in einem Simulationsmodell implementieren, welches die Beschleunigung und Verzögerungsvorgänge der Fahrzeuge berücksichtigt. Grundlage für die Ausführung der entwickelten Strategien ist die stete Abfrage, ob der nächste Knoten entlang der Route eines Fahrzeuges durch das Fahrzeug befahren werden darf. Ist ein vorhergehendes Fahrzeug verspätet und hat den nächsten Knoten noch nicht freigegeben, so muss das Fahrzeug unmittelbar zum Stillstand kommen und auf dem aktuellen Knoten warten. Eine Verzögerungsstrecke wird dabei nicht berücksichtigt.

In keiner Arbeit, weder FTS im Allgemeinen betreffend, noch im Bereich der FTF-basierten Kommissioniersysteme, wird eine Sequenzierung berücksichtigt, obwohl eine Auslagerung in Sequenz eine häufige Anforderung im Bereich der Kommissionierung ist und diese erhebliche Auswirkung auf den erzielbaren Durchsatz hat.

In der Tabelle 3-1 ist dargestellt, auf welche Aspekte in den unterschiedlichen Ansätzen eingegangen wird.

Tabelle 3-1: Arbeiten zur simulationsbasierten Durchsatzanalyse.

	Ansätze									
	FTS			Shuttle-System					RMFS	
	[Sch-2000]	[Rab-2002]	[Maz-2005]	[Ekr-2010]	[Pen-2014]	[Kum-2014]	[Roy-2016]	[Sch-2019]	[Mer-2018]	[Hvé-2018]
Detaillierungsgrad Layout	+	+	+	+	-	+	+	+	+	+
Deadlock-Handling	CARP		CARP		CARP				WHCA*	CARP
Beschleunigung und Verzögerung		x			(x)				x	
Verfügbarkeit und Störungen			(x)							
Sequenzierung										

Legende

x : berücksichtigt

(x) : eingeschränkt berücksichtigt

+/- : hoch / niedrig

Zusammenfassend bedarf es eines Vorgehens – einer Methodik – zur Simulation von FTF-basierten Kommissioniersystemen, welche

- für die Analyse unterschiedlicher Systemtypen herangezogen werden kann,
- das Layout in einem hohen Detaillierungsgrad abbildet,
- die Blockierungseffekte zwischen den Fahrzeugen abbildet,
- Beschleunigungs- und Verzögerungsvorgänge der Fahrzeuge berücksichtigt sowie
- die Verfügbarkeiten der Fahrzeuge modelliert.

Um eine ablauffähige Simulation und eine zielgerichtete Analyse zu ermöglichen, sollten in dieser Methodik Steuerungsstrategien berücksichtigt werden, welche

- einen hohen Durchsatz ermöglichen,
- einen robusten Betrieb sicherstellen (Umgang mit Deadlocks),
- mit Ausfällen und Verspätungen der Fahrzeuge umgehen können sowie
- eine Bereitstellung der Artikel in einer vordefinierten Sequenz ermöglichen.

Dafür ist zum einen eine geeignete Modellierung des Layouts sowie der Fahrzeuge zu entwickeln. Zum anderen sind entsprechende Steuerungsstrategien zu erarbeiten, um den zuvor erwähnten Anforderungen gerecht zu werden.

4 Konzept

Das Ziel der Arbeit besteht, wie im vorhergehenden Kapitel dargelegt, in der Entwicklung einer Methodik zur simulationsbasierten Durchsatzanalyse von FTF-basierten Kommissioniersystemen. Im Unterkapitel 4.1 wird zunächst auf die dafür erforderlichen Steuerungsstrategien eingegangen. Im anschließenden Unterkapitel 4.2 wird der übergeordnete Aufbau der Methodik erläutert und der Aufbau der nachfolgenden Kapitel vorgestellt.

4.1 Erforderliche Steuerungsstrategien

Im Unterkapitel 3.1 wurden die unterschiedlichen Aspekte, welche bei der Planung und Steuerung von FTS berücksichtigt werden müssen, vorgestellt. Zusammenfassend muss die Steuerung eines FTS im operativen Betrieb die folgenden Fragestellungen beantworten:

- Wann soll welches Fahrzeug wohin fahren? (Auftragsvergabe)
- Wie erreicht ein Fahrzeug schnell und sicher sein Ziel? (Routing und Deadlock-Handling)
- Wie verhält sich ein Fahrzeug, wenn es nach Beendigung eines Auftrags keinen neuen Folgeauftrag zugewiesen bekommt? (Leerfahrtstrategien)
- Wie kann eine die Bereitstellung in einer vordefinierten Sequenz sichergestellt werden? (Auftragsvergabe, Routing und Deadlock-Handling)
- Wie reagiert die verbleibende Fahrzeugflotte auf ausgefallene Fahrzeuge? (Störungsmanagement)

Die Auftragsvergabe kann anhand einfacher Regeln durchgeführt werden, lässt sich aber auch durch aufwändige Strategien optimieren (siehe Abschnitt 3.1.2). Für die Bestimmung des erzielbaren Durchsatzes ist von einer kontinuierlichen Auftragslast auszugehen, bei welcher sich der Auftragspool niemals leert. Sobald ein Fahrzeug einen Auftrag ausgeführt hat, bekommt es unmittelbar einen nächsten Auftrag zugewiesen. Die Auftragsvergabe erfolgt somit Fahrzeug-initiiert. Darüber hinaus sind aufgrund des Überhangs an Aufträgen keine Strategie für Fahrzeuge ohne zugewiesene Aufträge notwendig.

Das Routing und Deadlock-Handling erfordert bereits bei einer geringen Fahrzeuganzahl komplexe Strategien, um einen robusten Betrieb zu ermöglichen. Diese Strategien sind notwendig, um die Ablauffähigkeit eines Simulationsmodells sicherzustellen. Kommt es in der Simulation zu einem Deadlock und kann dieser aufgrund im Simulationsmodell fehlender Strategien nicht aufgelöst werden, so bricht die Simulation ab und es können keine Ergebnisse generiert werden. Um das Routing und Deadlock-Handling umzusetzen, wird auf den zeitenfensterbasierten Ansatz nach ter Mors et al. [Mor-2007] zurückgegriffen.

Wie im Abschnitt 3.1.3 erläutert erfolgt bei diesem Ansatz das Routing strikt sequentiell. Eine Optimierung, die das Routing mehrerer Fahrzeuge simultan betrachtet, wird nicht vorgenommen. Dennoch wird dieser Ansatz für FTF-basierte Kommissioniersysteme als geeignet erachtet. Die Situation, in welcher mehrere Fahrzeuge zum selben Zeitpunkt zu routen sind, ist im Normalbetrieb kein Regelfall. Vielmehr trifft immer mal wieder ein Fahrzeug im Lift auf einer Lagerebene ein oder ist nach der Beendigung eines Pickvorgangs abfahrbereit und muss geroutet werden.

Für die Umsetzung des zeitenfensterbasierten Routings ist das Layout durch einen Graphen zu modellieren. Dabei werden sowohl die Quellen, Senken und Kreuzungen als auch die Kanten, entlang welcher die Fahrzeuge verfahren können, durch Knoten repräsentiert. Diese Knoten, die auch als Ressourcen bezeichnet werden, sind durch unterschiedliche Längen charakterisiert. Die Kanten zwischen den Knoten haben hingegen keine Gewichtung und geben lediglich über die Vorgänger- und Nachfolgerbeziehungen der Knoten Auskunft. [Mor-2009, S. 32]

Zum einen ermöglicht der zeitenfensterbasierte Ansatz die Berechnung von zeiteffizienten, deadlockfreien Routen. Zum anderen dient das Konzept als Ausgangsbasis sowohl für die Sequenzierung als auch für die Notfallstrategien. Durch die Reservierung der Zeitefenster für die Fahrzeuge ist vor dem Beginn einer Fahrt bekannt, wann ein Fahrzeug planmäßig an seinem Ziel eintrifft. Somit ist auch die Reihenfolge bekannt, in welcher mehrere Fahrzeuge, die zu demselben Ziel geroutet worden sind, dort eintreffen. Soll die Bereitstellung an diesem Ziel in einer vorgegebenen Sequenz erfolgen, kann diese nach dem Routing eines Fahrzeuges überprüft und durch entsprechende Maßnahmen wie beispielsweise das erneute Routing mit einem späteren Abfahrzeitpunkt korrigiert werden.

Fällt ein Fahrzeug aus und blockiert es den Knoten, auf welchem es sich aktuell befindet, kann ebenfalls auf Basis der Zeitefenster reagiert werden. Zum einen können betroffene Fahrzeuge über die bestehenden Reservierungen auf dem blockierten Knoten identifiziert werden, zum anderen kann durch Reservierungen mit nach hinten offenen

Zeitfenstern verhindert werden, dass blockierte Knoten für das Routing anderer Fahrzeuge genutzt werden.

Das zeitfensterbasierte Routing wird im Rahmen der vorliegenden Arbeit derart erweitert, dass Beschleunigungs- und Verzögerungsvorgänge in den Routingprozess integriert werden. Dafür werden die Fahrbewegungen der Fahrzeuge nicht mehr nur von freiem Zeitfenster zu freiem Zeitfenster und damit von Knoten zu Knoten geplant, sondern durch mehrere freie Zeitfenster hindurch und somit über mehrere Knoten hinweg. Denn die Erreichbarkeit freier Zeitfenster hängt wesentlich auch von der aktuellen Geschwindigkeit ab, mit welcher sich das Fahrzeug bewegt. Die Route wird somit in einzelne Segmente heruntergebrochen, welche durch geradlinige Fahrbewegungen der Fahrzeuge ausgeführt werden. Diese umfassen eine Beschleunigungsphase, gegebenenfalls eine Fahrt mit konstanter Geschwindigkeit sowie eine Verzögerungsphase. Die Ausführung dieser Segmente ist so zu gestalten, dass es weder zu Kollisionen noch zu Deadlocks kommt, auch wenn sich einzelne Fahrzeuge verspäten und nicht die von ihnen reservierten Zeitfenster treffen.

Darüber hinaus wird das zeitfensterbasierte Routing dahingehend modifiziert, dass das Verfahren der Fahrzeuge auf einem rechtwinkligen Wegenetz und die damit notwendigen Richtungsänderungen um 90 Grad abgebildet werden können.

4.2 Aufbau der Methodik

Abbildung 4-1 visualisiert den Aufbau der Methodik zur simulationsbasierten Durchsatzanalyse von FTF-basierten Kommissioniersystemen.

Unterschiedliche Systemtypen von FTF-basierten Kommissioniersystemen, welche gemäß Abschnitt 2.3.2 in den Betrachtungsgegenstand dieser Arbeit fallen, werden derart modelliert, dass sich diese einheitlich in einer Simulationsumgebung abbilden lassen und der Ansatz des zeitfensterbasierten Routings verwendet werden kann. Diese Modellierung wird im nachfolgenden Kapitel 5 beschrieben. Gegenstand der Modellierung ist zum einen das zugrundeliegende Layout (Unterkapitel 5.1), zum anderen die Bewegungen der Fahrzeuge unter der Berücksichtigung von Beschleunigungs- und Verzögerungsvorgängen (Unterkapitel 5.2.).

Über eine entsprechende Parametrisierung des Layouts, der Fahrzeuge sowie peripherer Systeme wird ein Simulationsmodell erstellt. In diesem Simulationsmodell sind

Steuerungsstrategien hinterlegt, welche für den effizienten und robusten Betrieb notwendig sind. Diese Strategien, welche allgemeingültig und für jeden Systemtyp anwendbar sind, werden im Kapitel 6 beschrieben.

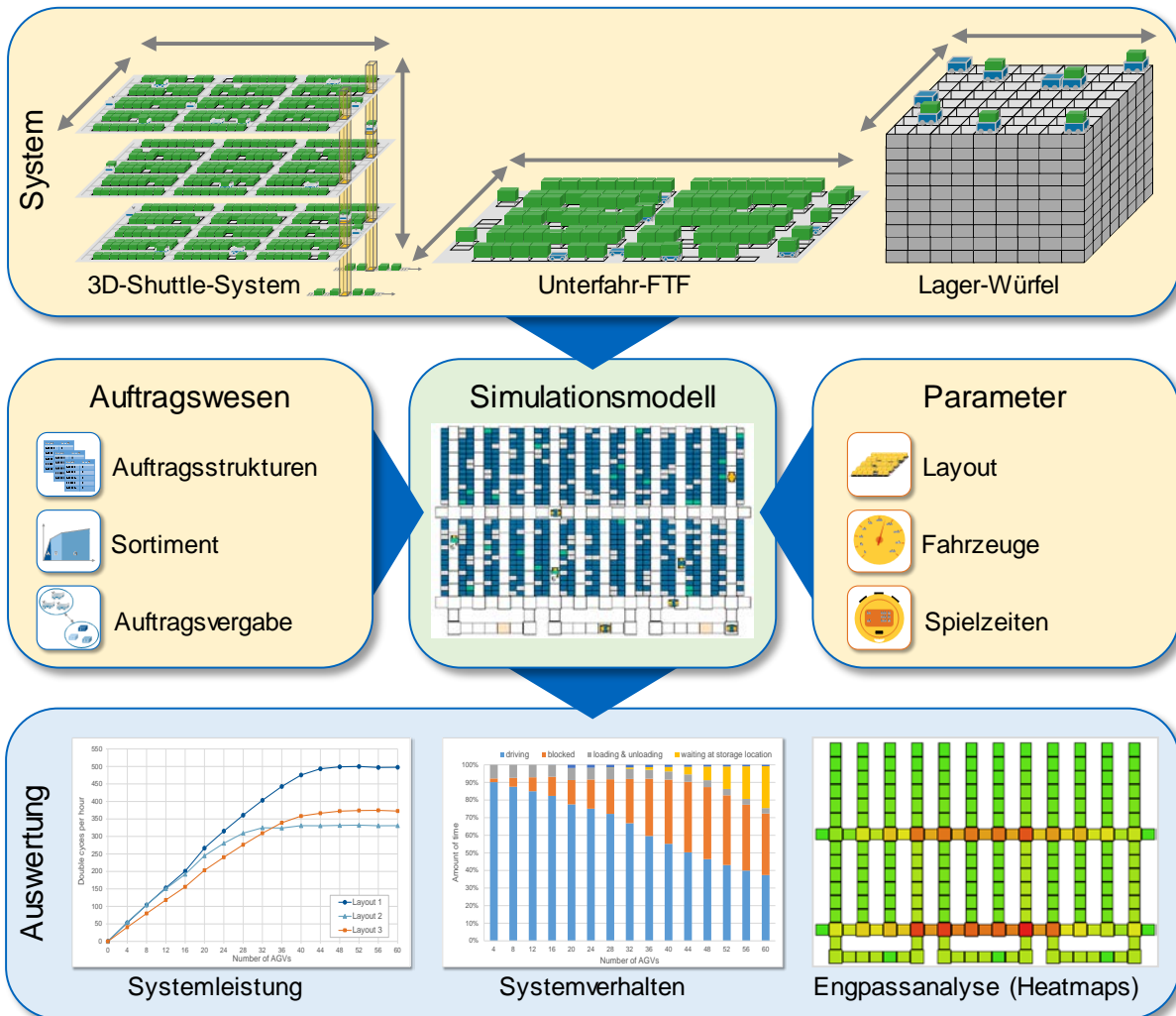


Abbildung 4-1: Aufbau der Methodik.

Bei den Strategien handelt es sich um:

- die Erweiterungen des zeitfensterbasierten Routings (Unterkapitel 6.1),
- die routingbasierte Sequenzierung (Unterkapitel 6.2),
- die Ausführung der berechneten Routen (Unterkapitel 6.3) sowie
- Strategien für den Umgang mit Störungen (Unterkapitel 6.4).

Die Schnittstelle zum Auftragswesen hingegen wird derart gestaltet, dass das Auftragswesen für jeden Anwendungsfall angepasst werden kann. Das Sortiment, die Auftragsstrukturen, Belegungsstrategien sowie auch die Auftragsvergabe werden sich je

nach Anwendungsfall unterscheiden und können nicht allgemeingültig abgebildet werden.

Das parametrisierte Simulationsmodell des betrachteten Systems ist aufgrund der hinterlegten Steuerungsstrategien ablauffähig und kann für die Ermittlung von Kennzahlen herangezogen werden. Neben der Systemleistung, welche durch die charakteristischen Durchsatzverläufe (siehe Abschnitt 3.1.7) beschrieben wird, sollen jedoch weitere Kennzahlen erfasst werden, welche ein tiefgehendes Systemverständnis sowie die Ermittlung von Engpässen und Optimierungspotentialen ermöglichen.

Das Kapitel 7 beinhaltet die Umsetzung der Modellierung und Steuerungsstrategien in Form eines modularen Bausteinkastens, mit Hilfe dessen sich aufwandsarm unterschiedliche Systeme abbilden und simulieren lassen. Dafür wird das Simulationsmodell in unterschiedliche Module untergliedert, welche sich an der in Abbildung 4-1 dargestellten Struktur orientieren.

5 Modellierung

In diesem Kapitel wird die Modellierung des FTS erläutert. Um das Fahrweglayout in eine Simulationsumgebung zu überführen und das zeitfensterbasierte Routing anzuwenden, wird dieses als Graph modelliert. Diese Modellierung wird im Unterkapitel 5.1 beschrieben. Das Unterkapitel 5.2 erläutert die Modellierung der Bewegungen der Fahrzeuge. In diesem Kapitel wird auf Ansätze zurückgegriffen, welche bereits in [Lie-2017a] und [Lie-2018a] beschrieben worden sind und federführend vom Verfasser der vorliegenden Arbeit erarbeitet wurden.

5.1 Modellierung des Layouts

Das Wegenetz des betrachteten Systems wird durch einen gerichteten, zusammenhängenden Graphen $G = (V, E)$ modelliert. Die Menge V enthält Knoten, welche einzelne Layoutabschnitte repräsentieren, auf denen sich die Fahrzeuge bewegen. Die Menge $E \subseteq V \times V$ enthält gerichtete Kanten, die Auskunft über die unmittelbaren Vorgänger und Nachfolger der einzelnen Knoten geben.

Die Knoten bilden ein rechtwinkliges Wegenetz. Die Fahrzeuge bewegen sich in diesem Wegenetz parallel zur X-Achse und zur Y-Achse. Die Ausrichtung A bezeichnet allgemein die Parallelität zu diesen beiden Achsen.

A : Ausrichtung mit $A \in \{X, Y\}$.

Bezogen auf die Fahrzeuge gibt die Ausrichtung A an, in welcher Richtung ein Fahrzeug derzeit verfahren kann (parallel zur X-Achse oder parallel zur Y-Achse). Bezogen auf die Knoten gibt die Ausrichtung A an, in welcher Richtung eine Fahrt über den Knoten möglich ist.

Über das gesamte Layout wird ein Raster gelegt und den einzelnen Knoten die jeweiligen Koordinaten im Raster zugewiesen, durch welche die Knoten eindeutig bestimmt sind. Anhand paarweiser Vergleiche der Koordinaten kann die Lage der Knoten zueinander bestimmt werden.

Einem Knoten $i \in V$ sind die folgenden Attribute zugeordnet:

- $(X, Y)_i$: Koordinaten des Knoten i in der Rasterung.
- R_i : Quadrupel mit Booleschen Variablen, welches die Informationen enthält, in welcher Fahrtrichtung der Knoten i verlassen werden kann (Norden, Osten, Süden, Westen), um die Fahrtrichtung entlang einzelner Layoutabschnitte zu parametrisieren.
- A_i : Menge der Ausrichtungen, welche angibt, in welcher Richtung der Knoten i befahren werden kann. Knoten können parallel zur X-Achse oder parallel zur Y-Achse befahren werden. Knoten, welche eine Kreuzung repräsentieren und einen Wechsel der Ausrichtung zulassen, können sowohl in Richtung der X-Achse als auch in Richtung der Y-Achse befahren werden.
 $A_i \in \{\{X\}, \{Y\}, \{X, Y\}\}$
- L_i^A : Ausdehnung des Knotens i in Ausrichtung $A \in A_i$.

Im Folgenden wird ein Knoten i mit den Koordinaten $(X, Y)_i$ als Knoten $K_{X,Y}$ bezeichnet.

In Abbildung 5-1 ist ein beispielhaftes Layout dargestellt. Die Knoten sind mit den entsprechenden Attributen parametrisiert.

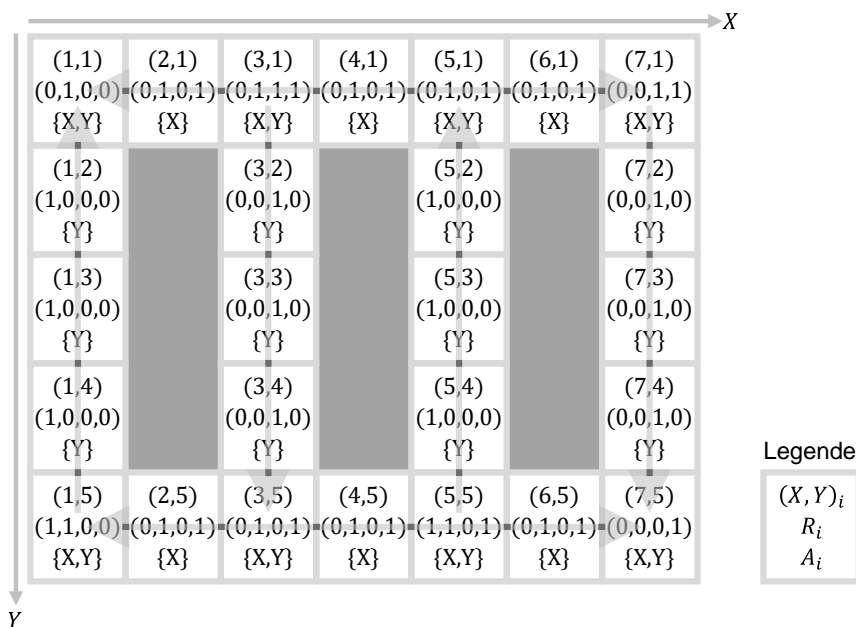


Abbildung 5-1: Modellierung eines Layouts mit unidirektionalen Lagergassen und bidirektionalen Quergassen durch entsprechende Belegung der Knotenattribute [Lie-2017a].

Für die Unterteilung des Wegenetzes in die einzelnen Knoten werden folgende Einschränkungen berücksichtigt:

1. Jeder Knoten kann ein Fahrzeug vollständig aufnehmen.
2. Kreuzungsknoten erlauben den Wechsel der Ausrichtung, ohne dass benachbarte Knoten durch das Fahrzeug belegt werden.
3. Auf jedem Knoten kann sich zu jedem Zeitpunkt nur ein einziges Fahrzeug befinden.

Die erste Bedingung verhindert, dass ein Fahrzeug in der Bewegung mehr als zwei Knoten gleichzeitig belegt. Damit kann, wie auch mit der zweiten Bedingung, die Komplexität des zeitenfensterbasierten Routings beschränkt werden. Die dritte Bedingung stellt sicher, dass keine Koordination mehrerer Fahrzeuge auf einem Knoten notwendig ist. Um eine hohe Ressourcenauslastung zu erzielen, ist es erstrebenswert, die Knoten so klein wie möglich zu gestalten.

Das Wegenetz wird in der Simulationsumgebung nicht nur graphisch hinterlegt, sondern auch formal über Adjazenzlisten beschrieben. Diese beinhalten sämtliche Nachfolger eines Knotens.

Adj_i : Menge der Nachfolger des Knoten i .

Für die Formalisierung des Layouts werden zwei unterschiedliche Typen von Adjazenzlisten herangezogen. In dem ersten Typ sind die Knoten unabhängig von ihrer Ausrichtung gespeichert. Über den paarweisen Vergleich der Knotenkoordinaten und unter der Berücksichtigung in welcher Fahrtrichtung die Knoten verlassen werden können, werden diese Adjazenzlisten erstellt. Sie enthalten die räumliche Ausdehnung und Ausrichtung eines Knoten i sowie die unmittelbaren Nachfolger samt Entfernung $s_{i,j}$ zwischen den Knotenmittelpunkten des Knoten i und den Nachfolgern $j \in Adj_i$.

$s_{i,j}$: Euklidischer Abstand der Mittelpunkte der Knoten i und j .

Tabelle 5-1 enthält beispielhaft die Adjazenzlisten für die Knoten $K_{1,4}$ und $K_{1,5}$ aus dem Layout in Abbildung 5-1.

Tabelle 5-1: Adjazenzlisten für die Knoten $K_{1,4}$ und $K_{1,5}$ aus dem Layout in Abbildung 5-1 ohne Berücksichtigung der Ausrichtung.

i	A_i	L_i^X	L_i^Y	$j \in Adj_i$	$s_{i,j}$
$K_{1,4}$	$\{Y\}$	-	$L_{K_{1,4}}^Y$	$K_{1,3}$	$1/2(L_{K_{1,4}}^Y + L_{K_{1,3}}^Y)$
$K_{1,5}$	$\{X, Y\}$	$L_{K_{1,5}}^X$	$L_{K_{1,5}}^Y$	$K_{2,5}$	$1/2(L_{K_{1,5}}^X + L_{K_{2,5}}^X)$
				$K_{1,4}$	$1/2(L_{K_{1,5}}^Y + L_{K_{1,4}}^Y)$

Da die minimale Fahrzeit zwischen zwei Knoten von der aktuellen Ausrichtung des Fahrzeuges auf dem Startknoten abhängt, wird der zweite Typ der Adjazenzlisten benötigt, welche die Ausrichtung der Knoten berücksichtigt. Für jeden Knoten $i \in V$ und für jede Ausrichtung $A \in A_i$, in welcher der Knoten i befahren werden kann, wird eine Adjazenzliste mit dem Eintrag i^A erzeugt. Die Nachfolger, welche in der Liste abgelegt werden, können entweder über eine reine Fahrbewegung oder über einen Wechsel der aktuellen Ausrichtung, welcher die Zeitdauer t_{RW} erfordert, erreicht werden. Zusätzlich zu jedem Nachfolger wird die Fahrzeit t abgelegt, welche notwendig ist, diesen Nachfolger in der angegebenen Ausrichtung zu erreichen.

t_{RW} : Ausrichtungswechselzeit, die ein Fahrzeug benötigt, um seine Ausrichtung $X \rightarrow Y$, bzw. $Y \rightarrow X$ zu wechseln.

$t(s_{i,j})$: Fahrzeit zum Zurücklegen der Strecke zwischen den Knoten i und j .

Tabelle 5-2 enthält beispielhaft die Adjazenzlisten für die Knoten $K_{1,4}$ und $K_{1,5}$ aus dem Layout in Abbildung 5-1, in welchen die Ausrichtung der Knoten berücksichtigt wird.

Tabelle 5-2: Adjazenzlisten für die Knoten $K_{1,4}$ und $K_{1,5}$ aus Abbildung 5-1 mit Berücksichtigung der Ausrichtung.

i^A	$j \in \text{Adj}_{i^A}$	t
$K_{1,4}^Y$	$K_{1,3}^Y$	$t(s_{K_{1,4},K_{1,3}})$
$K_{1,5}^X$	$K_{1,5}^Y$	t_{RW}
	$K_{2,5}^X$	$t(s_{K_{1,5},K_{2,5}})$
$K_{1,5}^Y$	$K_{1,5}^X$	t_{RW}
	$K_{1,4}^Y$	$t(s_{K_{1,5},K_{1,4}})$

Mittels dieses zweiten Typs der Adjazenzlisten wird eine minimale Fahrzeitenmatrix aufgestellt, welche die minimalen Fahrzeiten zwischen allen Knoten unter der Berücksichtigung ihrer Ausrichtungen enthält. Diese werden für das zeitenfensterbasierte Routing benötigt. Es ist möglich, für die Modellierung eines Layouts mehrere Graphen heranzuziehen, welche unterschiedliche Wegenetze repräsentieren. Kann beispielsweise ein unbeladenes Fahrzeug andere Routen einschlagen als ein beladenes, werden für das Routing die entsprechenden Adjazenzlisten und die entsprechende minimale Fahrzeitenmatrix herangezogen.

5.2 Modellierung der Bewegungen der Fahrzeuge

Im folgenden Unterkapitel wird die Modellierung der Bewegungen der Fahrzeuge beschrieben. Die Bewegung der Fahrzeuge untergliedert sich in einen Wechsel der Ausrichtung (Abschnitt 5.2.1) sowie in lineare Fahrbewegungen (Abschnitt 5.2.2). Abschnitt 5.2.3 definiert charakteristische Zeitstempel und -dauern, welche für das zeitensterbasierte Routing herangezogen werden.

5.2.1 Wechsel der Ausrichtung

Die Fahrzeuge können in jeder Ausrichtung vorwärts und rückwärts verfahren. Bei einem Wechsel von vorwärts zu rückwärts oder rückwärts zu vorwärts, für den keine zusätzliche Zeit angenommen wird, ändert sich die Ausrichtung nicht.

Soll die Ausrichtung geändert werden, muss ein Wechsel der Ausrichtung vollzogen werden. Im Rahmen der Modellierung werden hinsichtlich der Art und Weise, wie Fahrzeuge einen Wechsel der Ausrichtung durchführen, zwei Kategorien von Fahrzeugen unterschieden. Fahrzeuge der ersten Kategorie führen einen Wechsel der Ausrichtung durch eine 90 Grad Rotation um die eigene Achse aus (Abbildung 5-2, (a)). Fahrzeuge der zweiten Kategorie hingegen verfügen über einen Mechanismus zum seitlichen Verfahren, wie beispielsweise ein zusätzliches Seitenfahrwerk oder schwenkbare Räder (Abbildung 5-2, (b)). Die Zeitdauer für den Wechsel der Ausrichtung wird, wie im vorhergehenden Abschnitt beschrieben, durch t_{RW} wiedergegeben.

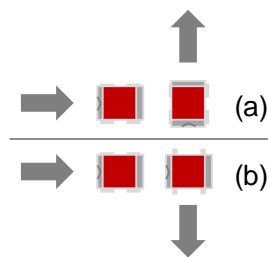


Abbildung 5-2: Wechsel der Ausrichtung von Fahrzeugen beider Kategorien [Lie-2017a].

Bei Fahrzeugen der zweiten Kategorie ändert sich nach einem Ausrichtungswechsel somit gegebenenfalls die Länge L_F^A des Fahrzeuges in Fahrtrichtung.

L_F^A : Länge des Fahrzeuges in Fahrtrichtung, wenn sich das Fahrzeug in Ausrichtung A bewegt.

Für Fahrzeuge der ersten Kategorie (Abbildung 5-2, (a)) gilt hingegen $L_F^X = L_F^Y$.

5.2.2 Fahrzeiten

Die Bewegung der Fahrzeuge ist neben der Ausrichtungswechselzeit durch die Beschleunigung, maximale Geschwindigkeit sowie die Verzögerung charakterisiert. Vereinfachend wird angenommen, dass die Beschleunigung und die Verzögerung gleichförmig verlaufen und darüber hinaus identisch sind. Eine Erweiterung, um unterschiedliche Werte für die Beschleunigung und die Verzögerung zu hinterlegen, ist möglich. Hierfür müssen lediglich nachfolgend hergeleitete Funktionen entsprechend angepasst werden. Ferner wird vorausgesetzt, dass Fahrzeuge entweder beschleunigen, mit maximaler Geschwindigkeit verfahren oder verzögern, aber niemals mit einer konstanten reduzierten Geschwindigkeit verfahren.

v_{max}^A : Maximale Geschwindigkeit eines Fahrzeuges, wenn sich das Fahrzeug in Ausrichtung A bewegt. Eine Differenzierung hinsichtlich einer Vorwärts- oder Rückwärtsbewegung erfolgt nicht.

a^A : Beschleunigung und Verzögerung eines Fahrzeuges, wenn sich das Fahrzeug in Ausrichtung A bewegt.

Im Rahmen der Modellierung können unterschiedliche Werte je nach Ausrichtung des Fahrzeuges berücksichtigt werden. Für die folgenden Ausführungen hingegen wird von einem einzigen Wert unabhängig von der Ausrichtung ausgegangen.

In Abhängigkeit der Länge der zurückzulegenden Fahrstrecke, der Geschwindigkeit und Beschleunigung entstehen zwei unterschiedliche Geschwindigkeits-Zeit-Verläufe. Wird beim Zurücklegen der Fahrstrecke die maximale Geschwindigkeit erreicht, wird die Fahrt als Trapezfahrt bezeichnet. Setzt sich die Fahrt hingegen lediglich aus Anfahren und Abbremsen zusammen, handelt es sich um eine Dreiecksfahrt. Für die Bestimmung der Fahrzeiten ist daher eine Fallunterscheidung zu treffen [Arn-2009, S. 204f.]. Zunächst werden folgende Variablen definiert:

l : Gesamtstrecke.

$t(l)$: Fahrzeit für das Zurücklegen der Gesamtstrecke l .

s_b : Beschleunigungsstrecke, Verzögerungsstrecke.

s_c : Strecke, welche mit konstanter (maximaler) Geschwindigkeit zurückgelegt wird.

t_b : Beschleunigungszeit, Verzögerungszeit.

t_c : Zeitdauer der Fahrt mit konstanter (maximaler) Geschwindigkeit.

Für die Trapezfahrt gelten die folgenden Zusammenhänge:

$$t_b = \frac{v_{max}}{a} \quad (5-1)$$

$$s_b = \frac{1}{2} a t_b^2 = \frac{1}{2} \frac{v_{max}^2}{a} \quad (5-2)$$

$$s_c = l - 2s_b = l - \frac{v_{max}^2}{a} \quad (5-3)$$

$$t_c = \frac{s_c}{v_{max}} = \frac{l}{v_{max}} - \frac{v_{max}}{a} \quad (5-4)$$

Wird die maximale Geschwindigkeit nicht erreicht und entfällt die Fahrt mit konstanter Geschwindigkeit, gilt hingegen:

$$t_b = \sqrt{\frac{l}{a}} \quad (5-5)$$

$$s_b = \frac{1}{2} a t_b^2 = \frac{l}{2} \quad (5-6)$$

Die Fahrzeit für das Zurücklegen der Strecke l ergibt sich somit zu:

$$t(l) = \begin{cases} 2t_b + t_c & \text{für } l \geq 2s_b = \frac{v_{max}^2}{a} \\ 2t_b & \text{für } l < 2s_b = \frac{v_{max}^2}{a} \end{cases} \quad (5-7)$$

$$= \begin{cases} \frac{l}{v_{max}} + \frac{v_{max}}{a} & \text{für } l \geq \frac{v_{max}^2}{a} \\ 2\sqrt{\frac{l}{a}} & \text{für } l < \frac{v_{max}^2}{a} \end{cases}$$

Abbildung 5-3 gibt beispielhaft die Geschwindigkeits-Zeit-Verläufe für eine Trapez- sowie für eine Dreiecksfahrt wieder.

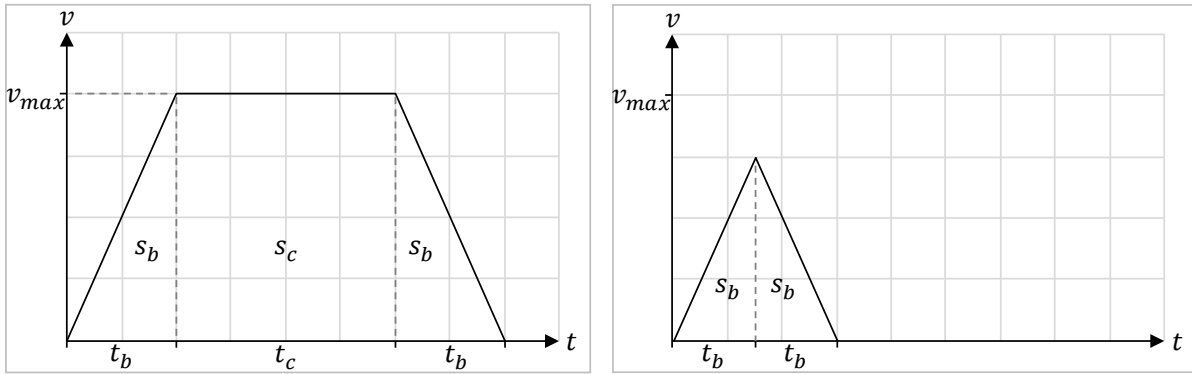


Abbildung 5-3: Geschwindigkeits-Zeit-Verlauf bei einer Trapezfahrt (links) und Dreiecksfahrt (rechts).

Für das zeitfensterbasierte Routing ist es nicht nur notwendig, die Fahrzeiten für das Zurücklegen einer Strecke zu kennen, sondern auch die Zeitpunkte, zu welchen bestimmte Teilstrecken zurückgelegt worden sind, wenn eine Gesamtstrecke gefahren wird. Die Funktion zur Bestimmung dieser Zeitpunkte wird im Folgenden hergeleitet.

- $s(t, l)$: Teilstrecke, welche nach der Zeit t zurückgelegt ist, wenn die Gesamtstrecke l gefahren wird (mit $s \leq l$).
- $t(s, l)$: Zeitdauer, welche notwendig ist, die Teilstrecke s zurückzulegen, wenn die Gesamtstrecke l gefahren wird.
- $v(t, l)$: Geschwindigkeit, mit welcher das Fahrzeug nach der Zeit t fährt, wenn die Gesamtstrecke l gefahren wird.

Zunächst wird die Funktion $s(t, l)$ für das Zurücklegen einer gegebenen Strecke l hergeleitet. Diese ist abschnittsweise definiert und setzt sich analog zum Geschwindigkeits-Zeit-Verlauf aus einer Beschleunigungsphase, einer Fahrt mit konstanter Geschwindigkeit sowie einer Verzögerungsphase zusammen. Für die zeitabhängige Geschwindigkeit $v(t, l)$ gilt zunächst:

$$v(t, l) = \begin{cases} at & \text{für } t < t_b = \frac{v_{max}}{a} \\ v_{max} & \text{für } \frac{v_{max}}{a} \leq t \leq t_b + t_c = \frac{l}{v_{max}} \\ v_{max} - a(t - t_b - t_c) & \text{für } \frac{l}{v_{max}} < t \end{cases} \quad (5-8)$$

Durch Integration erhält man:

$$s(t, l) = \begin{cases} \frac{1}{2} at^2 & \text{für } t < \frac{v_{\max}}{a} \\ s_b + v_{\max}(t - t_b) & \text{für } \frac{v_{\max}}{a} \leq t \leq \frac{l}{v_{\max}} \\ s_b + s_c + v_{\max}(t - t_b - t_c) - \frac{a}{2}(t - t_b - t_c)^2 & \text{für } \frac{l}{v_{\max}} < t \end{cases} \quad (5-9)$$

$$= \begin{cases} \frac{1}{2} at^2 & \text{für } t < \frac{v_{\max}}{a} \\ -\frac{1}{2} \frac{v_{\max}^2}{a} + v_{\max}t & \text{für } \frac{v_{\max}}{a} \leq t \leq \frac{l}{v_{\max}} \\ -\frac{1}{2} \frac{v_{\max}^2}{a} - \frac{a}{2} \frac{l^2}{v_{\max}^2} + v_{\max}t - \frac{a}{2} t^2 + at \frac{l}{v_{\max}} & \text{für } \frac{l}{v_{\max}} < t \end{cases}$$

Bildet man die entsprechende Umkehrfunktion durch Auflösen nach t , ergibt sich die gesuchte, ebenfalls abschnittsweise definierte Funktion $t(s, l)$:

$$t(s, l) = \begin{cases} \sqrt{2 \frac{s}{a}} & \text{für } s < s_b = \frac{1}{2} \frac{v_{\max}^2}{a} \\ \frac{1}{2} \frac{v_{\max}}{a} + \frac{s}{v_{\max}} & \text{für } \frac{1}{2} \frac{v_{\max}^2}{a} \leq s \leq s_b + s_c = l - \frac{1}{2} \frac{v_{\max}^2}{a} \\ \frac{l}{v_{\max}} + \frac{v_{\max}}{a} - \sqrt{\frac{2}{a}(l - s)} & \text{für } l - \frac{1}{2} \frac{v_{\max}^2}{a} < s \end{cases} \quad (5-10)$$

Abbildung 5-4 zeigt beispielhafte Funktionen $s(t, l)$ und $t(s, l)$ für eine Trapezfahrt.

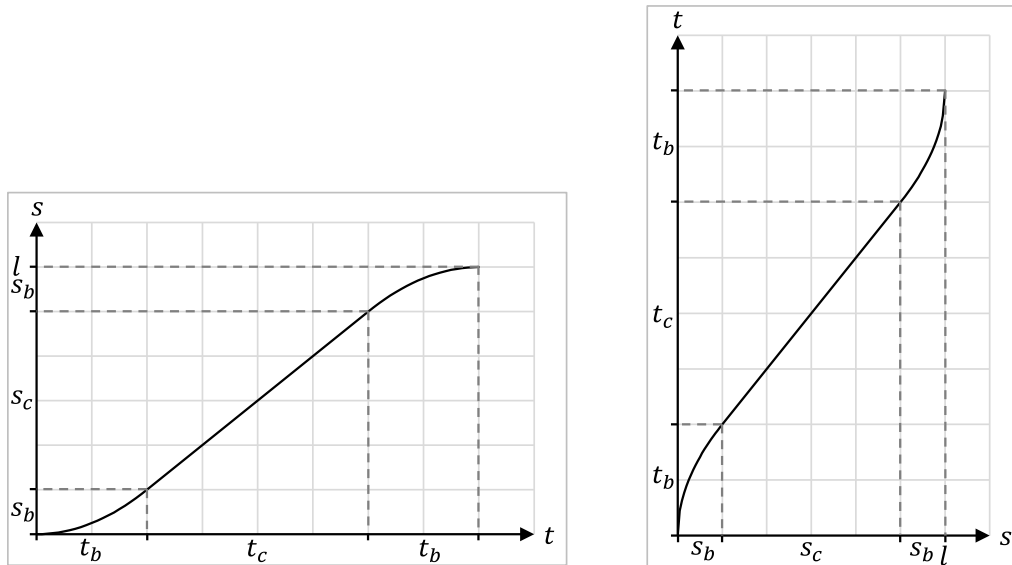


Abbildung 5-4: Funktion $s(t, l)$ und zugehörige Umkehrfunktion $t(s, l)$ für eine Trapezfahrt über die Gesamtstrecke l .

Analog erfolgt das Vorgehen für die Dreiecksfahrt, bei welcher die Fahrt mit konstanter Geschwindigkeit entfällt:

$$v(t, l) = \begin{cases} at & \text{für } t < t_b = \sqrt{\frac{l}{a}} \\ at_b - a(t - t_b) & \text{für } t \geq \sqrt{\frac{l}{a}} \end{cases} \quad (5-11)$$

$$s(t, l) = \begin{cases} \frac{1}{2}at^2 & \text{für } t < \sqrt{\frac{l}{a}} \\ s_b + at_b(t - t_b) - \frac{1}{2}a(t - t_b)^2 & \text{für } t \geq \sqrt{\frac{l}{a}} \end{cases} \quad (5-12)$$

$$= \begin{cases} \frac{1}{2}at^2 & \text{für } t < \sqrt{\frac{l}{a}} \\ 2t\sqrt{al} - \frac{1}{2}at^2 - l & \text{für } t \geq \sqrt{\frac{l}{a}} \end{cases}$$

$$t(s, l) = \begin{cases} \sqrt{2\frac{s}{a}} & \text{für } s < s_b = \frac{l}{2} \\ 2\sqrt{\frac{l}{a}} - \sqrt{2\frac{l-s}{a}} & \text{für } s \geq \frac{l}{2} \end{cases} \quad (5-13)$$

Abbildung 5-5 zeigt beispielhafte Funktionen $s(t, l)$ und $t(s, l)$ für eine Dreiecksfahrt.

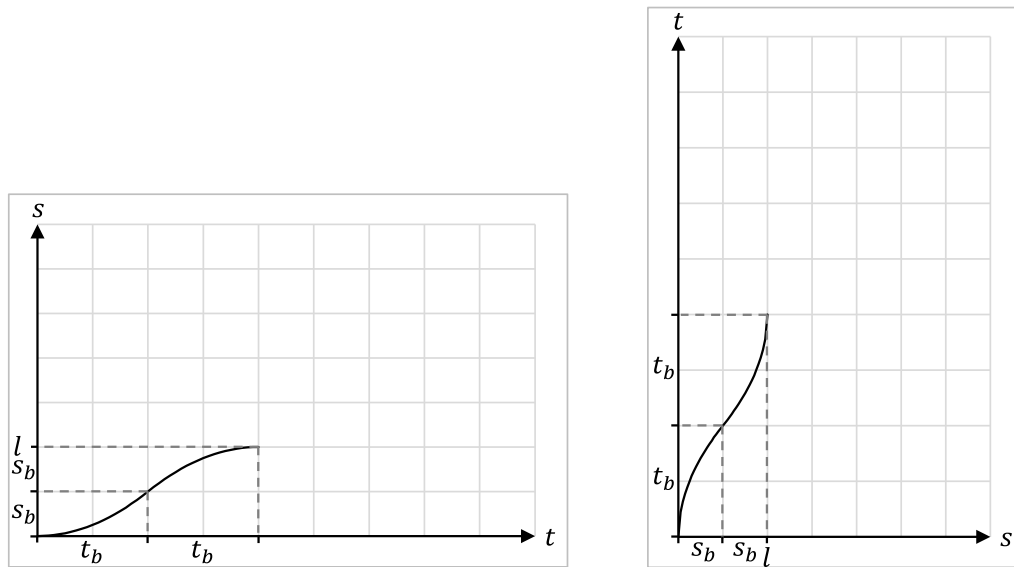


Abbildung 5-5: Funktion $s(t, l)$ und zugehörige Umkehrfunktion $t(s, l)$ für eine Dreiecksfahrt über die Gesamtstrecke l .

Um eine gegebene Strecke der Länge l zurückzulegen, beschleunigt ein Fahrzeug, bis es entweder die maximale Geschwindigkeit erreicht hat oder es aber bereits wieder verzögern muss, falls es sich um eine Dreiecksfahrt handelt. Bei einer Trapezfahrt erfolgt die Verzögerung erst nach einer Fahrt mit konstanter Geschwindigkeit. Um die Fahrt im Modell abzubilden, wird die Zeitdauer t_{VZ} , nach welcher die Verzögerung einzuleiten ist, bei der Abfahrt bestimmt ($v_{akt} = 0$).

t_{VZ} : Zeitdauer, nach welcher die Verzögerung eingeleitet wird.

v_{akt} : Aktuelle Geschwindigkeit des Fahrzeuges (mit $0 \leq v_{akt} \leq v_{max}$).

Diese Zeitdauer ergibt sich, je nachdem ob es sich um eine Trapez- oder Dreiecksfahrt handelt, zu:

$$t_{VZ}(v_{akt}) = \begin{cases} \frac{l}{v_{max}} & \text{für } l \geq \frac{v_{max}^2}{a} \\ \sqrt{\frac{l}{a}} & \text{für } l < \frac{v_{max}^2}{a} \end{cases} \quad (5-14)$$

Muss eine ursprünglich geplante Fahrstrecke während der Ausführung zum Zeitpunkt T auf die Restlänge l' verkürzt werden, muss die Zeitdauer bis zur Verzögerung erneut bestimmt werden.

l' : Verbleibende, verkürzte Fahrstrecke.

Hat das Fahrzeug bereits die maximale Geschwindigkeit erreicht ($v_{akt} = v_{max}$), gilt für die Zeitdauer bis zum Einleiten der Verzögerung:

$$t_{VZ}(v_{akt}) = \frac{l' - s_b}{v_{max}} = \frac{l' - \frac{1}{2} \frac{v_{max}^2}{a}}{v_{max}} = \frac{l'}{v_{max}} - \frac{1}{2} \frac{v_{max}}{a} \quad (5-15)$$

Zu beachten ist, dass t_{VZ} negative Werte annehmen kann, falls die verbleibende, verkürzte Fahrstrecke l' die Verzögerungsstrecke s_b unterschreitet. Dies ist durch die übergeordneten Steuerungsstrategien zu verhindern.

Befindet sich das Fahrzeug hingegen noch in der Beschleunigungsphase ($0 < v_{akt} < v_{max}$), muss eine Fallunterscheidung getroffen werden. Im ersten Fall erreicht das Fahrzeug die maximale Geschwindigkeit, bevor die Verzögerung eingeleitet werden muss. Im zweiten Fall ist die verbleibende Strecke l' nicht mehr ausreichend, damit das Fahrzeug die maximale Geschwindigkeit erreichen kann. Für die Bestimmung von t_{VZ} werden folgende Variablen definiert:

- t_{zb} : Zeitdauer, welche das Fahrzeug bereits beschleunigt hat.
- s_{zb} : Strecke, welche das Fahrzeug während der Beschleunigungsphase bereits zurückgelegt hat.
- t_{vb} : Verbleibende Zeitdauer während der Beschleunigungsphase, bis das Fahrzeug seine maximale Geschwindigkeit v_{max} erreicht.
- s_{vb} : Verbleibende Strecke, welche während der Beschleunigungsphase zurückgelegt wird, bis das Fahrzeug seine maximale Geschwindigkeit v_{max} erreicht.

Diese werden folgendermaßen bestimmt:

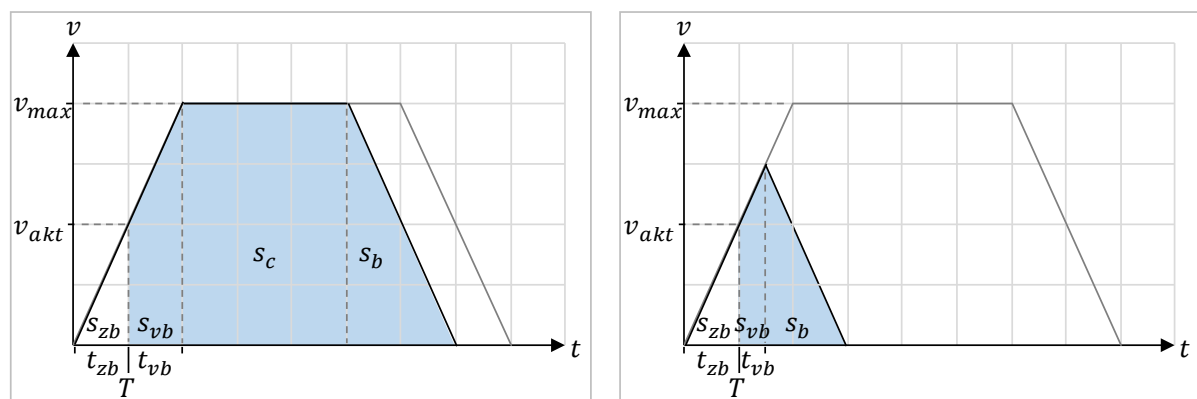
$$t_{zb} = \frac{v_{akt}}{a} \quad (5-16)$$

$$s_{zb} = \frac{1}{2} v_{akt} t_{zb} = \frac{1}{2} \frac{v_{akt}^2}{a} \quad (5-17)$$

$$t_{vb} = \frac{1}{a} (v_{max} - v_{akt}) \quad (5-18)$$

$$s_{vb} = t_{vb} v_{akt} + \frac{1}{2} t_{vb} (v_{max} - v_{akt}) = \frac{1}{2a} (v_{max}^2 - v_{akt}^2) \quad (5-19)$$

Abbildung 5-6 veranschaulicht die Bedeutung dieser Variablen für beide Fälle.



Legende

Verbleibende, verkürzte Fahrstrecke l'

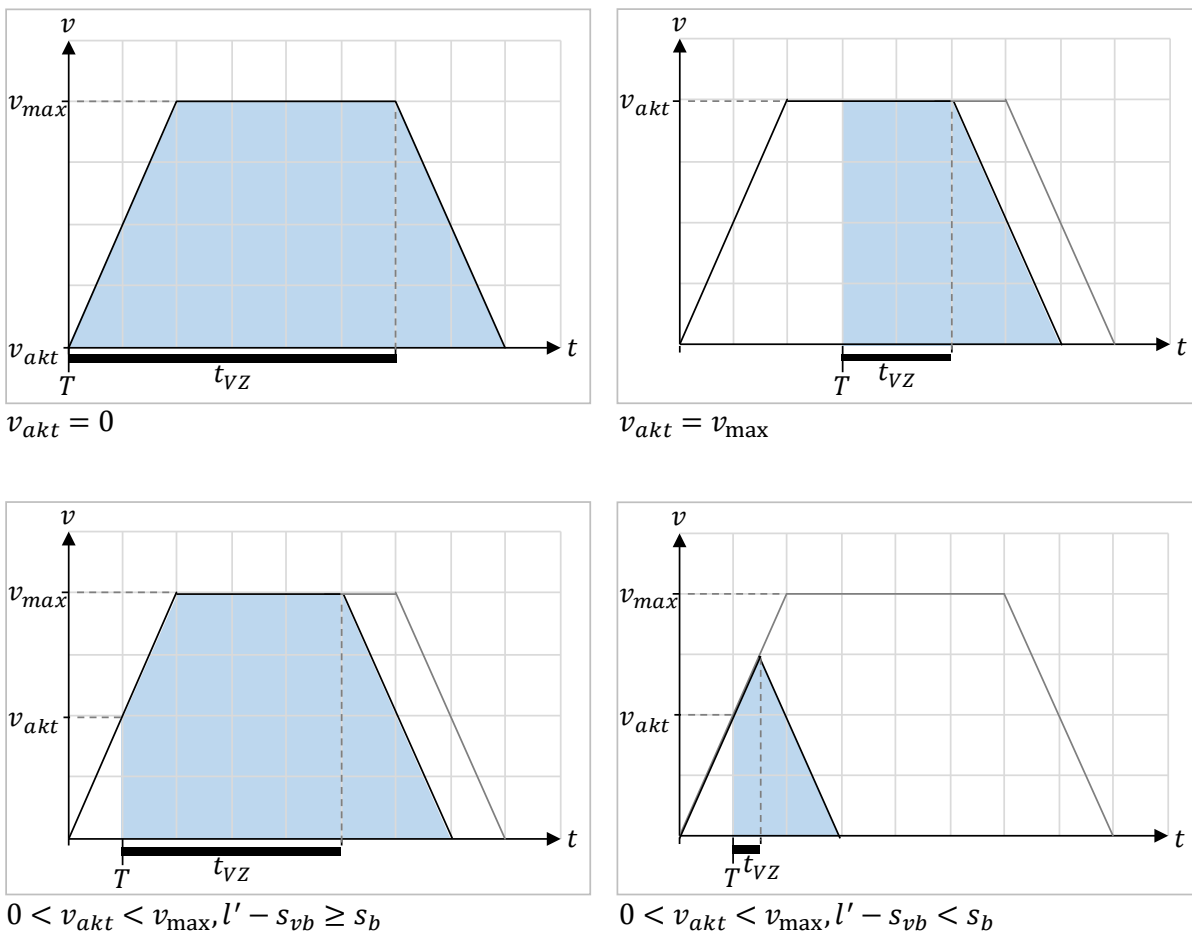
Abbildung 5-6: Fallunterscheidung: Im ersten Fall (links) erreicht das Fahrzeug die maximale Geschwindigkeit, bevor die Verzögerung eingeleitet werden muss, im zweiten Fall (rechts) hingegen nicht.

Mithilfe der definierten Variablen lässt sich die Zeitdauer bis zur Verzögerung bestimmen. Im ersten Fall wird dafür die Zeitdauer der Fahrt mit konstanter Geschwindigkeit ermittelt und der verbleibenden Zeitdauer der Beschleunigungsphase hinzugefügt. Im zweiten Fall lässt sich die Zeitdauer bis zur Verzögerung über (5-14) herleiten.

$$t_{VZ}(v_{akt}) = \begin{cases} t_{vb} + \frac{l' - s_{vb} - s_b}{v_{max}} & \text{für } l' - s_{vb} \geq s_b \\ \sqrt{\frac{l' + s_{zb}}{a}} - t_{zb} & \text{für } l' - s_{vb} < s_b \end{cases} \tag{5-20}$$

$$= \begin{cases} \frac{l'}{v_{max}} + \frac{1}{2a} \left(\frac{v_{akt}^2}{v_{max}} - 2v_{akt} \right) & \text{für } l' - s_{vb} \geq s_b \\ \sqrt{\frac{l'}{a} + \frac{1}{2} \frac{v_{akt}^2}{a^2}} - \frac{v_{akt}}{a} & \text{für } l' - s_{vb} < s_b \end{cases}$$

Abbildung 5-7 zeigt einen Geschwindigkeits-Zeit-Verlauf für eine ursprünglich geplante Fahrt. Darüber hinaus sind drei weitere Fälle dargestellt, bei welchen die Fahrstrecke während der Fahrtdauer zu unterschiedlichen Zeitpunkten T um unterschiedliche Distanzen verkürzt wurde.



Legende

Verbleibende, verkürzte Fahrstrecke l'

Abbildung 5-7: Zeitdauer bis zum Einleiten der Verzögerung für unterschiedliche Fälle.

5.2.3 Zeitstempel und Zeitdauern

Die Bewegung eines Fahrzeuges über die Knoten wird anhand von Zeitstempeln beschrieben, welche definierte Zeitpunkte wiedergeben.

- $T_{ES,i}$: Beginn des Eintritts eines Fahrzeuges auf dem Knoten i .
- $T_{An,i}$: Ankunftszeitpunkt eines Fahrzeuges auf dem Knoten i , mittig positioniert.
- $T_{Ab,i}$: Abfahrtszeitpunkt eines Fahrzeuges von dem Knoten i .
- $T_{AE,i}$: Ende des Austritts eines Fahrzeuges aus dem Knoten i .

Abbildung 5-8 verdeutlicht diese Zeitstempel anhand einer beispielhaften Bewegung eines Fahrzeuges über den Knoten i , einmal mit einem Stopp auf diesem Knoten (Abbildung 5-8, links), einmal ohne anzuhalten (Abbildung 5-8, rechts), einschließlich des jeweiligen Geschwindigkeits-Zeit-Verlaufs.

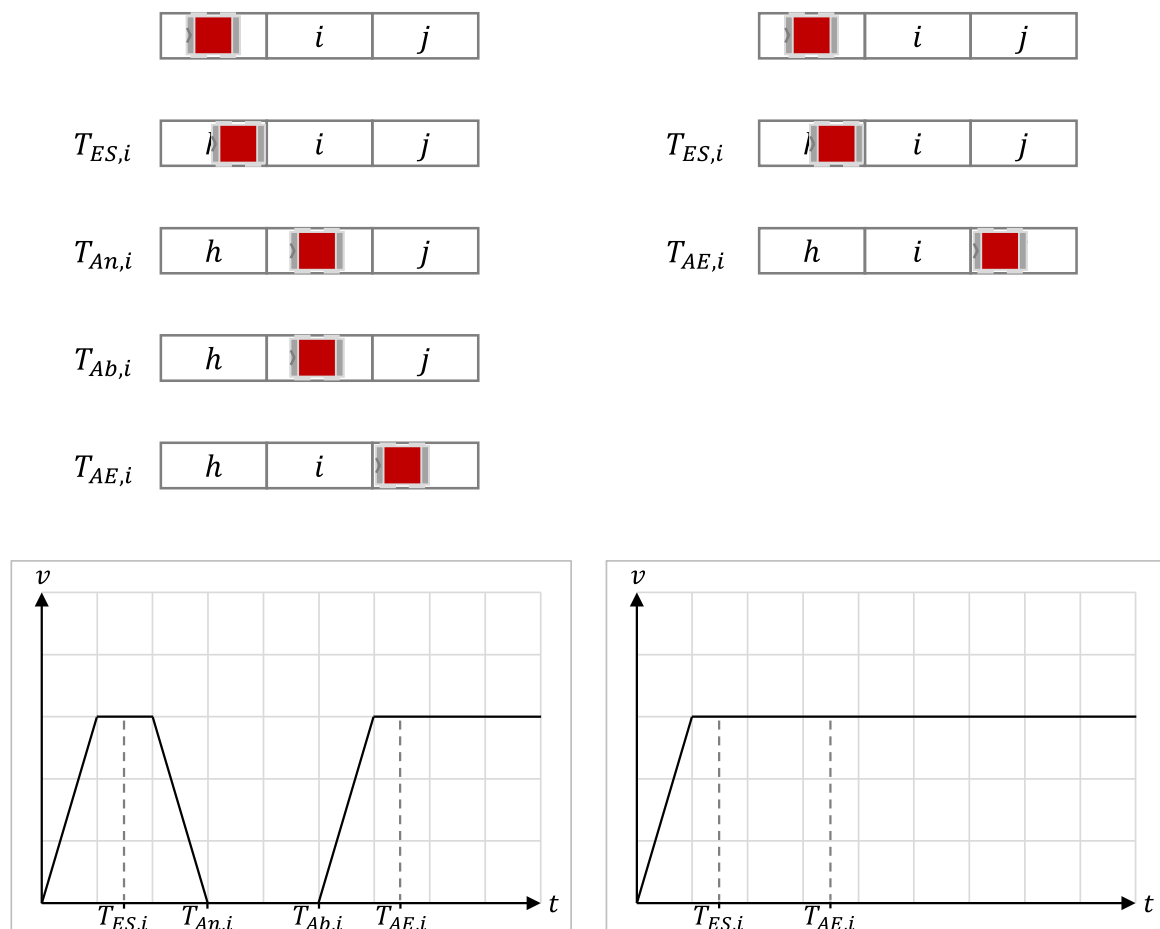


Abbildung 5-8: Zeitstempel und zugehörige qualitative Geschwindigkeits-Zeit-Verläufe für das Anfahren eines Knotens i (links) und das Durchfahren eines Knotens i (rechts).

Abschließend soll noch auf zwei Größen verwiesen werden, auf welche beim zeitfensterbasierten Routing zurückgegriffen wird:

$s_{Ex,i}^A$: Austrittsdistanz, welche von einem Fahrzeug zurückgelegt werden muss, bis es ausgehend von der Ankunftsposition in Ausrichtung A den Knoten i vollständig verlassen hat.

$t_{Ex,i}^A$: Kürzest mögliche Austrittsdauer, welche ein Fahrzeug benötigt, um einen Knoten i , ausgehend von der Ankunftsposition in Ausrichtung A , vollständig zu verlassen.

Die Austrittsdistanz ergibt sich aus der Ausdehnung eines Knotens sowie der Länge der Fahrzeuge bei der Fahrt in entsprechender Ausrichtung:

$$s_{Ex,i}^A = \frac{L_i^A - L_F^A}{2} + L_F^A = \frac{1}{2}(L_i^A + L_F^A) \quad (5-21)$$

Die kürzest mögliche Austrittsdauer für Knoten, welche nur in einer Ausrichtung befahren werden können, wird ermittelt, indem eine hinreichend lange Fahrdistanz angenommen wird, sodass Verzögerungsvorgänge nicht beachtet werden müssen.

$$t_{Ex,i}^A = t(s_{Ex,i}^A, \infty) \quad (5-22)$$

Für Kreuzungsknoten gilt hingegen:

$$t_{Ex,i}^A = \min\{t(s_{Ex,i}^A, \infty), t_{RW} + t(s_{Ex,i}^{A'}, \infty)\} \quad (5-23)$$

$$\text{mit } A' = \begin{cases} X & \text{für } A = Y \\ Y & \text{für } A = X \end{cases}$$

6 Steuerungsstrategien

In diesem Kapitel werden die Steuerungsstrategien beschrieben, welche im Simulationsmodell hinterlegt werden. Dabei handelt es sich um das zeitfensterbasierte Routing (Unterkapitel 6.1), die routingbasierte Sequenzierung (Unterkapitel 6.2), die Ausführung berechneter Routen (Unterkapitel 6.3) sowie Strategien für den Umgang mit Störungen (Unterkapitel 6.4).

6.1 Zeitfensterbasiertes Routing

Im Abschnitt 6.1.1 erfolgt die Definition der Zeitfenster. Im Abschnitt 6.1.2 werden die Eingaben in den Algorithmus beschrieben, während im Abschnitt 6.1.3 der übergeordnete Ablauf dargelegt wird. Der Abschnitt 6.1.4 beschreibt detailliert die Erreichbarkeit freier Zeitfenster während des Routingprozesses, während der Exkurs 6.1.5 auf alternative Reservierungsformen eingeht.

Grundlegende Bestandteile des Routings wurden bereits in [Lie-2017a] und [Lie-2017b] beschrieben, wobei hier die Beschleunigungs- und Verzögerungsvorgänge der Fahrzeuge noch nicht berücksichtigt wurden. In [Lie-2018a] wurde das Routing um die Beschleunigungs- und Verzögerungsvorgänge erweitert, jedoch erfolgte die Reservierung der benötigten Zeitfenster restriktiv (siehe Exkurs 6.1.5, [Lie-2020]).

6.1.1 Definition der Zeitfenster

Für jeden Knoten $i \in V$ unterhält der Algorithmus eine Liste mit reservierten Zeitfenstern (ZF), welche bereits von einzelnen Fahrzeugen reserviert worden sind sowie mit freien Zeitfenstern, durch welche Fahrzeuge geroutet werden können.

Ein freies Zeitfenster $f_{i,l}$ auf einem Knoten i ist durch das Ende des vorhergehenden reservierten Zeitfensters sowie durch den Beginn des nachfolgenden reservierten Zeitfensters bestimmt (Abbildung 6-1).

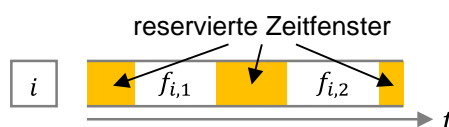


Abbildung 6-1: Reservierte und freie Zeitfenster auf einem Knoten i [Lie-2017a].

$f_{i,l}$:	l -tes freies Zeitfenster auf dem Knoten i .
$T_{Start,f_{i,l}}$:	Beginn des l -ten freien Zeitfensters auf dem Knoten i .
$T_{Ende,f_{i,l}}$:	Ende des l -ten freien Zeitfensters auf dem Knoten i .
F_i :	Menge aller freien Zeitfenster auf dem Knoten i .

Das letzte freie Zeitfenster auf einem Knoten ist nach hinten unbegrenzt, es sei denn das letzte reservierte Zeitfenster ist von unbestimmter Dauer.

Für den Routingprozess werden die in Abschnitt 5.2.3 definierten Zeitstempel nicht nur mit einem Knoten, sondern auch mit dem Zeitfenster assoziiert, in welchem Eintritt, Ankunft, Abfahrt und Austritt erfolgen.

6.1.2 Eingaben in das zeitfensterbasierte Routing

Für die Durchführung des zeitfensterbasierten Routings werden dem Algorithmus verschiedene Parameter übergeben. Diese umfassen den Startknoten, die Startzeit sowie die zu erreichenden Ziele. In vielen Anwendungsfällen stehen bereits vor Beginn einer Fahrt mehrere Ziele fest, welche von einem Fahrzeug in einer vorgegebenen Reihenfolge angefahren werden müssen. Bei der Durchführung eines Doppelspiels wird beispielsweise zunächst der Einlagerplatz angefahren, bevor die Fahrt zum Auslagerplatz und abschließend zum Übergabeplatz erfolgt. Die Eingabe in den Algorithmus umfasst daher eine beliebig lange Liste mit Zielen Z_1, \dots, Z_z , welche

- in einer vorgegebenen Reihenfolge (beispielsweise Ziel Z_{n-1} vor Ziel Z_n vor Ziel Z_{n+1}) angefahren werden;
- derart angefahren werden, dass eine Aktion – beispielweise ein Lagerspiel – mit einer vorgegebenen Verweildauer $t_{v,n}$ an jedem dieser Ziele ausführbar ist.

Liegt ein Ziel Z_n auf einem Kreuzungsknoten, welcher in beiden Ausrichtungen befahrbar ist, kann durch den Parameter A_n vorgegeben werden, in welcher Ausrichtung das Ziel erreicht werden muss. Das letzte anzufahrende Ziel Z_z , bei welchem die Fahrt endet, wird als finales Ziel bezeichnet.

Z_n :	n -tes Ziel in der Liste der übergebenen Ziele.
$t_{v,n}$:	Vorgegebene Verweildauer am n -ten Ziel, bevor die Fahrt fortgesetzt werden kann.
A_n :	Ausrichtung, in welcher das n -te Ziel erreicht werden muss.

Ein Ziel entspricht nicht zwingend einem einzelnen Knoten $i \in V$. Vielmehr ist ein Ziel durch eine Halteposition definiert, welche ein Fahrzeug einnimmt, um dieses Ziel zu erreichen. Diese Halteposition befindet sich entweder ausschließlich auf einem Knoten, oder aber höchstens auf zwei benachbarten Knoten (Abbildung 6-2), da gemäß der im Unterkapitel 5.1 formulierten Bedingung jeder Knoten ein Fahrzeug vollständig aufnehmen kann.

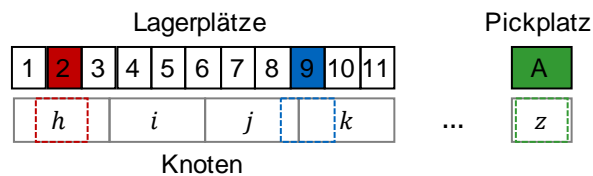


Abbildung 6-2: Haltepositionen für den Zugriff auf unterschiedliche Lagerplätze und einen Pickplatz.

Alle Knoten, welche beim Erreichen des Ziels Z_n durch das Fahrzeug belegt sind, werden in der Zielknotenmenge ZKM_n zusammengefasst.

ZKM_n : Zielknotenmenge des Ziels Z_n , welche alle Knoten beinhaltet, die beim Erreichen des n -ten Ziels durch ein Fahrzeug belegt werden.

Tabelle 6-1 stellt bezogen auf Abbildung 6-2 eine beispielhafte Zielliste für einen Routingprozess eines Doppelspiels dar. Zunächst soll der Einlagerplatz 2 angefahren werden. Für den Zugriff belegt das Fahrzeug lediglich den Knoten h , auf welchem es mindestens die Zeit für ein Lagerspiel verweilt, bevor der Auslagerplatz angefahren werden kann. Die Zielknotenmenge für den Auslagerplatz 9 hingegen enthält zwei Knoten. Die Fahrt endet bei einem Pickplatz. Da nicht bekannt ist, wann das Fahrzeug den zugehörigen Knoten z wieder verlassen wird, muss zunächst von einer unendlich langen Verweildauer ausgegangen werden.

Tabelle 6-1: Beispielhafte Liste mit Zielen für einen Routingprozess.

n	Ziel Z_n	Zielknotenmenge ZKM_n	Verweildauer $t_{V,n}$	Ausrichtung A_n
1	Einlagerplatz 2	$\{h\}$	$t_{\text{Lagerspiel}}$	-
2	Auslagerplatz 9	$\{j, k\}$	$t_{\text{Lagerspiel}}$	-
3	Pickplatz A	$\{z\}$	∞	-

Neben der Liste mit den Zielen werden zum einen der aktuelle Knoten, von welchem die Fahrt aus startet, zuzüglich der Ausrichtung, in welcher die Fahrt begonnen wird, übergeben. Zum anderen wird die Startzeit benötigt. Die Startzeit entspricht der frühestmöglichen Abfahrtszeit des zu routenden Fahrzeuges von dem Startknoten.

T_{start} : Frühestmögliche Abfahrtszeit des zu routenden Fahrzeuges von dem Startknoten.

Es ist möglich, ein Fahrzeug für eine Fahrt zu routen, welche erst zu einem späteren Zeitpunkt ausgeführt werden kann. Die Reihenfolge, in welcher Fahrzeuge routen, hat – wie im Abschnitt 3.1.3 bereits dargelegt – einen Einfluss auf die Fahrdauern der Fahrzeuge. Wird das erste Fahrzeug geroutet, sind keine Reservierungen vorhanden und der schnellstmögliche Weg wird reserviert. Beim Routingvorgang für weitere Fahrzeuge muss nun auf die bestehenden Reservierungen Rücksicht genommen werden. Es ist möglich, zu routende Fahrzeuge unterschiedlich zu priorisieren. Durch den Startzeit-Parameter lassen sich höher priorisierte Fahrzeuge bereits routen, auch wenn diese erst nach Fahrzeugen mit einer niedrigeren Priorität abfahrtsbereit sind.

Für das Routing werden die folgenden Annahmen getroffen:

1. Die Zielknotenmenge des finalen Ziels umfasst lediglich einen Knoten.
2. Das Verfahren eines Fahrzeuges auf einem einzelnen Zielknoten, um eine Halteposition einzunehmen, welche nicht mit der zentrierten Ankunftsposition übereinstimmt, wird vernachlässigt.
3. Jeder Abschnitt einer Fahrt endet immer auf dem hinteren Knoten, sofern die Zielknotenmenge zwei Knoten umfasst.

Die erste Annahme stellt sicher, dass ein untätiges Fahrzeug niemals zwei Knoten für unbestimmte Zeit blockiert. Ferner ist gewährleistet, dass eine neue Fahrt immer von einem Knoten aus startet, sodass im Algorithmus keine zusätzlichen Fallunterscheidungen getroffen werden müssen. Die zweite Annahme stellt eine Vereinfachung dar. Befindet sich die Halteposition nur auf einem einzelnen Knoten und stimmt diese nicht mit der zentrierten Ankunftsposition überein, wird die Differenz der Wegstrecken zwischen den beiden Positionen nicht berücksichtigt (Abbildung 6-3). Die notwendige Feinpositionierung auf einem Knoten wird hinsichtlich des Gesamtdurchsatzes als vernachlässigbar erachtet und in dem Modell zugunsten der reduzierten Komplexität somit unterschlagen.

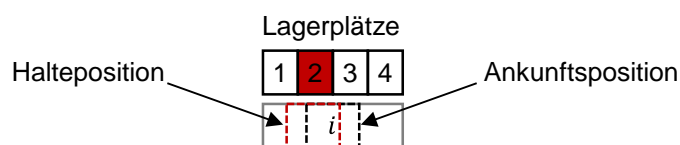


Abbildung 6-3: Um auf den Lagerplatz 2 zuzugreifen, wird nur der Knoten i belegt. Es wird vereinfachend angenommen, dass der Lagerplatz 2 von der zentrierten Ankunftsposition erreicht werden kann. Das Verfahren auf einem Knoten wird nicht betrachtet.

Die dritte Annahme ist ebenfalls eine Vereinfachung. Liegt eine Halteposition auf zwei benachbarten Knoten, wird nach dem Erreichen des Ziels eine Fahrbewegung in der vorhergehenden Fahrtrichtung eingeplant, um die Ankunftsposition auf dem hinteren der beiden Knoten zu erreichen (Abbildung 6-4). Dadurch werden, analog zur ersten Annahme, zusätzliche Fallunterscheidungen im Algorithmus vermieden. Findet nach dem Erreichen eines Ziels ein Ausrichtungswechsel statt, führt das Fahrzeug im Modell somit zunächst eine kurze Fahrbewegung in ursprünglicher Fahrtrichtung aus, bevor die Fahrt in entgegengesetzter Fahrtrichtung fortgesetzt wird.

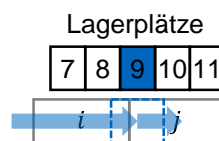


Abbildung 6-4: Nach dem Erreichen der Halteposition für den Zugriff auf den Lagerplatz 9 folgt eine Fahrbewegung, bis die Ankunftsposition auf dem Knoten j erreicht ist.

6.1.3 Ablauf des zeitfensterbasierten Routings

Der Algorithmus unterscheidet vier unterschiedliche Mengen an Zeitfenstern. Die Menge der

- *freien Zeitfenster*, welche alle freien Zeitfenster auf allen Knoten enthält,
- *erreichbaren Zeitfenster*, welche alle freien Zeitfenster enthält, die sich im Verlauf des Routings für das zu routende Fahrzeug als erreichbar herausgestellt haben und für die nächste Iteration ausgewählt werden können,
- *untersuchten Zeitfenster*, welche alle erreichbaren Zeitfenster enthält, für die die Untersuchung der Erreichbarkeit weiterer freier Zeitfenster bereits abgeschlossen ist,
- *reservierten Zeitfenster*, welche alle reservierten Zeitfenster bereits gerouteter Fahrzeuge enthält.

Abbildung 6-5 verdeutlicht den Bezug der Mengen untereinander. Stellt sich ein freies Zeitfenster im Laufe des Routings als erreichbar heraus, wird dieses in die Menge der erreichbaren Zeitfenster aufgenommen. Wird dieses erreichbare Zeitfenster in einer nächsten Iteration untersucht, verlässt es die Menge der erreichbaren Zeitfenster und wird der Menge der untersuchten Zeitfenster hinzugefügt. Ist das Zeitfenster am Ende des Routings Bestandteil der gefundenen Route, wird das Zeitfenster reserviert, wodurch es die Menge der freien Zeitfenster verlässt. Nachdem das Fahrzeug den Knoten passiert hat, wird das reservierte Zeitfenster gelöscht und der Menge der freien Zeitfenster hinzugefügt.

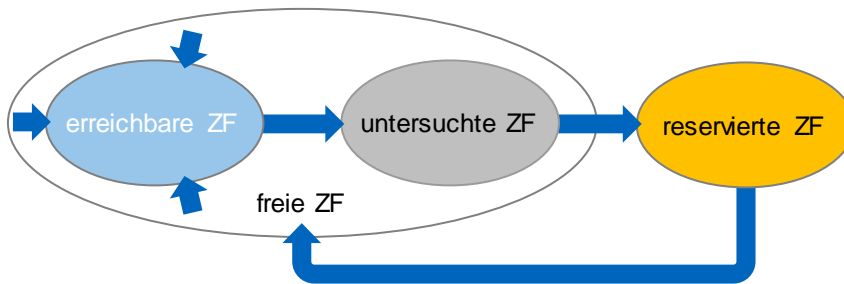


Abbildung 6-5: Unterschiedliche Mengen der Zeitfenster.

Der zeitfensterbasierte Routingalgorithmus untergliedert sich in fünf Phasen, von welchen die Phasen zwei bis vier iterativ ablaufen. Der Algorithmus besteht aus:

1. einer Initialisierung,
2. der Auswahl eines erreichbaren Zeitfensters für die nächste Iteration,
3. der Prüfung, ob das gewählte Zeitfenster bestimmte Bedingungen erfüllt, falls ein Ziel erreicht wird,
4. der Prüfung der Erreichbarkeit freier Zeitfenster aus dem gewählten Zeitfenster heraus,
5. einer Aufbereitung der Route, nachdem das finale Ziel erreicht worden ist.

Stellt sich ein freies Zeitfenster $f_{i,l}$ im Verlauf des Routings als erreichbar heraus, werden folgende Merkmale gespeichert:

- $ID_{f_{i,l}}$: Eindeutige ID, welche zur Unterscheidung notwendig ist, da freie Zeitfenster während eines Routings mehrmals erreicht und in die Menge der erreichbaren Zeitfenster aufgenommen werden können.
- $T_{An,f_{i,l}}$: Ankunftszeitpunkt in dem freien Zeitfenster $f_{i,l}$.
- $A_{f_{i,l}}$: Ausrichtung, in welcher die Ankunft in dem freien Zeitfenster $f_{i,l}$ erfolgt.
- $D_{f_{i,l}}$: Schätzer für die kürzeste Gesamtfahrzeit vom Startknoten zum finalen Ziel Z_z bei einer Fahrt, bei der der Knoten i im Zeitfenster $f_{i,l}$ angefahren wird.
- $n_{f_{i,l}}$: Zähler, welcher angibt, zu welchem Ziel aus der Liste der übergebenen Ziele aus dem freien Zeitfenster $f_{i,l}$ geroutet wird. Somit ist bekannt, welche Ziele $Z_1, \dots, Z_{n_{f_{i,l}}-1}$ vor der Ankunft im freien Zeitfenster $f_{i,l}$ bereits erreicht worden sind. $n_{f_{i,l}} = 1, \dots, z + 1$
- $B_{f_{i,l}}$: Zeiger (Backpointer), welcher auf das freie Zeitfenster zeigt, aus welchem das neu aufgenommene freie Zeitfenster $f_{i,l}$ erreicht wird, um nach Abschluss des Routings die Route konstruieren zu können.

Das Zeitfenster für die nächste Iteration im Algorithmus wird nach dem Schätzer $D_{f_{i,l}}$ ausgewählt. Er ist ein Maß für die Wahrscheinlichkeit, dass das Zeitfenster $f_{i,l}$ auf dem schnellsten Weg zum Ziel durchfahren wird und setzt sich wie folgt zusammen:

$$D_{f_{i,l}} = (T_{An,f_{i,l}} - T_{Start}) + h_{f_{i,l}} \quad (6-1)$$

Der Term $(T_{An,f_{i,l}} - T_{Start})$ entspricht der Zeitdauer zwischen dem Beginn der Fahrt und der Ankunft im Zeitfenster $f_{i,l}$. Der Wert $h_{f_{i,l}}$ hingegen schätzt die verbleibende Dauer bis zur Ankunft am finalen Ziel Z_z , einschließlich der Verweildauern an allen zuvor noch zu besuchenden Zielen $Z_{n_{f_{i,l}}}, \dots, Z_{z-1}$.

$h_{f_{i,l}}$: Schätzer für die verbleibende Gesamtfahrzeit bis zur Ankunft am finalen Ziel Z_z , ausgehend von der Ankunft im freien Zeitfenster $f_{i,l}$.

Sofern eine Lösung existiert, findet der Algorithmus stets die optimale Lösung (den schnellsten Weg), wenn der Schätzer $h_{f_{i,l}}$ eine untere Schranke für die tatsächlich verbleibende Gesamtfahrzeit ist [Nil-1971, S. 59]. Die Dauer bis zum Eintreffen am finalen Ziel darf demnach nicht überschätzt werden, wenn die Optimalität erhalten bleiben soll. In diesem Fall wird die Schätzung als zulässig bezeichnet. Die Verweildauern $t_{v,n_{f_{i,l}}}, \dots, t_{v,z-1}$ sind als Teil der Eingabe bekannt, womit lediglich die Fahrzeiten zwischen den einzelnen Zielen geschätzt werden müssen. Werden diese vernachlässigt, ist die Schätzung zulässig, allerdings erfolgt in diesem Fall die Suche durch die freien Zeitfenster nicht zielgerichtet und die Laufzeit des Algorithmus verlängert sich. Wird für den Schätzer die größte mögliche untere Schranke gewählt, durchläuft der Algorithmus die geringste mögliche Anzahl an Iterationen um den schnellsten Weg zu finden [Nil-1971, S. 66]. Eine solche größte untere Schranke für eine zulässige Schätzfunktion liefern die minimal verbleibenden Fahrzeiten, welche sich ergeben, wenn im System keine reservierten Zeitfenster vorhanden sind. Diese sind einmalig nach der Layouterstellung zu bestimmen und in der unter 5.1 beschriebenen minimalen Fahrzeitenmatrix abzulegen. Je nach Layout und Modellierung sind diese jedoch nur aufwändig zu bestimmen, falls Beschleunigungs- und Verzögerungsvorgänge der Fahrzeuge berücksichtigt werden.

Es ist möglich, eine Schätzfunktion zu wählen, welche nicht zulässig ist. Durch das Überschätzen der verbleibenden Fahrzeiten kann nicht mehr garantiert werden, dass der schnellste Weg gefunden wird, jedoch lässt sich dadurch die Laufzeit des Algorithmus durch eine zielgerichtete Suche verkürzen. Beispielsweise kann die Fahrzeit angenommen werden, welche sich ergibt, wenn auf jedem Knoten angehalten wird. In der Ausführung ist dies natürlich nicht der Fall und ein Fahrzeug erreicht schneller das

Ziel, jedoch erfolgt dadurch die Suche zielgerichtet. In den Fallbeispielen im Kapitel 8 wird dieser Ansatz gewählt.

Abbildung 6-6 beschreibt den übergeordneten Routingprozess.

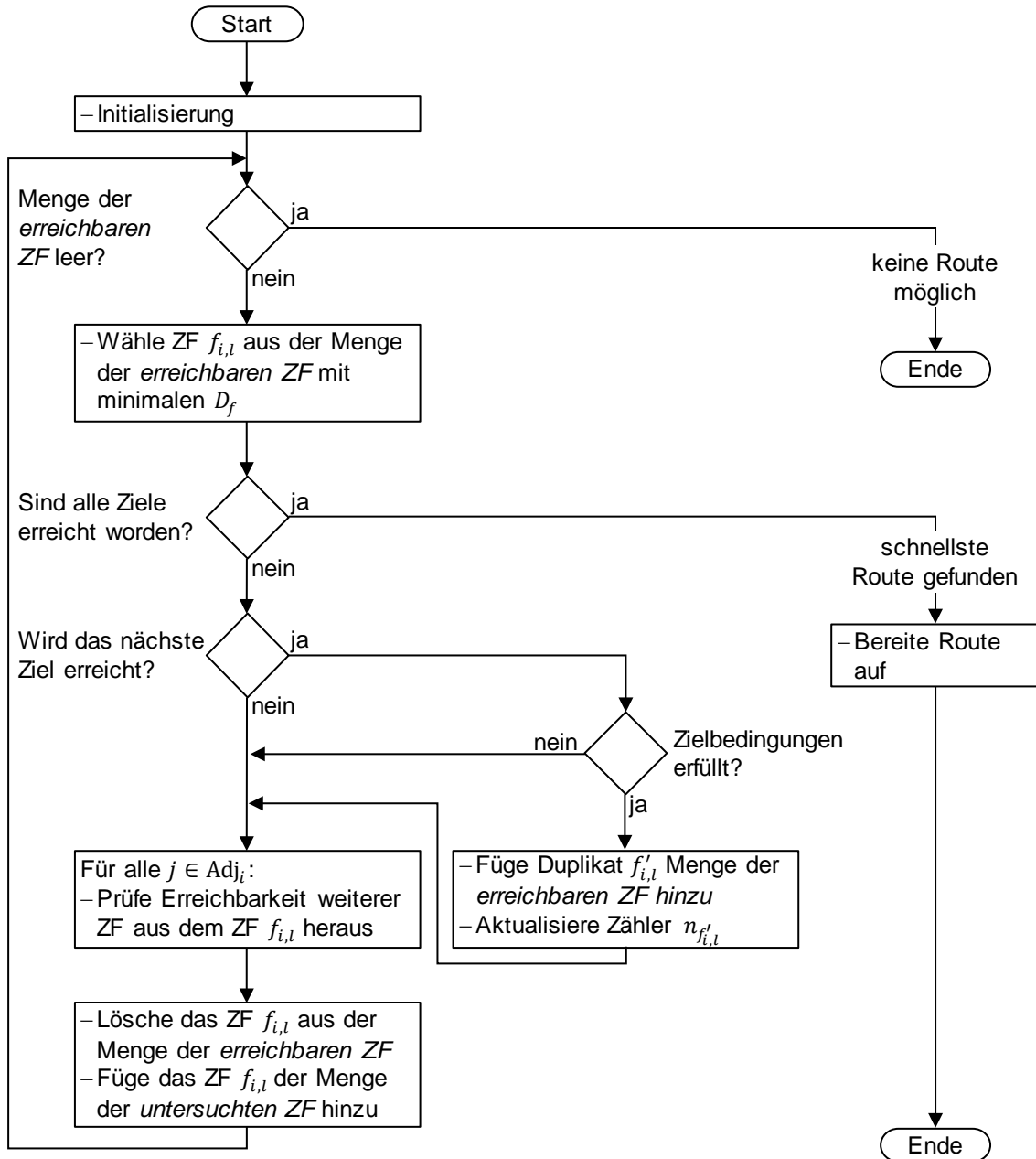


Abbildung 6-6: Zeitfensterbasiertes Routing (übergeordneter Prozess).

Im Rahmen der Initialisierung wird das Zeitfenster, aus welchem die Fahrt startet, der Menge der *freien Zeitfenster* hinzugefügt und die zuvor aufgeführten Merkmale entsprechend gespeichert. Dabei wird die Ankunftszeit mit der übergebenen Startzeit T_{Start} belegt. Somit enthält die Menge der erreichbaren Zeitfenster ein Zeitfenster und die Iterationen können beginnen.

Zu Beginn einer jeden Iteration wird das vielversprechendste erreichbare Zeitfenster ausgewählt. Es ist möglich, dass mehrere erreichbare Zeitfenster den minimalen Wert D aufweisen, wodurch ein weiteres Auswahlkriterium notwendig ist. Möglich sind eine zufällige Auswahl oder die Wahl des Zeitfensters, welches am längsten, bzw. am kürzesten in der Menge der erreichbaren Zeitfenster enthalten ist. Denkbar ist auch, das Zeitfenster derart auszuwählen, dass die zurückgelegte Distanz minimal ist. Damit werden kürzere Routen begünstigt, welche jedoch längere Wartezeiten auf den Knoten entlang der Route aufweisen.

Ist die Menge der erreichbaren Zeitfenster leer, so ist keine Route, welche alle übergebenen Ziele verbindet, möglich. Dieser Fall kann zum einen dann auftreten, wenn Fahrzeuge einen Knoten für unbestimmte Zeit reservieren können. Dieses Vorgehen kann beispielsweise angewendet werden, wenn ein untätiges Fahrzeug gepuffert werden soll und nicht bekannt ist, wann dieses Fahrzeug den Pufferplatz wieder verlässt. Zum anderen kann dieser Fall auftreten, wenn das Routing aus einem Zeitfenster heraus erfolgt, welches beschränkt ist und es somit für das Fahrzeug unmöglich ist, eine hinreichend lange Zeitdauer auf dem Knoten zu warten, bis eine Route zum Ziel möglich ist.

Wurden vor der Ankunft im gewählten Zeitfenster $f_{i,l}$ noch nicht alle übergebenen Ziele Z_1, \dots, Z_z erreicht ($n_{f_{i,l}} \leq z$) und ist das gewählte Zeitfenster keinem Knoten der Zielknotenmenge des aktuellen Ziels $Z_{n_{f_{i,l}}}$ zugeordnet ($i \notin ZKM_{n_{f_{i,l}}}$), erfolgt die Untersuchung der Erreichbarkeit weiterer freier Zeitfenster aus dem gewählten Zeitfenster $f_{i,l}$ heraus. Dieser Prozess wird im folgenden Abschnitt 6.1.4 behandelt. Abschließend wird, wie bereits erläutert, das gewählte Zeitfenster $f_{i,l}$ der Menge der erreichbaren Zeitfenster entnommen und der Menge der untersuchten Zeitfenster hinzugefügt.

Wird für die nächste Iteration ein Zeitfenster $f_{i,l}$ ausgewählt, welches einem Knoten der Zielknotenmenge des aktuellen Ziels zugeordnet ist und besteht diese Zielknotenmenge nur aus einem einzigen Knoten ($i \in ZKM_{n_{f_{i,l}}} \wedge |ZKM_{n_{f_{i,l}}}| = 1$), wird ein nächstes Ziel im Routing erreicht und die Zielbedingungen sind zu prüfen. Vorerst wird als einzige Bedingung lediglich eine ausreichende Restgröße des freien Zeitfensters $f_{i,l}$ berücksichtigt. Dafür muss zunächst der Zeitpunkt der Ankunft auf dem Knoten nach Vollzug der vorgesehenen Verweildauer berechnet werden. Dabei ist zu berücksichtigen, ob ein Wechsel der Ausrichtung notwendig ist, falls das Ziel auf einem Kreuzungsknoten liegt.

$$T_{Anf'_{i,l}} = \begin{cases} T_{An,f_{i,j}} + t_{RW} + t_{V,n_{f_{i,l}}} & , \text{für } A_{f_{i,l}} \neq A_{n_{f_{i,l}}} \\ T_{An,f_{i,j}} + t_{V,n_{f_{i,l}}} & , \text{sonst} \end{cases} \quad (6-2)$$

Anschließend muss überprüft werden, ob der Knoten i vollständig verlassen werden kann, bevor das gewählte Zeitfenster $f_{i,l}$ endet:

$$T_{Anf'_{i,l}} + t_{Ex,i}^{A_{f'_{i,l}}} \leq T_{Ende,f_{i,l}} \quad (6-3)$$

Ist dies der Fall, wird ein Duplikat $f'_{i,l}$ des Zeitfensters $f_{i,l}$ in der Menge der erreichbaren Zeitfenster erstellt, wobei eine eindeutige ID zugewiesen wird und der aktualisierte Ankunftszeitpunkt $T_{An,f'_{i,l}}$ gespeichert wird. Da ein Ziel erreicht wurde, wird von dem neu aufgenommenen Zeitfenster in Richtung des nächsten Ziels $Z_{n_{f_{i,l}+1}}$ in der Liste der übergebenen Ziele geroutet:

$$n_{f'_{i,l}} = n_{f_{i,l}} + 1 \quad (6-4)$$

Ist die Bedingung (6-3) nicht erfüllt, kann in dem gewählten Zeitfenster die vorgesehene Zielaktion nicht ausgeführt werden. Der Routingalgorithmus läuft weiter, bis ein späteres Zeitfenster auf dem Zielknoten i gefunden wird, welches eine ausreichende Länge aufweist.

Wird für die nächste Iteration ein Zeitfenster $f_{i,l}$ ausgewählt, bei dessen Erreichen bereits alle übergebenen Ziele besucht worden sind ($n_{f_{i,l}} = z + 1$), so wurden alle übergebenen Ziele erreicht und die schnellste Route ist gefunden. Diese kann über die Rückverfolgung der Zeiger B erstellt werden. Abschließend sind die benötigten Zeitfenster zu reservieren und es kann mit der Ausführung der Route begonnen werden.

Ein vereinfachtes Beispiel soll den übergeordneten Routingprozess verdeutlichen. Dabei werden Zeitfenster und Ausrichtung vernachlässigt. Von jedem Knoten ist jeder benachbarte Knoten jederzeit erreichbar.

Gegeben ist der ungerichtete Graph in Abbildung 6-7.

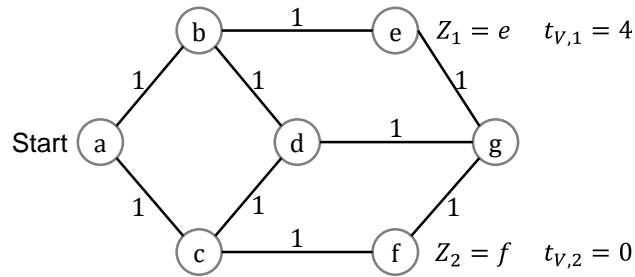


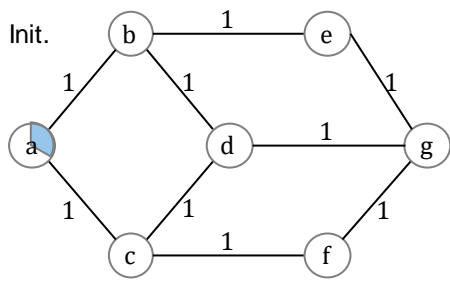
Abbildung 6-7: Ungerichteter Graph.

Ausgehend vom Startknoten a soll zunächst der Knoten e erreicht werden, auf welchem eine Verweildauer von 4 Zeiteinheiten (ZE) vorgesehen ist. Anschließend soll der Knoten f angefahren werden, auf dem das Routing endet. Die Kantengewichtungen entsprechen den Fahrzeiten in Zeiteinheiten. In Abbildung 6-8 sind die Iterationen des Algorithmus dargestellt. Für neu aufgenommene, erreichbare Knoten sind neben dem Zähler n auch die Schätzer D aufgeführt. Zu beachten ist, dass bei diesem vereinfachten Beispiel die kürzesten Fahrzeiten zwischen zwei Knoten bekannt sind.

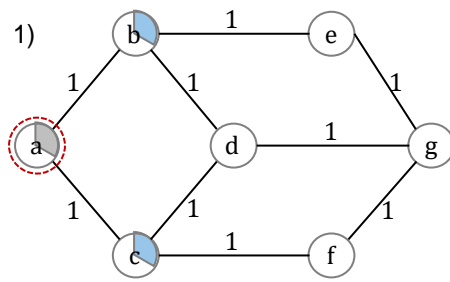
In der Initialisierung wird der Startknoten a der Menge der erreichbaren Knoten hinzugefügt. Ausgehend von diesem wird zunächst zum ersten Ziel geroutet ($n_a = 1$). Der Schätzer D_a für die kürzeste Gesamtfahrzeit ergibt sich aus der Fahrzeit bis zum Startknoten (0 ZE) zuzüglich des Schätzers für die verbleibende Gesamtfahrzeit bis zur Ankunft am finalen Ziel, welche sich aus der kürzest möglichen Fahrzeit zum nächsten Ziel (2 ZE), der dortigen Verweildauer (4 ZE) sowie der kürzest möglichen Fahrzeit vom ersten zum zweiten und letzten Ziel (2 ZE) ergibt. Erreichbar sind die Nachbarknoten b und c , welche in der ersten Iteration in die Menge der erreichbaren Knoten aufgenommen werden.

Für die zweite Iteration wird der Knoten b ausgewählt. Erreichbar sind die Nachbarknoten a , e und d . Die Knoten d und e werden in die Menge der erreichbaren Knoten aufgenommen. Der Knoten a hingegen wird nicht aufgenommen, da er im Verlaufe des Routings bereits mit einem geringeren Schätzer ($D_a = 8$ gegenüber $D_a = 10$) bei gleichem Zähler ($n_a = 1$) in die Menge der erreichbaren Knoten aufgenommen wurde. Eine weiterführende Beschreibung zu diesem Vorgehen erfolgt im nächsten Abschnitt. Für die dritte Iteration wird der Knoten e ausgewählt, welcher selbst das nächste Ziel im Routing darstellt. Somit wird der Menge der erreichbaren Knoten ein Duplikat hinzugefügt, wobei der Zähler n_e aktualisiert wird.

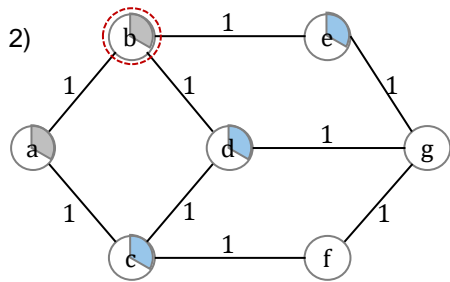
Nach der sechsten Iteration wird der Knoten f ausgewählt. Da nun $n_f = 2 + 1$ gilt, wurden alle Ziele erreicht und der Routingvorgang ist beendet.



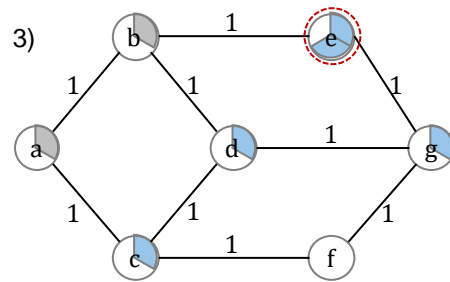
$n_a = 1 \quad D_a = 0 + (2 + 4 + 2) = 8 \leftarrow$



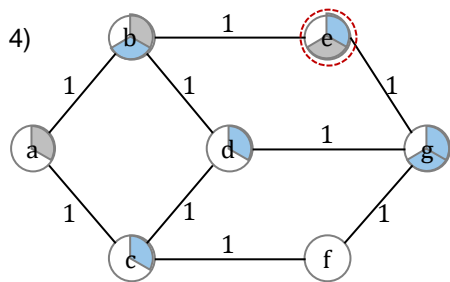
$n_b = 1 \quad D_b = 1 + (1 + 4 + 2) = 8 \leftarrow$
 $n_c = 1 \quad D_c = 1 + (3 + 4 + 2) = 10$



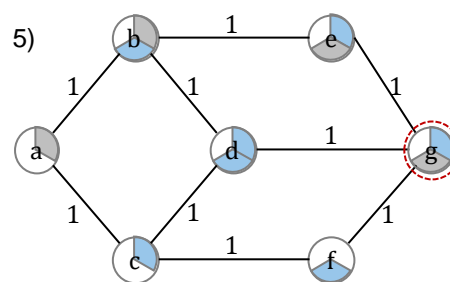
$n_d = 1 \quad D_d = 2 + (2 + 4 + 2) = 10$
 $n_e = 1 \quad D_e = 2 + (0 + 4 + 2) = 8 \leftarrow$



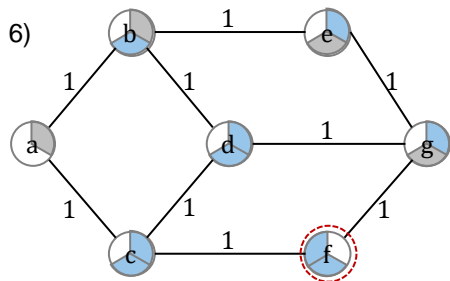
$n_g = 1 \quad D_g = 3 + (1 + 4 + 2) = 10$
 $n_e = 2 \quad D_e = 6 + 2 = 8 \leftarrow$



$n_b = 2 \quad D_b = 7 + 3 = 10$
 $n_g = 2 \quad D_g = 7 + 1 = 8 \leftarrow$



$n_d = 2 \quad D_e = 8 + 2 = 10$
 $n_f = 2 \quad D_f = 8 + 0 = 8 \leftarrow$



$n_c = 2 \quad D_c = 9 + 1 = 10$
 $n_f = 3 \quad D_f = 8 + 0 = 8 \leftarrow$

Legende

- Knoten ist erreichbar
- Knoten ist untersucht
- Abschnitte im Routing
- Auswahl des Knotens für die aktuelle Iteration
- Auswahl des Knotens für die nächste Iteration

Abbildung 6-8: Routingprozess zum Erreichen mehrerer Ziele.

6.1.4 Erreichbarkeit freier Zeitfenster

In jeder Iteration des zeitfensterbasierten Routingalgorithmus (Abbildung 6-6) wird nach freien Zeitfenstern gesucht, welche aus dem gewählten freien Zeitfenster $f_{i,l}$ heraus durch eine lineare Fahrbewegung erreichbar sind. Dafür wird für jeden Nachbarknoten $j \in \text{Adj}_i$ untersucht, wie weit sich eine Fahrbewegung durch die freien Zeitfenster auf dem Knoten j hindurch ausdehnen lässt (Abbildung 6-9).

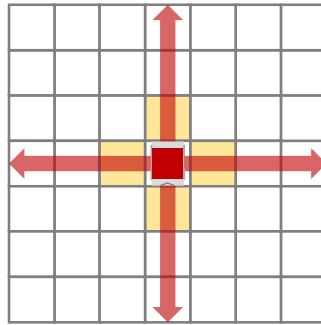


Abbildung 6-9: Ausdehnung linearer Fahrbewegungen über alle Nachbarknoten (gelb).

Eine lineare Fahrbewegung wird durch ein Segment beschrieben. Dieses umfasst alle freien Zeitfenster, welche bei der Fahrt aus dem gewählten Zeitfenster $f_{i,l}$ bis zum Erreichen des freien Zeitfensters $f_{k,p}$ durchfahren werden, einschließlich der Zeitfenster $f_{i,l}$ und $f_{k,p}$. Jedem Segment ist eine Ausrichtung A zugewiesen, entlang welcher die Fahrbewegung erfolgt.

$S_{f_{k,p}}$: Segment, welches alle Zeitfenster umfasst, die bei der Fahrt in das freie Zeitfenster $f_{k,p}$ durchfahren werden.

A_S : Ausrichtung des Segmentes.

Abbildung 6-10 zeigt ein Segment und die entsprechende Trajektorie eines Fahrzeuges beim Zurücklegen der Strecke von dem Knoten i zum Knoten k über den Nachbarknoten j .

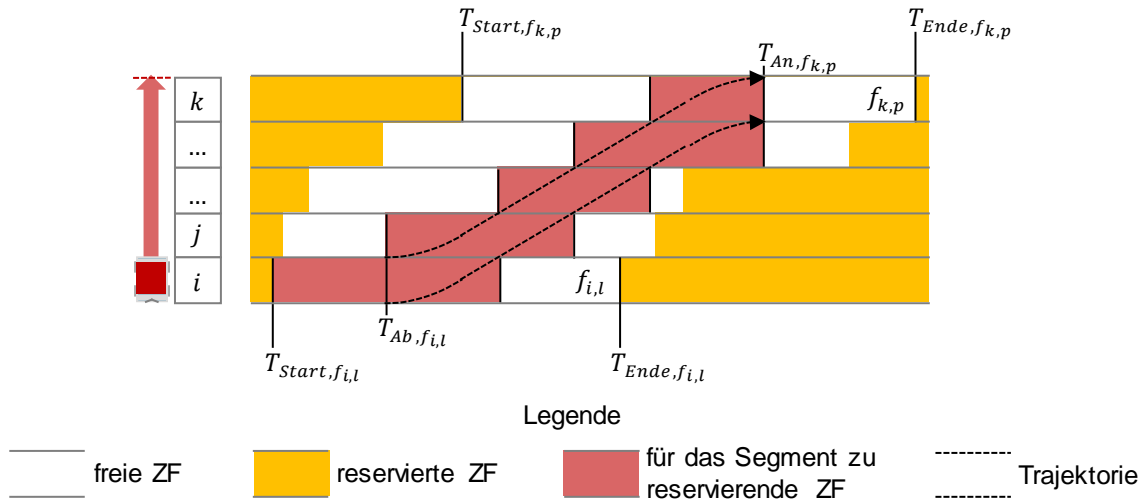


Abbildung 6-10: Segment.

In dem Beispiel aus Abbildung 6-10 beginnen die zu reservierenden Zeitfenster mit dem Eintritt des Fahrzeuges auf die Knoten und enden, sobald das Fahrzeug diese vollständig verlassen hat. Um eine robuste Ausführung zu ermöglichen, auch wenn sich einzelne Fahrzeuge verspäten und somit nicht ihre reservierten Zeitfenster treffen, werden die zu reservierenden Zeitfenster nach vorne verlängert. Es wird ein zeitlicher Puffer verwendet, der es einem Fahrzeug ermöglicht, sicher anzuhalten, falls ein vorhergehendes Fahrzeug verspätet ist. Dafür werden zwei weitere Zeitstempel eingeführt, welche den Beginn und das Ende einer Reservierung eines freien Zeitfensters beschreiben. Der zeitliche Puffer entspricht der Differenz aus dem Beginn des geplanten Eintritts in das freie Zeitfenster sowie dem Beginn der Reservierung.

$T_{RS, f_{k,p}}$: Beginn der Reservierung des freien Zeitfensters $f_{k,p}$.

$T_{RE, f_{k,p}}$: Ende der Reservierung des freien Zeitfensters $f_{k,p}$.

Abbildung 6-11 zeigt das Segment mit zusätzlichen Pufferzeiten. Auf dem zweiten Knoten eines Segmentes – in Abbildung 6-11 ist dies der Knoten j – wird kein zusätzlicher Puffer benötigt, da die Reservierung auf diesem Knoten immer mit der Abfahrt des Fahrzeuges beginnt. Auf die Bestimmung der notwendigen Pufferzeiten wird im Folgenden eingegangen.

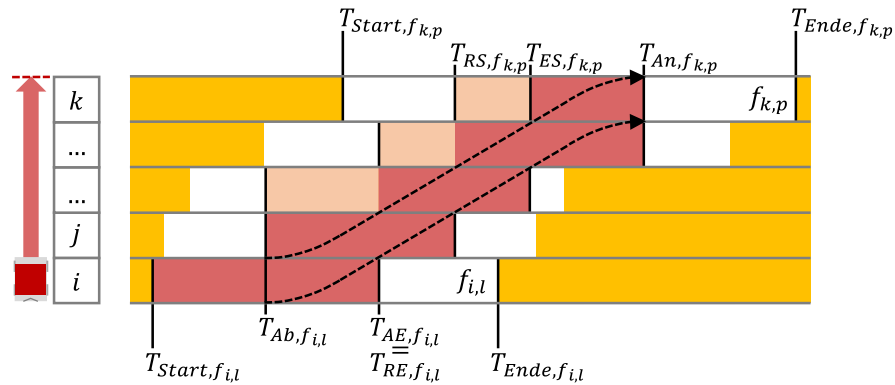


Abbildung 6-11: Segment mit zeitlichen Puffern (hellrot).

Sollte sich das Fahrzeug, welches das vorhergehende Zeitfenster auf dem Knoten k reserviert hat, verspäten, muss sichergestellt werden, dass das zu routende Fahrzeug rechtzeitig auf dem vorhergehenden Knoten anhalten kann, um eine Kollision zu verhindern. Dafür wird der Frage nachgegangen, wann hierfür die Verzögerung eingeleitet werden muss. Es werden die folgenden Annahmen getroffen:

1. Für die Bestimmung der Verzögerungsstrecke wird stets eine Fahrt mit maximaler Geschwindigkeit angenommen.
2. Die Abfrage, ob ein vorhergehendes Fahrzeug verspätet ist, erfolgt beim Passieren der Ankunftsposition eines jeden Knotens.

Ausgehend vom vorletzten Knoten des Segmentes, auf welchem das Fahrzeug im Fall einer Verspätung zum Stillstand kommen muss, werden die Distanzen zwischen den Knoten solange aufsummiert, bis die Verzögerungsstrecke erreicht ist. Sei der Knoten a der Knoten, bei dem diese aufsummierte Distanz die Verzögerungsstrecke erreicht oder überschreitet. Die Reservierung des freien Zeitfensters $f_{k,p}$ muss beginnen, sobald die Halteposition des Knoten a passiert wird. Abbildung 6-12 verdeutlicht das Vorgehen für zwei unterschiedlich lange Verzögerungsstrecken.

Für das zweite und dritte Zeitfenster eines Segmentes gilt: Die Reservierung beginnt stets mit der Abfahrt. Denn wenn auf einem dieser beiden Knoten das vorhergehende Fahrzeug verspätet ist, darf das ausführende Fahrzeug nicht beschleunigen.

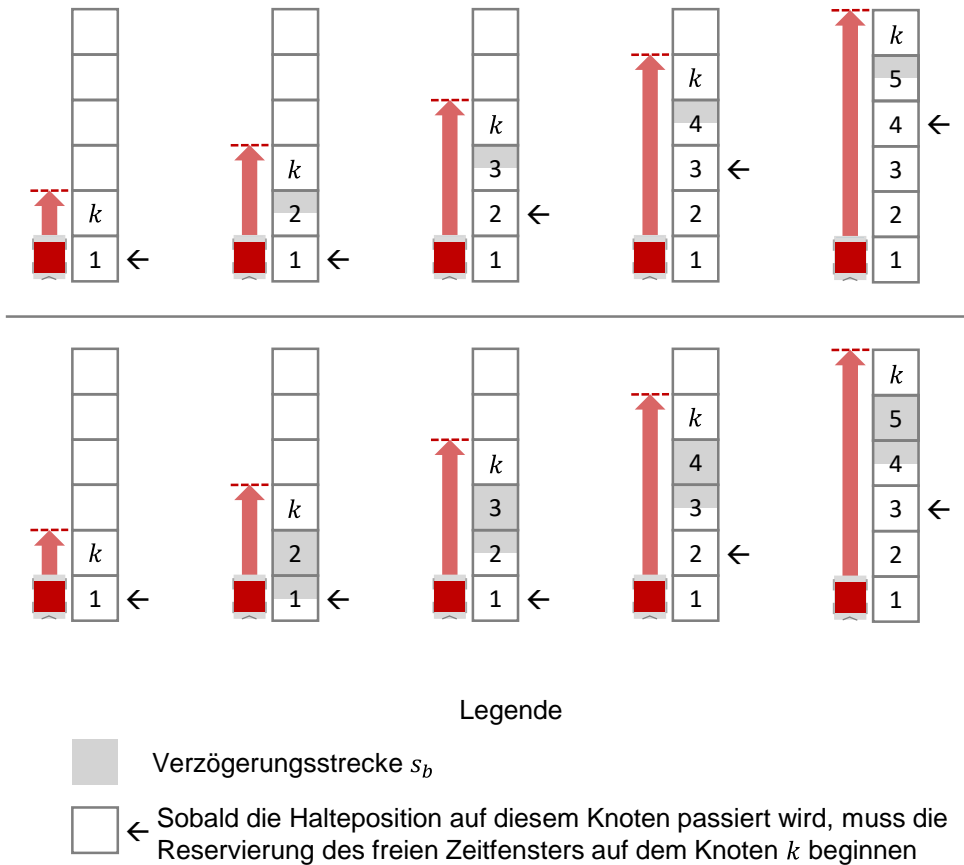


Abbildung 6-12: Bestimmung des Beginns der Reservierung.

Der Zeitpunkt, zu welchem das freie Zeitfenster $f_{k,p}$ spätestens beginnen muss, wird als der notwendige Start bezeichnet.

$T_{notwstart, f_{k,p}}$: Zeitpunkt, zu welchem das freie Zeitfenster $f_{k,p}$ spätestens beginnen muss, damit dieses bei gegebener Abfahrt $T_{Ab, f_{i,l}}$ angefahren werden kann.

Die Erstellung eines Segmentes erfolgt iterativ. In jeder Iteration wird versucht, das Segment um ein freies Zeitfenster zu erweitern und die lineare Fahrbewegung somit um einen Knoten zu verlängern. Dieser Vorgang startet mit einem einzigen Segment – dem initialen Segment – welches nur das gewählte Zeitfenster $f_{i,l}$ enthält.

Es ist möglich, dass ein Knoten durch zwei unterschiedliche Segmente erreicht werden kann, welche sich hinsichtlich der Abfahrtszeitpunkte und der zu durchfahrenden, freien Zeitfenster unterscheiden (Abbildung 6-13). Dabei kann die Ankunft auf dem Knoten k in ein und demselben oder aber auch in zwei unterschiedlichen freien Zeitfenstern erfolgen.

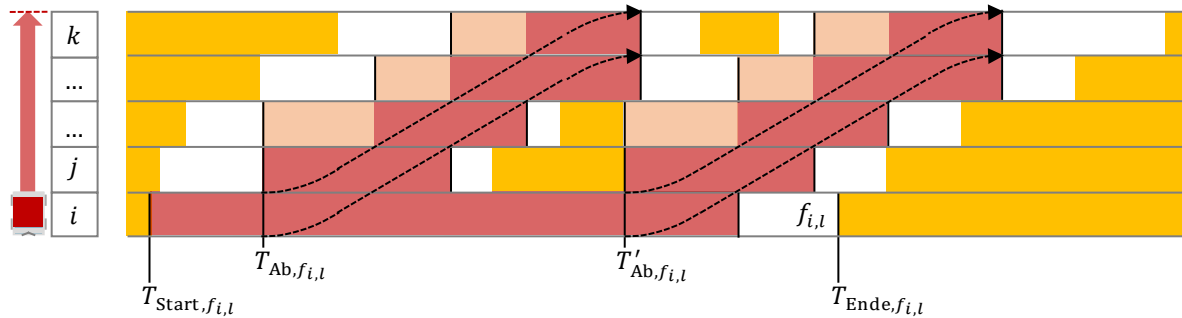


Abbildung 6-13: Der Knoten k ist aus dem gewählten Zeitfenster $f_{i,l}$ durch zwei unterschiedliche Segmente zu erreichen.

Die unterschiedlichen Segmente werden in der Menge der *aktiven Segmente* gespeichert. In jeder Iteration wird versucht, jedes aktive Segment zu erweitern. Die dafür notwendige Vorgehensweise ist in dem Ablaufdiagramm in Abbildung 6-14 dargestellt.

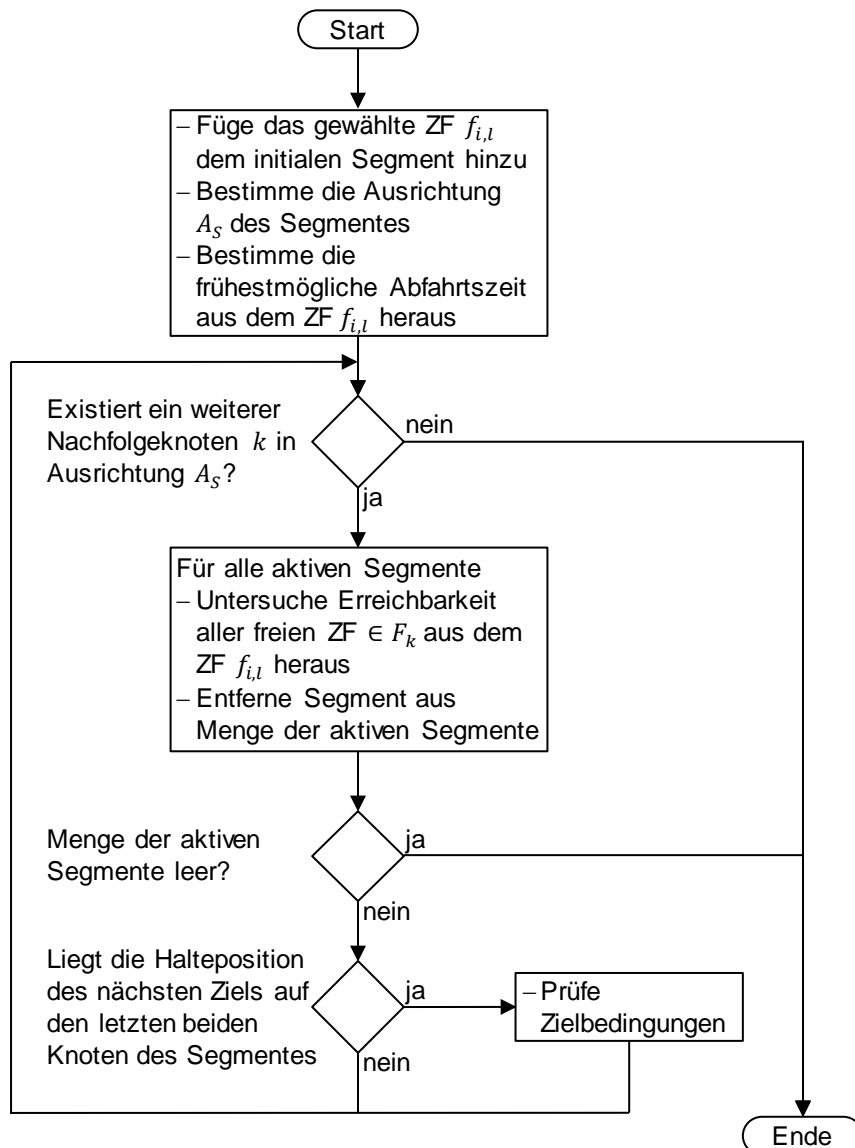


Abbildung 6-14: Erstellung eines Segmentes.

Für eine nächste Iteration des zeitenfensterbasierten Routingalgorithmus wird das erreichbare Zeitfenster $f_{i,l}$ ausgewählt (Abbildung 6-15). Die Ankunftszeit sowie die Ausrichtung, in welcher die Ankunft erfolgt, sind bekannt.

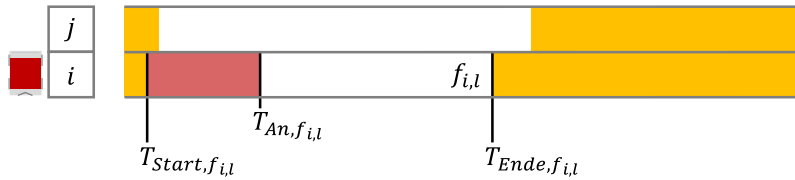


Abbildung 6-15: Ausgangslage für die Erstellung eines Segmentes.

Zunächst wird das gewählte Zeitfenster dem initialen Segment hinzugefügt, welches in der Menge der aktiven Segmente gespeichert wird. Dafür werden die in Tabelle 6-2 aufgeführten Attribute abgelegt.

Tabelle 6-2: *Initiales Segment.*

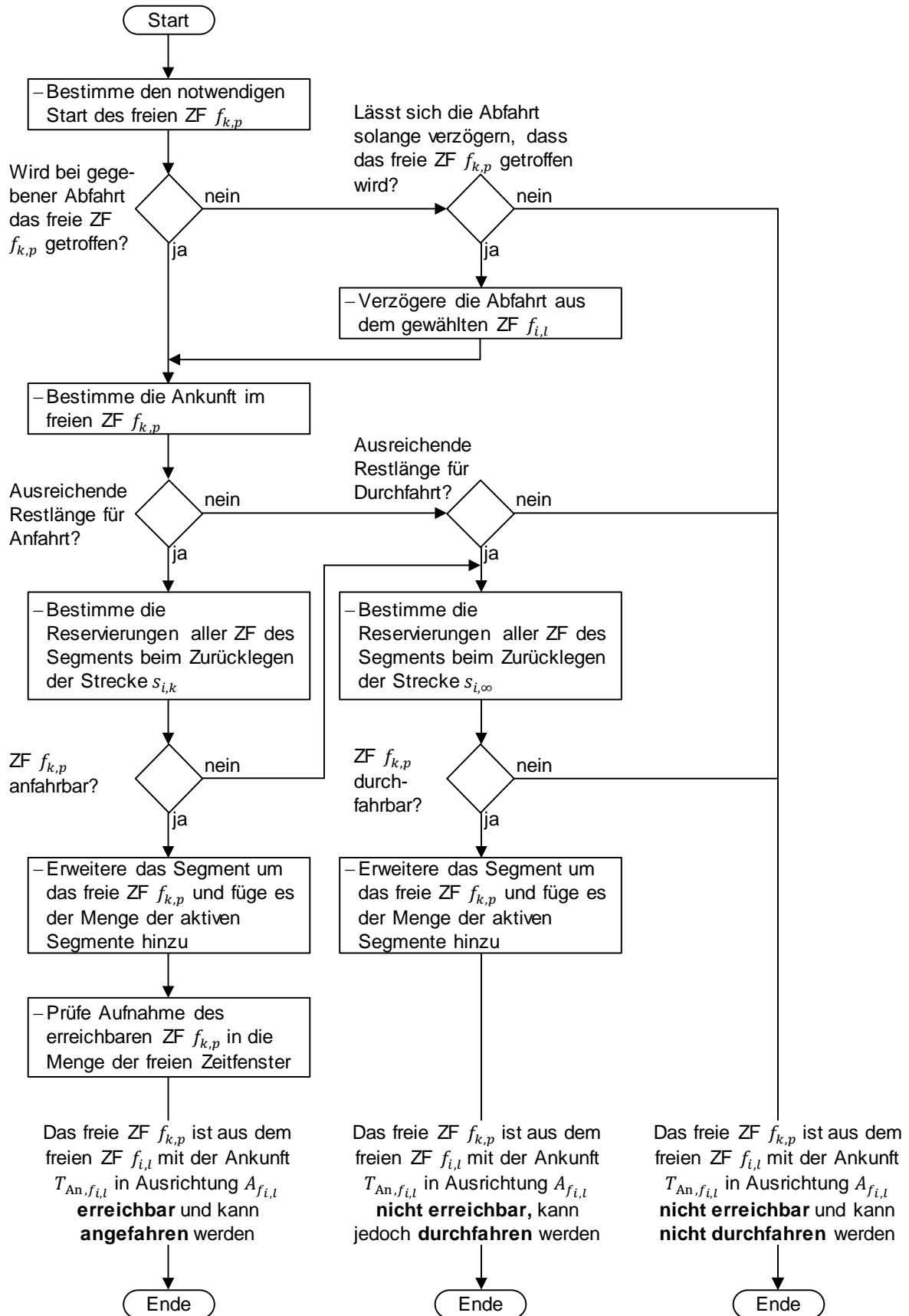
Knoten	T_{Start}	T_{Ende}	T_{RS}	T_{ES}	T_{AE}	T_{RE}	T_{Ab}	T_{An}
i	$T_{Start,f_{i,l}}$	$T_{Ende,f_{i,l}}$	$T_{RS,f_{i,l}}$	$T_{ES,f_{i,l}}$	-	-	-	-

Anschließend ist die Ausrichtung A_S des Segmentes zu bestimmen, welche sich aus der Lage der Knoten i und j zueinander ergibt. Die frühestmögliche Abfahrtszeit aus dem gewählten Zeitfenster $f_{i,l}$ heraus bestimmt sich folgendermaßen:

$$T_{Ab,f_{i,l}} = \begin{cases} \max \{ T_{An,f_{i,l}}, T_{Start,f_{j,m}} \} & , \text{ ohne Wechsel der Ausrichtung} \\ \max \{ T_{An,f_{i,l}} + t_{RW}, T_{Start,f_{j,m}} \} & , \text{ mit Wechsel der Ausrichtung} \end{cases} \quad (6-5)$$

Die Abfahrt aus dem gewählten Zeitfenster $f_{i,l}$ kann nicht eher als die Ankunft in diesem erfolgen und erst sobald das freie Zeitfenster $f_{j,m}$ auf dem Nachbarknoten begonnen hat. Gegebenenfalls muss vor der Abfahrt noch ein Wechsel der Ausrichtung vollzogen werden.

Nach dieser Initialisierung werden die Iterationen des Algorithmus durchlaufen. In jeder Iteration wird versucht, die lineare Fahrbewegung um einen Knoten zu verlängern, sofern ein weiterer Nachfolgeknoten k in Ausrichtung des Segmentes existiert. Dafür wird für jedes freie Zeitfenster auf dem Knoten k geprüft, ob es aus dem gewählten Zeitfenster $f_{i,l}$ heraus erreichbar ist oder zumindest durchfahren werden kann, sollte das Zeitfenster ein Anhalten und erneutes Beschleunigen des Fahrzeuges nicht erlauben. Das Ablaufdiagramm in Abbildung 6-16 gibt dafür das Vorgehen für das freie Zeitfenster $f_{k,p}$ wieder.

Abbildung 6-16: Test auf Erreichbarkeit eines freien Zeitfensters $f_{k,p}$.

Zunächst wird der notwendige Startzeitpunkt $T_{\text{notwStart},f_{k,p}}$ des freien Zeitfensters $f_{k,p}$ bestimmt. Fällt dieser in das freie Zeitfenster ($T_{\text{Start},f_{k,p}} \leq T_{\text{notwStart},f_{k,p}} \leq T_{\text{Ende},f_{k,p}}$), bleibt der Abfahrtszeitpunkt aus dem gewählten Zeitfenster erhalten. Der Beginn der Reservierung entspricht dem notwendigen Startzeitpunkt:

$$T_{RS,f_{k,p}} = T_{\text{notwStart},f_{k,p}} \quad (6-6)$$

Wird das freie Zeitfenster $f_{k,p}$ hingegen nicht getroffen, beginnt dieses aber nach dem notwendigen Beginn ($T_{\text{Start},f_{k,p}} \geq T_{\text{notwStart},f_{k,p}}$), wird die Abfahrt aus dem gewählten Zeitfenster entsprechend verzögert:

$$T_{Ab,f_{i,l}} = T_{Ab,f_{i,l}} + \left(T_{\text{Start},f_{k,p}} - T_{\text{notwStart},f_{k,p}} \right) \quad (6-7)$$

Der Beginn der Reservierung entspricht in diesem Fall dem Beginn des freien Zeitfensters:

$$T_{RS,f_{k,p}} = T_{\text{Start},f_{k,p}} \quad (6-8)$$

Nachdem die Abfahrt aus dem gewählten Zeitfenster bestimmt worden ist, lässt sich die Ankunft im freien Zeitfenster errechnen:

$$T_{An,f_{k,p}} = T_{Ab,f_{i,l}} + t(s_{i,k}) \quad (6-9)$$

Anschließend erfolgt eine Fallunterscheidung: Es ist zu prüfen, ob die Restlänge des freien Zeitfensters $f_{k,p}$ nach der Ankunft ausreichend ist, um den Knoten k wieder vollständig zu verlassen, bevor das freie Zeitfenster endet. Dies ist dann möglich, falls gilt:

$$T_{An,f_{k,p}} + t_{Ex,k}^{As} \leq T_{\text{Ende},f_{k,p}} \quad (6-10)$$

Im ersten Fall ist die Restlänge des freien Zeitfensters $f_{k,p}$ für eine Anfahrt ausreichend. Nun muss sichergestellt werden, dass auf keinem Knoten des Segmentes ein bereits reserviertes Zeitfenster – in den Abbildungen in Orange dargestellt – verletzt wird. Dafür werden die notwendigen Reservierungen der Zeitfenster des Segmentes – in den Abbildungen in Rot dargestellt – bestimmt. Der Beginn der Reservierung eines Zeitfensters im Segment entspricht, mit Ausnahme des ersten Zeitfensters, dem notwendigen Start dieses Zeitfensters. Ein Zeitfenster wird genau solange reserviert, bis das Fahrzeug den Knoten vollständig verlassen hat. Dieser Zeitpunkt ergibt sich aus dem Abfahrtszeitpunkt aus dem gewählten Zeitfenster $T_{Ab,f_{i,l}}$ zuzüglich der Zeitdauer,

welche notwendig ist, die Teilstrecke $s_{i,x} + s_{Ex,x}^{AS}$ zurückzulegen, wenn die Gesamtstrecke $s_{i,k}$ gefahren wird:

$$T_{RE,f_{x,y}} = T_{AE,f_{x,y}} = T_{Ab,f_{i,l}} + t(s_{i,x} + s_{Ex,x}^{AS}, s_{i,k}) \quad (6-11)$$

Wird kein bereits reserviertes Zeitfenster auf einem Knoten des Segmentes verletzt ($T_{AE,f_{x,y}} \leq T_{Ende,f_{x,y}}$), kann das Segment um das freie ZF $f_{k,p}$ erweitert werden (Abbildung 6-17, Tabelle 6-3).

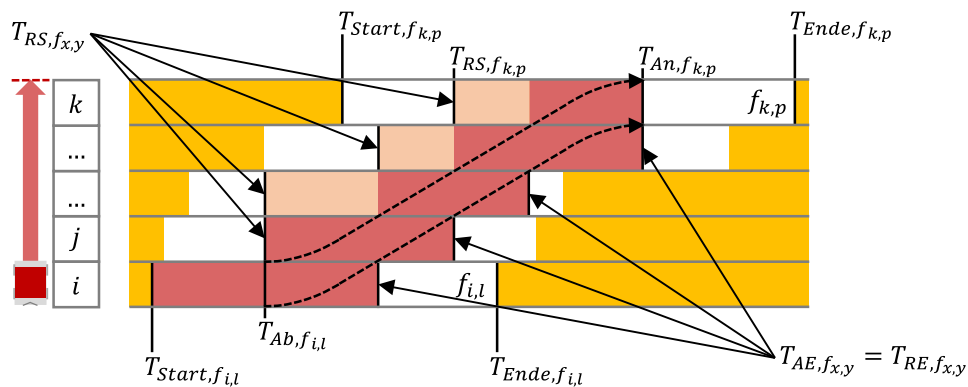


Abbildung 6-17: Das freie Zeitfenster $f_{k,p}$ ist anfahrbar. Das Segment kann verlängert werden.

Tabelle 6-3: Um das anfahrbare Zeitfenster $f_{k,p}$ erweitertes Segment.

Knoten	T_{Start}	T_{Ende}	T_{RS}	T_{ES}	T_{AE}	T_{RE}	T_{Ab}	T_{An}
i	$T_{Start,f_{i,l}}$	$T_{Ende,f_{i,l}}$	$T_{RS,f_{i,l}}$	$T_{ES,f_{i,l}}$	$T_{AE,f_{i,l}}$	$T_{AE,f_{i,l}}$	$T_{Ab,f_{i,l}}$	-
j	$T_{Start,f_{j,m}}$	$T_{Ende,f_{j,m}}$	$T_{RS,f_{j,m}}$	$T_{ES,f_{j,m}}$	$T_{AE,f_{j,m}}$	$T_{AE,f_{j,m}}$	-	-
...	-	-
k	$T_{Start,f_{k,p}}$	$T_{Ende,f_{k,p}}$	$T_{RS,f_{k,p}}$	$T_{ES,f_{k,p}}$	-	-	-	$T_{An,f_{k,p}}$

Abschließend ist die Aufnahme des erreichbaren Zeitfensters $f_{k,p}$ in die Menge der erreichbaren Zeitfenster zu prüfen. Ein erreichbares Zeitfenster wird genau dann in die Menge der erreichbaren Zeitfenster aufgenommen, wenn

- es mit gleichem Zähler $n_{f_{k,p}}$ weder in der Menge der erreichbaren Zeitfenster, noch in der Menge der untersuchten Zeitfenster vorhanden ist oder
- es mit gleichem Zähler $n_{f_{k,p}}$ bereits in der Menge der erreichbaren Zeitfenster vorhanden ist, aber einen geringeren Schätzer $D_{f_{k,p}}$ für die kürzeste Gesamtfahrzeit aufweist, als das bereits vorhandene Zeitfenster.

Wird ein erreichbares Zeitfenster in die Menge der erreichbaren Zeitfenster aufgenommen, werden die unter 6.1.3 aufgeführten Attribute gespeichert. Dabei ist die Information zu übergeben, zu welchem Ziel aus dem aufgenommenen Zeitfenster $f_{k,p}$ geroutet wird:

$$n_{f_{k,p}} = n_{f_{i,l}} \quad (6-12)$$

Darüber hinaus ist der Zeiger des erreichbaren Zeitfensters $f_{k,p}$ auf das gewählte Zeitfenster $f_{i,j}$ zu setzen, um nach Erreichen des finalen Ziels die Route zu konstruieren:

$$B_{f_{k,p}} = f_{i,l} \quad (6-13)$$

Dafür wird auch das Segment $S_{f_{k,p}}$ gespeichert, um die Informationen zu erhalten, durch welche freien Zeitfenster das erreichbare Zeitfenster $f_{k,p}$ erreicht worden ist.

Im zweiten Fall ist die Restlänge des freien Zeitfensters $f_{k,p}$ für eine Anfahrt nicht ausreichend. Es wird geprüft, ob das freie Zeitfenster zumindest eine Durchfahrt erlaubt. Damit das freie Zeitfenster bei einer Fahrt mit maximaler Geschwindigkeit verlassen werden kann, bevor dieses endet, muss gelten:

$$T_{Ab,f_{i,l}} + t(s_{i,k} + s_{Ex,k}^{AS}, \infty) \leq T_{Ende,f_{k,p}} \quad (6-14)$$

Auch in diesem Fall ist sicherzustellen, dass auf keinem Knoten des Segmentes ein bereits reserviertes Zeitfenster verletzt wird. Für die Bestimmung der Austritte wird eine Fahrt über eine unendlich lange Distanz angenommen:

$$T_{RE,f_{x,y}} = T_{AE,f_{x,y}} = T_{Ab,f_{i,l}} + t(s_{i,x} + s_{Ex,x'}^{AS}, \infty) \quad (6-15)$$

Weist im ersten Fall das freie Zeitfenster $f_{k,p}$ nach der Ankunft eine ausreichende Restlänge auf (6-10), wird aber ein bereits reserviertes Zeitfenster auf einem Knoten des Segmentes verletzt, wird ebenso geprüft, ob das freie Zeitfenster $f_{k,p}$ nicht zumindest durchfahren werden kann.

Wird kein vorhergehendes Zeitfenster des Segmentes verletzt ($T_{AE,f_{x,y}} \leq T_{Ende,f_{x,y}}$), kann das Segment um das freie Zeitfenster $f_{k,p}$ erweitert werden (Abbildung 6-18, Tabelle 6-4).

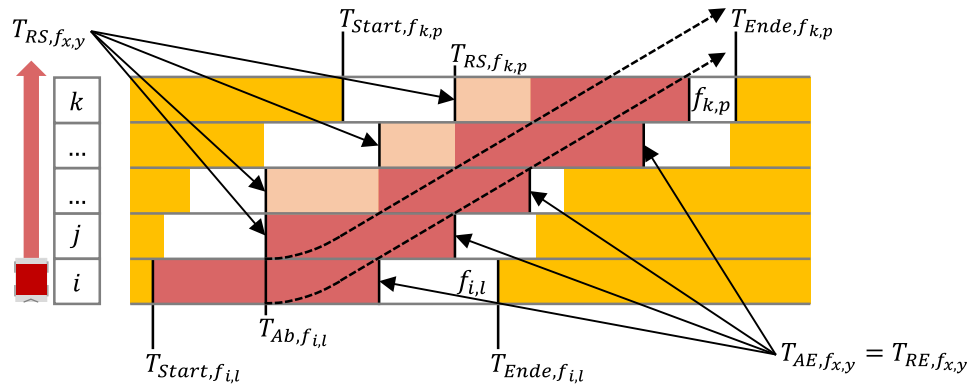


Abbildung 6-18: Das freie Zeitfenster $f_{k,p}$ ist nicht anfahrbar, jedoch durchfahrbar. Das Segment kann verlängert werden.

Tabelle 6-4: Um das durchfahrbare Zeitfenster $f_{k,p}$ erweiterte Segment.

Knoten	T_{Start}	T_{Ende}	T_{RS}	T_{ES}	T_{AE}	T_{RE}	T_{Ab}	T_{An}
i	$T_{Start, f_{i,l}}$	$T_{Ende, f_{i,l}}$	$T_{RS, f_{i,l}}$	$T_{ES, f_{i,l}}$	$T_{AE, f_{i,l}}$	$T_{AE, f_{i,l}}$	$T_{Ab, f_{i,l}}$	-
j	$T_{Start, f_{j,m}}$	$T_{Ende, f_{j,m}}$	$T_{RS, f_{j,m}}$	$T_{ES, f_{j,m}}$	$T_{AE, f_{j,m}}$	$T_{AE, f_{j,m}}$	-	-
...	-	-
k	$T_{Start, f_{k,p}}$	$T_{Ende, f_{k,p}}$	$T_{RS, f_{k,p}}$	$T_{ES, f_{k,p}}$	-	$T_{AE, f_{k,p}}$	-	-

Da das freie Zeitfenster $f_{k,p}$ lediglich durchfahrbar und nicht anfahrbar und somit auch nicht erreichbar ist, wird dieses auch nicht in die Menge der erreichbaren Zeitfenster aufgenommen.

Ist ein freies Zeitfenster weder erreichbar noch durchfahrbar, kann das Segment nicht verlängert werden und es wird kein erweitertes Segment der Menge der aktiven Segmente hinzugefügt. Kann in einer Iteration kein einziges aktives Segment erweitert werden, leert sich die Menge der aktiven Segmente und das Verfahren endet.

Wurde ein Segment in einer Iteration verlängert, erfolgt abschließend die Abfrage, ob die letzten beiden Knoten des Segmentes in der Zielknotenmenge des aktuellen Ziels enthalten sind. Da im übergeordneten Routingalgorithmus (Abbildung 6-6) lediglich die Ziele berücksichtigt werden, für deren Erreichen nur ein einzelner Knoten belegt wird, ist die Überprüfung, ob das nächste Ziel im Routing erreicht wird, an dieser Stelle notwendig.

Sind die letzten beiden Knoten des Segmentes in der Zielknotenmenge des aktuellen Ziels enthalten, folgt die Prüfung der Zielbedingungen: Für jedes aktive Segment wird geprüft, ob das letzte freie Zeitfenster $f_{k,p}$ erreichbar bleibt, wenn die Fahrt zwischen den Knoten i und k durch einen Aufenthalt an der Halteposition des aktuellen Ziels

$Z_{n_{f_{i,l}}}$ der Dauer $t_{V,n_{f_{i,l}}}$ unterbrochen wird (Abbildung 6-19). Dabei wird wie zuvor beschrieben vorgegangen, lediglich ergeben sich eine verzögerte Ankunftszeit im freien Zeitfenster $f_{k,p}$ sowie andere Austrittszeiten aus den vorhergehenden Zeitfenstern. Bleibt das freie Zeitfenster $f_{k,p}$ erreichbar und wird dieses in die Menge der erreichbaren Zeitfenster aufgenommen, ist der Zähler n zu aktualisieren:

$$n_{f_{k,p}} = n_{f_{i,l}} + 1 \tag{6-16}$$

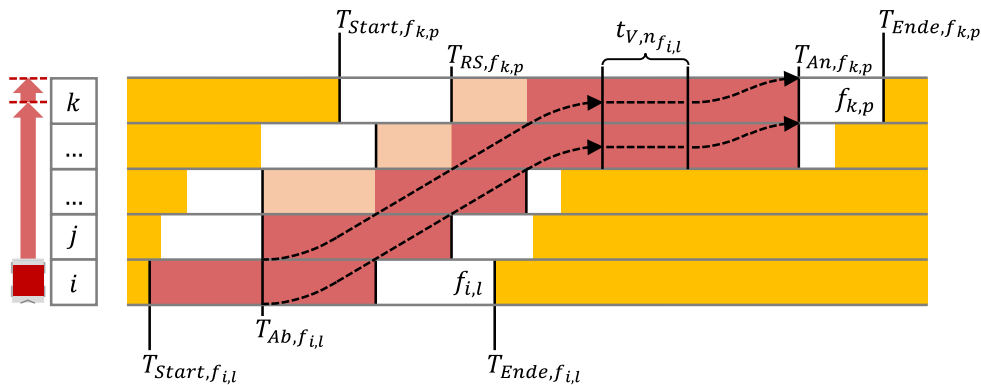


Abbildung 6-19: Segment, bei welchem das nächste Ziel erreicht wird.

Um die Robustheit des Routings zu erhöhen, ist es möglich, Pufferzeiten zu berücksichtigen, welche zwischen zwei Reservierungen auf einem Knoten eingefügt werden. Durch solche Pufferzeiten können geringe Verspätungen abgefangen werden, ohne dass nachfolgende Fahrzeuge von ihrer geplanten Route abweichen müssen.

t_p : Zeitlicher Puffer, welcher zwischen zwei reservierten Zeitfenstern auf einem Knoten eingefügt wird.

Diese Pufferzeiten sind zu berücksichtigen bei

- der Abfrage, ob ein Zielknoten nach der Verweildauer vollständig verlassen werden kann (6-3):

$$T_{Anf_{i,l}} + t_{Ex,i}^{A_{f_{i,l}}} \leq T_{Ende,f_{i,l}} - t_p \tag{6-17}$$

- der Abfrage, ob die Restlänge des freien Zeitfensters nach der Ankunft ausreichend ist, um den Knoten wieder vollständig zu verlassen (6-10):

$$T_{An,f_{k,p}} + t_{Ex,k}^{AS} \leq T_{Ende,f_{k,p}} - t_P \quad (6-18)$$

- der Abfrage, ob ein freies Zeitfenster die Durchfahrt erlaubt, falls dieses nicht erreichbar ist (6-14):

$$T_{Ab,f_{i,l}} + t(s_{i,k} + s_{Ex,k}^{AS}, \infty) \leq T_{Ende,f_{k,p}} - t_P \quad (6-19)$$

- der Bestimmung der Reservierungen der Zeitfenster eines Segmentes (6-11), (6-15):

$$T_{RE,f_{x,y}} = T_{AE,f_{x,y}} + t_P \quad (6-20)$$

- sowie bei der Überprüfung, ob auf einem Knoten des Segmentes ein bereits reserviertes Zeitfenster verletzt wird:

$$T_{AE,f_{x,y}} \leq T_{Ende,f_{x,y}} - t_P \quad (6-21)$$

6.1.5 Exkurs: Erreichbarkeit freier Zeitfenster bei weiteren Reservierungsformen

Bei der Erreichbarkeit freier Zeitfenster und deren Reservierung wird eine Pufferzeit verwendet, welche das sichere Anhalten der Fahrzeuge ermöglicht, falls ein vorhergehendes Fahrzeug verspätet ist. Ferner wird vorausgesetzt, dass die Reservierung auf einem Knoten endet, sobald das Fahrzeug diesen vollständig verlassen hat ($T_{RE,f_{x,y}} = T_{AE,f_{x,y}}$). Somit ergeben die Reservierungen der Zeitfenster eines Segmentes eine Treppenform. In diesem Exkurs werden zwei weitere Reservierungsformen aufgezeigt, welchen andere Annahmen zu Grunde liegen.

Kann die Ausführung eines Segmentes – beispielsweise aufgrund eingeschränkter Kommunikationsmöglichkeiten – nicht mehr abgebrochen werden sobald ein Fahrzeug mit der Ausführung begonnen hat, muss sichergestellt werden, dass es zu keinen Kollisionen mit anderen Fahrzeugen kommen kann, auch wenn diese sich verspäten. Dies wird dadurch erreicht, dass die Reservierung aller Zeitfenster eines Segmentes mit der Abfahrt des Fahrzeuges beginnen ($T_{RS,f_{x,y}} = T_{Ab,f_{i,j}}$). Damit ergeben die Reservierung der Zeitfenster eine Dreiecksform (Abbildung 6-20).

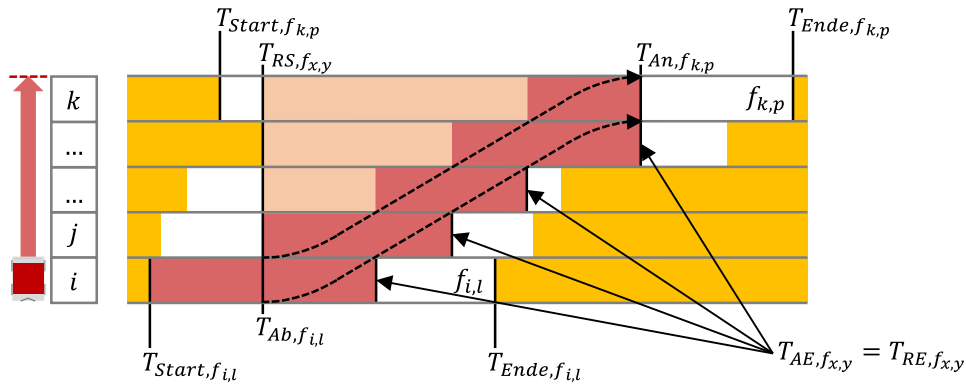


Abbildung 6-20: Segment mit der Dreiecksform.

Das Vorgehen für die Überprüfung der Erreichbarkeit eines freien Zeitfensters unter diesen Bedingungen ist in Abbildung 6-21 beschrieben. Modifikationen gegenüber dem zuvor beschriebenen Vorgehen sind nur zu Beginn vorzunehmen.

Voraussetzung für die Erreichbarkeit ist zunächst, dass sich das gewählte Zeitfenster $f_{i,l}$ und das freie Zeitfenster $f_{k,p}$ überlappen:

$$T_{Ab, f_{i,l}} < T_{End, f_{k,p}} \tag{6-22}$$

$$T_{Start, f_{k,p}} < T_{End, f_{i,l}} \tag{6-23}$$

Die Abfahrt aus dem ersten Zeitfenster $f_{i,l}$ muss erfolgen, bevor das freie Zeitfenster $f_{k,p}$ endet und das freie Zeitfenster $f_{k,p}$ muss beginnen, bevor das erste Zeitfenster $f_{i,l}$ endet. In diesem Fall kann die frühestmögliche Abfahrtszeit bestimmt werden:

$$T_{Ab, f_{i,l}} = \max \{ T_{An, f_{i,l}}, T_{Start, f_{k,p}} \} \tag{6-24}$$

Beginnt das freie Zeitfenster $f_{k,p}$ später als die Ankunft im gewählten Zeitfenster $f_{i,l}$ ergibt sich eine Wartezeit auf dem Knoten i . Nun lässt sich gemäß (6-9) die Ankunftszeit in dem freien Zeitfenster $f_{k,p}$ bestimmen. Das weitere Vorgehen zur Überprüfung der Erreichbarkeit entspricht dem beschriebenen Vorgehen aus Abbildung 6-16, wobei lediglich bei der Bestimmung der Reservierungen darauf zu achten ist, dass diese mit der Abfahrt aus dem ersten Zeitfenster $f_{i,l}$ beginnen.

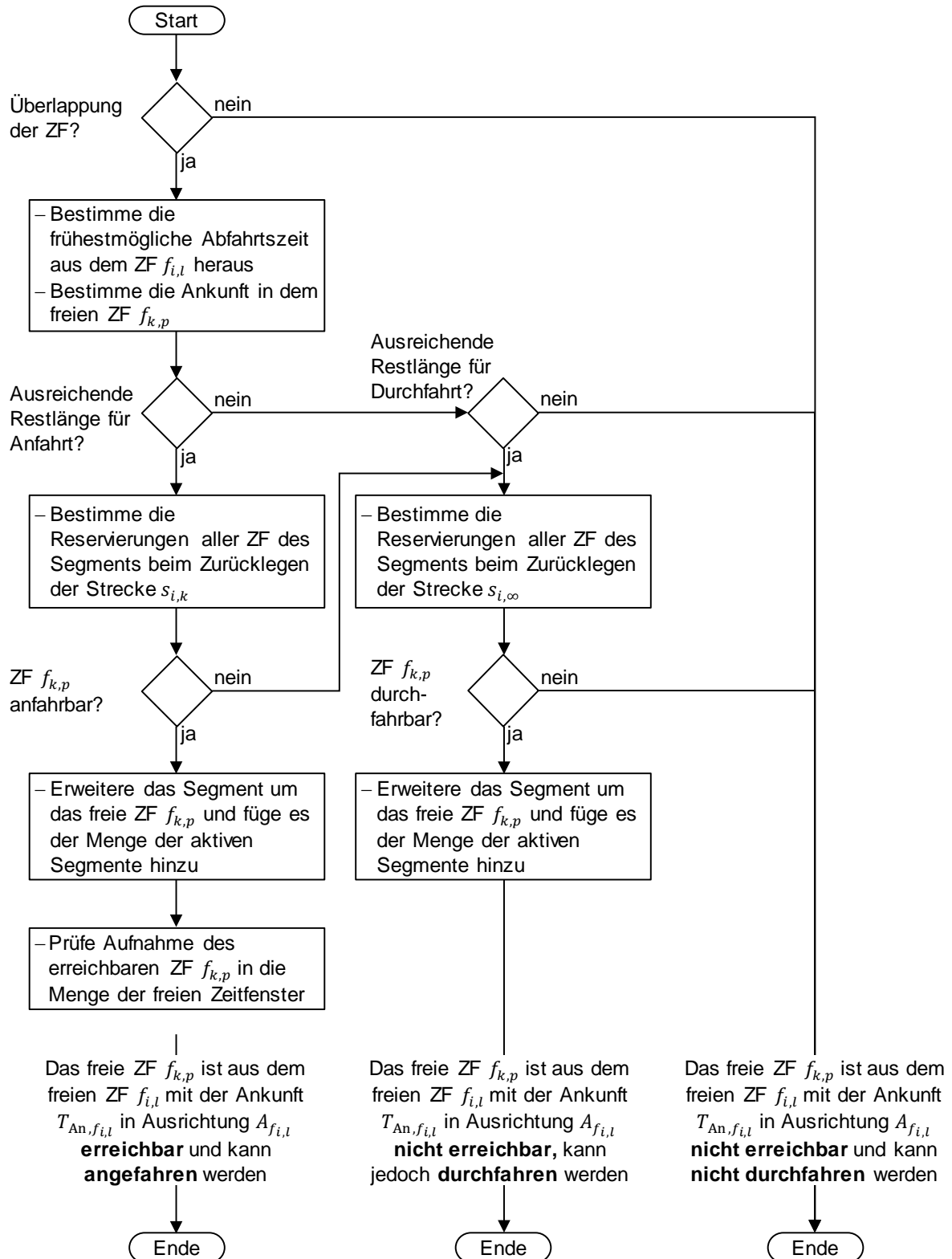


Abbildung 6-21: Test auf Erreichbarkeit eines freien Zeitfensters $f_{k,p}$ bei der Dreiecksform (bzw. Blockform).

Ist es nicht möglich die reservierten Zeitfenster freizugeben sobald ein Fahrzeug einen Knoten vollständig verlassen hat, sondern erst dann, wenn ein Fahrzeug wieder zum

Stillstand kommt und ein Segment beendet hat, enden die Reservierungen eines Segmentes mit der Ankunft im letzten Zeitfenster des Segmentes ($T_{RE,f_{x,y}} = T_{An,f_{k,p}}$). In diesem Fall ergeben die reservierten Zeitfenster eines Segmentes eine Blockform (Abbildung 6-22).

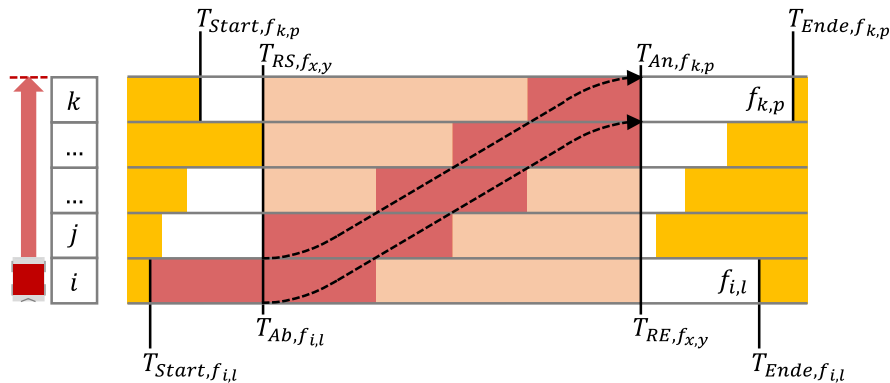


Abbildung 6-22: Segment mit Blockform.

Das Vorgehen zur Überprüfung der Erreichbarkeit entspricht bis auf die Bestimmung der Reservierungen aller Zeitfenster des Segmentes dem zuvor beschriebenen Vorgehen aus Abbildung 6-21, wobei wiederum die Reservierungen entsprechend anzupassen sind.

Sowohl bei der Dreiecksform, wie auch bei der Blockform, hat die maximale Segmentlänge eine Auswirkung auf den erzielbaren Durchsatz. Kurze Segmente führen zu häufigen Verzögerungs- und Beschleunigungsvorgängen. Längere Segmente erlauben den Fahrzeugen hingegen, die Ziele mit weniger Zwischenstopps zu erreichen. Die Knoten sind jedoch auch für längere Zeit reserviert. Die Bestimmung der optimalen maximalen Segmentlängen muss daher für diese Reservierungsformen durch eine Simulationsstudie, wie in [Lie-2018a] durchgeführt, ermittelt werden.

6.2 Routingbasierte Sequenzierung

Eine Anforderung gemäß Unterkapitel 3.3 ist es, eine Sequenz am Übergabepunkt sicherzustellen. Der Punkt, an welchem die Transporteinheiten in einer definierten Reihenfolge bereitgestellt werden müssen, wird als Sequenzpunkt bezeichnet. Jeder Sequenzpunkt im System wird durch einen eigenen Sequenzstrom versorgt. Somit können in einem System mehrere Sequenzströme existieren, welche voneinander unabhängig sind. Innerhalb eines Stromes sind die bereitzustellenden Transporteinheiten mit Sequenznummern versehen und nach aufsteigender Sequenznummer bereitzu-

stellen. Dabei können mehrere Transporteinheiten die gleiche Sequenznummer haben, etwa wenn diese einen Batch bilden, innerhalb dessen die Reihenfolge nicht relevant ist. In diesem Unterkapitel wird auf einen Ansatz zurückgegriffen, welcher in [Lie-2018b] beschrieben worden ist.

Durch die Anwendung des zeitfensterbasierten Routings ist nicht nur die geplante Ankunftszeit der Fahrzeuge an einem Ziel bekannt, sondern auch die Reihenfolge, in welcher die Fahrzeuge das Ziel erreichen. Dies wird für die Sequenzierung genutzt, welche in das zeitfensterbasierte Routing integriert wird. Dafür wird zunächst die Eingabe in den Algorithmus um die Sequenznummer S_n erweitert.

S_n : Sequenznummer, welche angibt, an welcher Position im Sequenzstrom das n -te Ziel zu erreichen ist.

Für jeden Sequenzpunkt wird eine Liste gespeichert, in welcher die Ankunftszeiten der einzelnen Sequenznummern hinterlegt werden. Diese Ankunftszeiten ergeben sich aus dem Routingprozess.

$T_{An,k}^{S_n}$: Ankunftszeitpunkt der Sequenznummer S_n am Sequenzpunkt k .

Um die Sequenz sicherzustellen, werden die Zielbedingungen um eine weitere Bedingung ergänzt:

$$T_{An,k}^{S_{n-1}} < T_{An,k}^{S_n} \quad (6-25)$$

Wird ein nächstes Ziel im Routing erreicht, ist neben der ausreichenden Restdauer des Zeitfensters (6-3) zu prüfen, ob die Ankunft der vorhergehenden Sequenznummer früher erfolgt als die Ankunft der aktuell zu routenden Sequenznummer. Ist diese Bedingung nicht erfüllt, kann in dem gewählten Zeitfenster das nächste Ziel nicht erreicht werden. Voraussetzung für dieses Vorgehen ist, dass die Fahrzeuge nach aufsteigenden Sequenznummern geroutet werden.

Das Vorgehen wird im Folgenden anhand zweier Beispiele verdeutlicht. Der Knoten k stellt den Sequenzpunkt und das Ziel des Routings dar. Das Routing der vorhergehenden Sequenznummer $S_n - 1$ ist bereits erfolgt und der Ankunftszeitpunkt gespeichert.

In dem ersten Beispiel (Abbildung 6-23 bis Abbildung 6-25) erfolgt die Ankunft der vorhergehenden Sequenznummer aus der entgegengesetzten Richtung. Für die

nächste Iteration im Routingprozess für die Sequenznummer S_n wird das Zeitfenster $f_{i,l}$ ausgewählt. Es erfolgt die Erstellung der Segmente.

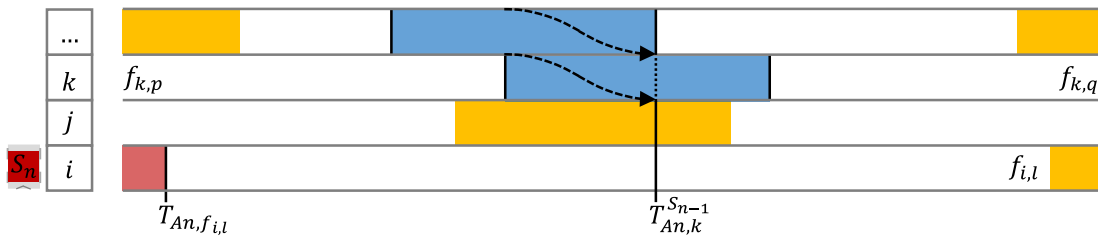


Abbildung 6-23: Die Sequenznummer $S_n - 1$ erreicht den Sequenzpunkt aus der entgegengesetzten Richtung.

Das freie Zeitfenster $f_{k,p}$ ist aus dem gewählten Zeitfenster $f_{i,l}$ heraus erreichbar. Wird dieses jedoch in einer späteren Iteration ausgewählt, ist die Sequenzbedingung nicht erfüllt und das Ziel Z_n kann in dem Zeitfenster $f_{k,p}$ nicht erreicht werden.

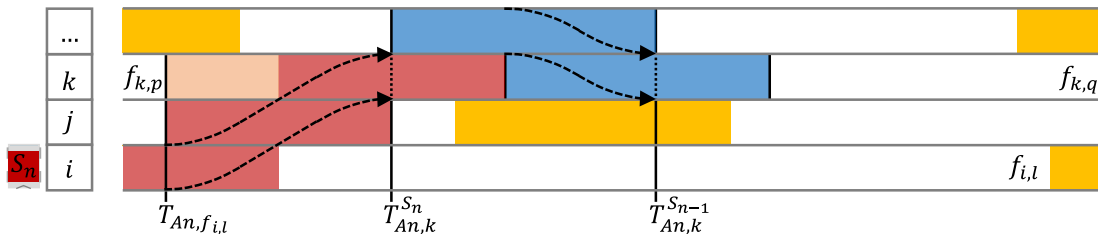


Abbildung 6-24: In dem erreichbaren Zeitfenster $f_{k,p}$ kann das Ziel nicht erreicht werden, da die Sequenz verletzt ist.

Neben dem Zeitfenster $f_{k,p}$ ist auch das freie Zeitfenster $f_{k,q}$ erreichbar. Hierfür ist eine längere Wartezeit in dem gewählten Zeitfenster $f_{i,l}$ notwendig. Wird in einer späteren Iteration das erreichbare Zeitfenster $f_{q,k}$ ausgewählt, ist auch die Sequenzbedingung erfüllt und das Ziel kann erreicht werden.

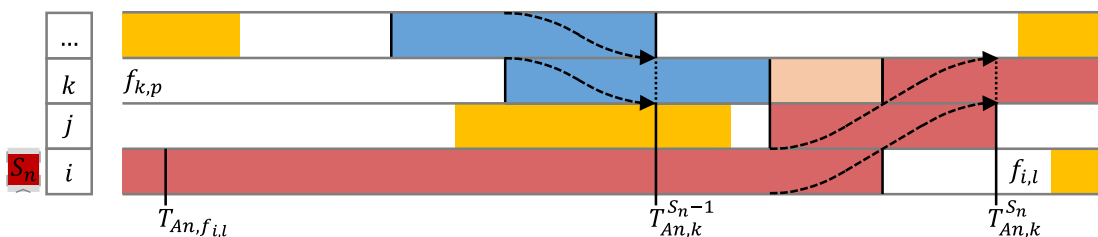


Abbildung 6-25: Das später erreichbare Zeitfenster $f_{k,q}$ hingegen ermöglicht das Erreichen des nächsten Ziels.

In dem zweiten Beispiel (Abbildung 6-26 bis Abbildung 6-28) erfolgt die Ankunft der vorhergehenden Sequenznummer aus der gleichen Richtung. Für die nächste Iteration im Routingprozess für die Sequenznummer S_n wird erneut das Zeitfenster $f_{i,l}$ ausgewählt. Es erfolgt die Erstellung der Segmente.

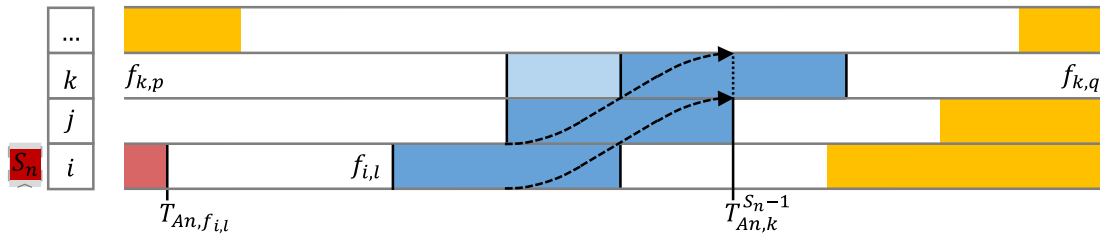


Abbildung 6-26: Die Sequenznummer $S_n - 1$ erreicht den Sequenzpunkt aus der gleichen Richtung.

Das freie Zeitfenster $f_{k,p}$ ist aus dem gewählten Zeitfenster $f_{i,l}$ heraus erreichbar, allerdings wird abermals die Sequenzbedingung verletzt.

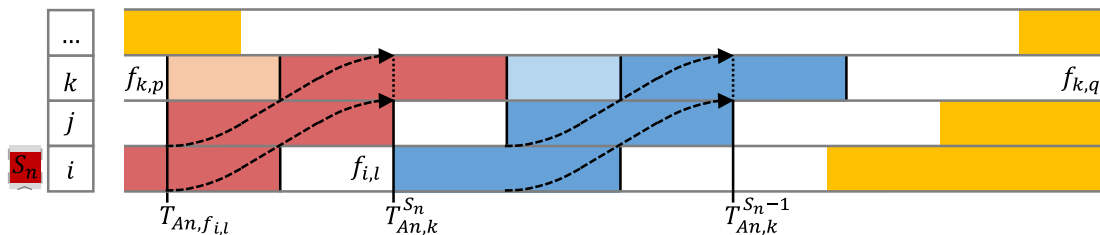


Abbildung 6-27: In dem erreichbaren Zeitfenster $f_{k,p}$ kann das Ziel abermals nicht erreicht werden, da die Sequenz verletzt ist.

Da das freie Zeitfenster $f_{k,p}$ erreichbar ist, kann das Segment verlängert werden. In einer nachfolgenden Iteration wird das entsprechende erreichbare Zeitfenster ausgewählt, aus welchem nun das freie Zeitfenster $f_{k,q}$ erreichbar ist. Auch in diesem Fall ist nun die Sequenz korrekt.

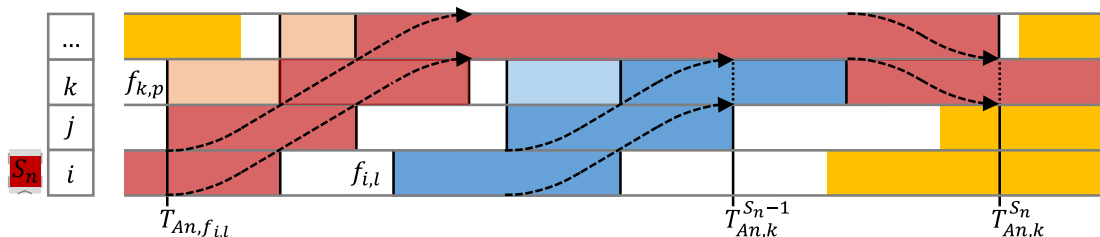


Abbildung 6-28: Durch eine Ausweichbewegung auf den Nachbarknoten ist das freie Zeitfenster $f_{k,q}$ erreichbar, in welchem das nächste Ziel erreicht werden kann.

Nach dem erfolgreichen Routingvorgang wird die Ankunft der Sequenznummer S_n am Sequenzpunkt k gespeichert und kann für das korrekte Routing der Sequenznummer $S_n + 1$ herangezogen werden.

Um die während des Routings geplante Sequenz einzuhalten, warten die Fahrzeuge (wie beim ersten aufgeführten Beispiel), oder führen Ausweichbewegungen durch (wie beim zweiten aufgeführten Beispiel), um vorhergehende Sequenznummern passieren zu lassen. Eine Bereitstellung der Transporteinheiten in einer definierten Sequenz führt aufgrund dieser Wartevorgänge und Ausweichbewegungen zu Durchsatzverlusten.

Der Untersuchung der Durchsatzverluste wird in einem Fallbeispiel in Kapitel 8 nachgegangen.

6.3 Ausführung einer berechneten Route

Das Ergebnis des Routingprozesses ist eine Liste mit Segmenten, welche nacheinander abgefahren werden, um alle übergebenen Ziele deadlockfrei zu erreichen. Dabei ist es nicht ausreichend, die Fahrzeuge die Segmente lediglich entsprechend der im Routing bestimmten Abfahrtszeiten zurücklegen zu lassen. Für das Routing der Fahrzeuge wird mit exakten Werten bezüglich der maximalen Geschwindigkeit, der Beschleunigung, der Zeitdauer für den Wechsel der Ausrichtung sowie den vorgegebenen Verweildauern an den einzelnen Zielen gerechnet. Jedoch kann die Einhaltung dieser Werte im realen Betrieb aus unterschiedlichen Gründen nicht garantiert werden. Je nach Zustand der Rollen oder auch der Fahrspuren können Schwankungen bei der Beschleunigung oder auch bei der maximalen Geschwindigkeit entstehen. Denkbar ist auch, dass ein Lagerspiel mehr Zeit in Anspruch nimmt, als beim Routing ursprünglich geplant war. Die Konsequenz daraus ist, dass sich die Fahrzeuge bezüglich ihrer reservierten Zeitfenster verspäten können oder auch dem Zeitplan voraus sind. Eine strikte Orientierung an den Abfahrtszeiten der einzelnen Segmente kann daher zu Deadlocks und schlimmer noch, zu Kollisionen führen. Wird hingegen die Reihenfolge eingehalten, in welcher die einzelnen Fahrzeuge die Zeitfenster auf einem Knoten reserviert haben, können die Routen auch im Fall von Verspätungen deadlockfrei ausgeführt werden. Das Vorgehen hierfür wird im Abschnitt 6.3.1 beschrieben. Wichtiger Bestandteil dabei ist die Beanspruchung der Knoten. Dieser Prozess wird im anschließenden Abschnitt 6.3.2 erläutert. Der Abschnitt 6.3.3 geht auf die Zugriffe auf die Datenstruktur der Zeitfenster ein.

6.3.1 Ausführung eines Segmentes

Das Beispiel in Abbildung 6-29 verdeutlicht den eingangs beschriebenen Sachverhalt. Der Abfahrtszeitpunkt für das rote FTF ist erreicht, um den Knoten 2 zu befahren. Das vorhergehende blaue FTF hat sich jedoch verspätet und das reservierte Zeitfenster auf dem Knoten 2 noch nicht freigegeben. Wenn das rote FTF nun seine Fahrt fortsetzt, begegnen sich die Fahrzeuge zwischen den Knoten 2 und 4 und ein Deadlock entsteht.

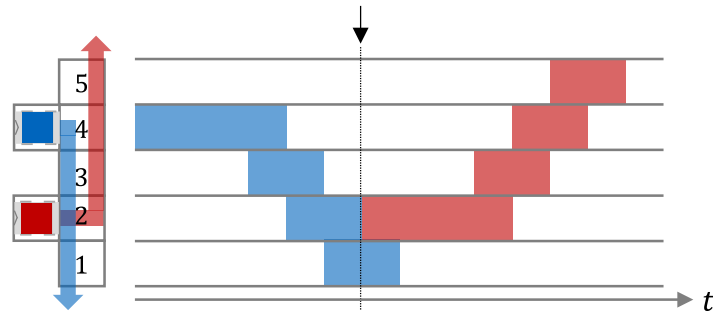


Abbildung 6-29: Das Einhalten der Reihenfolge garantiert Deadlockfreiheit.

Deadlocks können jedoch ausgeschlossen werden, wenn die Reihenfolge auf den Knoten eingehalten wird, welche sich durch die Reservierungen der Zeitfenster ergibt [Maz-2005].

Damit ein Fahrzeug einen Knoten befahren darf, müssen die beiden folgenden Bedingungen erfüllt sein:

1. Ein Fahrzeug darf einen Knoten nur dann befahren, wenn seine Reservierung auf diesem Knoten bereits begonnen hat.
2. Ein Fahrzeug darf einen Knoten nur dann befahren, wenn es auf diesem das nächste Zeitfenster reserviert hat.

Die erste Bedingung verhindert, dass Fahrzeuge einen Knoten zu früh befahren. Dies ist aus zwei Gründen nicht zulässig: Zum einen ist für eine korrekte Ausführung eines Segmentes notwendig, dass die im Routing bestimmte Abfahrtszeit aus dem ersten Zeitfenster des Segmentes nicht unterschritten wird. In dem Beispiel in Abbildung 6-30 hat das rote FTF auf dem zweiten Knoten des Segmentes das erste Zeitfenster reserviert. Beginnt das rote FTF nun unmittelbar mit der Ausführung des Segmentes, muss es diese unterbrechen, da es den Knoten 4 zu früh erreicht und das vorhergehende blaue FTF diesen Knoten noch nicht passiert hat.

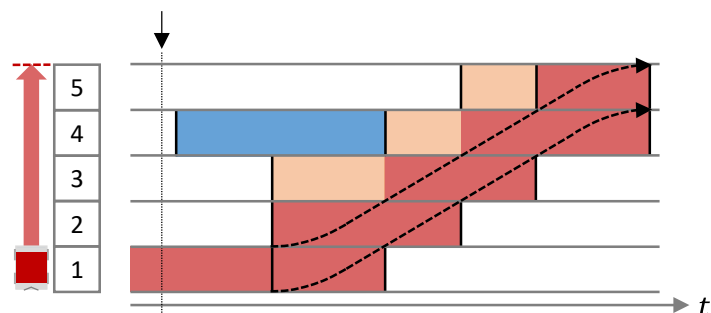


Abbildung 6-30: Eine verfrühte Abfahrt verhindert die korrekte Ausführung eines geplanten Segmentes.

Zum anderen entstehen Inkonsistenzen bezüglich des Systemzustandes. Ist ein Fahrzeug seinen reservierten Zeitfenstern voraus und erfolgt das Routing für ein weiteres Fahrzeug, können Deadlocks entstehen. In dem Beispiel in Abbildung 6-31 wird das blaue FTF geroutet. Unter Berücksichtigung der existierenden Reservierungen, kann das blaue FTF den Knoten 2 passieren, bevor das reservierte Zeitfenster für das rote FTF beginnt. Dieses hat den Knoten jedoch verfrüht befahren. Somit kommt es erneut zu einer Begegnung der beiden Fahrzeuge zwischen den Knoten 2 und 4 und damit zu einem Deadlock.

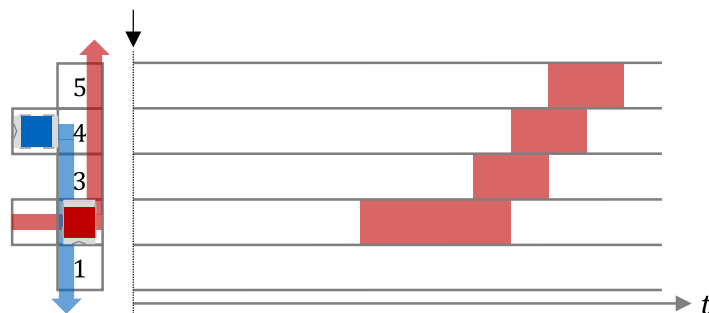


Abbildung 6-31: Ein FTF befährt einen Knoten zu früh.

Die zweite Bedingung stellt die geplante Reihenfolge der Fahrzeuge sicher. Im eingangs angeführten Beispiel in Abbildung 6-29 wird der Deadlock dadurch verhindert. Das rote FTF darf den Knoten 2 erst befahren, wenn das blaue FTF diesen Knoten passiert und das reservierte Zeitfenster gelöscht hat.

Die Vorgehensweise zur Ausführung einer berechneten Route ist in Abbildung 6-32 dargestellt. Das dargestellte Flussdiagramm wird durchlaufen, sobald ein Fahrzeug eine neue Route zugewiesen bekommen hat und wann immer es ein Segment dieser Route beendet hat.

Zunächst wird geprüft, ob das Fahrzeug sein nächstes Ziel im korrekten Zeitfenster erreicht. Es ist durchaus möglich, dass das Fahrzeug den Knoten nochmal verlassen muss, bevor es seine Zielaktion dort ausführen kann, welche eine längere Zeitdauer erfordert. Erreicht ein Fahrzeug sein nächstes Ziel im korrekten Zeitfenster, wird die Zielaktion angestoßen, welche beispielsweise die Ausführung eines Lagerspiels darstellen kann. Endet mit dem Erreichen des Ziels die Route, wird also das finale Ziel erreicht, ist die Ausführung der Route abgeschlossen.

Anderenfalls wird ein Wechsel der Ausrichtung vollzogen, falls dieser für die Ausführung des nächsten Segmentes notwendig ist.

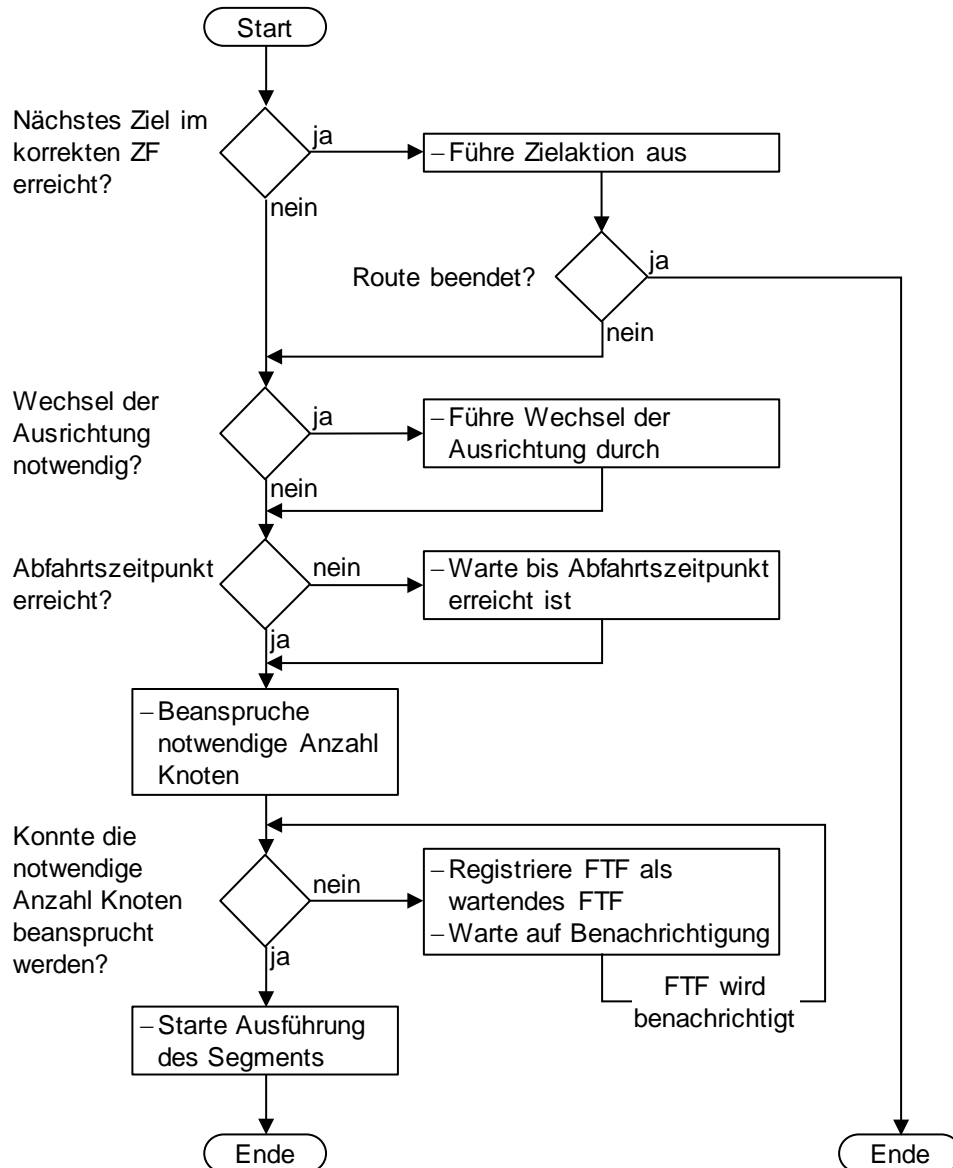


Abbildung 6-32: Ausführung einer berechneten Route.

Ist der Abfahrtszeitpunkt des nächsten Segmentes erreicht, erfolgt die Beanspruchung der notwendigen Anzahl der Knoten. Durch diesen Prozess, welcher im folgenden Abschnitt erläutert wird, wird die zweite Bedingung – die Einhaltung der Reihenfolge – sichergestellt. Konnte die notwendige Anzahl Knoten beansprucht werden, wird mit der Ausführung des Segmentes begonnen. Das Fahrzeug beginnt zu beschleunigen, wobei die Dauer bis zum Einleiten der Verzögerung gemäß (5-14) zu bestimmen ist.

Konnte die notwendige Anzahl Knoten hingegen nicht beansprucht werden, wird das Fahrzeug an dem Knoten, welcher nicht beansprucht werden konnte, als wartend registriert.

Sobald ein Fahrzeug einen Knoten vollständig verlassen hat, wird zum einen das entsprechende reservierte Zeitfenster gelöscht. Zum anderen wird geprüft, ob das nachfolgende Fahrzeug bereits als wartendes Fahrzeug bei dem verlassenen Knoten registriert ist. In diesem Fall wird das wartende Fahrzeug benachrichtigt, welches erneut mit der Beanspruchung der Knoten beginnt.

6.3.2 Beanspruchung von Knoten

Ist der Abfahrtszeitpunkt für ein Segment erreicht, beansprucht ein Fahrzeug die nächsten Knoten. Ein Fahrzeug kann einen Knoten genau dann beanspruchen, wenn es das erste Zeitfenster auf diesem Knoten reserviert hat und somit das Fahrzeug ist, welches den Knoten als nächstes passiert.

Dabei müssen genau so viele Knoten beansprucht werden, dass ein Fahrzeug sicher zum Stillstand kommt, falls ein vorhergehendes Fahrzeug verspätet ist. Ausgehend von der Ankunftsposition des nächsten Knotens, muss das Minimum aus Verzögerungsstrecke und verbleibender Segmentlänge beansprucht werden. Abbildung 6-33 verdeutlicht dies anhand zweier Beispiele mit unterschiedlich langen Verzögerungsstrecken.

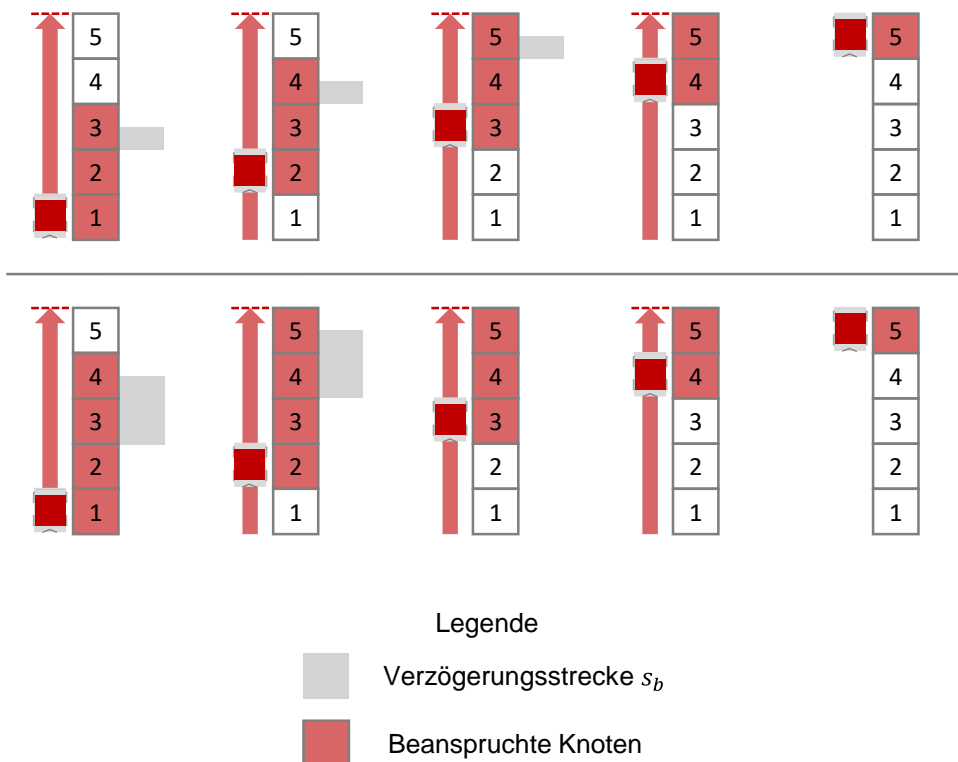


Abbildung 6-33: Beanspruchung von Knoten.

Die Beanspruchung der nächsten Knoten erfolgt zum einen zu Beginn eines Segmentes, zum anderen sobald die Ankunftsposition eines Knoten erreicht wird. Somit entsteht ein Sicherheitskorridor, welcher dynamisch von einem Fahrzeug vor sich hergeschoben wird.

In dem Beispiel in Abbildung 6-34 erreicht das rote FTF die Ankunftsposition des zweiten Knoten im Segment. Um die Ausführung des Segmentes fortzusetzen, muss nun der Knoten 4 beansprucht werden. Auf diesem beginnt das reservierte Zeitfenster, jedoch ist das vorhergehende blaue FTF verspätet. Somit muss das rote FTF auf dem Knoten 3 einen ungeplanten Stopp einlegen. Dafür ist die Dauer bis zur Einleitung der Verzögerung erneut zu bestimmen, welche sich in Abhängigkeit der aktuellen Geschwindigkeit gemäß (5-15), bzw. (5-20) errechnen lässt. Das Konzept der Beanspruchung sowie die Reservierung der zeitlichen Puffer ermöglichen das sichere Anhalten des Fahrzeuges.

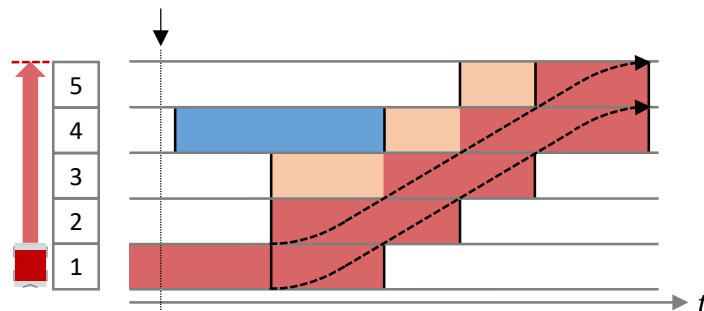


Abbildung 6-34: Ein geplantes Segment wird verkürzt.

Erreicht das rote FTF den Knoten 3, wird dem Ablauf nach der Beendigung eines Segmentes (Abbildung 6-32) gefolgt, wobei zunächst lediglich das zuvor unterbrochene Segment beendet wird.

Für die in Abschnitt 6.1.5 dargestellten alternativen Reservierungsformen Dreieck und Block gilt hingegen, dass vor der Ausführung eines Segmentes sämtliche Knoten des Segmentes beansprucht werden müssen. Somit erfolgt die Abfahrt erst dann, wenn das FTF auf allen Knoten des Segmentes das erste Zeitfenster reserviert hat. Weitere Abfragen während der Ausführung eines Segmentes sind jedoch nicht notwendig.

6.3.3 Zugriffe auf die Datenstruktur der Zeitfenster

Die reservierten Zeitfenster werden für alle Knoten in einer zentralen Datenstruktur vorgehalten. Während des Routings werden die freien Zeitfenster ausgelesen und auf Erreichbarkeit überprüft. Nach dem erfolgreichen Routing werden die benötigten Zeitfenster reserviert, was einen Schreibvorgang erfordert.

Während der Ausführung einer reservierten Route erfolgen Lesezugriffe auf die Datenstruktur, um festzustellen, ob ein Fahrzeug das erste Zeitfenster auf einem Knoten reserviert hat. Nach dem Verlassen eines Knotens wird das entsprechende Zeitfenster gelöscht. Abbildung 6-35 stellt die unterschiedlichen Zugriffe auf die Datenstruktur der Zeitfenster dar.

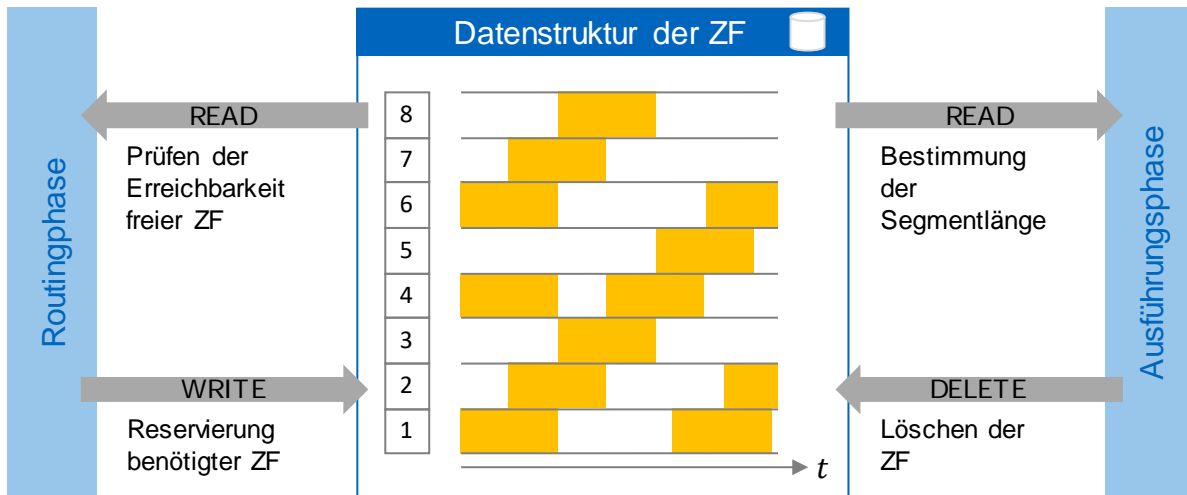


Abbildung 6-35: Zugriffe auf die Datenstruktur der Zeitfenster.

Der Routingprozess kann – je nach Anzahl der Knoten im Graphen, Anzahl der Fahrzeuge im System und Länge der übergebenen Zielliste – Rechenzeit in Anspruch nehmen, während derer die Ausführung berechneter Routen voranschreitet. Da während der Ausführung jedoch lediglich Zeitfenster ausgelesen oder gelöscht werden, lassen sich die benötigten Zeitfenster in jedem Fall reservieren. Routing und Ausführung können somit simultan erfolgen, während hingegen mehrere Routingprozesse sequentiell abgewickelt werden müssen.

6.4 Strategien für den Umgang mit Störungen

In dem vorangegangenen Unterkapitel wurde die Ausführung einer berechneten Route beschrieben. Diese erfolgt deadlockfrei, auch wenn sich Fahrzeuge verspäten. Wenn ein Fahrzeug sich jedoch nicht nur verspätet, sondern sich in einem Störungszustand befindet, kann das zu einem Stillstand des gesamten Systems führen. In diesem Unterkapitel werden vier unterschiedliche Handlungsalternativen für den Umgang mit Störungen aufgezeigt und die entsprechenden Strategien dokumentiert. Diese Strategien sind generisch und nicht abhängig von einem zugrundeliegenden Layout. Durch die Integration der Störungsstrategien in das Simulationsmodell lässt sich dieses zusätzlich durch die Verfügbarkeit sowie eine durchschnittliche Reparaturzeit (MTTR – Mean

Time To Repair) parametrisieren und der Durchsatz auch im Fall von Störungen analysieren.

Störung werden gemäß den Parametern nur dann erzeugt, wenn ein Fahrzeug weitere Knoten für die Ausführung seiner Route beansprucht. Somit ist die Anzahl der möglichen Zustände, welche ein System beim Auftreten und nach dem Beheben einer Störung, einnehmen kann, beschränkt. Dadurch lässt sich die Komplexität der Strategien reduzieren, da nur auf eine begrenzte Menge an möglichen Systemzuständen reagiert werden muss. Dennoch lassen sich die Verfügbarkeiten der Fahrzeuge und somit des Gesamtsystems hinreichend genau evaluieren.

Geht ein Fahrzeug in Störung, kann dieses seine Fahrt nicht fortsetzen und blockiert den Knoten, auf welchem es sich befindet. Vereinfachend wird angenommen, dass das Fahrzeug nach der hinterlegten MTTR wieder einsatzbereit ist und seine Fahrt fortsetzen kann.

Um den Störungen zu begegnen, wird zum einen in die Ausführung der Routen eingegriffen. Zum anderen werden vorhandene Routen gelöscht und neue Routen berechnet. In die Auftragsvergabe, welche in jedem Anwendungsfall individuell ausgestaltet ist, wird hingegen nicht eingegriffen. Die nachfolgend beschriebenen Strategien beruhen auf [Lie-2019].

6.4.1 Störungen ignorieren

Die erste Handlungsalternative besteht darin, dass auftretende Störungen ignoriert werden. Tritt bei einem Fahrzeug eine Störung auf, kann dieses die geplante Route nicht mehr unmittelbar fortsetzen und blockiert den Knoten, auf welchem es sich befindet. Somit werden weitere Fahrzeuge aufgehalten, welche entweder den blockierten Knoten passieren sollen oder aber einen Knoten entlang der verbleibenden Route des gestörten Fahrzeuges, sofern dieses ein früheres Zeitfenster auf diesem Knoten reserviert hat. In beiden Fällen registrieren sich die Fahrzeuge als wartende Fahrzeuge und warten auf die Benachrichtigung (siehe Abbildung 6-32). Die wartenden Fahrzeuge können ihre Route ebenfalls nicht fortsetzen, was wiederum weitere Fahrzeuge an der Ausführung ihrer Routen behindert. Auf diese Art und Weise kann sich eine Störung über das System ausbreiten bis zum Extremfall, in welchem kein einziges Fahrzeug seine Fahrt fortsetzen kann.

Ist die Störung behoben, setzt das entstörte Fahrzeug seine Fahrt fort, gibt die reservierten Zeitfenster frei und benachrichtigt die wartenden Fahrzeuge. Nach und nach

setzen die wartenden Fahrzeuge ihre Fahrt ebenfalls fort und der Systemzustand normalisiert sich wieder. Um diese Strategie umzusetzen, sind bis auf die Integration der Störungen keine weiteren Eingriffe notwendig.

6.4.2 System pausieren

Die zweite Handlungsalternative besteht darin, sämtliche Fahrzeuge im Fall einer Störung anzuhalten und das System somit zu pausieren. Wann immer ein Fahrzeug ein Segment seiner Route beendet hat und die nächsten Knoten für die Weiterfahrt beansprucht (siehe Abbildung 6-32), wird geprüft, ob eine Störung vorliegt und das Fahrzeug somit seine Fahrt unterbrechen muss. In diesem Fall wird die Beanspruchung der Knoten und somit die Weiterfahrt abgebrochen. Nach und nach unterbrechen alle Fahrzeuge somit ihre Fahrt bis das gesamte System zum Stillstand kommt.

Ist die Störung behoben, beanspruchen alle Fahrzeuge die nächsten Knoten und die Fahrten werden entlang der ursprünglich berechneten Routen – wenn auch mit Verspätung – fortgesetzt.

6.4.3 System neustarten

Der Grundgedanke der dritten Handlungsalternative besteht darin, alle Fahrzeuge im Fall einer Störung anzuhalten und anschließend erneut zu routen.

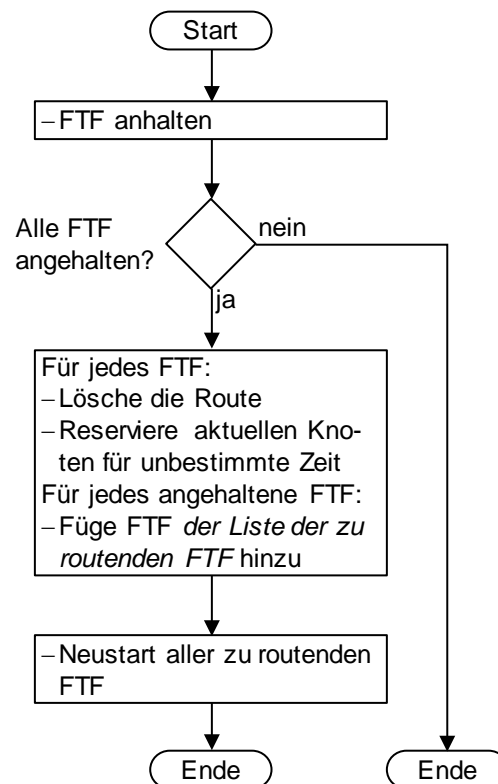


Abbildung 6-36: FTF anhalten und Systemneustart vorbereiten.

Analog zur zweiten Handlungsalternative wird beim Beanspruchen weiterer Knoten überprüft, ob eine Störung im System vorliegt, die Ausführung der Route abzubrechen und das Fahrzeug anzuhalten ist. Anschließend wird geprüft, ob bereits alle Fahrzeuge angehalten worden sind und – wie in Abbildung 6-36 dargestellt – mit der Vorbereitung des Neustarts fortgefahren werden kann.

Vor dem Beginn des Neustarts werden zunächst für alle Fahrzeuge, einschließlich des gestörten Fahrzeuges, die aktuellen Routen und damit auch die reservierten Zeitfenster gelöscht. Anschließend wird auf jedem Knoten, auf welchem sich ein Fahrzeug befindet, ein nach hinten offenes Zeitfenster reserviert. Somit kann dieser Knoten von anderen Fahrzeugen nicht genutzt werden. Abschließend wird jedes angehaltene Fahrzeug einer Liste, *der Liste der zu routenden Fahrzeuge*, hinzugefügt.

Nun kann der Neustart aller angehaltenen Fahrzeuge beginnen. Dieser Vorgang ist in Abbildung 6-37 beschrieben. Bei einem solchen Neustart können Deadlocks auftreten, wenn zwei Fahrzeuge nach hinten offene Zeitfensters reserviert haben, die verhindern, dass das jeweils andere Fahrzeug sein nächstes Ziel erreichen kann. Daher ist für diese Situation ein Vorgehen zur Auflösung des Deadlocks anzuwenden, welches in Abbildung 6-41 beschrieben ist.

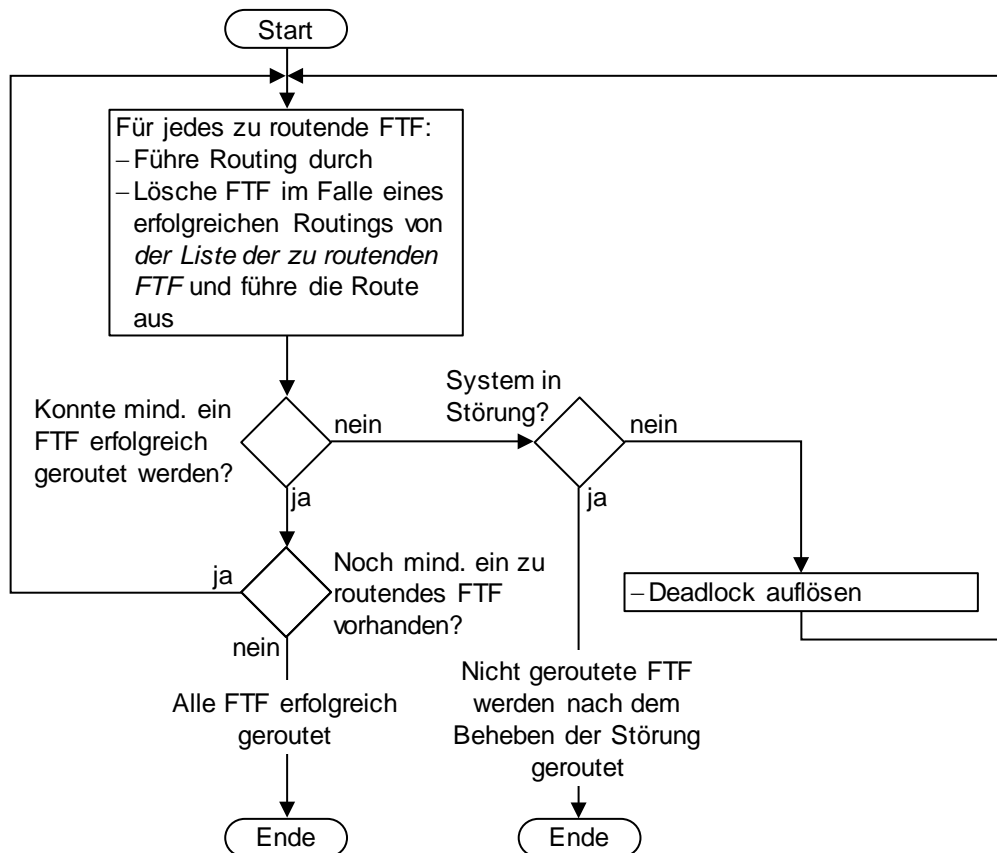


Abbildung 6-37: Neustart angehaltener Fahrzeuge.

Für den Neustart wird für jedes zu routende Fahrzeug das Routing erneut durchgeführt. Ist der Routingprozess erfolgreich, wird das Fahrzeug von *der Liste der zu routenden Fahrzeuge* entfernt und mit der Ausführung der Route fortgefahren. In diesem Fall wird auch das unbegrenzte Zeitfenster gelöscht und durch ein Zeitfenster endlicher Dauer ersetzt. Ist der Routingprozess hingegen nicht erfolgreich, weil beispielsweise ein Ziel von der unbegrenzten Reservierung eines noch nicht gerouteten Fahrzeuges blockiert wird, verbleibt das Fahrzeug in *der Liste der zu routenden Fahrzeuge* und wird in einer nachfolgenden Iteration erneut geroutet.

In dem Beispiel in Abbildung 6-38 symbolisieren die farbigen Kreise die Ziele der jeweiligen FTF sowie die eingefärbten Knoten die für unbestimmte Zeit reservierten Knoten. Das grüne FTF ist gestört; sowohl das blaue als auch das rote FTF sind in *der Liste der zu routenden Fahrzeuge* gespeichert. Wird zunächst das blaue FTF geroutet, ist der Routingprozess nicht erfolgreich, da der Weg durch das unbegrenzte Zeitfenster des roten FTF versperrt ist. Erfolgt nun das Routing für das rote FTF, ist dieses erfolgreich und das unbegrenzte Zeitfenster wird durch eines mit endlicher Dauer ersetzt. Ein erneutes Routing des blauen FTF in einer weiteren Iteration führt nun ebenso zum Erfolg.

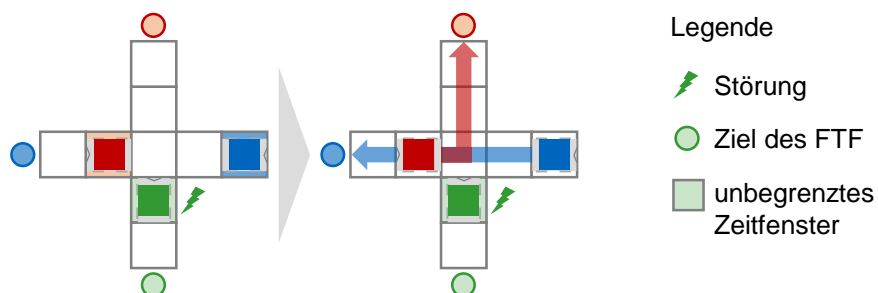


Abbildung 6-38: Das rote und das blaue FTF müssen neu geroutet werden.

Wurden alle FTF erfolgreich geroutet, ist der Neustart beendet. Es kann jedoch der Fall auftreten, dass in einer gesamten Iteration kein weiteres Fahrzeug geroutet werden kann. Dies geschieht dann, wenn das Erreichen eines Ziels für mindestens ein Fahrzeug durch das gestörte Fahrzeug unmöglich ist. Dieses Fahrzeug kann wiederum weitere Fahrzeuge blockieren. In dem Beispiel in Abbildung 6-39 blockiert das gestörte, grüne FTF das rote FTF, welches wiederum das blaue FTF blockiert. Alle blockierten Fahrzeuge können in solch einem Fall erst mit dem Beheben der Störung ihre Fahrt fortsetzen.

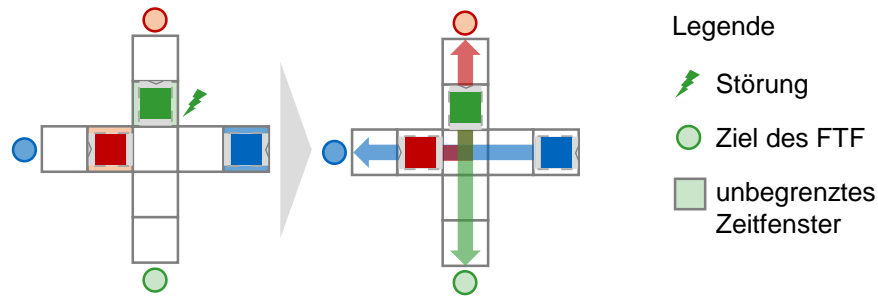


Abbildung 6-39: Ein gestörtes FTF (grün) verhindert das Routing eines angehaltenen FTF (rot), wodurch ein weiteres FTF (blau) blockiert wird.

Ist die Störung behoben, erfolgt gemäß dem Vorgehen in Abbildung 6-40 das Routing für das entstörte Fahrzeug. Verläuft dieses erfolgreich, wird die Ausführung der Route gestartet und noch vorhandene, zu routende Fahrzeuge werden neu gestartet. Bei dem zuvor aufgeführten Beispiel wird das unbegrenzte Zeitfenster des entstörten, grünen FTF gelöscht und das Routing des roten sowie des blauen FTF verlaufen erfolgreich.

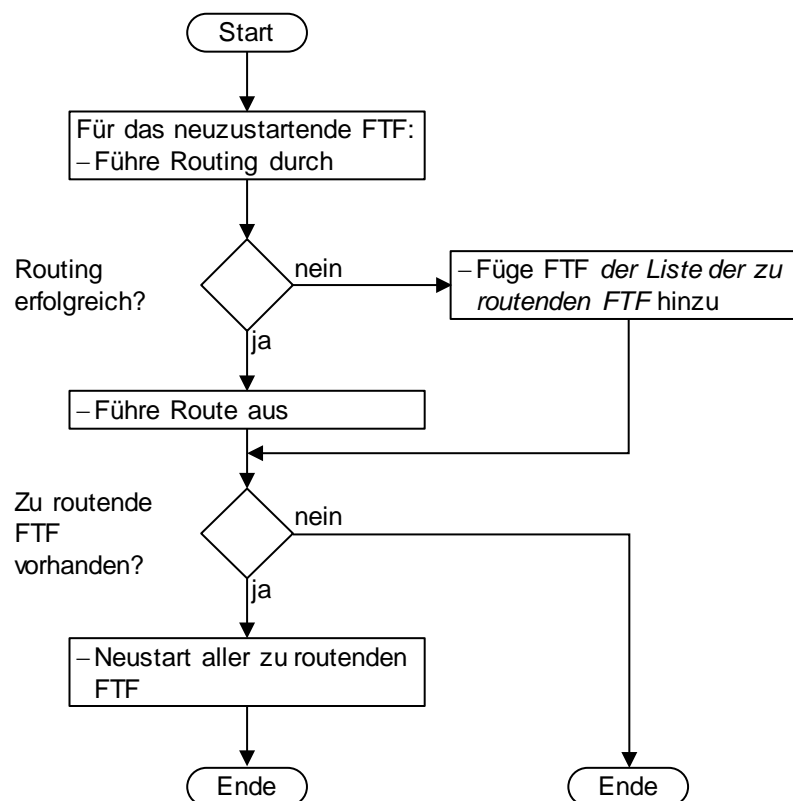


Abbildung 6-40: Entstörtes FTF neustarten.

Verläuft das Routing für das entstörte Fahrzeug hingegen nicht erfolgreich, wird das Fahrzeug nun selber in *der Liste der zu routenden Fahrzeuge* gespeichert und der Neustart aller zu routenden Fahrzeuge wird durchgeführt. Konnte in einer Iteration kein

weiteres Fahrzeug erfolgreich geroutet werden und ist im gesamten System kein gestörtes Fahrzeug vorhanden, welches die Fahrzeuge blockiert, liegt ein Deadlock vor. Dieser ist gemäß dem Vorgehen in Abbildung 6-41 aufzulösen.

Der Deadlock wird dadurch aufgelöst, dass einem der beteiligten Fahrzeuge ein alternatives Zwischenziel zugewiesen wird, zu welchem eine Route berechnet werden kann. Das Fahrzeug löscht das nach hinten offene Zeitfenster und der Knoten kann von den anderen Fahrzeugen wieder genutzt werden.

Für die Auflösung eines Deadlocks wird *die Liste der zu routenden Fahrzeuge* der Reihe nach durchgegangen. In jeder Iteration werden mögliche Zwischenziele in einer Liste gesammelt. Als ein Zwischenziel eignen sich insbesondere Knoten, welche im Normalbetrieb nicht oder nur selten von anderen Fahrzeugen angefahren werden, wie beispielsweise Pufferplätze für Fahrzeuge ohne zugewiesenen Auftrag.

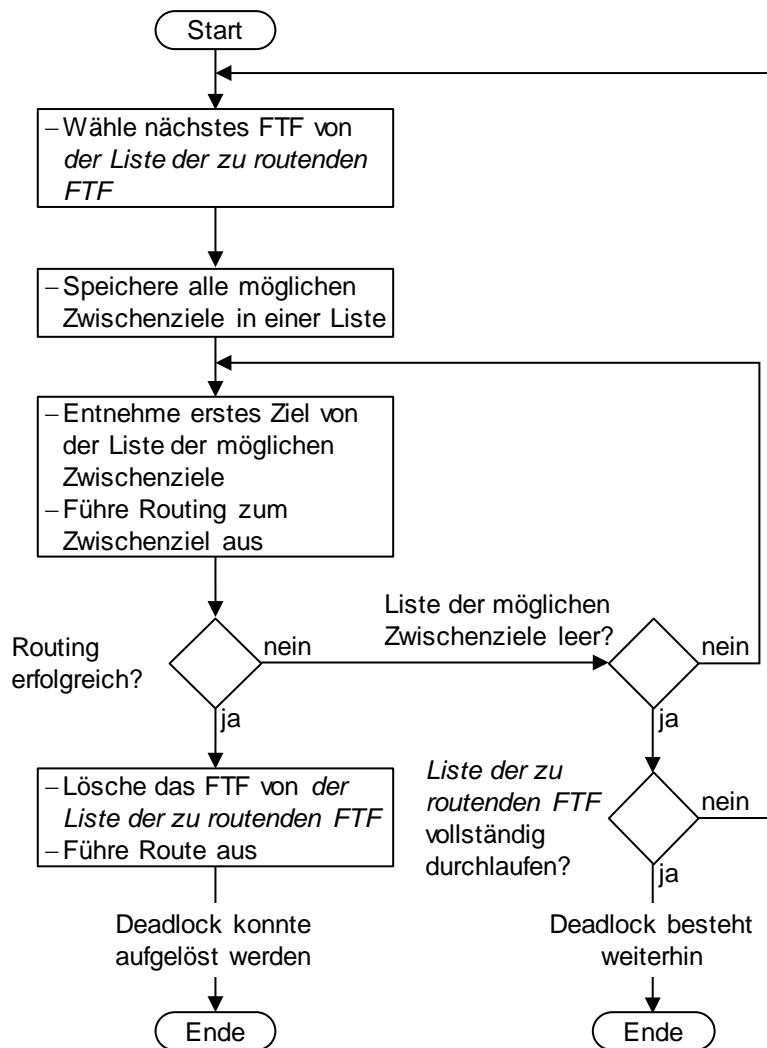


Abbildung 6-41: Auflösung eines Deadlocks.

Nun wird versucht, das blockierte Fahrzeug zum ersten Zwischenziel auf der Liste mit den gesammelten Zwischenzielen zu routen. Dabei wird auf dem Zielknoten eine unendlich lange Verweildauer eingeplant, um einen erfolgreichen Anschlussroutingprozess zu garantieren. Ist der Routingprozess erfolgreich, wird das Fahrzeug von der *Liste der zu routenden Fahrzeuge* gelöscht und die Auflösung des Deadlocks ist beendet. Eine neue Iteration im Neustart aller zu routenden Fahrzeuge beginnt. Ist das Zwischenziel hingegen nicht erreichbar, wird mit dem nächsten Zwischenziel fortgefahren.

Ist kein einziges Zwischenziel erreichbar, wiederholt sich das Vorgehen mit dem nächsten blockierten Fahrzeug. Konnte kein einziges blockiertes Fahrzeug ein mögliches Zwischenziel erreichen, konnte der Deadlock nicht aufgelöst werden. In diesem Fall sind die möglichen Kandidaten für die Zwischenziele zu restriktiv gewählt. Abbildung 6-42 verdeutlicht das beschriebene Vorgehen.

Nach dem Beheben der Störung kann das grüne FTF nicht erfolgreich geroutet werden (a). Nun wird das grüne FTF der *Liste der zu routenden Fahrzeuge* hinzugefügt. Da beim Neustart kein einziges blockiertes FTF geroutet werden kann und keine weitere Störung vorliegt, startet die Prozedur zum Auflösen des Deadlocks. Dafür wird dem grünen FTF ein Zwischenziel zugewiesen (b), das anschließende Routing verläuft erfolgreich, was wiederum dem roten FTF ermöglicht, sein Ziel zu erreichen (c). Sobald das FTF das Zwischenziel erreicht, wird zum ursprünglichen Ziel geroutet (d).

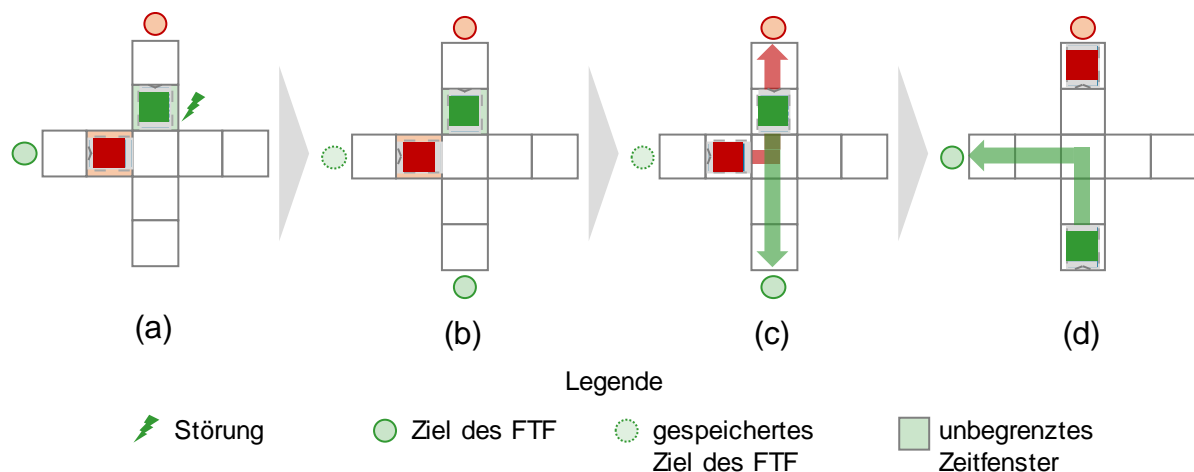


Abbildung 6-42: Durch die Störung entsteht ein Deadlock, welcher aufgelöst wird.

6.4.4 Betroffene Fahrzeuge umleiten

In der vierten Handlungsalternative wird davon Abstand genommen, alle Fahrzeuge anzuhalten. Vielmehr wird versucht, lediglich unmittelbar durch die Störung betroffene

Fahrzeuge dynamisch umzuleiten. Tritt bei einem Fahrzeug eine Störung auf, wird die Fahrt abgebrochen und wie in Abbildung 6-43 dargestellt vorgegangen.

Zunächst sind alle Fahrzeuge zu identifizieren, welche durch die Störung unmittelbar betroffen sind. Das sind jene Fahrzeuge, welche über den Knoten geroutet worden sind, auf welchem sich das gestörte Fahrzeug befindet. Die betroffenen Fahrzeuge werden über die reservierten Zeitfenster auf diesem Knoten bestimmt und in einer Liste, *der Liste der umzuleitenden Fahrzeuge*, gespeichert. Anschließend wird die Route des gestörten Fahrzeuges gelöscht. Dabei ist zu beachten, dass beim Löschen der Zeitfenster mögliche wartende Fahrzeuge zu benachrichtigen sind. Abschließend werden sämtliche Zeitfenster auf dem aktuellen Knoten gelöscht und ein unbeschränktes Zeitfenster auf dem Knoten reserviert.

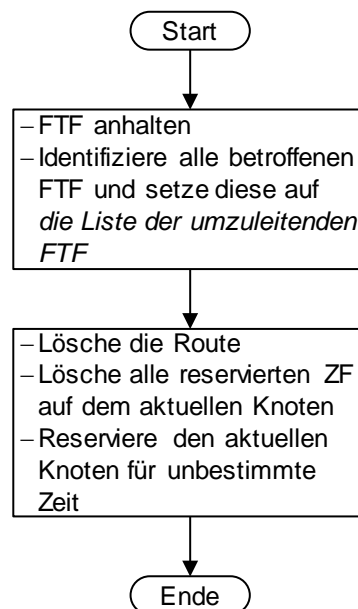


Abbildung 6-43: Fahrt abbrechen.

Sobald ein Fahrzeug ein Segment seiner Route beendet hat und die nächsten Knoten für die Weiterfahrt beansprucht, wird überprüft, ob das Fahrzeug von der Störung betroffen ist und sich somit auf *der Liste der umzuleitenden Fahrzeuge* befindet. Ist ein Fahrzeug umzuleiten, wird wie in Abbildung 6-44 beschrieben vorgegangen.

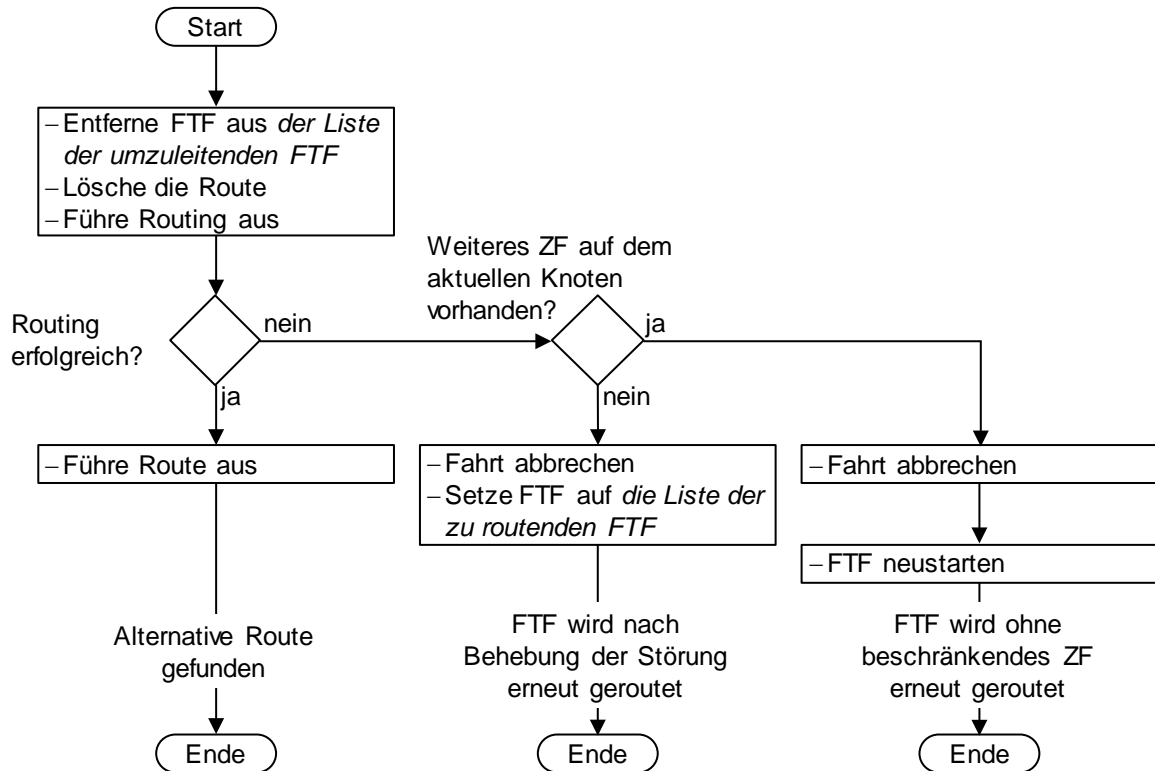


Abbildung 6-44: FTF umleiten.

Zunächst ist das Fahrzeug aus *der Liste der umzuleitenden Fahrzeuge* zu entfernen und die ursprünglich geplante Route zu löschen. Anschließend wird das Routing erneut ausgeführt. Eine neue Route kann nicht über den durch das gestörte Fahrzeug blockierten Knoten führen, da dieser für unbestimmte Zeit reserviert ist. Wird eine alternative Route gefunden, wird mit der Ausführung dieser Route fortgefahren und das Umleiten ist beendet. Ist das Routing hingegen nicht erfolgreich, kann dies zwei Ursachen haben. Zum einen kann der aktuelle Knoten nicht mehr rechtzeitig verlassen werden, weil ein weiteres Zeitfenster auf dem Knoten reserviert ist. In diesem Fall wird die Fahrt des umzuleitenden Fahrzeuges gemäß dem Vorgehen aus Abbildung 6-43 abgebrochen und damit auch das beschränkende Zeitfenster gelöscht. Anschließend wird das Fahrzeug – wie in Abbildung 6-40 beschrieben – neu gestartet. Zum anderen ist es möglich, dass der Weg zum Ziel durch ein unbeschränktes Zeitfenster blockiert ist. In diesem Fall ist das Fahrzeug *der Liste der zu routenden Fahrzeuge* hinzuzufügen und die Fahrt ebenfalls abzubrechen. Ein erneutes Routing erfolgt erst nach dem Beheben der Störung und dem Neustart des entstörten Fahrzeuges (Abbildung 6-40).

Ein Vergleich der vier beschriebenen Strategien erfolgt im Rahmen der Evaluation in Kapitel 8.

7 Umsetzung im Simulationsmodell

In diesem Kapitel wird die Umsetzung der Modellierung sowie der Steuerungsstrategien in einer Simulationsumgebung beschrieben. Dafür wird der Aufbau des Simulationsmodells in sechs Hauptkomponenten gegliedert, welche im Unterkapitel 7.1 vorgestellt werden. Das Unterkapitel 7.2 beschreibt eine konkrete Implementierung in der Simulationsumgebung *Tecnomatix Plant Simulation*.

7.1 Module des Simulationsmodells

Abbildung 7-1 gibt den strukturellen Aufbau des Modells wieder und visualisiert den Informationsaustausch zwischen den einzelnen Komponenten.

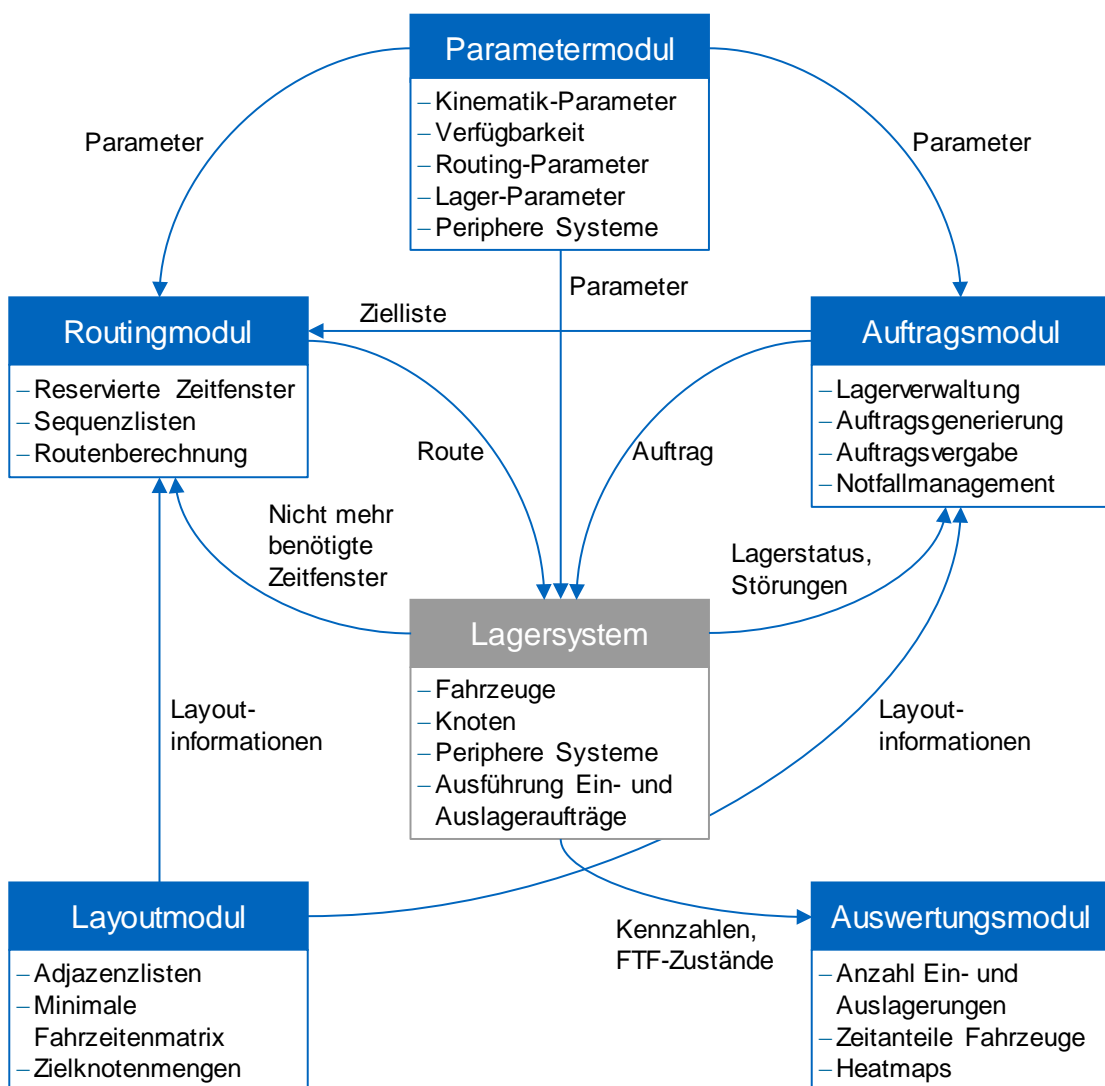


Abbildung 7-1: Schematischer Aufbau des Simulationsmodells.

Das Simulationsmodell umfasst neben dem betrachteten Lagersystem (Abschnitt 7.1.6) fünf Module, welche zur Datenhaltung (Layoutmodul, Abschnitt 7.1.1), Parametrisierung (Parametermodul, Abschnitt 7.1.2), Ausführung der Steuerungsstrategien (Auftragsmodul und Routingmodul, Abschnitte 7.1.3 und 7.1.4) sowie Aufbereitung der Simulationsergebnisse (Auswertungsmodul 7.1.5) herangezogen und im Folgenden beschrieben werden.

7.1.1 Layoutmodul

Das Layoutmodul enthält sämtliche Informationen über das Layout des Systems sowie die Methoden zur Generierung dieser Informationen.

Die beiden in Unterkapitel 5.1 beschriebenen Adjazenzlisten werden im Layoutmodul generiert und gespeichert. Mithilfe der Adjazenzliste, welche die Ausrichtungen der Knoten berücksichtigt, wird durch Anwendung des Dijkstra-Algorithmus [Dij-1959] eine minimale Fahrzeitenmatrix zwischen allen Knoten erstellt, welche für das Routing herangezogen wird, um die verbleibende Gesamtfahrzeit bis zur Ankunft am letzten Ziel zu schätzen. Darüber hinaus können die Layoutinformationen für die Vergabe von Aufträgen an die Fahrzeuge herangezogen werden.

Entsprechen die Ziele nicht einzelnen Knoten im zugrundeliegenden Graphen und werden für die Zugriffe auf die Lagerplätze mehrere Knoten gleichzeitig belegt, sind im Layoutmodul die Zielknotenmengen aller Ziele hinterlegt.

7.1.2 Parametermodul

Das Parametermodul dient der Parametrierung des Simulationsmodells. Die Parameter lassen sich in folgende Kategorien einteilen.

Parameter der Fahrzeugkinematik:

- maximale Geschwindigkeit eines Fahrzeuges, wenn sich das Fahrzeug in Ausrichtung X bewegt (v^X)
- maximale Geschwindigkeit eines Fahrzeuges, wenn sich das Fahrzeug in Ausrichtung Y bewegt (v^Y)
- Beschleunigung und Verzögerung eines Fahrzeuges, wenn sich das Fahrzeug in Ausrichtung X bewegt (a^X)
- Beschleunigung und Verzögerung eines Fahrzeuges, wenn sich das Fahrzeug in Ausrichtung Y bewegt (a^Y)
- Ausrichtungswechselzeit, die ein Fahrzeug benötigt, um seine Ausrichtung $X \rightarrow Y$, bzw. $Y \rightarrow X$ zu wechseln (t_{RW})

- Spielzeit eines Lastwechsels ($t_{Lagerspiel}$)

Parameter zur technischen Verfügbarkeit und dem Umgang mit Störungen:

- Verfügbarkeit eines Fahrzeuges
- MTTR eines Fahrzeuges
- Strategie für den Umgang mit Störungen (*ignorieren, pausieren, neustarten, umleiten*)

Parameter, die den Routingprozess betreffen:

- zeitlicher Puffer zwischen zwei Reservierungen (t_p)
- Reservierungsform (*Treppe, Dreieck, Block*)
- maximale Segmentlänge

In dem Parametermodul sind auch die Anzahl der Fahrzeuge festzulegen, welche zu Beginn eines Simulationslaufes im Lager erzeugt werden. Darüber hinaus können weitere Größen zur individuellen Modellparametrisierung übergeben werden, wie die Kinematik peripherer Systeme (beispielsweise Lifte), Pickzeiten, initialer Lagerfüllgrad oder eine Zonierung des Lagers.

7.1.3 Auftragsmodul

Im Auftragsmodul ist zum einen die Lagerverwaltung abgebildet, zum anderen erfolgt in dem Modul die Zuordnung der Aufträge zu den Fahrzeugen. Bei diesen Aufträgen kann es sich um Ein-, Aus- oder auch Umlageraufträge sowie reine Fahraufträge (beispielsweise zu einem Pufferplatz) handeln. Wird einem Fahrzeug ein Auftrag zugewiesen, wird die Zielliste (siehe Abschnitt 6.1.2) erstellt und dem Routingmodul für die Berechnung der Route übergeben.

Das Auftragsmodul ist je nach Anwendungsfall individuell auszugestalten. Insbesondere betrifft dies die Auftragsvergabe, welcher unterschiedliche Strategien zu Grunde liegen können (siehe Abschnitt 3.1.2). Für die Umsetzung dieser Strategien werden die notwendigen Informationen bezüglich der Lagerbelegung, vorhandener Aufträge und verfügbarer Fahrzeuge in verschiedenen Tabellen abgelegt.

Die erste Tabelle enthält sämtliche Lagerplätze sowie deren aktuellen Inhalt. Wird ein Lagerplatz von einem Fahrzeug für eine Ein- oder Auslagerung angefahren, wird dieses Fahrzeug ebenfalls in der Tabelle vermerkt. Darüber hinaus lassen sich weitere Attribute erfassen, wie beispielsweise die Lagergasse oder –ebene sowie Zeitstempel

einer Ein- und Auslagerung. Durch das entsprechende Filtern der Tabelle lassen sich geeignete Lagerplätze für Ein- und Auslagerungen identifizieren. Beispielsweise können alle leeren Lagerplätze in einem bestimmten Lagerbereich identifiziert werden, welche noch nicht von einem Fahrzeug angefahren werden und daher für eine Einlagerung zur Verfügung stehen.

Analog zur Tabelle der Lagerplätze können im Auftragsmodul weitere Tabellen für Sonderknoten wie beispielsweise Puffer- oder Übergabepunkte angelegt werden, welche ebenfalls für die Erzeugung und Vergabe von Aufträgen herangezogen werden können.

Der Auftragspool enthält Ein- und Auslageraufträge, welche zur Ausführung freigegeben sind und an ein Fahrzeug vergeben werden können. Für einen Einlagerauftrag ist in der Tabelle der Aufträge der Übergabepunkt, an welchem die Lagereinheit von dem Fahrzeug aufgenommen wird sowie der Einlagerplatz, sofern dieser bereits bestimmt wurde, vermerkt. Für einen Auslagerauftrag enthält die Tabelle den Auslagerplatz sowie den Übergabepunkt, an welchem die Lagereinheit bereitgestellt werden muss. Darüber hinaus können abermals weitere Informationen wie die Zeitstempel der Freigabe der Aufträge gespeichert werden. Die Aufträge können unter Berücksichtigung der aktuellen Lagerbelegung zufällig erzeugt werden oder aber auf realen Auftragsdaten basieren.

Alle vorhandenen Fahrzeuge sind in einer Tabelle referenziert, in welcher zusätzliche Informationen, wie beispielsweise der aktuell zugewiesene Auftrag, der aktuelle Standort sowie aufgetretene Störungen vermerkt werden können.

In dem Auftragsmodul ist auch der Störungsmanager angesiedelt, welcher die gewählten Notfallstrategien umsetzt. Darüber hinaus ist in dem Auftragsmodul auch die Methode zur initialen Erzeugung der Fahrzeuge platziert, welche zum Beginn einer Simulation ausgeführt wird. Für die Initialisierung lassen sich zwei Vorgehensweisen unterscheiden. Zum einen können die Fahrzeuge nacheinander auf einem fiktiven Knoten erzeugt und anschließend in das Lager geroutet werden. Zum anderen können alle Fahrzeuge (zufällig) im Lager verteilt werden. Der Beginn des Lagerbetriebes erfolgt anschließend durch die Durchführung eines Systemneustarts gemäß dem Vorgehen aus Abschnitt 6.4.3.

7.1.4 Routingmodul

Das Routingmodul enthält die Logik zur Wegfindung und verwaltet die Sequenzlisten für die Anwendung der routingbasierten Sequenzierung sowie die reservierten Zeitfenster. Für jeden Knoten wird eine Tabelle geführt, in welcher die reservierten Zeitfenster sowie die reservierenden Fahrzeuge chronologisch gespeichert sind. Für das Routing werden die freien Zeitfenster, welche vor, zwischen und nach den Reservierungen vorhanden sind, aus dieser Tabelle herausgelesen.

Soll für ein Fahrzeug eine neue Route berechnet werden, werden die im Abschnitt 6.1.2 beschriebenen Parameter benötigt. Nach der Berechnung der Route wird diese an das ausführende Fahrzeug übergeben und die benötigten Zeitfenster in der entsprechenden Tabelle gespeichert. Verläuft das Routing hingegen nicht erfolgreich, so wird dieses an das Auftragsmodul zurückgemeldet.

7.1.5 Auswertungsmodul

Im Auswertungsmodul werden vorab definierte Kennzahlen gesammelt und aufbereitet. Neben dem erzielten Durchsatz wird für jedes eingesetzte Fahrzeug aufgeschlüsselt, wie lange es sich in welchem Zustand befindet. Mögliche Fahrzeugzustände sind nachfolgend aufgeführt.

- Fahrend: Das Fahrzeug bewegt sich fort oder führt einen Wechsel der Ausrichtung aus.
- Lastwechselnd: Das Fahrzeug vollzieht einen Lastwechsel, eine Transporteinheit wird aufgenommen oder abgegeben.
- Wartend: Das Fahrzeug kann seine Fahrt nicht fortsetzen, da entweder das reservierte Zeitfenster auf dem nächsten Knoten noch nicht begonnen hat oder aber das vorhergehende Fahrzeug sich verspätet und das reservierte Zeitfenster noch nicht gelöscht hat.
- Gestört: Das Fahrzeug ist gestört und muss seine Fahrt unterbrechen.
- Angehalten: Das Fahrzeug ist aufgrund einer Störung im System angehalten und muss seine Fahrt unterbrechen.

Darüber hinaus können je nach simuliertem System weitere Zustände definiert werden, wie beispielsweise die Fahrt in einem Lift oder das Warten auf einem Pickplatz.

Während der Durchführung der Simulation wird mitgeschrieben, wie oft ein Fahrzeug einen Knoten befährt und wie lange es auf einem Knoten wartet. Diese Informationen

können nach Ablauf der Simulation für die Erzeugung von Heatmaps herangezogen werden.

7.1.6 Lagersystem

Das Lagersystem stellt den Kern der Simulation dar. Hier wird der Betrieb des Lagersystems abgebildet und animiert. Es enthält das Wegenetz, auf welchem die Fahrzeuge verfahren, um Ein- und Auslageraufträge durchzuführen. Dafür beinhaltet es die Methoden zur Ausführung der berechneten Routen gemäß dem Vorgehen aus dem Unterkapitel 6.3.

Die berechneten Routen werden den Fahrzeugen vor dem Beginn einer Fahrt vom Routingmodul übergeben. Nach der Durchführung einer Ein- oder Auslagerung ist die Tabelle der Lagerplätze im Auftragsmodul entsprechend zu aktualisieren. Ist ein Auftrag abgeschlossen, meldet sich ein Fahrzeug beim Auftragsmodul und bekommt einen Folgeauftrag und somit auch eine neue Route zugewiesen.

7.2 Implementierung

In diesem Unterkapitel wird die Umsetzung der Methodik in der Simulationsumgebung *Tecnomatix Plant Simulation* beschrieben. Ziel bei dieser Umsetzung ist es, einen Bausteinkasten zu erstellen, welcher für jeden Anwendungsfall als Vorlage herangezogen werden kann, um eine aufwandsarme Modellerstellung zu ermöglichen.

Zunächst wird die verwendete Simulationsumgebung vorgestellt (Abschnitt 7.2.1), bevor die Implementierungen der Module (Abschnitt 7.2.2), der Knoten (Abschnitt 7.2.3) und Fahrzeuge (Abschnitt 7.2.4) sowie des Lagerbereichs (Abschnitt 7.2.5) beschrieben werden. Abschließend wird auf die automatisierte Modellerstellung (Abschnitt 7.2.6) sowie auf die Visualisierung der Steuerungsalgorithmen (Abschnitt 7.2.7) eingegangen.

7.2.1 Verwendete Simulationsumgebung

Plant Simulation ist eine Software für die ereignisdiskrete Simulation, Visualisierung, Analyse und Optimierung von Produktions- und Logistiksystemen. Die Simulationsumgebung unterstützt einen objektorientierten Aufbau der Modelle und ermöglicht das Umsetzen des Programmierkonzeptes der Vererbung.

Für die Modellerstellung stehen eine Reihe vorgefertigter Bausteine zur Verfügung, mit Hilfe derer der Material- und Informationsfluss abgebildet werden kann. Jeder Baustein

stellt eine eigene Klasse dar. Instanzen dieser Klassen werden im Modell per Drag and Drop erzeugt, indem diese in sogenannte Netzwerke eingesetzt werden, welche die Behälter für ein Simulationsmodell darstellen. Netzwerke erlauben eine hierarchische Strukturierung des Modells. So kann eine gesamte Fabrik in einem Netzwerk abgebildet werden, welches wiederum Netzwerke enthält, die beispielsweise die verschiedenen Produktionsbereiche darstellen.

Die Software verfügt über eine eigene Programmiersprache. Diese ermöglicht es, das Verhalten der eingesetzten Bausteine zu verändern. Darüber hinaus lassen sich Methoden schreiben, um Kontrollstrukturen umzusetzen und den Programmablauf zu steuern.

Die bereitgestellte Klassenbibliothek vorgefertigter Bausteine kann beliebig erweitert werden. Neue Bausteine, insbesondere neue Netzwerke, welche sich aus mehreren vorgefertigten Bausteinen und selbst geschriebenen Methoden zusammensetzen, können als Klasse angelegt und in verschiedenen Modellen instanziiert werden. Dabei werden angelegte Elemente (Variablen, Methoden, Objekte und Netzwerke) vererbt, können in der Instanz jedoch überschrieben und verändert werden. Somit ist eine hohe Wiederverwendung erstellter Modellbestandteile möglich.

Plant Simulation stellt Werkzeuge für die Durchführung und Auswertung von Simulationsstudien zur Verfügung. Der Ereignisverwalter koordiniert die Ereignisse während des Simulationslaufes. Der Experimentverwalter unterstützt die automatisierte Durchführung von Parameterstudien.

7.2.2 Implementierung der Module

Die fünf im Unterkapitel 7.1 beschriebenen Module werden als eigenständige Netzwerke implementiert und in ein übergeordnetes Hauptnetzwerk eingesetzt. Für jeden Anwendungsfall kann dieser Aufbau als Vorlage herangezogen und angepasst werden. Eine Vielzahl der implementierten Methoden und Objekte kann für jeden Anwendungsfall unverändert übernommen werden. Einige Methoden sind zwar für jeden Anwendungsfall notwendig, unterscheiden sich jedoch derart, dass eine allgemeingültige Implementierung nicht möglich ist. Diese Methoden sind in den Netzwerken bereits angelegt, müssen aber ausprogrammiert werden. Darüber hinaus gibt es bei jedem Anwendungsfall Elemente, welche nur für eben diesen benötigt werden. Diese Elemente müssen somit für jeden Anwendungsfall individuell implementiert werden.

Für die gesamte Implementierung wird ein Farbschema gewählt, welches der Strukturierung dient und die Vererbung einzelner Modellkomponenten visualisiert:

- Orange: Dieser Bereich beinhaltet allgemeine Werkzeuge und Objekte, welche für die Datenhaltung genutzt werden.
- Grün: Im grün hinterlegten Bereich sind Elemente abgelegt, welche in jedem Anwendungsfall unverändert eingesetzt werden.
- Gelb: Im gelb hinterlegten Bereich hingegen sind Methoden abgelegt, welche in jedem Anwendungsfall zum Einsatz kommen, deren Inhalt jedoch überschrieben werden muss.
- Blau: Im blau hinterlegten Bereich sind individuelle Elemente abgelegt, welche für die Abbildung und Steuerung des betrachteten Systems notwendig sind.

Abbildung 7-2 zeigt das übergeordnete Netzwerk. Dieses enthält neben dem Ereignisverwalter und Experimentverwalter zwei Methoden, welche jeweils zu Beginn oder nach dem Ende eines Simulationslaufes automatisiert ausgeführt werden.



Abbildung 7-2: Simulationsmodell mit den fünf Modulen und dem Lagerbereich, welcher aus einer Lagerebene besteht.

In diesen Methoden ist der Lagerbetrieb zu initialisieren sowie die Auswertung eines Simulationslaufes anzustoßen. Der Lagerbereich wird in Form von Netzwerken eingesetzt. Dabei wird jeder eigenständige Lagerbereich (beispielsweise jede Lagerebene) durch ein eigenes Netzwerk repräsentiert.

Abbildung 7-3 zeigt die Implementierung des Layoutmoduls. Es enthält unter anderem die beiden Typen der Adjazenzlisten sowie die minimale Fahrzeitenmatrix. Wird ein neues Layout erzeugt, ist die Methode *Layout_initialisieren* auszuführen, welche die notwendigen Informationen generiert und in den Tabellen ablegt. Modellspezifische

Layoutinformationen wie beispielsweise die Anzahl und Anordnung von Pickplätzen müssen individuell implementiert werden.

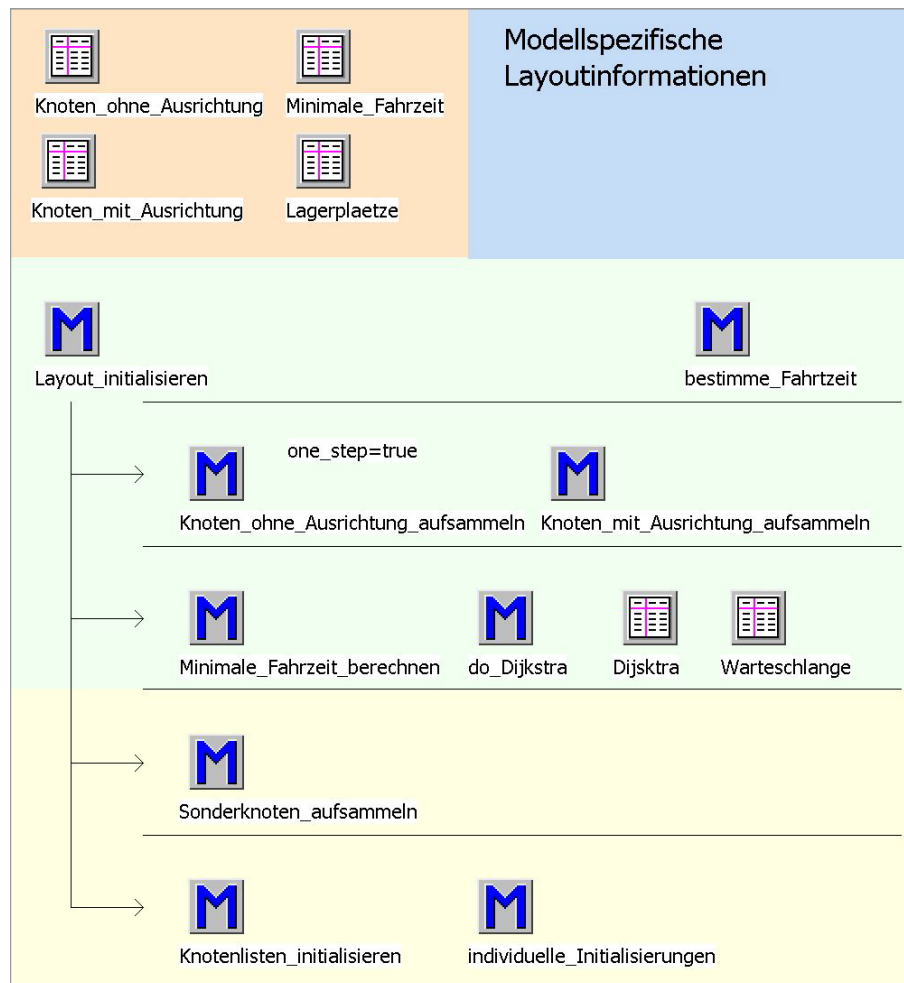


Abbildung 7-3: Implementierung des Layoutmoduls.

Abbildung 7-4 zeigt das implementierte Parametermodul. Es enthält die im Abschnitt 7.1.2 aufgeführten Parameter sowie die Methoden, um diese Parameter zu setzen. Anzumerken ist, dass die Dimensionen eines Fahrzeuges sowie die Art und Weise, wie ein Fahrzeug die Ausrichtung wechselt (siehe Abschnitt 5.2.1), grundsätzlich nicht vor der Durchführung einer Simulation verändert werden können. Die Veränderung dieser Parameter kann zu einer Verletzung der Voraussetzungen bei der Modellierung (Unterkapitel 5.1) führen. Eine erneute Bestimmung der Knotengrößen führt somit zu einer erneuten Modellerstellung (siehe Abschnitt 7.2.6).

Im Parametermodul ist ein Netzwerk enthalten, in welchem für jedes eingesetzte Fahrzeug eine Einzelstation – ein von *Plant Simulation* vorgefertigter Baustein – angelegt ist. Diese lassen sich mit einer Verfügbarkeit und MTTR parametrisieren und werden für die Erzeugung der Störungen bei den Fahrzeugen herangezogen.




Dimensionen	Bewegungsparameter	FTF	Routingparameter	Lagerparameter
FTF_Laenge_Y=0.8m FTF_Laenge_X=0.8m	FTF_Geschwindigkeit_X=1m/s FTF_Geschwindigkeit_Y=1m/s FTF_Beschleunigung_X=1m/s FTF_Beschleunigung_Y=1m/s	Anzahl_FTF=12 Verfuegbarkeit=99 MTTR=1:00.0000	max_Segmentlaenge_X=100 max_Segmentlaenge_Y=100 Mindestzeitabstand=0.1 Routingbasierte_Sequenzierung=false	Lagerfuellgrad=0.8
Typ	FTF_Richtungswechselzeit=2s FTF_Lastwechselzeit=5s	Störungsstrategie 3 - Neustarten	Reservierungsform Treppe	
Modellspezifische Parameter				
 allgemeine_Parameter_uebernehmen		 Stoerstationen	 spezifische_Parameter_uebernehmen	

Abbildung 7-4: Implementierung des Parametermoduls.

Bei der Implementierung des Routingmoduls (Abbildung 7-5) ist weder das Überschreiben von Methoden, noch das Hinzufügen individueller Elemente notwendig und daher nicht vorgesehen. Sollen jedoch während des Routingprozesses bestimmte Knotenarten ausgeschlossen werden – beispielsweise, wenn ein Pickplatz nur dann von einem Fahrzeug befahren werden darf, wenn dieser auch das nächste Ziel des Fahrzeuges ist, dann muss dieses sowohl im übergeordneten Routing, als auch in der Erstellung der Segmente vermerkt werden.

Je nach der im Parametermodul gewählten Reservierungsform werden die Erreichbarkeitsbedingungen für die Formen *Treppe*, *Dreieck* oder *Block* geprüft. Darüber hinaus bietet das Modul Möglichkeiten für die Visualisierung der Routen, welche insbesondere für Verifikationszwecke herangezogen werden können (siehe Abschnitt 7.2.7).

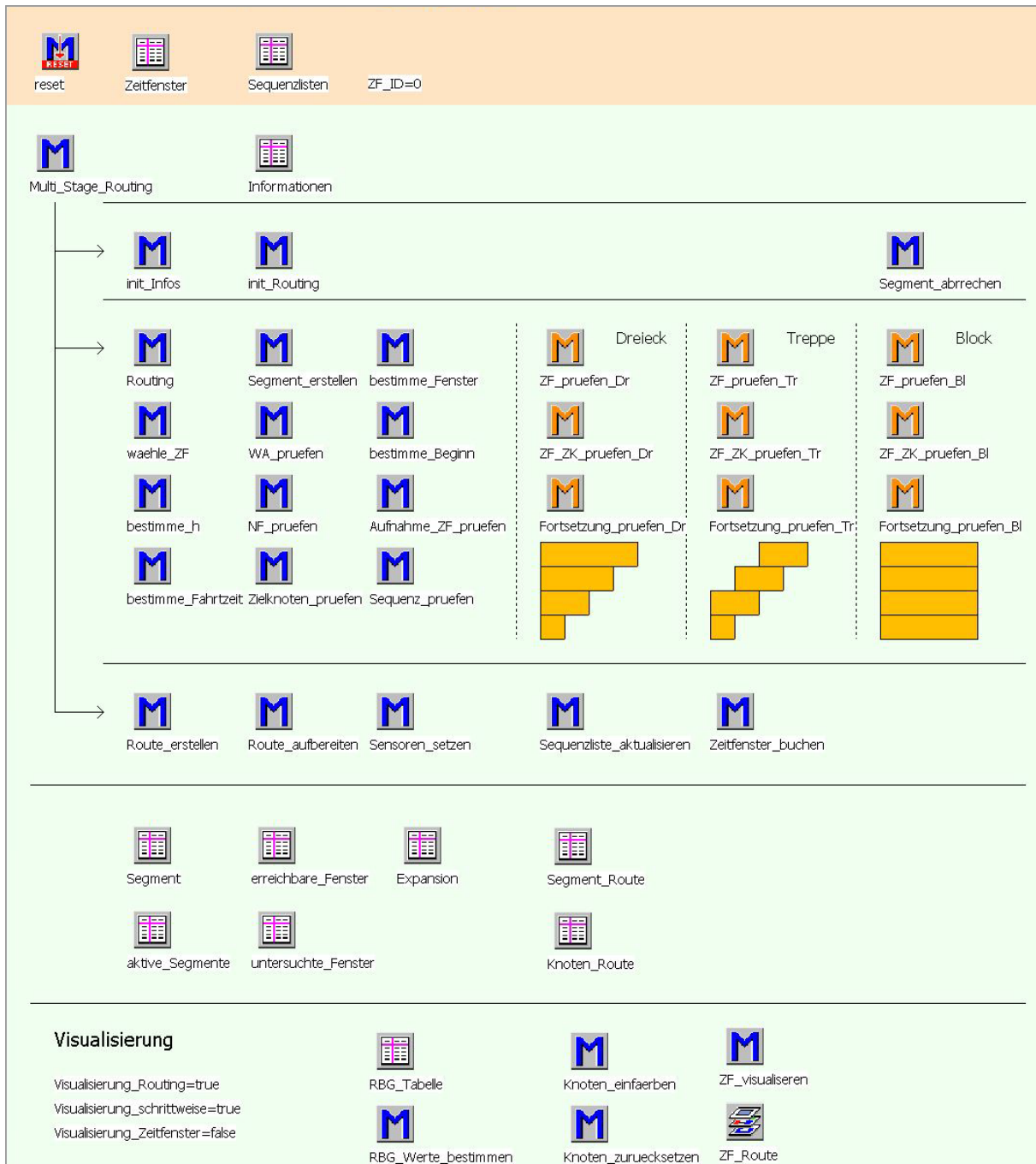


Abbildung 7-5: Implementierung des Routingmoduls.

Abbildung 7-6 zeigt die Implementierung des Auftragmoduls. Angelegt sind die Tabellen für die Lagerplätze und Pufferplätze, welche bei der Initialisierung des Layouts befüllt werden. Darüber hinaus ist die Tabelle für die Aufträge sowie die Tabelle der Fahrzeuge enthalten. Wie im Abschnitt 7.1.3 beschrieben, werden Aufträge durch geeignetes Filtern dieser Tabellen erzeugt und vergeben. Dafür werden die temporär gefüllten Tabellen herangezogen, welche nach dem Filtern alle Kandidaten enthalten, die die entsprechenden Kriterien erfüllen.

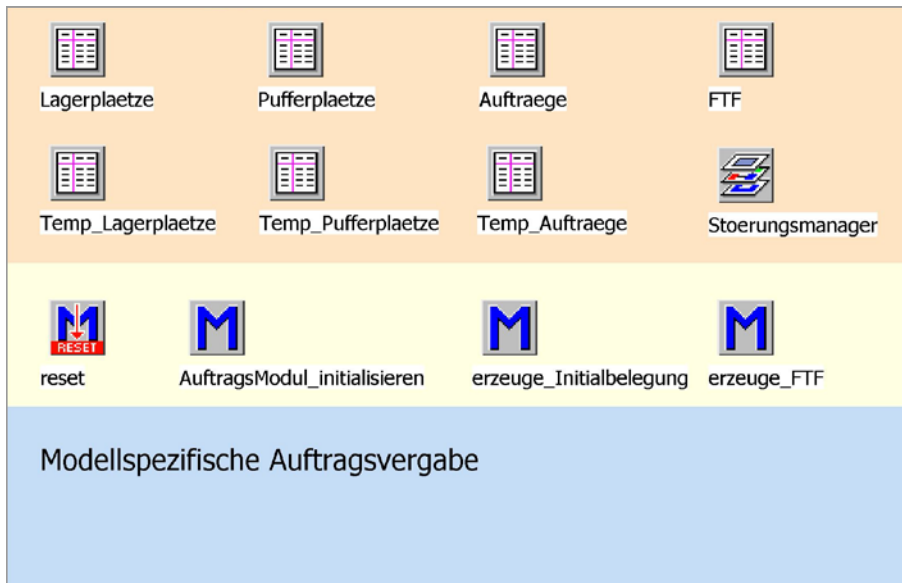


Abbildung 7-6: Implementierung des Auftragmoduls.

Die Implementierung des Auswertungsmoduls ist in Abbildung 7-7 dargestellt. Nach Abschluss eines Simulationslaufes wird die Simulation ausgewertet. Dabei werden alle erhobenen Kennzahlen in ein Protokoll geschrieben. Bei der Durchführung eines Experimentes erhält das Protokoll somit sämtliche Simulationsergebnisse. Darüber hinaus können im Auswertungsmodul die Heatmaps für die Anzahl der Knotenbelegung und Wartezeiten generiert werden.

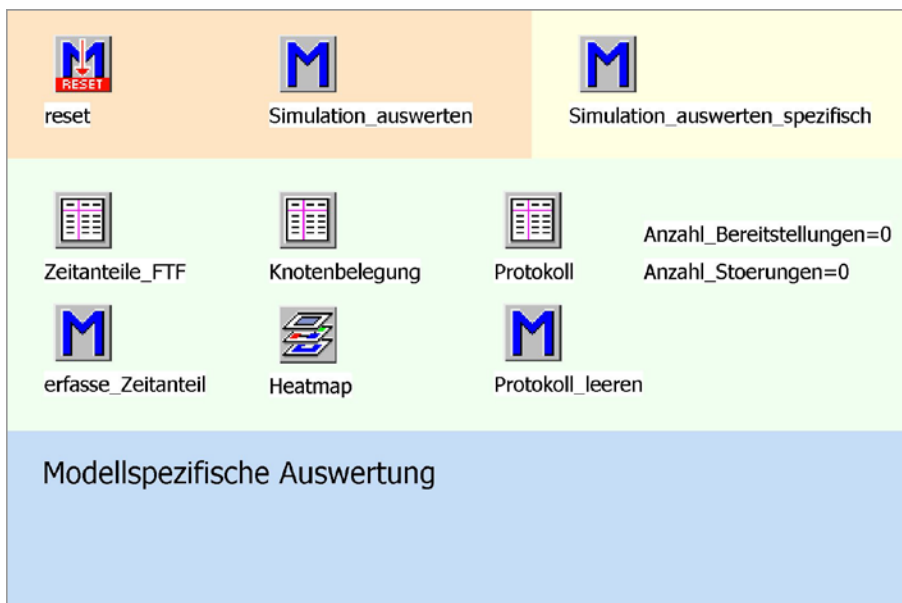


Abbildung 7-7: Implementierung des Auswertungsmoduls.

7.2.3 Implementierung der Knoten

Ein Knoten des Layoutgraphen ist als ein eigenständiges Netzwerk implementiert, welches beliebig oft in den Lagerbereich eingesetzt werden kann, um das Wegenetz zu erzeugen. Das Netzwerk, dargestellt in Abbildung 7-8, enthält die im Unterkapitel 5.1 aufgeführten Attribute. Das Netzwerk beinhaltet zwei Objekte vom Typ Wegstück, ein von *Plant Simulation* bereitgestellter Baustein. Über diese Wegstücke bewegen sich die Fahrzeuge fort. Eines der Wegstücke wird für die Bewegung in X-Ausrichtung genutzt, das andere für die Bewegung in Y-Ausrichtung. Auf diesen Wegstücken sind Sensoren platziert, bei deren Erreichen unterschiedliche Methoden aufgerufen werden. Diese Methoden werden aufgerufen, sobald ein Fahrzeug

- die Ankunftsposition (mittig zentriert auf dem Knoten) erreicht,
- an die Knotengrenze stößt und auf den nachfolgenden Knoten entlang der Route umgesetzt werden muss,
- den Knoten vollständig verlassen hat und dieser freigegeben werden kann.

Von dieser Klasse können weitere Klassen abgeleitet werden, welche lediglich eines der beiden Wegstücke enthalten und somit Knoten abbilden, welche keinen Wechsel der Ausrichtung erlauben.

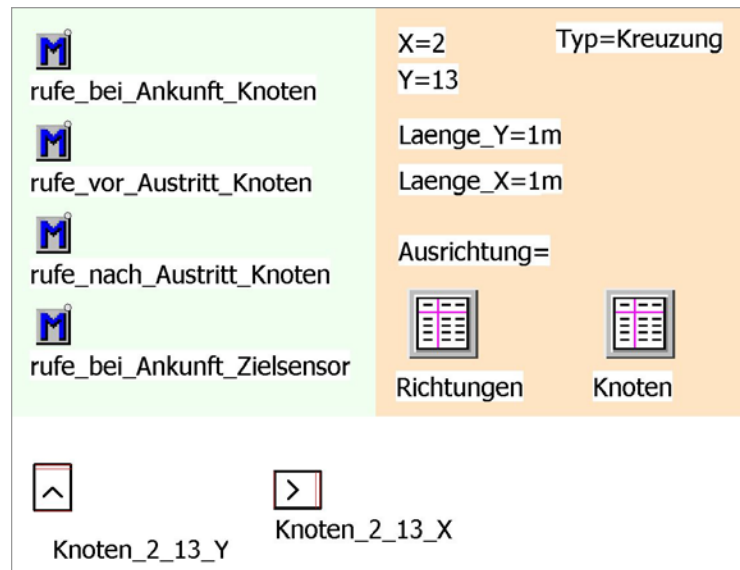


Abbildung 7-8: Implementierung eines Knotens.

7.2.4 Implementierung der Fahrzeuge

Für die Abbildung beweglicher Objekte stellt *Plant Simulation* unter anderem das Objekt Fahrzeug bereit. Dieses lässt sich mit Werten für die Beschleunigung, Verzögerung und Zielgeschwindigkeit parametrieren. Wird die Zielgeschwindigkeit gesetzt, beginnt das Fahrzeug gleichförmig zu beschleunigen, bis die Zielgeschwindigkeit erreicht

ist und die Fahrt mit konstanter Geschwindigkeit fortgesetzt wird. Analog dazu wird die Verzögerung abgebildet. Darüber hinaus lässt sich die aktuelle Geschwindigkeit eines Fahrzeuges herauslesen. Für jedes Fahrzeug kann angegeben werden, ob es sich vorwärts oder rückwärts auf einem Wegstück bewegen soll.

Zur Ausführung einer Route wird diese dem Fahrzeug übergeben. Eine solche Route zeigt Tabelle 7-1. Neben den zu befahrenden Knoten, den reservierten Zeitfenstern sowie den Abfahrtszeitpunkten der einzelnen Segmente ist spezifiziert, entlang welcher Ausrichtung sich das Fahrzeug bewegt, ob vor der Ausführung eines Segmentes ein Wechsel der Ausrichtung notwendig ist sowie in welche Fahrtrichtung ein Segment zurückzulegen ist. Mit Hilfe dieser Informationen wird die Fahrt in dem Simulationsmodell umgesetzt.

Tabelle 7-1: An ein Fahrzeug übergebene Route.

Knoten	Start	Ziel	resv_ZF_Start	Eintritt_Start	An	Abfahrt	Austritt_Ende	resv_ZF_Ende	Ausricht.	Wechsel d.A.	Fahrtrichtung	Distanz
Knoten_4_1	true		0,00		0,00	0,00	1,50	1,51	Y		rueck	4
Knoten_4_2			0,00	0,10			2,50	2,51	Y		rueck	
Knoten_4_3			0,00	1,51			3,50	3,51	Y		rueck	
Knoten_4_4			1,50	2,51			4,90	4,91	Y		rueck	
Knoten_4_5			2,50	3,51	5,00	7,00	8,50	8,51	Y	true	vor	7
Knoten_5_5			7,00	7,10			9,50	9,51	X		vor	
Knoten_6_5			7,00	8,51			10,50	10,51	X		vor	
Knoten_7_5			8,50	9,51			11,50	11,51	X		vor	
Knoten_8_5			9,50	10,51			12,50	12,51	X		vor	
Knoten_9_5			10,50	11,51			13,50	13,51	X		vor	
Knoten_10_5			11,50	12,51			14,90	14,91	X		vor	
Knoten_11_5		true	12,50	13,51	15,00			15,00	X			

Ausgehend vom Startknoten $K_{4,1}$ bewegt sich das Fahrzeug zunächst rückwärts über die Y -Wegstücke der Knotennetzwerke (in Abbildung 7-8: von oben nach unten). Nachdem das erste Segment beendet ist, muss das Fahrzeug auf dem Knoten $K_{4,5}$ einen Wechsel der Ausrichtung vollziehen. Anschließend bewegt sich das Fahrzeug vorwärts über die X -Wegstücke (in Abbildung 7-8: von links nach rechts), bis der Knoten $K_{11,5}$ und damit das Ziel erreicht ist.

In dem Bausteinkasten werden zwei Klassen für die Fahrzeuge angelegt: Eine für die Bewegung in Ausrichtung X , die andere für die Bewegung in Ausrichtung Y . Die Fahrzeuge werden über die Länge L_F^A des Fahrzeuges in Fahrtrichtung dimensioniert.

7.2.5 Implementierung des Lagerbereichs

Der Lagerbereich enthält die Instanzen der Knoten, über welche sich die Fahrzeuge bewegen, sowie die Methoden zur Ausführung von berechneten Routen (wie in Abschnitt 6.3.1 beschrieben) und Ein- und Auslageraufträgen. Diese sind in Abbildung

7-9 dargestellt. Lagerplätze sowie Lagereinheiten können zwecks Visualisierung abgebildet werden, sind für die Durchführung der Simulation aber nicht zwingend notwendig, da die Lagerhaltung über die entsprechende Tabelle im Auftragsmodul abgewickelt wird.

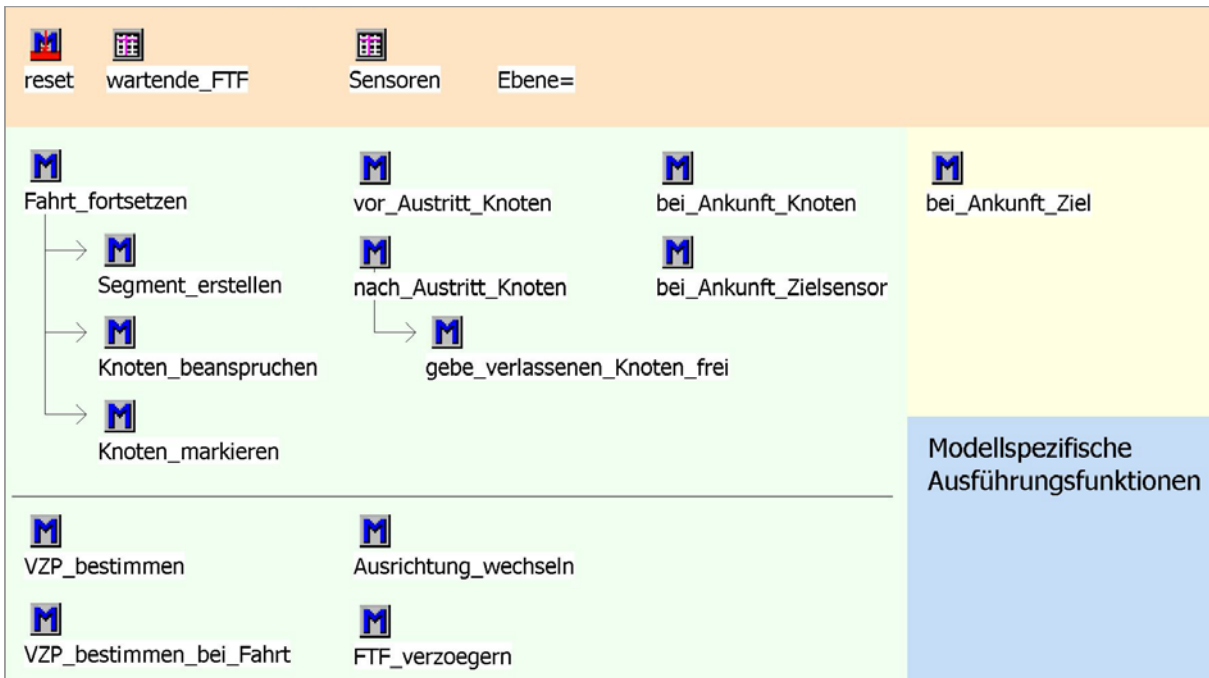


Abbildung 7-9: Implementierte Methoden zur Ausführung von berechneten Routen.

Beginnt die Ausführung eines Segmentes, wird der Verzögerungszeitpunkt bestimmt, zu welchem die Verzögerung des Fahrzeuges eingeleitet werden muss. Die Zielgeschwindigkeit wird auf die maximale Geschwindigkeit gesetzt und das Fahrzeug beginnt zu beschleunigen. Ist der Verzögerungszeitpunkt erreicht, wird die Zielgeschwindigkeit auf null gesetzt und das Fahrzeug kommt an der korrekten Position zum Stillstand.

Muss vor der Ausführung eines Segmentes ein Wechsel der Ausrichtung vollzogen werden, wird nach dem Verstreichen der notwendigen Zeitdauer eine neue Instanz der Fahrzeugklasse der entsprechenden Ausrichtung auf dem entsprechenden Wegstück im Knoten-Netzwerk erzeugt. Bevor das alte Fahrzeug-Objekt gelöscht werden kann, wird die Belegung sämtlicher Variablen auf das neue Fahrzeug übertragen.

7.2.6 Modellerstellung

Im Prinzip ist es möglich, den Lagerbereich manuell zu erstellen und anschließend das Layout zu initialisieren. Das erlaubt das Erstellen unterschiedlicher Layoutformen, er-

fordert jedoch einen hohen Aufwand. Um diesen zu reduzieren, werden von *Plant Simulation* bereitgestellte Methoden genutzt, mit welchen Instanzen automatisiert erzeugt werden können. Abbildung 7-10 zeigt die beispielhafte Implementierung eines Layoutgenerators.

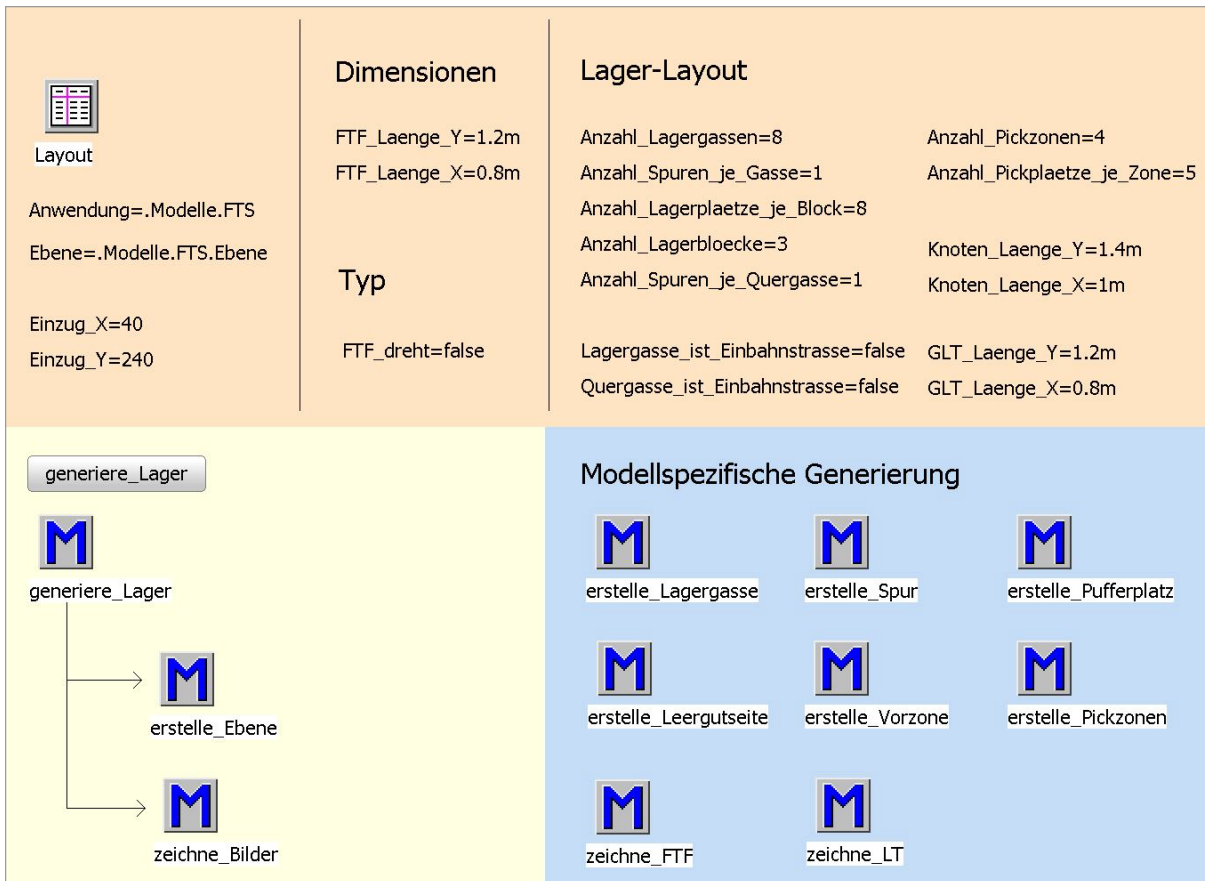


Abbildung 7-10: Beispielhafter Layoutgenerator.

Für die Modellerstellung sind die Dimensionen und der Typ eines Fahrzeuges festzulegen, um die korrekte Positionierung der Sensoren auf den Wegstücken vorzunehmen. Für die Erstellung des Layouts steht eine Reihe an Parametern zur Verfügung, um den Lagerbereich (Anzahl und Länge der Gassen, Anzahl Quergassen, Fahrtrichtung) zu definieren. Die Ausdehnung der Knoten ist festzulegen, kann aber bei Bedarf auch automatisiert berechnet werden, um beispielsweise die Knotenanzahl in einer Lagergasse zu maximieren. Die eingesetzten Knoten können aber auch unterschiedliche Ausdehnungen aufweisen.

7.2.7 Visualisierung der Steuerungsalgorithmen

Im Rahmen der Implementierung sind an verschiedenen Stellen im Simulationsmodell logische Überprüfungen mit dem Ziel der Verifikation integriert. So wird bei der Reser-

vierung der Zeitfenster sichergestellt, dass sich diese nicht mit vorhandenen Reservierungen auf dem entsprechenden Knoten überschneiden. Bei der Freigabe eines Knotens wird überprüft, ob die Simulationszeit mit dem Ende der Reservierung übereinstimmt. Werden keine Störungen simuliert, kann auf diesem Wege die korrekte Ausführung einer berechneten Route sichergestellt werden.

Um nicht nur die Ausführung einer Route überprüfen zu können, sondern auch die Berechnung der Route selbst, wird diese visualisiert. Die Animation stellt eine Verifikationstechnik dar, wobei das Ziel der Animation darin bestehen sollte, das Systemverhalten möglichst transparent darzustellen [Rab-2008, S. 96].

Abbildung 7-11 zeigt die Visualisierung des Routingprozesses. Ein Fahrzeug (orange) soll über einen Grid-Graphen, welcher abwechselnd von Quellen (blau) und Senken (grau) umgeben ist, zur hellblau markierten Senke geroutet werden (a).

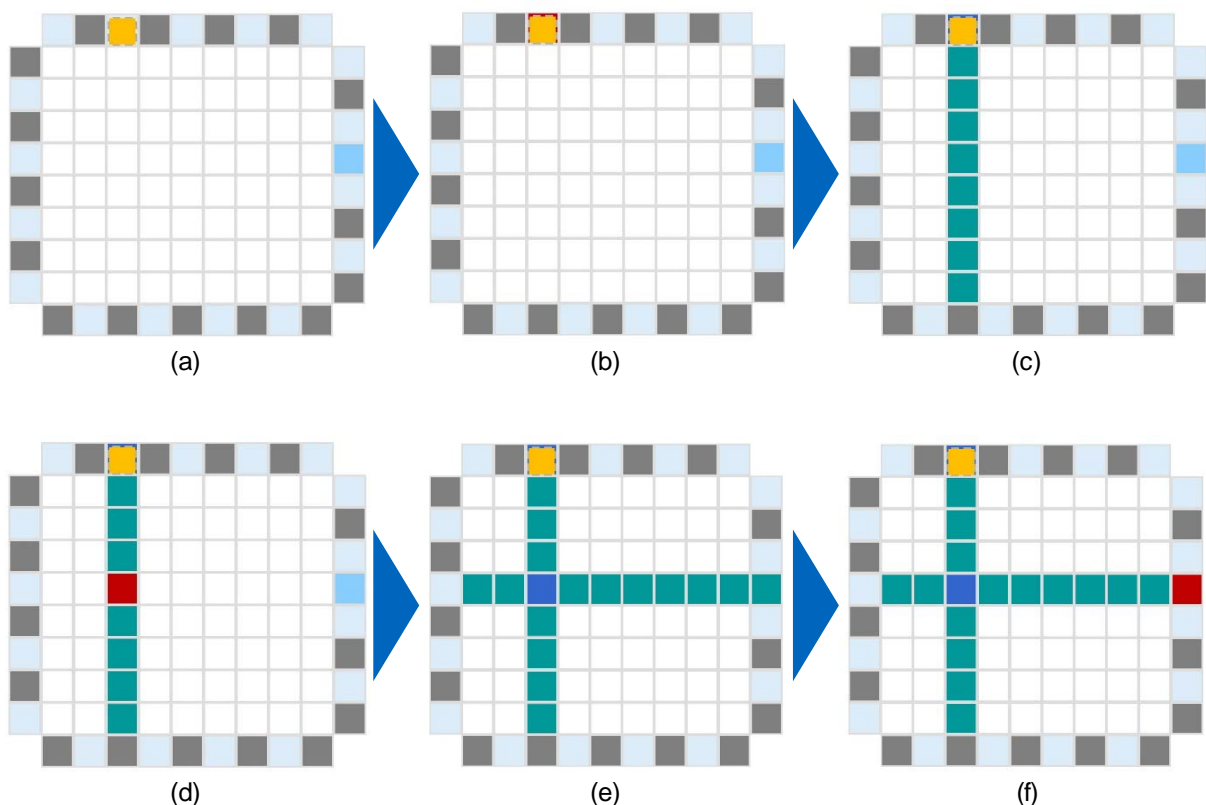


Abbildung 7-11: Visualisierung des Routingprozesses.

Visualisiert wird in jeder Iteration des zeitenfensterbasierten Routings der Knoten, auf welchem das für die nächste Iteration gewählte Zeitfenster liegt (rot, siehe (b), (d) und (f)), die Erstellung der Segmente (grün, siehe (c) und (e)) sowie die Knoten, auf welchen bereits untersuchte Zeitfenster liegen (dunkelblau, siehe (e) und (f)). Somit lässt

sich die Ermittlung der zeitenfensterbasierten Route schrittweise nachvollziehen und verifizieren.

Ist eine Route berechnet und werden die Zeitefenster für diese reserviert, werden die zugehörigen Knoten eingefärbt. Während der Ausführung einer berechneten Route werden die durch das Fahrzeug beanspruchten Knoten rot umrandet. Die in Abbildung 7-12 dargestellte Route entspricht der Beispielroute aus Tabelle 7-1.

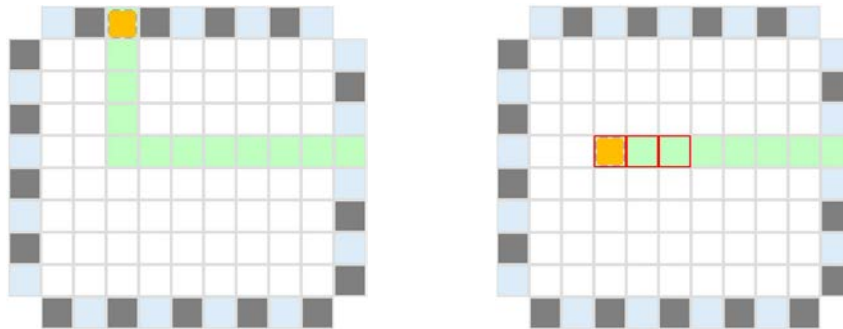


Abbildung 7-12: Visualisierung der Route und der beanspruchten Knoten bei der Ausführung.

Neben den reservierten Knoten lassen sich auch die reservierten Zeitefenster visualisieren. In einem separaten Netzwerk werden die Reservierungen in Anlehnung an ein Gantt-Diagramm aufgespannt. Abbildung 7-13 zeigt die reservierten Zeitefenster für die Route aus dem vorhergehenden Beispiel für alle drei Reservierungsformen. Die beiden Segmente sowie der Wechsel der Ausrichtung sind gut zu erkennen.

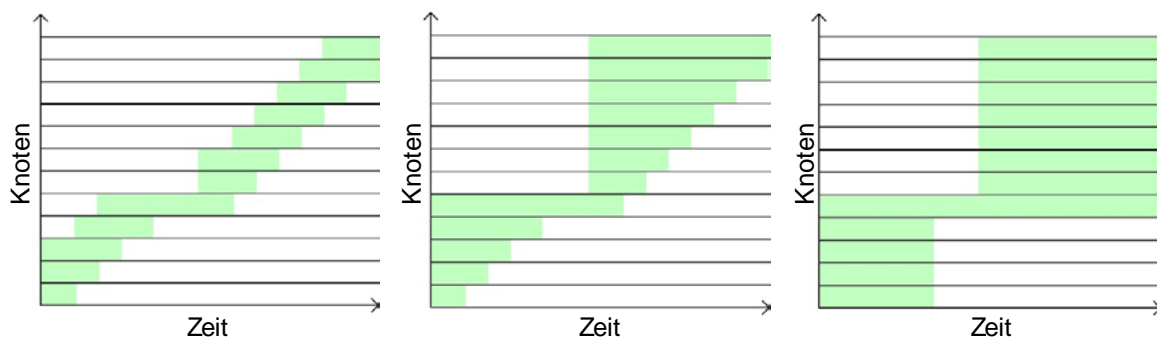


Abbildung 7-13: Visualisierung der reservierten Zeitefenster bei den unterschiedlichen Reservierungsformen Treppe, Dreieck und Block.

Für die Visualisierung der Routen ist jedem Fahrzeug eine eigene Farbe zugeordnet. Die reservierten Knoten werden derart eingefärbt, dass die sichtbare Farbe dem Fahrzeug zugeordnet ist, welches auf dem Knoten das erste Zeitefenster reserviert hat und somit das Fahrzeug ist, welches den Knoten als nächstes passieren wird. In dem Bei-

spiel in Abbildung 7-14 passiert das Fahrzeug mit der blauen Routenfarbe einige Knoten entlang seiner Route vor dem Fahrzeug mit der gelben Routenfarbe, was sich auch an der Darstellung der reservierten Zeitfenster erkennen lässt.

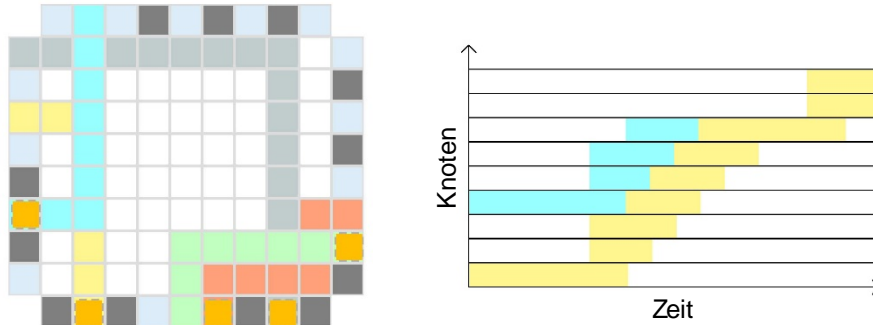


Abbildung 7-14: Visualisierung mehrerer Routen für unterschiedliche Fahrzeuge.

8 Evaluierung

In diesem Kapitel erfolgt die Evaluierung der entwickelten Methodik, welche sich aus der Modellierung, der Steuerungsstrategien sowie der Umsetzung in einer Simulationsumgebung zusammensetzt. Die Evaluierung wird anhand zweier Fallstudien durchgeführt. Im Unterkapitel 8.1 werden unterschiedliche Layoutvarianten für ein Kommissioniersystem mit Unterfahr-FTF untersucht sowie die im Unterkapitel 6.4 vorgestellten Strategien für den Umgang mit Störungen verglichen. Im Unterkapitel 8.2 wird ein Shuttle-System mit gassen- und ebene wechselnden Shuttle-Fahrzeugen betrachtet und die Auswirkung einer Auslagerung in Sequenz analysiert. In beiden Fallstudien werden die charakteristischen Durchsatzverläufe erstellt und ausgewertet.

8.1 Durchsatzanalyse eines automatisierten Kommissioniersystems mit Unterfahr-FTF

In der ersten Fallstudie wird ein Kommissioniersystem betrachtet, welches auf Basis einer Flotte von Unterfahr-FTF betrieben wird. Das betrachtete System sowie die verwendeten Auftragsvergabe-strategien beruhen auf einem geplanten System für die Kommissionierung in der Produktion bei einem Automobilhersteller. Die im Rahmen dieser Arbeit vorgestellte Methodik wurde in der Angebotsphase angewendet, um die Flottengröße zur Erreichung eines gegebenen Durchsatzes auf einem vorgegebenen Layout zu bestimmen. Im Abschnitt 8.1.1 wird das betrachtete System beschrieben, bevor im Abschnitt 8.1.2 die Modellierung erläutert wird. Abschnitt 8.1.3 beschreibt die Durchführung der Simulationsstudie, bei welcher zum einen unterschiedliche Layoutvarianten gegenübergestellt, zum anderen die im Unterkapitel 6.4 vorgestellten Strategien für den Umgang mit Störungen verglichen werden.

Eine Studie zum Vergleich verschiedener Layoutvarianten des betrachteten Systems findet sich in [Lie-2018c]. In dieser wurde jedoch ein Modell herangezogen, welches nicht die Berücksichtigung von Beschleunigungs- und Verzögerungsvorgängen umfasst. Die Betrachtung des Systems unter dem Einfluss von Störungen findet sich in [Lie-2019], wobei nicht nur die verschiedenen Strategien für den Umgang mit Störungen verglichen wurden, sondern auch die Auswirkungen unterschiedlicher Verfügbarkeiten der FTF analysiert wurden.

8.1.1 Systembeschreibung

Die gelagerten, zu kommissionierenden Artikel befinden sich auf Regalen, welche in 14 Regalreihen mit je 24 Regalen angeordnet sind. Am Fue der Regalreihen befindet sich die Pickzone. In vier Pickbereichen werden die Regale auf jeweils fnf Pickpltzen durch die FTF bereitgestellt. Dafr unterfahren die FTF die auszulagernden Regale, heben diese an und transportieren sie in die Pickzone. Beladene FTF knnen sich somit lediglich in den Lager- und Quergassen zwischen den Regalreihen fortbewegen. Unbeladene FTF hingegen knnen auch unter gelagerten Regalen hindurchfahren. Abbildung 8-1 zeigt ein beispielhaftes System.

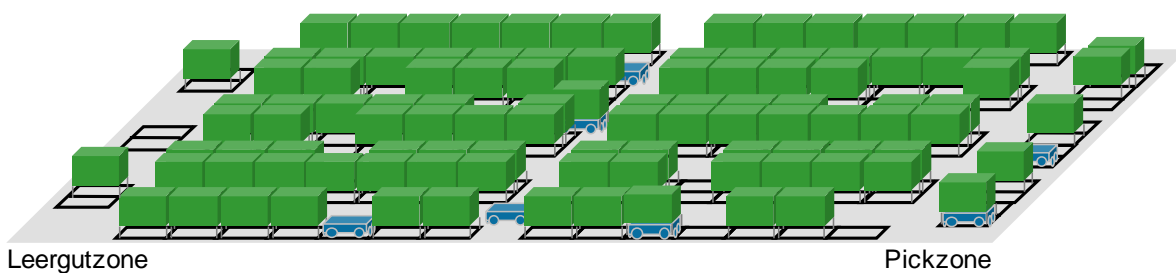


Abbildung 8-1: Kommissionierlager mit Unterfahr-FTF.

Die FTF fhren, beginnend mit der Fahrt in der Pickzone, drei unterschiedliche Spiele aus. Kann ein Regal nach dem Pickvorgang wieder eingelagert werden, wird dieses von einem FTF aufgenommen. Anschließend wird dem FTF das als nchstes auszulagernde Regal zugewiesen. Im Standardspiel erfolgt nun die Einlagerung. Das aufgenommene Regal wird auf einem zufllig gewhlten freien Lagerplatz in derjenigen Gasse eingelagert, in welcher das Regal zuvor entnommen wurde. Es folgt eine Leerfahrt zum Auslagerplatz, wo das auszulagernde Regal aufgenommen wird. Das Spiel schliet mit der Fahrt und der Abgabe des Regals in der gleichen Pickzone, in welcher das Spiel begonnen hat. Jedes FTF ist einer Pickzone somit fest zugeordnet.

Ist nach dem Pickvorgang eine Nachbefllung des Regals mit dem entnommenen Artikel notwendig, wird der zweite Spieltyp ausgefhrt. Das Regal ist nicht einzulagern, sondern auf einen freien Platz in der auf der gegenberliegenden Seite befindlichen Leergutzone zu bringen, wo die Regale aufgefllt werden. Es folgt die Leerfahrt zum Auslagerplatz sowie die Bereitstellung in der Pickzone.

Beim dritten Spieltyp wird ein nachbeflltes Regal zwischen der eigentlichen Ein- und Auslagerung eingelagert, sofern dieses in derselben Gasse einzulagern ist, der das nchste Regal entnommen wird. Das Ablaufdiagramm in Abbildung 8-2 zeigt die Entstehung sowie die einzelnen Bestandteile der drei unterschiedlichen Spieltypen.

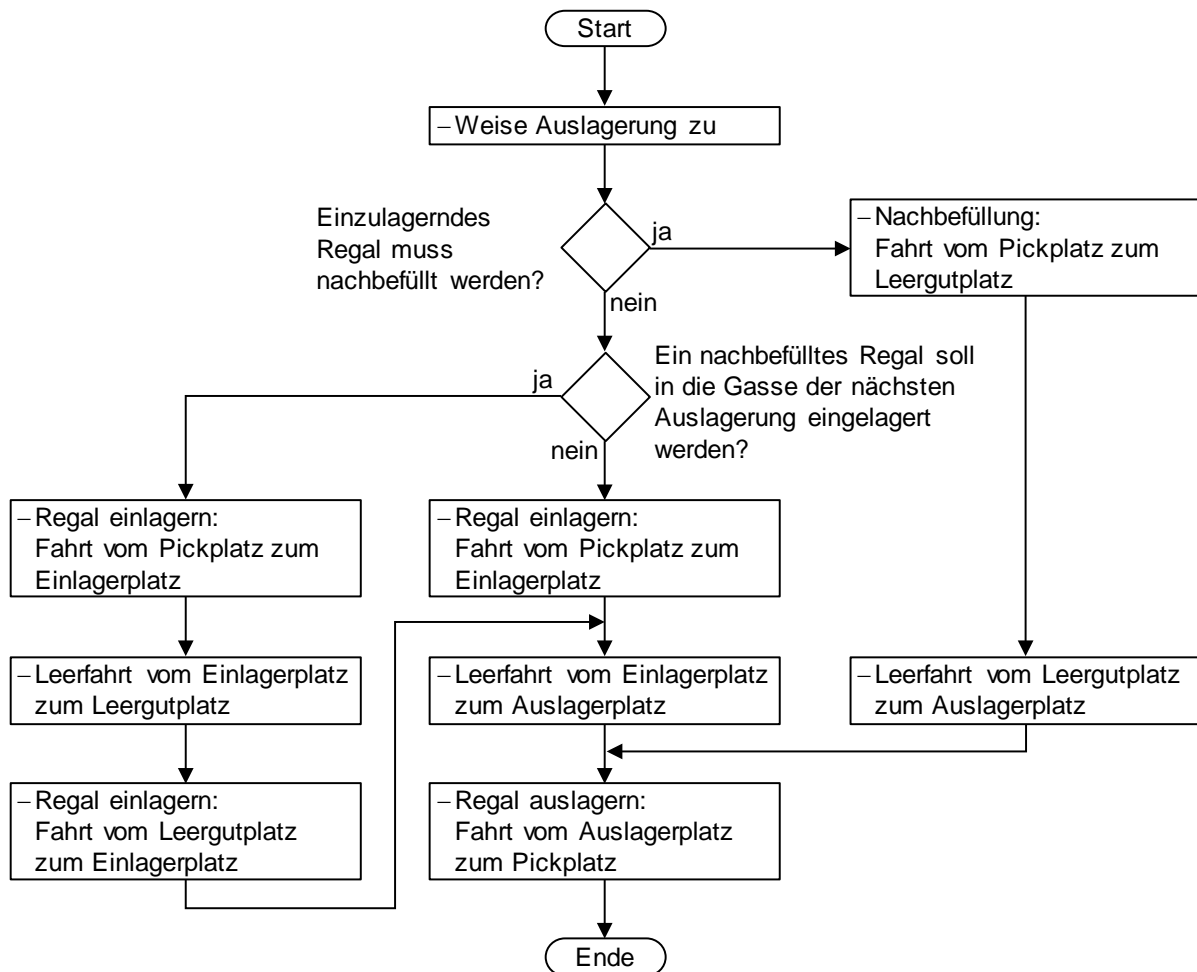


Abbildung 8-2: Zusammensetzung der unterschiedlichen Spiele, beginnend mit der Aufnahme eines einzulagernden Regals in der Pickzone.

Das Routing erfolgt nach der Bestimmung des Spieltyps und endet in jedem Fall zunächst am Auslagerplatz, da zum Zeitpunkt des Routings noch nicht abzusehen ist, ob in der entsprechenden Pickzone ein freier Pickplatz verfügbar ist. Somit wird für das Erreichen des finalen Ziels ein nach hinten offenes Zeitfenster benötigt, da nicht bekannt ist, wann das FTF den Auslagerplatz wieder verlassen wird. Nach der Aufnahme des auszulagernden Regals erfolgt das Routing in die Pickzone, sofern ein freier Platz verfügbar ist. Anderenfalls trägt sich das FTF in einer Warteliste ein und wird benachrichtigt, sobald ein FTF die Pickzone verlässt und somit ein Pickplatz frei wird.

Bis zu einer Anzahl von vier FTF je Pickzone stellt ein FTF das ausgelagerte Regal auf dem Pickplatz ab und fährt in derselben Pickzone zu dem Regal, welches bereits am längsten auf die Einlagerung wartet. Ab einer Anzahl von fünf FTF je Pickzone verweilen die FTF hingegen unter dem Regal, bis der Pickvorgang abgeschlossen ist.

8.1.2 Modellierung

Das gesamte Layout, einschließlich der Lager-, Leergut- und Pickplätze wird als ein Grid modelliert, in dem alle Knoten die gleichen Ausdehnungen aufweisen. Die Knotengröße entspricht dabei der Größe eines Lagerplatzes ($L_i^X = 1m, L_i^Y = 1,4m$). Da alle Knoten Kreuzungsknoten sind, wird lediglich eine Knotenklasse benötigt. Über die Belegung eines Attributes können die verschiedenen Typen (Lagerplatz-, Gassen-, Quergassen-, Pickplatz- und Leergutplatzknoten) unterschieden werden.

Abbildung 8-3 zeigt einen Screenshot aus dem Simulationsmodell. Jedes Rechteck entspricht einem Knoten. Gelagerte und einzulagernde Regale sind dunkelgrau gefärbt, auszulagernde Regale hingegen hellgrau.

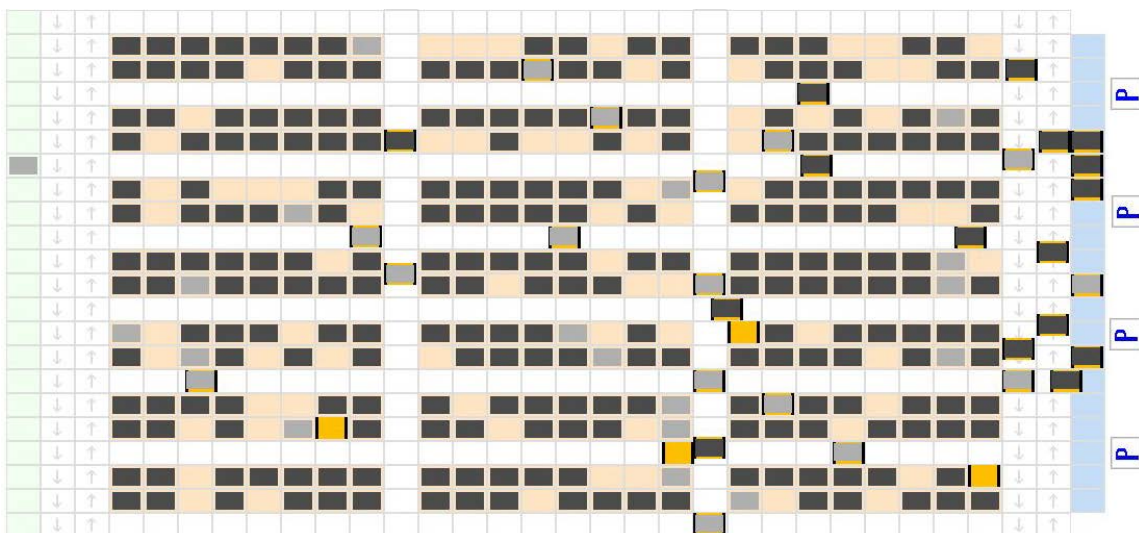


Abbildung 8-3: Screenshot aus dem Simulationsmodell (um 90 Grad gedreht): Abgebildet ist der gesamte Lagerbereich einschließlich Pickzone (blau) und Leergutzone (grün).

Da unbeladenen FTF ein anderes Wegenetz als beladenen FTF zur Verfügung steht, werden für das Routing zwei minimale Fahrzeitenmatrizen herangezogen. Darüber hinaus wird bei der Erstellung der Segmente (siehe Abschnitt 6.1.4) geprüft, ob der als nächstes zu betrachtende Knoten überhaupt von dem FTF mit dem aktuellen Status (beladen, unbeladen) befahren werden darf.

Lagerplatzknoten dürfen von beladenen FTF nur dann befahren werden, sofern diese das nächste Ziel im Routingprozess darstellen. Diese Einschränkung stellt eine Vereinfachung bei der Steuerung des Gesamtsystems dar, welche den Lagerablauf jedoch robuster gestaltet. Es ist möglich, dass beladene Fahrzeuge Lagerplatzknoten auch dann befahren dürfen, wenn diese nicht das nächste Ziel im Routingprozess darstellen, sofern sich auf diesen Lagerplätzen keine Regale befinden. Dafür müsste der Zustand des Lagerplatzes zum Zeitpunkt zu dem Fahrzeug den Knoten befährt bereits

zum Zeitpunkt des Routings bekannt sein und dürfte sich anschließend nicht mehr ändern.

Eine ähnliche Einschränkung gilt für Pick- und Leergutplätze sowohl für beladene als auch für unbeladene FTF. Diese sollen nur von Fahrzeugen befahren werden, deren nächstes Ziel auf diesen Knoten liegen. Dadurch werden diese Zielknoten nicht von anderen Fahrzeugen blockiert.

Das Modell wird initialisiert, indem die FTF reihum auf freien Knoten in den vier Pickzonen erzeugt werden. Sind alle Pickplätze belegt – im betrachteten Layout sind es 20 – werden weitere FTF auf belegten Lagerplatzknoten erzeugt. Zu Beginn der Simulation nehmen diese dann das entsprechende Regal auf und melden sich in der Pickzone an.

Eine Lagerhaltung wird nicht abgebildet. Die Generierung und Zuweisung von Auslageraufträgen erfolgt auf Regal- und nicht auf Artikelebene. Ob ein Regal nach dem Pickvorgang wieder befüllt werden muss, wird über eine Wahrscheinlichkeit abgebildet, welche einen modelspezifischen Parameter darstellt.

8.1.3 Simulationsstudie

In der Simulationsstudie werden zunächst die Auswirkungen dreier unterschiedlicher Layoutvarianten auf den Durchsatz untersucht. Die Anzahl und Länge der Lagergassen bleiben konstant, ebenso die Anzahl der Pick- und Leergutplätze. Die zu untersuchenden Layoutvarianten unterscheiden sich hinsichtlich der Anzahl der Fahrspuren je Lagergasse sowie deren Fahrtrichtung (Abbildung 8-4). Verglichen werden ein doppelspuriges Layout (Layout 1), ein einspuriges Layout mit bidirektionalen Spuren (Layout 2, dargestellt in Abbildung 8-3) sowie ein einspuriges Layout mit unidirektionalen Spuren (Layout 3). In einem zweiten Schritt werden die Varianten mit zwei eingezogenen Quergassen (nach acht und sechzehn Lagerplätzen) analysiert, wobei die Anzahl der Spuren je Quergasse sowie deren Fahrtrichtung analog zu den Lagergassen ausgestaltet sind.

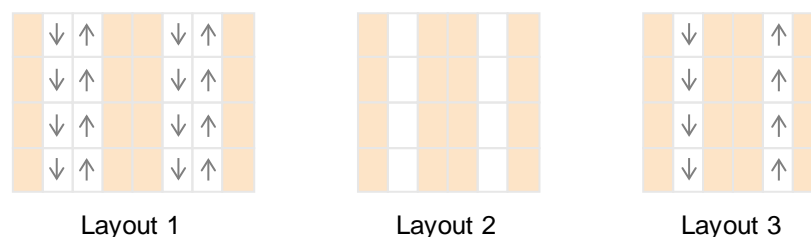


Abbildung 8-4: Untersuchte Layoutvarianten.

Es ist anzunehmen, dass das Layout 1 den höchsten Durchsatz erzielt, da sich entgegenkommende FTF in den Lagergassen nicht beeinflussen. Im Layout 2 hingegen müssen entgegenkommende FTF entweder warten oder aber einen Umweg fahren. Im Layout 3 können sich FTF innerhalb einer Lagergasse nicht begegnen, jedoch verlängern sich die Fahrstrecken in dem Einbahnstraßensystem. Für eine geringere Anzahl an FTF sollte sich daher im Layout 2 ein höherer Durchsatz erzielen lassen. Mit zunehmender Anzahl an FTF wird der Vorteil der kürzeren Fahrstrecken jedoch vermutlich durch erhöhte Wartezeiten kompensiert. Anzumerken ist, dass der Platzbedarf für das Layout 2 und Layout 3 identisch ist, während das Layout 1 aufgrund der doppelspurigen Lagergassen einen deutlich erhöhten Platzbedarf aufweist.

Die Modellparameter sind Tabelle 8-1 zu entnehmen:

Tabelle 8-1: Modellparameter Unterfahr-FTF.

Allgemeine Parameter		Modellspezifische Parameter	
Parameter	Belegung	Parameter	Belegung
L_F^X	1,2m	Lagerfüllgrad	80%
L_F^Y	0,8m	Pickzeit	5s
v^X	0,5m/s	Leergutquote	5%
v^Y	0,5m/s	Füllzeit Leergut	100s
a^X	0,5m/s ²		
a^Y	0,5m/s ²		
t_{RW}	5s		
$t_{Lastwechsel}$	5s		
Reservierungsform	Treppe		
Max Segmentlänge	∞		
t_p	0,01s		

Die Anzahl der FTF wird beginnend mit vier FTF – für jede Pickzone eines – schrittweise um vier FTF bis 60 FTF erhöht. Der Lagerbetrieb wird über eine Dauer von 24 Stunden simuliert. Mit jeder Parameterkonfiguration werden fünf Replikationen erstellt. Die sich zum Konfidenzniveau von 95% ergebenden Konfidenzintervalle werden als hinreichend klein erachtet. Über alle 90 durchgeführten Experimente weisen die Konfidenzintervalle im Mittel eine Breite von $\pm 0,36\%$ um den Mittelwert auf, die maximale Breite eines Konfidenzintervalls beträgt $\pm 1,19\%$.

Das Diagramm in Abbildung 8-5 zeigt die Durchsatzverläufe für die drei Layoutvarianten. Bis zu einer Anzahl von zwölf FTF werden mit dem Layout 2 die größten Durchsätze erzielt. Dies ist auf die verkürzten Fahrstrecken gegenüber Layout 1 (durch doppelspurige Lagergassen) sowie Layout 3 (durch Einbahnstraßensystem) zurückzuführen.

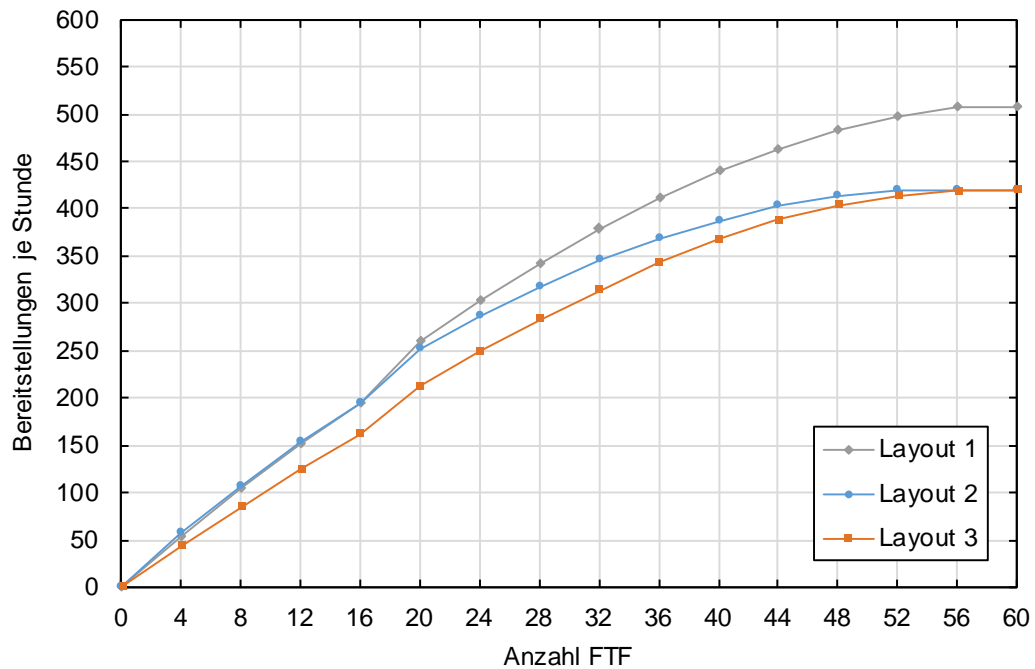


Abbildung 8-5: Durchsatzverläufe für die betrachteten Layoutvarianten.

Der Sprung in den Durchsatzverläufen zwischen 16 und 20 FTF ist auf das zuvor beschriebene, veränderte Verhalten der FTF in der Pickzone zurückzuführen. Für alle drei Layouts steigt der Durchsatz – abgesehen von dem erwähnten Sprung – zunächst im linearen Bereich an, flacht aber mit zunehmender Anzahl an FTF immer weiter ab, bis der Grenzdurchsatz erzielt wird. Der Grenzdurchsatz wird als erreicht betrachtet, wenn der Leistungsgewinn durch vier zusätzliche FTF unter einem Prozent liegt. Der höchste Grenzdurchsatz wird mit 507 Bereitstellungen je Stunde mit dem Layout 1 mit 56 FTF erreicht. Der Grenzdurchsatz von 419 Bereitstellungen je Stunde wird mit dem Layout 2 mit 52 FTF erreicht und entspricht dem Grenzdurchsatz, welcher mit dem Layout 3 erreicht wird, wenn 56 FTF im Einsatz sind.

Ein tiefergehendes Systemverständnis liefert die Analyse der Zeitanteile der FTF, welche im Diagramm in Abbildung 8-6 für das Layout 2 dargestellt sind. Neben den im Abschnitt 7.1.5 definierten Standardzeitanteilen wurde die Auswertung um zwei Zeitanteile ergänzt. Der Zeitanteil *wartend auf Pickplatz* wird erfasst, wenn ein FTF mit einem Regal auf einem Pickplatz eintrifft und unter dem Regal wartet, bis der Artikel gepickt wurde und das Regal wieder eingelagert werden kann. Der Zeitanteil *wartend auf Lagerplatz* wird erfasst, wenn ein FTF ein auszulagerndes Regal aufgenommen hat, aber die Fahrt in die Pickzone noch nicht starten kann, weil kein Pickplatz frei ist.

Bis einschließlich 16 FTF entfällt sämtliche Zeit auf die Zeitanteile *fahrend*, *wartend* und *lastwechselnd*. Mit zunehmender Anzahl an FTF steigt der Wartezeitanteil im System, welcher durch den zunehmenden Verkehr hervorgerufen wird. Ab 40 FTF nimmt

der Zeitanteil *wartend auf Pickplatz* deutlich zu. Die Anzahl der FTF, welche sich entweder auf einem Platz in der Pickzone befinden oder aber unterwegs zur der Pickzone sind, ist in dem betrachteten System auf fünf FTF beschränkt. Mit einer zunehmenden Anzahl an FTF dauert die Fahrt vom Auslagerplatz zum Pickplatz aufgrund des steigenden Verkehrs immer länger. Zusätzlich eingesetzte FTF bringen eine immer geringere Durchsatzsteigerung und führen dazu, dass die FTF immer länger auf den Lagerplätzen warten, bis die Fahrt zur Pickzone begonnen werden kann.

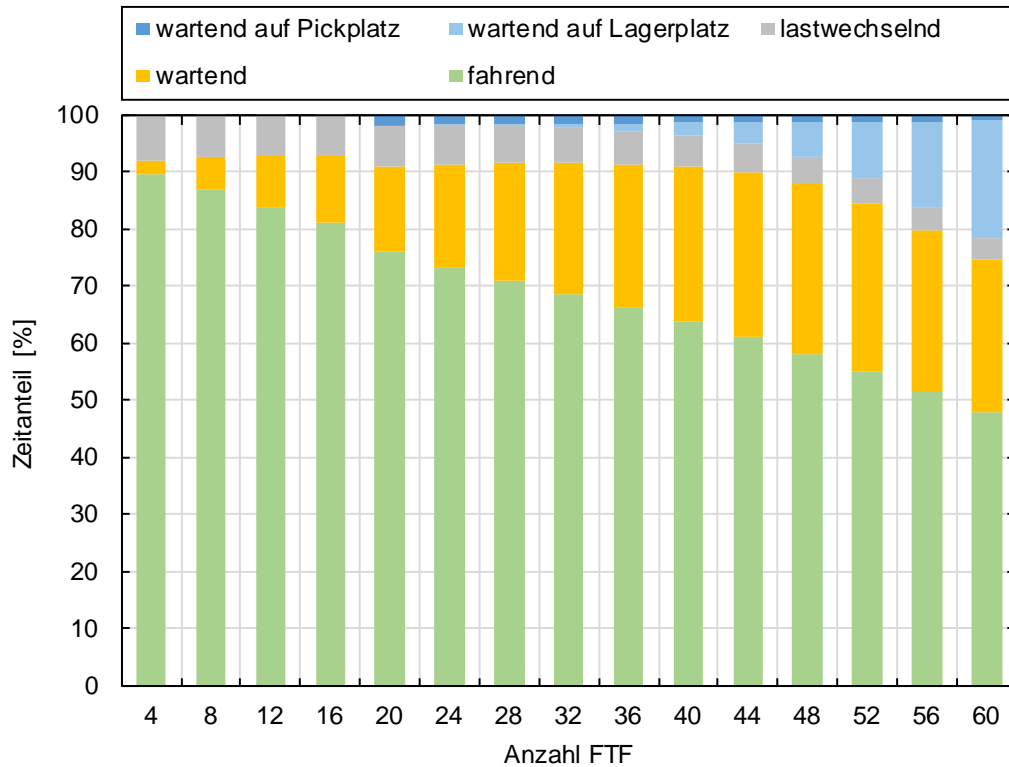


Abbildung 8-6: Zeitanteile der FTF für das Layout 2.

Abbildung 8-7 zeigt die Heatmap für die Anzahl der Knotenbelegungen im Layout 2. Der Verkehr konzentriert sich auf die Quergassen in der Vorzone, welche offensichtlich einen Engpass darstellen. Die Lagergassen sind auf der Heatmap deutlich zu erkennen. Die Anzahl der Belegung der einzelnen Knoten nimmt ab, je größer die Distanz von den Knoten zur Pickzone ist.

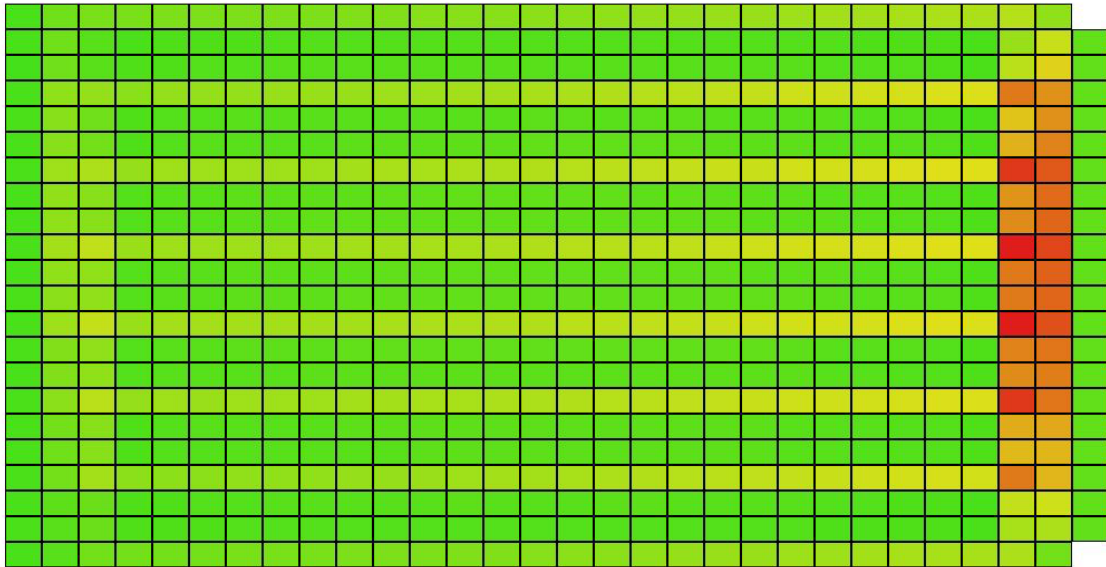


Abbildung 8-7: Heatmap für die Anzahl der Knotenbelegungen für das Layout 2.

Das Diagramm in Abbildung 8-8 zeigt die Durchsatzverläufe für die drei Layoutvarianten einschließlich der Varianten mit eingezogenen Quergassen. Es zeigt sich, dass durch die Quergassen der Grenzdurchsatz nochmals deutlich gesteigert werden kann. Dieser Effekt ist bei Layout 2 am deutlichsten zu beobachten.

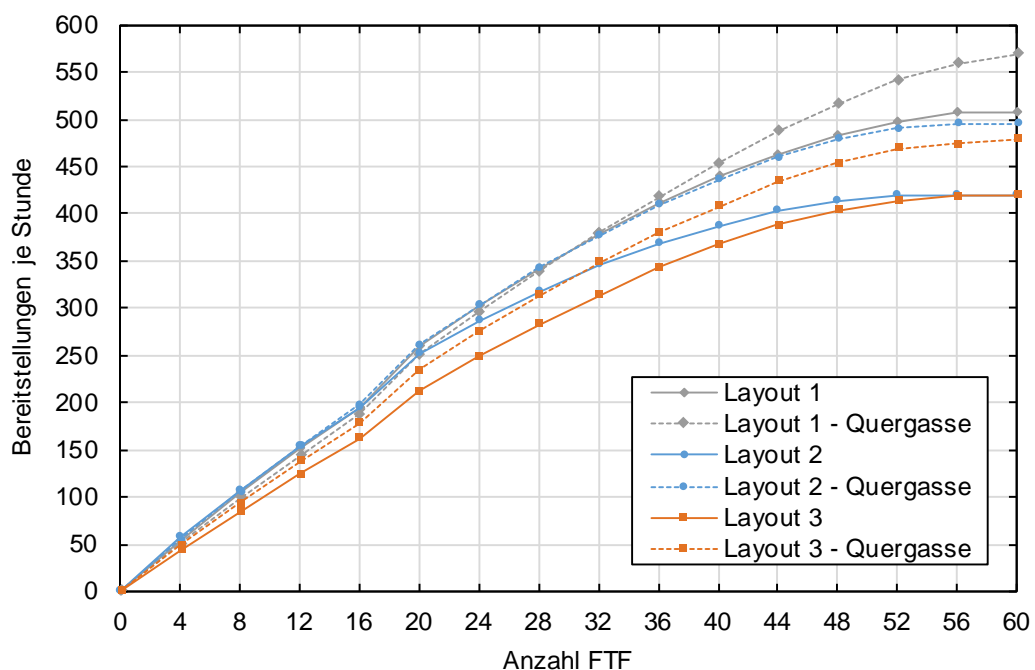


Abbildung 8-8: Durchsatzverläufe für die betrachteten Layoutvarianten einschließlich Varianten mit Quergassen.

In einer zweiten Versuchsreihe werden die in Unterkapitel 6.4 beschriebenen Strategien für den Umgang mit Störungen verglichen. Dafür wird jedem FTF eine Verfügbarkeit von 99 Prozent sowie eine mittlere MTTR von drei Minuten zugeordnet. Die nach

einer aufgetretenen Störung benötigte MTTR wird über eine Erlang-Verteilung (mit dem Erwartungswert von drei Minuten) bestimmt. Die Dauer zwischen dem Auftreten zweier Störungen ist exponentialverteilt (mit einem Erwartungswert von 297 Minuten). Aufgrund der großen Abstände zwischen dem Auftreten zweier Störungen eines FTF wird die Simulationszeit auf 72 Stunden hochgesetzt.

Mit jeder Parameterkonfiguration werden abermals fünf Replikationen erstellt. Die sich zum Konfidenzniveau von 95% ergebenden Konfidenzintervalle werden im Allgemeinen als hinreichend klein erachtet, auch wenn es zu einzelnen Ausreißern kommt und die Breite sich je nach angewendeter Strategie unterscheidet. Insbesondere bei den Strategien *Ignorieren* und *Pausieren* treten größere Streuungen auf.

Tabelle 8-2: Mittlere und maximale Breite der Konfidenzintervalle.

Strategie	Mittlere Breite	Maximale Breite
Ignorieren	± 3,32%	± 5,45%
Pausieren	± 3,72%	± 8,37%
Neustarten	± 0,92%	± 1,88%
Umleiten	± 3,72%	± 0,48%

Das Diagramm in Abbildung 8-9 zeigt die Durchsatzverläufe für die unterschiedlichen Strategien auf dem Layout 2 mit Quergassen.

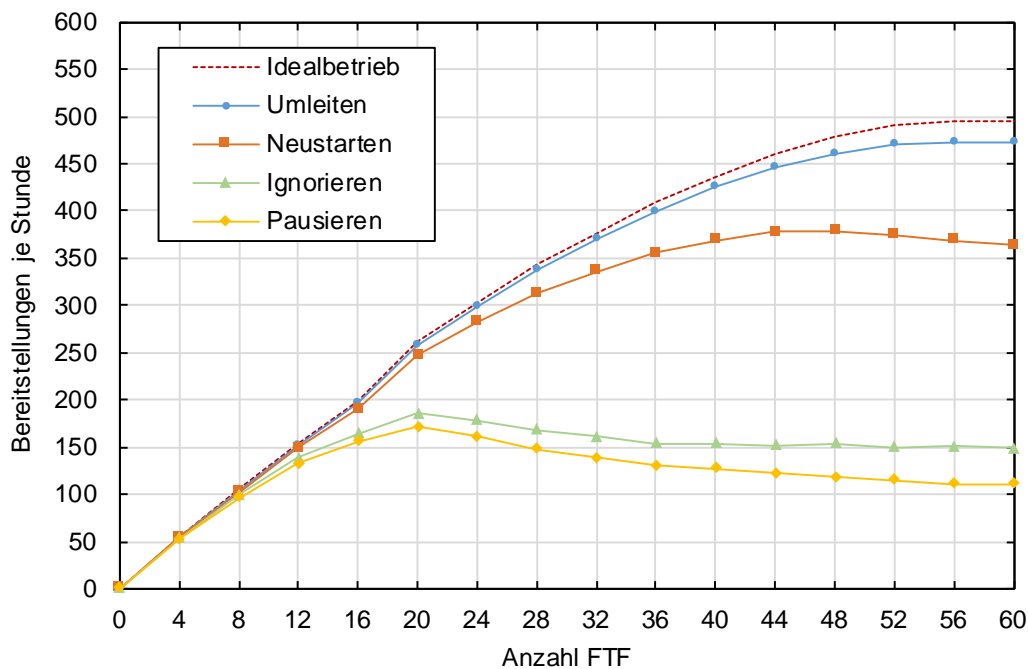


Abbildung 8-9: Durchsatzverläufe für unterschiedliche Strategien für den Umgang mit Störungen für das Layout 2 mit Quergassen.

Die Unterschiede zwischen den Strategien werden deutlicher, je mehr FTF im System arbeiten. Die Strategie *Pausieren* erzielt den niedrigsten Durchsatz. Dies war zu erwarten, da die Strategie die Redundanz der FTF-Flotte nicht nutzt. Die Strategie *Ignorieren* erzielt lediglich einen leicht höheren Durchsatz. Bei einer größeren Anzahl an FTF im System breitet sich eine Störung aufgrund der Notwendigkeit, die ursprünglich geplante Reihenfolge an den Knoten einzuhalten, schnell über das gesamte System aus. Sowohl *Pausieren* als auch *Ignorieren* erzielen mit 20 FTF den höchsten Durchsatz. Mit zunehmender Anzahl an FTF steigt die Wahrscheinlichkeit, dass mindestens ein FTF der Flotte gestört ist, immer weiter an und die gesamte Flotte wird zum Anhalten gezwungen oder zumindest an der Ausführung der Routen behindert.

Die Strategie *Neustarten* erzielt deutlich höhere Durchsätze als *Pausieren* und *Ignorieren* und die Durchsatzkurve weist nahezu auch die charakteristische Form auf. Jedoch werden auch bei dieser Strategie alle FTF zum Anhalten gezwungen, obgleich ein erneutes Routing unmittelbar nach dem Anhalten aller FTF erfolgt.

Der höchste Durchsatz unter der Berücksichtigung von Störungen wird mit der Strategie *Umleiten* erzielt. Das Diagramm in Abbildung 8-10 stellt den Durchsatzverlust gegenüber dem Idealbetrieb dar. Unter der Berücksichtigung der Verfügbarkeit von 99 Prozent liegt der Durchsatzverlust der Strategie *Umleiten* bis einschließlich 24 FTF nahe am Optimum von einem Prozent. Selbst bei einer Anzahl von 60 FTF im System liegt der Durchsatzverlust bei unter 4,5 Prozent.

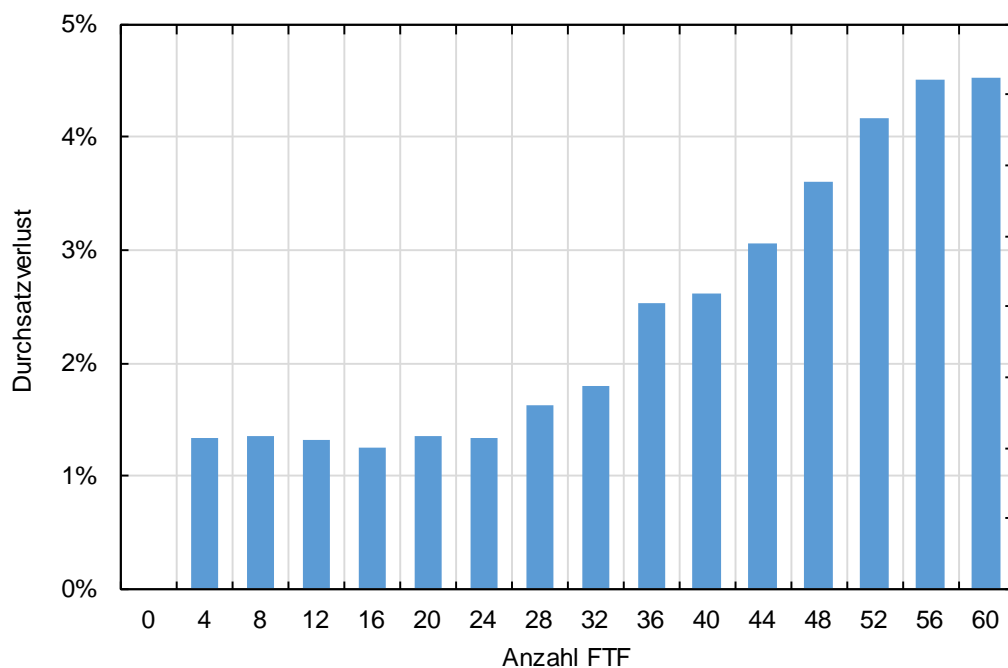


Abbildung 8-10: Durchsatzverlust beim Anwenden der Strategie *Umleiten* gegenüber dem Idealbetrieb.

8.2 Durchsatzanalyse eines Shuttle-Systems mit gassen- und ebenenwechselnden Shuttle-Fahrzeugen

In der zweiten Fallstudie wird ein exemplarisches Shuttle-System betrachtet, welches im Abschnitt 8.2.1 beschrieben wird. Im Abschnitt 8.2.2 wird die Modellierung erläutert, welche in Teilen [Lie-2017b] entnommen ist. Abschnitt 8.2.3 beschreibt die Durchführung der Simulationsstudie einschließlich der Ergebnisse. Eine analoge Studie findet sich in [Lie-2018b], wobei bei dieser beim zeitfensterbasierten Routing die Reservierungsform *Block* (siehe Exkurs 6.1.5) verwendet wurde.

8.2.1 Systembeschreibung

Betrachtet wird ein System mit gassen- und ebenenwechselnden Shuttle-Fahrzeugen der Variante 4 (siehe Klassifikation von Shuttle-Systemen in Abbildung 2-4 im Abschnitt 2.3.2). In Shuttle-Systemen der Variante 4 kann jedes Shuttle-Fahrzeug jeden Lagerplatz im System anfahren. Abbildung 8-11 zeigt ein beispielhaftes Shuttle-System dieser Variante.

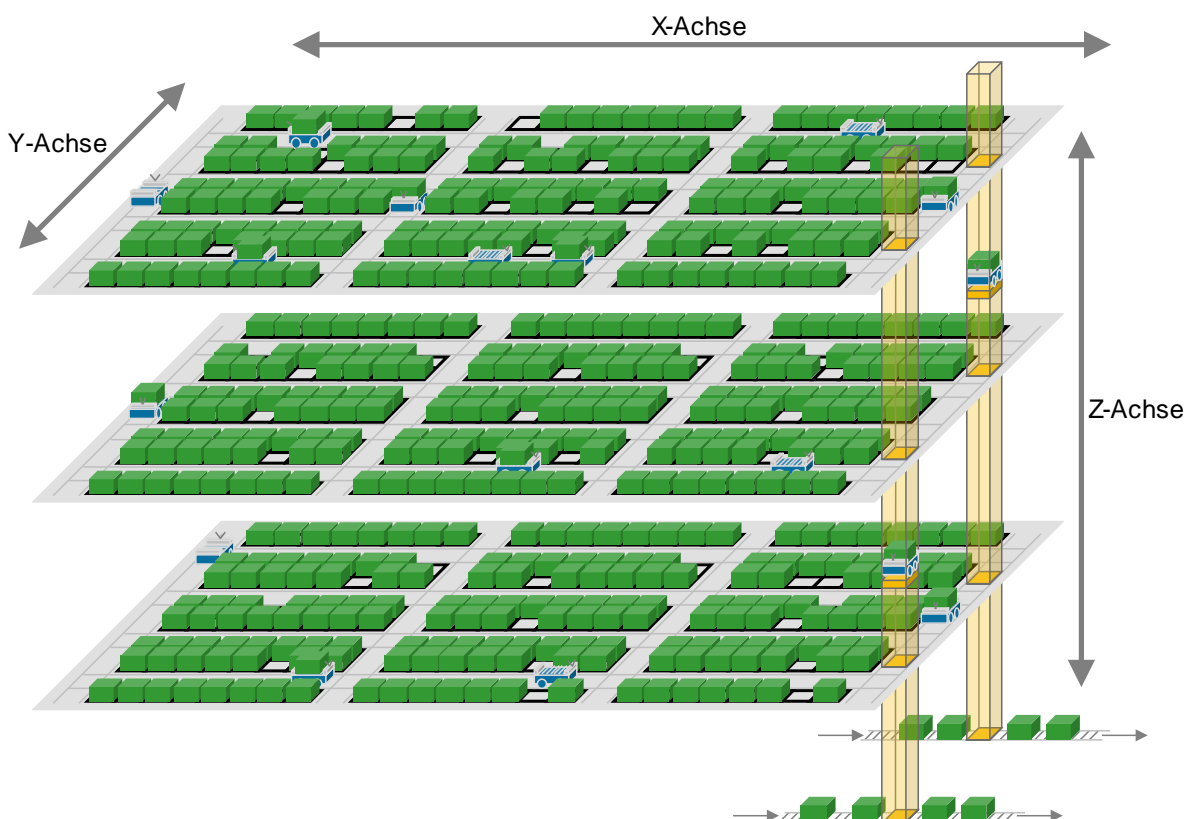


Abbildung 8-11: Shuttle-System der Variante 4.

Das System besteht aus zehn Lagerebenen, welche durch drei Lifte miteinander sowie mit der E/A-Ebene verbunden sind. Auf jeder Ebene sind zwölf Lagergassen angeordnet, welche durch zwei Quergassen miteinander verbunden sind. Eine Quergasse befindet sich unmittelbar vor den Liften, die andere in der Gassenmitte. Die Shuttle-Fahrzeuge fahren von einer Seite in den Lift hinein und verlassen diesen an der gegenüberliegenden Seite. Auf der E/A-Ebene findet die Übergabe der Lagereinheiten an die Fördertechnik der Vorzone statt, ohne dass die Fahrzeuge den Lift verlassen.

Die Shuttle-Fahrzeuge führen ausschließlich Doppelspiele durch. Nach Übergabe der ausgelagerten Lagereinheit auf der E/A-Ebene wird eine einzulagernde Lagereinheit aufgenommen. Darüber hinaus wird dem Shuttle-Fahrzeug der nächste Auslagerauftrag aus dem Auftragspool zugewiesen, welcher kontinuierlich mit neuen Auslageraufträgen nachbefüllt wird. Auf der Auslagerebene wird zufällig ein leerer Lagerplatz für die Einlagerung ermittelt. Somit entstehen Doppelspiele, welche sich stets auf eine Lagerebene beschränken.

Da in dem betrachteten System nicht nur jedes Shuttle-Fahrzeug jeden Lagerplatz im gesamten System anfahren kann, sondern auch jede Lagereinheit an jedem E/A-Punkt bereitgestellt werden kann, ist eine Auslagerung in exakter Sequenz direkt aus dem System heraus möglich. Jeder Lift versorgt einen eigenen Sequenzstrom. In dem betrachteten Lager existieren somit drei unabhängige Sequenzströme. Die Sequenzierung erfolgt zweistufig. Auf jeder Ebene müssen die Lagereinheiten einer jeden Sequenz an den Liften nach aufsteigender Sequenznummer bereitgestellt werden. Die finale Sequenz wird anschließend durch den Lift gebildet. Er zieht die Sequenznummern in strikter Reihenfolge von den einzelnen Ebenen ab (Abbildung 8-12).

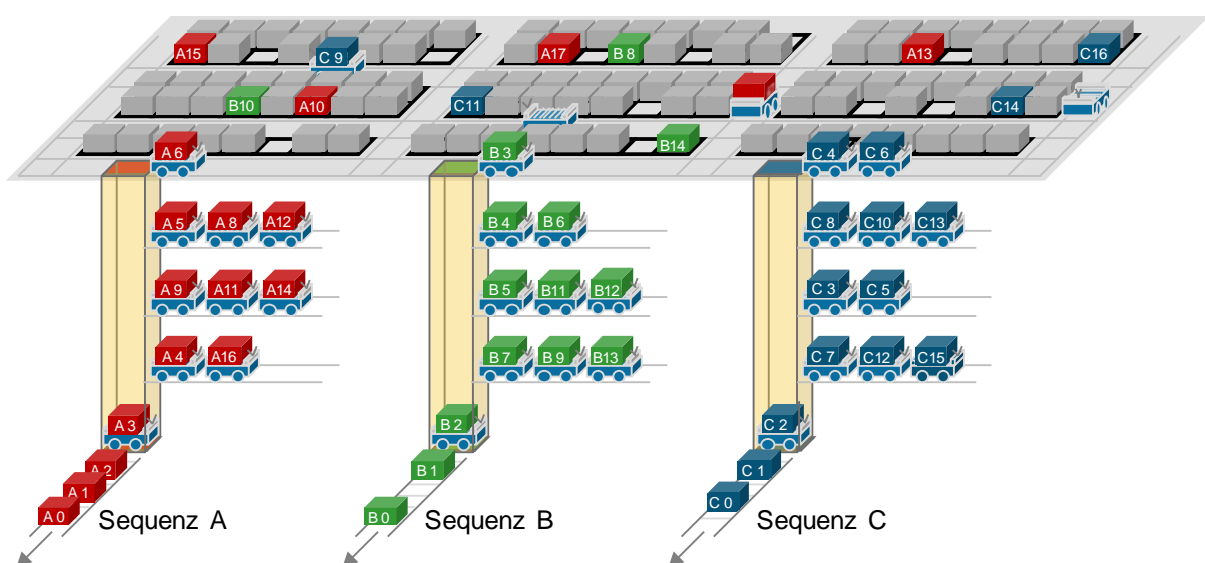


Abbildung 8-12: Jeder Lift versorgt einen eigenen Sequenzstrom.

8.2.2 Modellierung

Jede Ebene des Shuttle-Lagers wird, wie in Unterkapitel 5.1 beschrieben, durch einen gerichteten, zusammenhängenden Graphen modelliert. Dafür werden unterschiedliche Knotenklassen angelegt, welche sich hinsichtlich der Ausrichtungen und Ausdehnungen unterscheiden:

- Lagergassen-knoten: Diese Knoten repräsentieren Layoutabschnitte in den Lagergassen und können lediglich entlang der X-Achse befahren werden.
- Quergassen-knoten: Diese Knoten repräsentieren Layoutabschnitte in den Quergassen und können lediglich entlang der Y-Achse befahren werden.
- Kreuzungskno-ten: Kreuzungsknoten verbinden Lagergassen mit Quergassen und erlauben einen Wechsel der Ausrichtung.
- Pufferknoten: Pufferknoten werden vor und nach den Liften platziert und dienen als Eingangs- und Ausgangspuffer der Lifte.
- Liftknoten: Liftknoten repräsentieren den Lift und können nur befahren werden, sofern der Lift leer ist und sich auf der entsprechenden Ebene befindet.

Jeder Lift ist als Zustandsautomat modelliert, welcher die in Abbildung 8-13 dargestellten Zustände einnehmen kann.

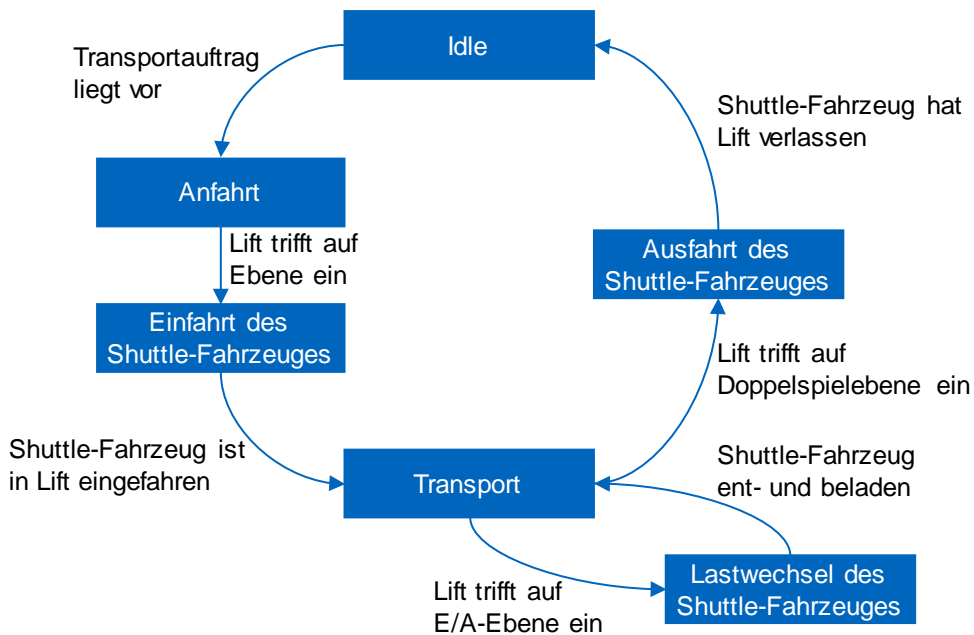


Abbildung 8-13: Modellierung eines Liftes.

Trifft ein Fahrzeug auf einer Ebene ein, erfolgt das Routing ausgehend vom Lift zum Einlagerplatz, zum Auslagerplatz sowie zu dem Lift, über welchen die Lagerebene verlassen wird. Dabei wird als letztes Ziel der Pufferknoten unmittelbar vor dem Lift übergeben. Trifft ein Shuttle-Fahrzeug an diesem Knoten ein, wird ein Transportauftrag für den Lift generiert und in eine Auftragsliste geschrieben. Jeder Lift arbeitet seine Auftragsliste entweder gemäß dem FIFO Prinzip ab oder aber strikt nach aufsteigenden Sequenznummern, sofern die Auslagerung in Sequenz gefordert ist.

Abbildung 8-14 zeigt einen Ausschnitt einer Lagerebene aus dem Simulationsmodell. Zu sehen sind der untere Teil der Lagergassen sowie die Anbindung der Lifte. Gelagerte sowie einzulagernde Lagereinheiten sind braun gefärbt. Auszulagernde Lagereinheiten sind mit der entsprechenden Sequenznummer versehen sowie in der Farbe des Sequenzstroms dargestellt. Zu erkennen ist die Bereitstellung der Lagereinheiten nach aufsteigender Sequenznummer in der roten Sequenz. In der grünen Sequenz warten die Shuttle-Fahrzeuge mit den Sequenznummern 10 und 11 vor dem Kreuzungsknoten bis das Shuttle-Fahrzeug mit der Sequenznummer 9 die Einfahrt zum Lift passiert hat.

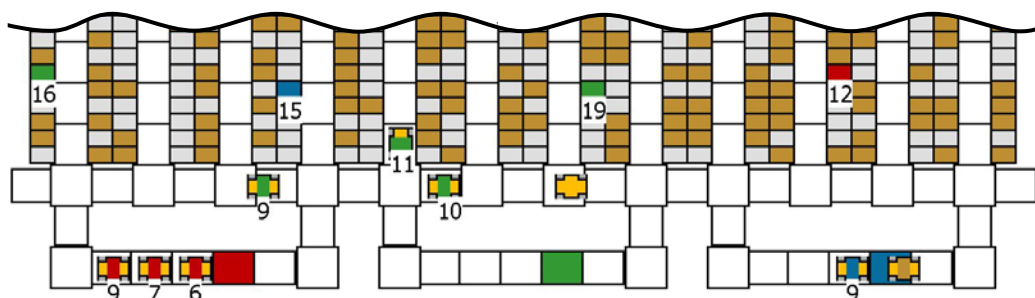


Abbildung 8-14: Screenshot aus dem Simulationsmodell: Abgebildet ist ein Teil einer Lagerebene einschließlich der Anbindung der Lifte.

Das Modell wird initialisiert, indem die vorgegebene Anzahl an Shuttle-Fahrzeugen auf einer virtuellen Lagerebene erzeugt wird und die Shuttle-Fahrzeuge nacheinander von den Liften in das Lager transportiert werden.

8.2.3 Simulationsstudie

Im Rahmen der Simulationsstudie wird die Auswirkung einer Auslagerung in strikter Sequenz auf den erzielbaren Durchsatz analysiert. Dafür wird der Durchsatz in Abhängigkeit der Anzahl der eingesetzten Shuttle-Fahrzeuge ermittelt. Das Modell ist wie in Tabelle 8-3 aufgeführt parametrisiert.

Tabelle 8-3: Modellparameter Shuttle-Lager.

Allgemeine Parameter		Modellspezifische Parameter	
Parameter	Belegung	Parameter	Belegung
L_F^X	0,8m	Lagerfüllgrad	80%
L_F^Y	0,8m	Übergabezeit EA-Punkt	4s
v^X	1m/s	Geschwindigkeit Lift	2m/s
v^Y	1m/s	Beschleunigung Lift	∞
a^X	1m/s ²	Abstand zweier Ebenen	1m
a^Y	1m/s ²		
t_{RW}	3s		
$t_{Lagerspiel}$	5s		
Reservierungsform	Treppe		
Max Segmentlänge	∞		
t_P	0,01s		

Die Anzahl der eingesetzten Shuttle-Fahrzeuge wird, beginnend mit zwei Fahrzeugen, schrittweise um zwei Fahrzeuge erhöht. In einer ersten Simulationsreihe erfolgt die Auslagerung chaotisch. In der anschließenden Simulationsreihe wird hingegen die routingbasierte Sequenzierung aktiviert und die Auftragsstrategie für die Lifte entsprechend umgestellt. Die Simulationszeit beträgt 24 Stunden. Mit jeder Parameterkonfiguration werden fünf Replikationen erstellt. Die sich zum Konfidenzniveau von 95% ergebenden Konfidenzintervalle werden als hinreichend klein erachtet. Über alle 40 durchgeführten Experimente weisen die Konfidenzintervalle im Mittel eine Breite von $\pm 0,28\%$ um den Mittelwert auf, die maximale Breite eines Konfidenzintervalls beträgt $\pm 0,70\%$.

Das Diagramm in Abbildung 8-15 zeigt den erzielten Durchsatz in Abhängigkeit der Anzahl der eingesetzten Shuttle-Fahrzeuge für eine chaotische Auslagerung sowie für eine Auslagerung in Sequenz. Der Durchsatz verläuft in beiden Fällen zunächst nahezu linear, flacht mit zunehmender Anzahl an Shuttle-Fahrzeugen jedoch ab bis der Grenzdurchsatz erreicht ist. Dieser ist durch die Lifte bestimmt: Wenn die Lifte ununterbrochen arbeiten, kann der Durchsatz durch zusätzliche Shuttle-Fahrzeuge nicht mehr gesteigert werden. Der Grenzdurchsatz lässt sich durch einen kombinatorischen Ansatz gemäß dem Vorgehen in der VDI-Richtlinie 2696 [VDI-2696] bestimmen. Die Herleitung für das betrachtete Lager kann dem Anhang entnommen werden.

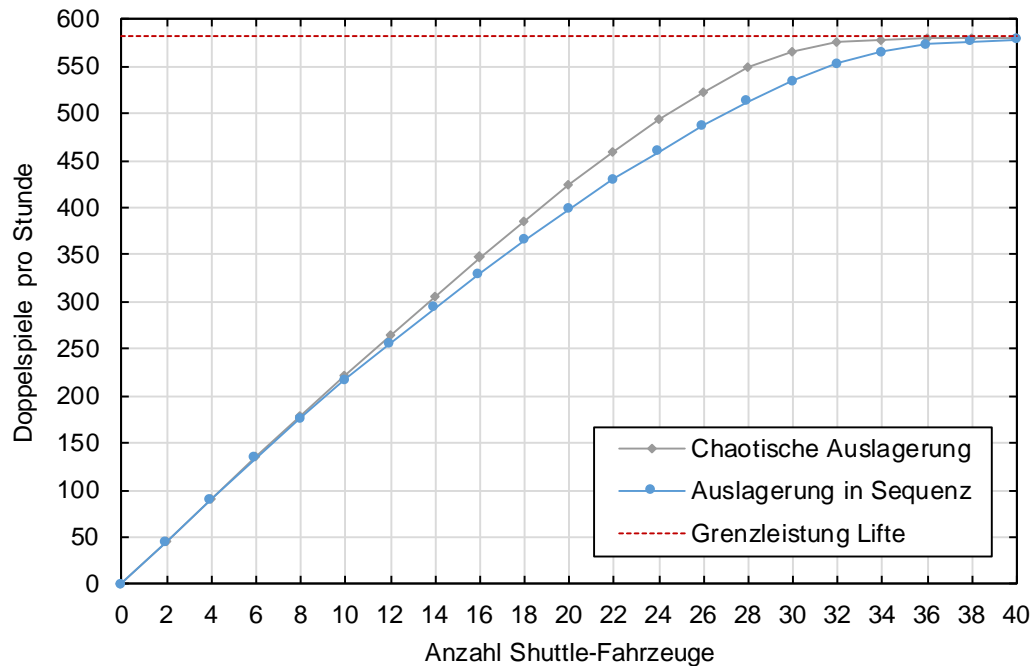


Abbildung 8-15: Durchsatzverläufe für eine chaotische Auslagerung sowie eine Auslagerung in Sequenz.

Das Diagramm in Abbildung 8-16 verdeutlicht den durch die Sequenzierung hervorgerufenen Durchsatzverlust, welcher der Fläche zwischen den beiden Graphen in Abbildung 8-15 entspricht. Der Durchsatzverlust steigt zunächst mit zunehmender Anzahl an Shuttle-Fahrzeugen und erreicht mit 7% das Maximum bei 24 Shuttle-Fahrzeugen.

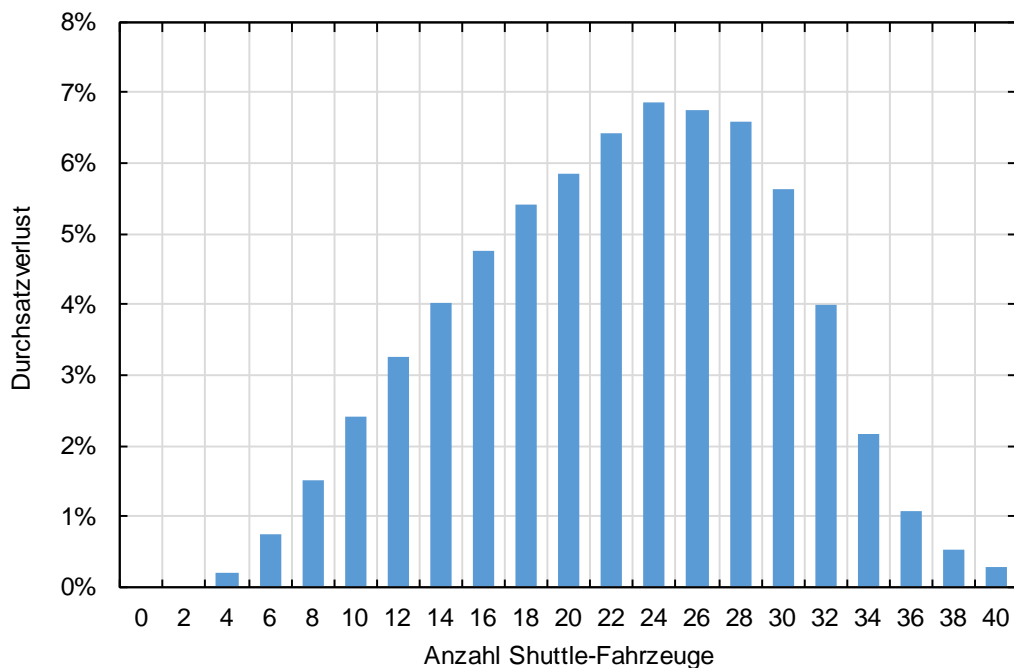


Abbildung 8-16: Durchsatzverlust durch eine Auslagerung in Sequenz gegenüber einer chaotischen Auslagerung.

Anschließend nähert sich der Durchsatz für die chaotische Auslagerung dem Grenzdurchsatz, wodurch sich der Durchsatzverlust wieder verringert bis dieser bei 40 Shuttle-Fahrzeugen nahezu verschwindet. Die Auslagerung in Sequenz erreicht nun ebenfalls die Grenzleistung.

Der Durchsatzverlust kann durch den Einsatz zusätzlicher Shuttle-Fahrzeuge kompensiert werden. Die Anzahl der hierfür notwendigen Shuttle-Fahrzeuge kann dem Diagramm in Abbildung 8-15 entnommen werden. Ist beispielsweise ein Durchsatz von 460 Doppelspielen pro Stunde gefordert, sind dafür 22 Shuttle-Fahrzeuge bei einer chaotischen Auslagerung notwendig. Kommt die Sequenzanforderung hinzu, sind hingegen 24 Shuttle-Fahrzeuge notwendig, um die geforderte Leistung zu erzielen.

Eine weitere Möglichkeit zur Kompensation des Durchsatzverlustes besteht in der Verwendung von Resequenzierpuffern, welche auf der E/A-Ebene vor den Bereitstellorten angeordnet sind. Dadurch wird es den Liften ermöglicht, den Transportauftrag mit einer höheren Sequenznummer auszuführen, falls das Shuttle-Fahrzeug mit der nächsten Sequenznummer noch nicht am Lift eingetroffen ist. Eine nähere Beschreibung sowie eine simulationsbasierte Untersuchung kann [Lie-2018b] entnommen werden.

9 Zielerfüllung der Methodik zur simulationsbasierten Durchsatzanalyse

In diesem Kapitel erfolgt eine Auseinandersetzung mit den erzielten Forschungsergebnissen. Zunächst wird die entwickelte Methodik hinsichtlich der formulierten Anforderungen überprüft (Unterkapitel 9.1). Anschließend erfolgt eine kritische Diskussion der erzielten Forschungsergebnisse (Unterkapitel 9.2).

9.1 Überprüfung der Anforderungen

Das Ziel der vorliegenden Arbeit bestand in der Entwicklung einer Methodik zur simulationsbasierten Durchsatzanalyse von FTF-basierten Kommissioniersystemen.

An die Methodik sowie an die zu entwickelnden Strategien wurden im Unterkapitel 3.3 unterschiedliche Anforderungen gestellt, welche im Folgenden hinsichtlich der Erfüllung geprüft werden.

Für die Evaluierung wurden im Rahmen zweier Fallstudien durch die Anwendung der Methodik zwei unterschiedliche Systemtypen betrachtet (Tabelle 9-1, Nr. 1). Für die Modellerstellung wurde der generische Bausteinkasten herangezogen. Lediglich individuelle Modellkomponenten wie beispielsweise die Pickbereiche in der ersten Fallstudie oder die Shuttle-Lifte in der zweiten Fallstudie mussten modelliert und implementiert werden.

In beiden Systemen bewegen sich die Fahrzeuge auf einem rechtwinkligen Wegenetz. Im ersten System wechseln die Unterfahr-FTF ihre Ausrichtung über einen Mechanismus zum seitlichen Verfahren (siehe Abbildung 5-2, (b)). Im Shuttle-System hingegen führen die Shuttle-Fahrzeuge einen Wechsel der Ausrichtung durch eine 90 Grad Rotation um die eigene Achse aus (Abbildung 5-2, (a)). Darüber hinaus positionieren sich die Fahrzeuge vor den Lagerplätzen um die Lagereinheiten aufzunehmen.

Bei der Modellierung des Layouts wurde beachtet, dass die Shuttle-Fahrzeuge für gewöhnlich länger sind, als die Lagereinheiten breit sind. Dafür musste auch das Routing angepasst werden, da ein Ziel somit nicht mehr zwingend einem einzelnen Knoten entspricht, sondern vielmehr durch eine Halteposition definiert ist, welche durch ein

Fahrzeug eingenommen wird. Dabei kann sich diese Halteposition auch über zwei benachbarte Knoten erstrecken (siehe Abbildung 6-2). Der Detaillierungsgrad des Layouts wird somit als hoch eingestuft (Tabelle 9-1, Nr. 2).

Die Systeme wurden derart modelliert, dass Blockierungseffekte zwischen den Fahrzeugen abgebildet werden. Da sich zu jedem Zeitpunkt auf jedem Knoten nur ein einziges Fahrzeug befinden kann, muss ein Fahrzeug seine Fahrt unterbrechen, sollte der nächste Knoten entlang der Route noch durch ein anderes Fahrzeug belegt sein. Die Fahrzeuge blockieren sich somit gegenseitig (Tabelle 9-1, Nr. 3).

Die Fahrzeuge lassen sich nicht nur über die Geschwindigkeit parametrisieren, sondern auch über die Beschleunigung und Verzögerung (Tabelle 9-1, Nr. 4). Um dies zu ermöglichen war insbesondere eine Weiterentwicklung des zeitfensterbasierten Routings notwendig. Darüber hinaus können auch Verfügbarkeiten für die Fahrzeuge hinterlegt werden, um Störungen einzelner Fahrzeuge zu simulieren und die Auswirkungen auf den Durchsatz zu analysieren (Tabelle 9-1, Nr. 5).

Durch den Einsatz des zeitfensterbasierten Routings werden zeiteffiziente und deadlockfreie Routen berechnet (Tabelle 9-1, Nr. 6). Eine restriktive Ressourcenzuteilung oder die Vorgabe eines lediglich unidirektionalen Verkehrs ist nicht notwendig. Durch die Art und Weise, wie die Routen ausgeführt werden, wird sichergestellt, dass es weder zu Kollisionen noch zu Verklemmungen zwischen den Fahrzeugen kommt, auch wenn sich diese beispielsweise aufgrund von Störungen verspäten und ihre reservierten Zeitfenster nicht einhalten (Tabelle 9-1, Nr. 8). Dieses Verhalten konnte insbesondere in den Simulationsexperimenten für den Vergleich der Strategien für den Umgang mit Störungen in der ersten Fallstudie verifiziert werden.

Darüber hinaus wurde – ebenfalls in der ersten Fallstudie – gezeigt, wie durch den Einsatz entsprechender Strategien die Fahrzeugflotte beim Ausfall eines Fahrzeuges effizient weiterarbeiten kann (Tabelle 9-1, Nr. 9). Somit konnte auch der Vorteil einer hohen Redundanz, welche durch FTF-basierte Kommissioniersysteme ermöglicht wird, bestätigt werden.

In der zweiten Fallstudie wurde die routingbasierte Sequenzierung genutzt, um die Auslagerung in einer vorgegebenen Sequenz sicherzustellen (Tabelle 9-1, Nr. 10). Dabei wurde die sequenzgerechte Bereitstellung an den Liften verifiziert.

Tabelle 9-1 fasst die Anforderungen und deren Erfüllung zusammen. Die im Unterkapitel 3.3 identifizierte Forschungslücke wird mit der vorliegenden Arbeit als geschlossen erachtet.

Tabelle 9-1: Überprüfung der Anforderungen.

Ziele der Methodik zur simulationsbasierten Durchsatzanalyse		Erfüllung
1.	Anwendung zur Analyse unterschiedlicher Systemtypen	✓
2.	Hoher Detaillierungsgrad des Layouts	✓
3.	Berücksichtigung von Blockierungseffekten zwischen den Fahrzeugen	✓
4.	Berücksichtigung von Beschleunigungs- und Verzögerungsvorgängen	✓
5.	Berücksichtigung von Verfügbarkeiten	✓
6.	Ermöglichung eines hohen Durchsatzes	✓
7.	Sicherstellung eines robusten Betriebes (Umgang mit Deadlocks)	✓
8.	Umgang mit Verspätungen der Fahrzeuge	✓
9.	Umgang mit Ausfällen der Fahrzeuge	✓
10.	Sequenzierung	✓

9.2 Kritische Diskussion der Forschungsergebnisse

Die entwickelte Methodik enthält detaillierte Steuerungsstrategien, insbesondere für das Routing und Deadlock-Handling der Fahrzeuge. Bei der Erstellung dieser Strategien wurde darauf geachtet, dass sich diese grundsätzlich auch auf reale Systeme übertragen lassen und nicht nur in einer Simulationsumgebung den reibungslosen Betrieb sicherstellen. In einer ereignisdiskreten Simulation werden die Ereignisse streng sequenziell abgearbeitet, auch wenn diese zur gleichen Simulationszeit stattfinden. Weisen zwei oder mehrere Ereignisse den gleichen Eintrittszeitpunkt auf, wird immer zunächst das Ereignis abgearbeitet, welches als erstes in den Ereigniskalender eingetragen wurde (siehe Abschnitt 2.4.2). In einem realen System hingegen finden mehrere Prozesse simultan statt. Dies ist insbesondere für die Berechnung der Routen sowie deren Ausführung relevant. Wird in der Simulation eine neue Route für ein Fahrzeug berechnet, verändert sich der Systemzustand während dieser Berechnung nicht. Die Ausführung berechneter Routen durch andere Fahrzeuge schreitet nicht voran. In einem realen System hingegen bewegen sich Fahrzeuge während der Routenberechnung fort, auch wenn die Berechnung nur eine kurze Zeitdauer in Anspruch nimmt. Durch die Gestaltung der Zugriffe auf die Datenstruktur der Zeitfenster (siehe Abschnitt 6.3.3) wird jedoch sichergestellt, dass berechnete Routen ausführbar sind, obwohl sich der Systemzustand während der Routenberechnung ändert.

Darüber hinaus wurden unterschiedliche Reservierungsformen diskutiert (siehe Abschnitt 6.1.5), um unterschiedlichen Kommunikationsmöglichkeiten zwischen der übergeordneten Steuerung und den einzelnen Fahrzeugen gerecht zu werden.

Durch die generische Umsetzung, welche sich von der Modellierung über die Strategien bis hin zur Strukturierung im Simulationsmodell erstreckt, lässt sich der Aufwand

bei der Modellerstellung gegenüber einer individuellen Modellerstellung erheblich reduzieren.

Gleichwohl die entwickelte Methodik auf die Planung und Auslegung von FTF-basierten Systemen zielt, ist die Untersuchung der Übertragbarkeit der Steuerungsstrategien auf ein reales System eine ausstehende Aufgabe. Damit einher geht die Validierung der Simulation und die Beantwortung der Frage, inwieweit die erzielten Simulationsergebnisse mit den Durchsätzen übereinstimmen, welche mit dem realen System erreicht wurden.

Wie im Abschnitt 3.1.3 erläutert, ermöglicht das zeitenfensterbasierte Routing die Berechnung der schnellsten Route, wobei auf bestehende Reservierungen bereits gerouteter Fahrzeuge Rücksicht genommen werden muss. Beim Routing erfolgt somit keine globale Optimierung, vielmehr hängt das Gesamtergebnis von der Reihenfolge ab, in welcher die Fahrzeuge geroutet werden. Das nachfolgend dargestellte Beispiel verdeutlicht diesen Sachverhalt. Erfolgt das Routing in der Reihenfolge rotes FTF, blaues FTF, grünes FTF, gelbes FTF (Abbildung 9-1), müssen die FTF jeweils warten, bis das vorhergehende FTF die Knoten in der entgegengesetzten Richtung passiert hat, bevor mit der Ausführung der Route begonnen werden kann.

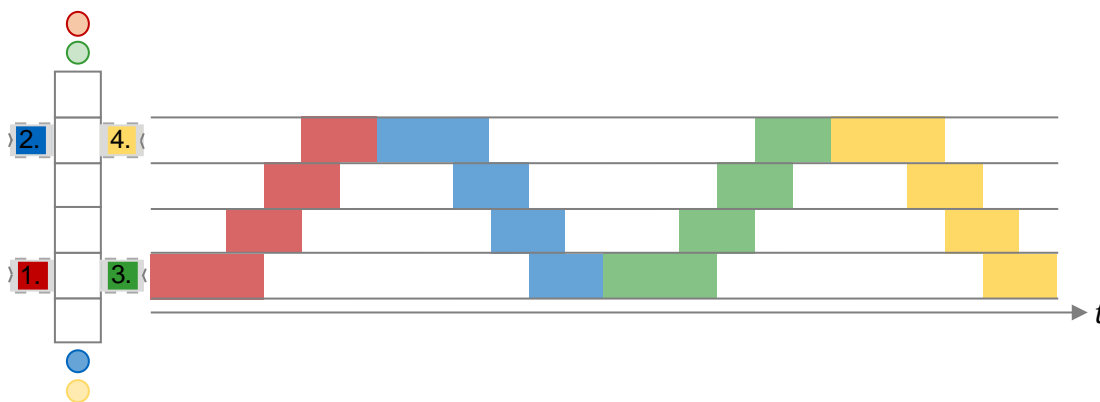


Abbildung 9-1: Vier Fahrzeuge müssen zum gleichen Zeitpunkt geroutet werden: ungünstige Reihenfolge beim Routing.

Erfolgt hingegen das Routing in der Reihenfolge rotes FTF, grünes FTF, blaues FTF, gelbes FTF (Abbildung 9-2), erreicht das letzte der vier Fahrzeug deutlich früher sein Ziel.

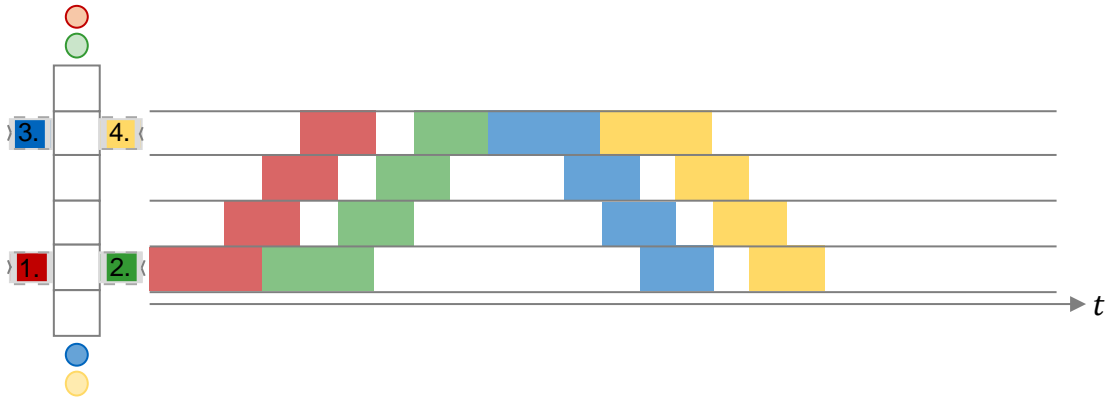


Abbildung 9-2: Vier Fahrzeuge müssen zum gleichen Zeitpunkt geroutet werden: optimale Reihenfolge beim Routing.

Auch wenn Situationen, in welchen mehrere Fahrzeuge zum gleichen Zeitpunkt abfahrbereit sind und geroutet werden müssen, in FTF-basierten Kommissioniersystemen nicht der Regelfall sind, existiert beim Routing – wie aufgezeigt – noch Optimierungspotential. Denkbar ist beispielsweise, dass das Routing für mehrere Fahrzeuge durchgeführt wird, welche innerhalb eines bestimmten Zeitintervalls abfahrbereit sind. Über den Startzeitparameter (siehe Abschnitt 6.1.2) lässt sich ein solches Vorgehen realisieren.

Ein weiterer Ansatzpunkt für eine Optimierung beim Routing besteht in der Veränderung bereits geplanter und reservierter Routen. Für das Routing eines Fahrzeuges müssten dann die reservierten Zeitfenster bereits gerouteter Fahrzeuge nicht als gegeben hingenommen werden, sondern könnten verschoben oder sogar gelöscht werden. Zu beachten ist jedoch, dass die erfolgreiche Neuberechnung von Routen von Fahrzeugen während der Ausführung nicht gewährleistet ist (siehe Abschnitt 6.4.4). Ansätze für eine globale Optimierung beim Routing sind in [Mer-2017] und [Hvě-2018] zu finden.

Das Ziel der vorliegenden Arbeit bestand in der Entwicklung einer Methodik zur simulationsbasierten Durchsatzanalyse von FTF-basierten Kommissioniersystemen, welche sich unabhängig von einer konkreten Simulationsumgebung anwenden lässt. Die Evaluierung erfolgt in der vorliegenden Arbeit anhand eines Bausteinkastens, welcher in der Simulationsumgebung *Plant Simulation* implementiert wurde. Um zu zeigen, dass die Modellierung, die Strategien sowie die Strukturierung des Modells unabhängig von der Simulationsumgebung anwendbar sind, sollte die Umsetzung in einer weiteren ereignisdiskreten Simulationsumgebung erfolgen.

10 Zusammenfassung und Ausblick

In diesem letzten Kapitel wird die vorliegende Arbeit zunächst zusammengefasst (Unterkapitel 10.1). Anschließend wird ein Ausblick gegeben, in welchem zum einen weitere Einsatzmöglichkeiten der entwickelten Methodik aufgezeigt werden (Unterkapitel 10.2 und 10.3), zum andern erläutert wird, wie die entwickelte Methodik erweitert werden kann (Unterkapitel 10.4).

10.1 Zusammenfassung

Eine Herausforderung bei der Planung von FTF-basierten Kommissioniersystemen besteht in der Bestimmung der Anzahl der Fahrzeuge, welche notwendig ist, um den geforderten Durchsatz zu erzielen. Der erzielbare Durchsatz einer FTF-Flotte lässt sich aufgrund der dynamischen Prozesse durch den Einsatz analytischer Methoden nur näherungsweise bestimmen. Zur Absicherung der Planung ist daher die Durchführung einer Simulation notwendig. Die Modellerstellung hierfür ist aufwändig: Die Systeme müssen modelliert und detaillierte Steuerungsstrategien entwickelt und implementiert werden. Insbesondere für das Routing und Deadlock-Handling sind komplexe Strategien notwendig, um die Ablauffähigkeit des Simulationsmodells sicherzustellen.

In der vorliegenden Arbeit wurde eine Methodik zur Simulation von FTF-basierten Kommissioniersystemen entwickelt. Diese umfasst zum einen die Modellierung der Systeme, sodass sich diese in eine Simulationsumgebung überführen lassen. Zum anderen beinhaltet die Methodik detaillierte Steuerungsstrategien, welche einen effizienten und robusten Betrieb ermöglichen und die Ablauffähigkeit der Simulation sicherstellen. Den Kern dieser Strategien bildet das Konzept des zeitfensterbasierten Routings, welches derart erweitert wurde, dass sich Beschleunigungs- und Verzögerungsvorgänge der Fahrzeuge berücksichtigen lassen. Das zeitfensterbasierte Routing dient auch als Grundlage für die Sequenzierung sowie für den Umgang mit Störungen.

Für die Umsetzung der Modellierung sowie der Steuerungsstrategien wurde das Simulationsmodell in einzelne Module unterteilt, welche unterschiedliche Funktionalitäten für die Durchführung der Simulation enthalten. Diese Module wurden einschließlich ihrer Schnittstellen erläutert.

Anschließend wurde die beispielhafte Umsetzung in der Simulationsumgebung *Tecnomatrix Plant Simulation* beschrieben. Die Umsetzung erfolgte in Form eines Bausteinkastens. Dieser kann für die Analyse unterschiedlicher Systeme, welche in den Betrachtungsgegenstand der vorliegenden Arbeit fallen, herangezogen werden.

Der Bausteinkasten wurde abschließend für die Evaluierung genutzt. In der ersten Fallstudie wurde ein Kommissioniersystem betrachtet, welches auf Basis einer Flotte von Unterfahr-FTF betrieben wurde. In dieser Fallstudie wurden zum einen unterschiedliche Layoutvarianten untersucht, zum anderen die im Unterkapitel 6.4 beschriebenen Strategien für den Umgang mit Störungen verglichen. In der zweiten Fallstudie wurde ein Shuttle-System mit gassen- und ebenenwechselnden Shuttle-Fahrzeugen betrachtet und die Durchsatzminderung durch die Auslagerung in Sequenz gegenüber einer chaotischen Auslagerung ermittelt.

10.2 Anwendung der Methodik auf weitere Systemtypen

Die im Rahmen dieser Arbeit entwickelte Methodik kann in weiteren Forschungsarbeiten dazu genutzt werden, um weitere Systemtypen zu analysieren. So wird beispielsweise im Forschungsprojekt *PALSA* (Paletten Aufnehmen, Lagern, Sequenzieren, Abgeben) – Entwicklung eines dynamischen Lagersystems zur Speicherung und sequenzierten Abgabe von Paletten – auf den Simulationsbausteinkasten zurückgegriffen, um ein neuartiges Lagersystem, welches unterschiedliche Lagertechnologien kombiniert, zu dimensionieren. In diesem System werden Paletten in einem Kanallager, welches über Satellitenfahrzeuge bedient wird, gelagert. Regalbediengeräte übernehmen den Vertikaltransport der Paletten zwischen den einzelnen Lagerebenen sowie der untersten Lagerebene, auf welcher gassenwechselnde Shuttle-Fahrzeuge verfahren (Abbildung 10-1). Diese Ebene wird als Kurzzeitpuffer sowie für eine Sequenzierung genutzt. Somit kann zur Steuerung der Shuttle-Fahrzeuge auf die entwickelten Strategien zurückgegriffen werden.

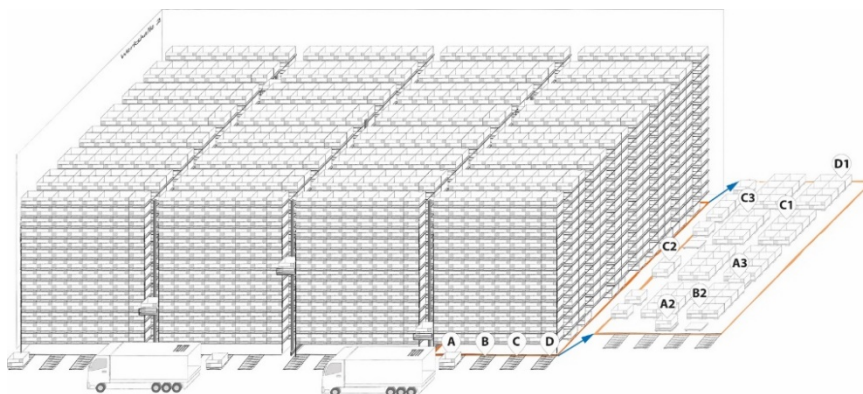


Abbildung 10-1: Kombiniertes Lagersystem [Quelle: Lehrstuhl fml].

Azadeh et al. betrachten ein Kommissioniersystem mit analytischen Methoden, in welchem die Fahrzeuge, die in diesem Kontext als Kommissionierroboter zu bezeichnen sind, sich nicht in der Ebene bewegen, sondern entlang einer Regalwand klettern [Aza-2018]. Da die Roboter dabei ein rechtwinkliges Wegenetz nutzen, kann auch dieser Systemtyp mithilfe der entwickelten Methodik abgebildet und simuliert werden. Abbildung 10-2 zeigt einen Screenshot des erstellten Simulationsmodells.

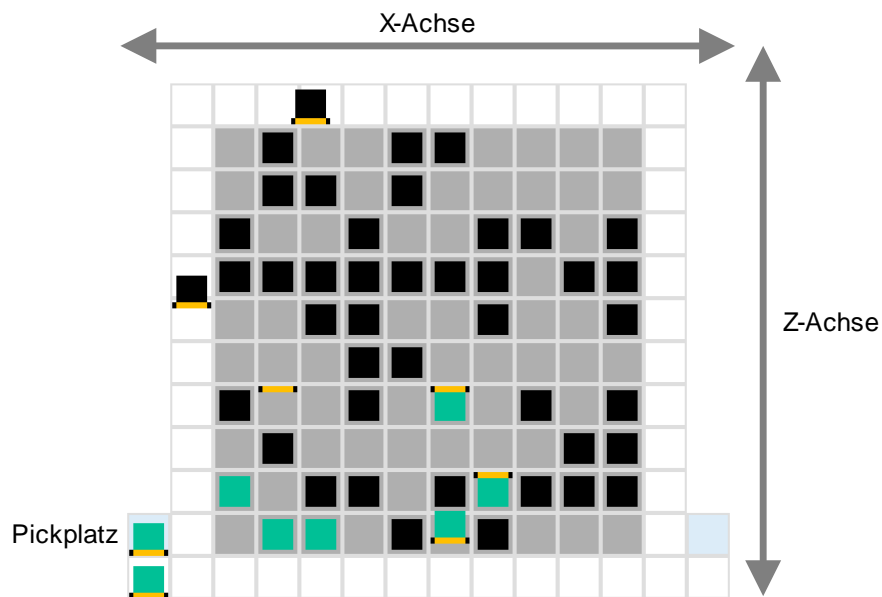


Abbildung 10-2: Regalwand, entlang welcher sich die Roboter vertikal und horizontal bewegen um die Ein- und Auslagerungen durchzuführen.

10.3 Anwendung der Methodik in fachfremden Bereichen

Die Modellierung sowie die entwickelten Steuerungsstrategien können auch in anderen Anwendungsbereichen zum Einsatz kommen, in welchen sich Objekte über ein rechtwinkliges Wegenetz bewegen. Das in dieser Arbeit erweiterte zeitfensterbasierte Routing wurde beispielsweise für die simulationsbasierte Durchsatzanalyse eines automatisierten Laborsystems herangezogen.

Das betrachtete Laborsystem besteht aus mehreren Analysegeräten, welche über eine Transportfläche miteinander verbunden sind. Diese Transportfläche ist schachbrettartig in einzelne Felder unterteilt. Über diese Felder bewegen sich sogenannten Sample Container Carrier (SCC), welche Blut- und Urinproben – die Samples – zwischen den Quellen, den Analysegeräten sowie den Senken des Systems transportieren. Dabei können sich die SCC parallel zu den beiden Hauptachsen in der Ebene von Feld zu Feld bewegen. Sobald ein SCC die Ausführung eines Segments beginnt, kann diese

aus Kommunikationsgründen nicht mehr unterbrochen werden. Daher wurde für das zeitfensterbasierte Routing die Dreiecksform (siehe Abschnitt 6.1.5) verwendet.

Eine Herausforderung bei der Betrachtung des beschriebenen Systems liegt darin, dass die Beschleunigungs- und Verzögerungswerte sowie die maximale Geschwindigkeit nicht präzise angegeben werden können. Je nach Zustand der Transportoberfläche bewegen sich die SCC mit unterschiedlichen Geschwindigkeiten von Feld zu Feld. Erfolgt die Routenplanung mit eher niedrigen Werten, so bewegen sich die SCC während der Ausführung der Routen möglicherweise schneller als geplant über die Transportfläche und müssen zwischen der Ausführung zweier Segmente häufig warten, da die SCC nie ihren Reservierungen voraus sein dürfen (siehe Bedingungen bei der Ausführung von Routen in Abschnitt 6.3.1). Wird hingegen mit hohen Werten geplant, sind die SCC verspätet und die reservierten Zeitfenster stimmen nicht mehr mit den tatsächlichen Zeitfenstern überein. Diese Verspätungen verbreiten sich aufgrund der einzuhaltenden Reihenfolgen an jedem Feld schnell über sämtliche sich im System bewegendende SCC. Dadurch werden möglicherweise Routen gewählt, welche nicht die schnellsten Verbindungen zwischen dem Start- und dem Zielort darstellen. Gegenstand der Simulationsstudie war unter anderem auch die Ermittlung dieser optimalen Planungswerte. [Haf-2019]

10.4 Erweiterung der Methodik

Eine mögliche Erweiterung der Modellierung besteht darin, nicht nur Fahrzeuge oder Roboter abzubilden, welche sich in zwei, sondern in drei Dimensionen fortbewegen können. Dafür ist die Ausrichtung A (siehe Unterkapitel 5.1) um die Z -Achse zu erweitern. In einem entsprechenden Modell hat dann jeder Knoten bis zu sechs Nachfolger, über welche die Ausdehnung der Segmente durch die freien Zeitfenster hindurch zu prüfen ist (Abbildung 10-3.). Die entwickelten Algorithmen können jedoch mit geringen Anpassungen, beispielsweise einer Differenzierung bei der Ausrichtungswechselzeit, übernommen werden.

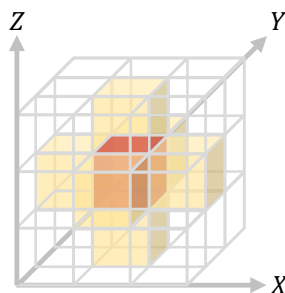


Abbildung 10-3: Wird die Bewegung der Roboter um die dritte Dimension erweitert, hat jeder Knoten bis zu sechs Nachfolger.

Eine andere Erweiterung der Modellierung besteht in der Berücksichtigung unterschiedlicher Geschwindigkeiten wie auch Beschleunigungen der Fahrzeuge, welche je nach Ladezustand variieren. Möglicherweise können unbeladene Fahrzeuge mit einer höheren Geschwindigkeit verfahren als Fahrzeuge, welche eine Transporteinheit befördern. Die unterschiedlichen Geschwindigkeiten und Beschleunigungen müssten sowohl beim Routing, als auch bei der Ausführung der Routen berücksichtigt werden.

Die Methodik kann auch als Ausgangsbasis für weitere Forschungsarbeiten im Bereich der Steuerungsstrategien herangezogen werden. Dabei können beispielsweise Auftragsvergabestrategien im Fokus stehen, aber auch Leerfahrtstrategien, welche im Rahmen der vorliegenden Arbeit nicht betrachtet worden sind.

Zudem können die Strategien für den Umgang mit Störungen wie folgt erweitert werden: Fällt ein Fahrzeug aus und blockiert einen Knoten, wird ein Sperrbereich eingerichtet. Dieser Sperrbereich umfasst neben dem blockierten Knoten weitere Knoten, über welche ein Zugang zu dem gestörten Fahrzeug eingerichtet werden kann, um dieses aus dem Lagersystem zu bergen. Dabei darf der Sperrbereich von anderen Fahrzeugen nicht befahren werden (Abbildung 10-4). [fml-2018]



Abbildung 10-4: Sperrbereich für die Evakuierung eines gestörten Fahrzeuges.

Darüber hinaus lässt sich die Methodik um weitere Aspekte wie beispielsweise die Betrachtung des Energiebedarfs der Fahrzeuge ergänzen. In der vorliegenden Arbeit wurde der Fokus auf die Berechnung von Routen gelegt, auf welchen die Fahrzeuge in möglichst kurzer Zeit zu ihrem Ziel kommen. Eine andere Herangehensweise ist die Berechnung energieeffizienter Routen, auf welchen die Fahrzeuge möglichst kurze Strecken mit möglichst wenigen Zwischenstopps zurücklegen. Eine interessante Forschungsaufgabe besteht in der Entwicklung der entsprechenden Strategien. Auch der Vergleich mit den im Rahmen dieser Arbeit entwickelten Strategien bezüglich des Durchsatzes sowie des Energiebedarfs kann Gegenstand zukünftiger Forschung sein.

Literaturverzeichnis

- [Arn-2002] Arnold, D.; Isermann, H.; Kuhn, A.; Tempelmeier, H.: Handbuch Logistik. Springer-Verlag, Berlin, Heidelberg, New York, 2002.
- [Arn-2009] Arnold, D.; Furmans, K.: Materialfluss in Logistiksystemen. Springer-Verlag, Berlin, Heidelberg, 2009.
- [Aza-2018] Azadeh, K.; Roy, D.; De Koster, R. M. B. M.: Design, Modeling, and Analysis of Vertical Robotic Storage and Retrieval Systems. In: *Transportation Science*, Jg. 53 (2018) Nr. 5, S. 1213–1234.
- [Aza-2019] Azadeh, K.; De Koster, R. M. B. M.; Roy, D.: Robotized and Automated Warehouse Systems: Review and Recent Developments. In: *Transportation Science*, Jg. 53 (2019) Nr. 4, S. 917–945.
- [Bel-1958] Bellman, R.: On a routing problem. In: *Quarterly of Applied Mathematics*, Jg. 16 (1958) Nr. 1, S. 87–90.
- [Ble-2017] Blesing, C.; Lünsch, D.; Stenzel, J.; Korth, B.: Concept of a multi-agent based decentralized production system for the automotive industry. In: *Proceedings of the 15th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS)* (2017)
- [Bus-2004] Busacker, T.: Steigerung der Flughafen-Kapazität durch Modellierung und Optimierung von Flughafen-Boden-Rollverkehr – Ein Beitrag zu einem künftigen Rollführungssystem. Dissertation. Institut für Luft- und Raumfahrt, Technische Universität Berlin, Berlin, 2004.
- [Cof-1971] Coffman, E. G.; Elphick, M.J., Shoshani, A.: System Deadlocks. In: *Computing Surveys*, Jg. 3 (1971) Nr. 2, S. 67–78.
- [De -2004] De Koster, R. M. B. M.; Le-Anh, T.; van der Meer, J. R.: Testing and classifying vehicle dispatching rules in three real-world settings. In: *Journal of Operations Management*, Jg. 22 (2004) Nr. 4, S. 369–386.
- [De -2007] De Koster, R. M. B. M.; Le-Duc, T.; Roodbergen, K. J.: Design and control of warehouse order picking: A literatur review. In: *European Journal of Operational Research*, Jg. 182 (2007) Nr. 2, S. 481–501.

- [Dij-1959] Dijkstra, E. W.: A note on two problems in connexion with graphs. In: Numerische Mathematik, Jg. 1 (1959) Nr. 1, S. 269–271.
- [Dom-2015] Domschke, W.; Drexl, A.; Klein, R.; Scholl, A.: Einführung in Operations Research. Springer-Gabler, Berlin, Heidelberg, 2015.
- [Ebb-2001] Ebben, M.: Logistic Control In Automated Transportation Networks. Dissertation, University of Twente, Twente, 2001.
- [Egb-1984] Egbelu, P. J.; Tanchoco, J.M.A.: Characterization of Automatic Guided Vehicle Dispatching Rules. In: International Journal of Production Research, Jg. 22 (1984) Nr. 3, S. 359–374.
- [Egb-1986] Egbelu, P. J.; Tanchoco, J.M.A.: Potentials for bi-directional guide-path for automated guided vehicle based systems. In: International Journal of Production Research, Jg. 24 (1986) Nr. 5, S. 1075–1097.
- [Egb-1993] Egbelu, P. J.: Positioning of automated guided vehicles in a loop layout to improve response time. In: European Journal of Operational Research, Jg. 71 (1993) Nr. 1, S. 32-44.
- [Ekr-2010] Ekren, B. Y.; Heragu, S. S.; Krishnamurthy, A.; Malmborg, C. J.: Simulation based experimental design to identify factors affecting performance of AVS/RS. In: Computers & Industrial Engineering, Jg. 58 (2010) Nr. 1, S. 175–185.
- [Ekr-2014] Ekren, B. Y.; Heragu, S. S.; Krishnamurthy, A.; Malmborg, C. J.: Matrix-geometric solution for semi-open queuing network model of autonomous vehicle storage and retrieval system. In: Computers & Industrial Engineering, Jg. 68 (2014), S. 78–86.
- [Flo-1962] Floyd, R. W.: Algorithm 97: Shortest path. In: Communications of the ACM, Jg. 5 (1962) Nr. 6, S. 345.
- [For-1956] Ford, L. R., JR.: Network flow theory. The RAND Corporation, Technischer Bericht, 1956.
- [Gan-1998] Ganesharajah, T.; Hall, N. G.; Sriskandarajah, C.: Design and operational issues in AGV-served manufacturing systems. In: Annals of Operations Research, Jg. 76 (1998) Nr. 0, S. 109–154.
- [Gro-1984] Großeschallau, W.: Materialflussrechnung. Springer-Verlag, Berlin, Heidelberg, 1984.

- [Gud-2010] Gudehus, T.: Logistik – Grundlagen Strategien Anwendungen. Springer-Verlag, Heidelberg, Dordrecht, London, New York, 2010.
- [Haf-2019] Hafner, Y.; Heinz, P.; Lienert, T.; Schnarwiler, D.; Taeymans, B. Wenzler, Florian: Method of operating a laboratory sample distribution system, laboratory sample distribution system and laboratory automation system. Patentschrift EP3537159 (A1), 2019.
- [Har-1968] Hart, P. E.; Nilsson, N. J.; Raphael B.: A Formal Basis for the Heuristic Determination of Minimum Cost Paths. In: IEEE Transactions on Systems Science and Cybernetics, Jg. 4 (1968) Nr. 2, S. 100–107.
- [Hat-2001] Hatzack, W.; Nebel, B.: The operational traffic problem: Computational complexity and solutions. In: Proceedings of the 6th European Conference on Planning (ECP-01) (2001)
- [Hom-2008] ten Hompel, M.; Heidenblut, V.: Taschenlexikon Logistik. Springer-Verlag, Berlin, Heidelberg, 2008.
- [Hom-2010] ten Hompel, M.; Sadowsky, V.; Beck, M.: Kommissionierung. Springer-Verlag, Berlin, Heidelberg, 2010.
- [Hua-2015] Huang, G. Q.; Chen, M. Z. Q.; Pan, J.: Robotics in ecommerce logistics. In: HKIE Transactions, Jg. 22 (2015) Nr. 2, S. 68–77.
- [Hvě-2018] Hvězda, J.; Rybecký, T.; Kulich, M.; Přeučil, L.: Context-Aware Route Planning for Automated Warehouses. In: Proceedings of the IEEE 21th International Conference on Intelligent Transportation Systems Conference (ITSC) (2018)
- [Kal-2011] Kalinovic, L.; Petrovic, T.; Bogdan, S.; Bobanc, V.: Modified Banker's algorithm for scheduling in multi-AGV systems. In: Proceedings of the IEEE International Conference on Automation Science and Engineering (2011)
- [Kar-2012] Kartnig, G.; Grösel, B.; Zrnić, N.: Past, state-of-the-art and future of intralogistics in relation to megatrends. In: FME Transactions, Jg. 40 (2012) Nr. 4, S. 193–200.
- [Kim-1991] Kim, C. W.; Tanchoco, J.M.A.: Conflict-free shortest-time bidirectional AGV routeing. In: International Journal of Production Research, Jg. 29 (1991) Nr. 12, S. 2377–2391.

- [Kim-1997] Kim, C. W.; Tanchoco, J.M.A.; Koo Pyung-Hoi: Deadlock Prevention in Manufacturing Systems With AGV Systems: Banker's Algorithm Approach. In: Journal of Manufacturing Science and Engineering, Jg. 119 (1997), S. 849–854.
- [Kle-1996] Klei, C. M.; Kim, J.: AGV dispatching. In: International Journal of Production Research, Jg. 34 (1996) Nr. 1, S. 95–110.
- [Kri-2018] Kriehn, T.; Schloz, F.; Wehking, K.-H.; Fittinghoff, M.: Impact of Class-Based Storage, Sequencing of Retrieval Requests and Warehouse Reorganisation on Throughput of Shuttle-Based Storage and Retrieval Systems. In: FME Transactions, Jg. 46 (2018) Nr. 3, S. 320–329.
- [Kum-2014] Kumar, A.; Roy, D.; Tiwari, M. K.: Optimal partitioning of vertical zones in vehicle-based warehouse systems. In: International Journal of Production Research, Jg. 52 (2014) Nr. 5, S. 1285-135.
- [Lam-2017] Lamballais, T.; Roy, D.; De Koster, R. M. B. M.: Estimating performance in a Robotic Mobile Fulfillment System. In: European Journal of Operational Research, Jg. 256 (2017) Nr. 3, S. 976–990.
- [Law-2015] Law, A. K.: Simulation Modeling and Analysis. McGraw-Hill Education, New York, 2015.
- [Le--2006] Le-Anh, T.; De Koster, R. M. B. M.: A review of design and control of automated guided vehicle systems. In: European Journal of Operational Research, Jg. 171 (2006) Nr. 1, S. 1–23.
- [Leh-2006] Lehmann, M.: Einsatzplanung von Fahrerlosen Transportsystemen in Seehafen-Containerterminals. Dissertation, Technische Universität Berlin, Berlin, Heidelberg, 2006.
- [Ler-2018] Lerher, T.: Aisle changing shuttle carriers in autonomous vehicle storage and retrieval systems. In: International Journal of Production Research, Jg. 56 (2018) Nr. 11, S. 3859–3879.
- [Lie-2016] Lienert, T.; Günthner, W. A.: Development and Simulation-Based Evaluation of an Algorithm for the Retrieval-in-Sequence for Shuttle-Systems. In: Proceedings of of the 28th European Modeling and Simulation Symposium (EMSS) (2016)

- [Lie-2017a] Lienert, T.; Fottner, J.: Entwicklung einer generischen Simulationsmethode für das zeitenfensterbasierte Routing Fahrerloser Transportfahrzeuge. In: Tagungsband zum 13. Fachkolloquium der Wissenschaftlichen Gesellschaft für Technische Logistik (2017)
- [Lie-2017b] Lienert, T.; Fottner, J.: No more Deadlocks - Applying the Time Window Routing Method to Shuttle Systems. In: Proceedings of the 31st European Conference on Modelling and Simulation (2017)
- [Lie-2018a] Lienert, T.; Wenzler, F.; Fottner, J.: Robust Integration of Acceleration and Deceleration Processes into the Time Window Routing Method. In: Proceedings of the 9th International Scientific Symposium on Logistics (2018)
- [Lie-2018b] Lienert, T.; Fottner, J.: Routing-based Sequencing Applied to Shuttle Systems. In: Proceedings of the IEEE 21th International Conference on Intelligent Transportation Systems Conference (ITSC) (2018)
- [Lie-2018c] Lienert, T.; Staab, T.; Ludwig, C.; Fottner, J.: Simulation-based Performance Analysis in Robotic Mobile Fulfillment Systems. In: Proceedings of the 8th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (2018)
- [Lie-2019] Lienert, T.; Stigler, L.; Fottner, J.: Failure-Handling Strategies for Mobile Robots in Automated Warehouses. In: Proceedings of the 33rd European Conference on Modelling and Simulation (2019)
- [Lie-2020] Lienert, T.; Wenzler, F.; Fottner, J.: Simulation-Based Evaluation of Reservation Mechanisms for the Time Window Routing Method. In: Proceedings of the 34th European Conference on Modelling and Simulation (2020)
- [Liu-2001] Liu, F.-H.; Hung, P.-C.: Real-time deadlock-free control strategy for single multi-load automated guided vehicle on a job shop manufacturing system. In: International Journal of Production Research, Jg. 39 (2001) Nr. 7, S. 1323–1342.
- [Mal-2002] Malmberg, C. J.: Conceptualizing tools for autonomous vehicle storage and retrieval systems. In: International Journal of Production Research, Jg. 40 (2002) Nr. 8, S. 1807–1822.

- [Maz-2005] Maza, S.; Castagna, P.: A performance-based structural policy for conflict-free routing of bi-directional automated guided vehicles. In: Computers in Industry, Jg. 56 (2005), S. 719–733.
- [McH-1995] McHaney, R.: Modelling battery constraints in discrete event automated guided vehicle simulations. In: International Journal of Production Research, Jg. 33 (1995) Nr. 11, S. 3023–3040.
- [Mer-2017] Merschformann, M.; Xie, L.; Erdmann, D.: Path planning for robotic mobile fulfillment systems. In: ArXiv e-prints (2017)
- [Mer-2018] Merschformann, M.; Xie, L.; Li, H.: RAWSim-O: A Simulation Framework for Robotic Mobile Fulfillment Systems. In: Logistics Research, Jg. 11 (2018) Nr. 1, S. 1–11.
- [Mor-2007] ter Mors, A. W.; Zutt, J.; Witteveen, C.: Context-aware logistic routing and scheduling. In: Proceedings of the Seventeenth International Conference on Automated Planning and Scheduling (2007)
- [Mor-2009] ter Mors, A. W.: The world according to MARP – Multi-Agent Route Planning. Dissertation, Technische Universiteit Delft, Delft, 2009.
- [Mor-2010] ter Mors, A. W.; Witteveen, C.; Zutt, J.; Kuipers, F. A.: Context-Aware Route Planning. In: Proceedings of the German Conference on Multi-agent System Technologies (MATES) (2010)
- [Mül-1983] Müller, T.: Automated Guided Vehicles. Springer-Verlag, Berlin, Heidelberg, New York, 1983.
- [Nav-2009] Nave, M.: Einführung und Grundlagen. In: Pulverich, M.; Schietinger, J. (Hrsg.): Handbuch Kommissionierung. Heinrich Vogel, München, 2009, S. 16–29.
- [Nil-1971] Nilsson, N. J.: Problem-solving methods in artificial intelligence. McGraw-Hill Pub. Co., New York, 1971.
- [Pen-2014] Penners, L.T.M.E.: Investigating the effect of layout and routing strategy on the performance of the Adapto system. Masterarbeit, Eindhoven University of Technology, Eindhoven, 2014.
- [Rab-2002] Rabe, M.: Modellierung von Layout und Steuerungsregeln für die Materialfluss-Simulation. Dissertation, Technische Universität Berlin, Berlin, 2002.

- [Rab-2008] Rabe, M.; Spieckermann, S.; Wenzel, S.: Verifikation und Validierung für die Simulation in Produktion und Logistik. Springer-Verlag, Berlin, Heidelberg, 2008.
- [Ros-2011] Rose, O.; März, L.: Simulation. In: März, L., et al. (Hrsg.): Simulation und Optimierung in Produktion und Logistik. Springer-Verlag, Berlin, Heidelberg, 2011, S. 13–19.
- [Roy-2015] Roy, D.; Krishnamurthy, A.; Heragu, S. S.; Malmberg, C. J.: Queuing models to analyze dwell-point and cross-aisle location in autonomous vehicle-based warehouse systems. In: European Journal of Operational Research, Jg. 242 (2015) Nr. 1, S. 72–87.
- [Roy-2016] Roy, D.; Krishnamurthy, A.; Heragu, S. S.; Malmberg, C. J.: A simulation framework for studying blocking effects in warehouse systems with autonomous vehicles. In: European Journal of Industrial Engineering, Jg. 10 (2016) Nr. 1, S. 51–80.
- [Sch-1989] Schmidt, F.: Komplexe Fahrerlose Transportsysteme – Fahrzeuganzahl, Investitionsaufwand, Wirtschaftlichkeit. Verlag TÜV Rheinland GmbH, Köln, 1989.
- [Sch-2000] Schrecker, A.: Planung und Steuerung Fahrerloser Transportsysteme – Ansätze zur Unterstützung der Systemgestaltung. Springer-Verlag, Wiesbaden, 2000.
- [Sch-2019] Schloz, F.; Kriehn, T.; Schulz, R.; Fittinghoff, M.: Entwicklung einer KI-basierten Reihenfolgestrategie für Hochregallager mit autonomen Fahrzeugen. In: Tagungsband zum 15. Fachkolloquium der Wissenschaftlichen Gesellschaft für Technische Logistik (2019)
- [Sil-2005] Silver, D.: Cooperative pathfinding. In: Proceedings of the First AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE) (2005)
- [Sin-1994] Sinriech, D.; Tanchoco, J.M.A.: SFT - Segmented Flow Topology. In: Tanchoco, J.M.A. (Hrsg.): Material Flow Systems in Manufacturing. Chapman & Hall, London, Glasgow, Weinheim, New York, Tokyo, Melbourne, Madras, 1994, S. 200–235.
- [Ste-2008] Stenzel, B.: Online Disjoint Vehicle Routing with Application to AGV Routing. Dissertation, Technische Universität Berlin, Berlin, 2008.

- [Tag-1995] Taghaboni-Dutta, F.; Tanchoco, J.M.A.: Comparison of dynamic routing techniques for automated guided vehicle system. In: International Journal of Production Research, Jg. 33 (1995) Nr. 10, S. 2653–2669.
- [Tap-2017] Tappia, E.; Roy, D.; De Koster, R. M. B. M.; Melancini, M.: Modeling, Analysis, and Design Insights for Shuttle-Based Compact Storage Systems. In: Transportation Science, Jg. 51 (2017) Nr. 1, S. 269–295.
- [Ull-2014] Ullrich, G.: Fahrerlose Transportsysteme. Springer-Vieweg, Wiesbaden, 2014.
- [VDI-2510] Verein Deutscher Ingenieure: Fahrerlose Transportsysteme (FTS). VDI-Richtlinie Nr. 2510, 2005.
- [VDI-2696] Verein Deutscher Ingenieure: Shuttle-Systeme für kleine Ladeeinheiten. VDI-Richtlinie Nr. 2696, 2015.
- [VDI-2710] Verein Deutscher Ingenieure: Ganzheitliche Planung von Fahrerlosen Transportsystemen (FTS). VDI-Richtlinie Nr. 2710, 2010.
- [VDI-3590] Verein Deutscher Ingenieure: Kommissioniersysteme. VDI-Richtlinie Nr. 3590, 1994.
- [VDI-3633] Verein Deutscher Ingenieure: Simulation von Logistik-, Materialfluss- und Produktionssystemen. VDI-Richtlinie Nr. 3633, 2014.
- [VDI-4451] Verein Deutscher Ingenieure: Kompatibilität von Fahrerlosen Transportsystemen (FTS) Nr. 4451, 2005.
- [Vis-2006] Vis, I. F.A.: Survey of research in the design and control of automated guided vehicle systems. In: European Journal of Operational Research, Jg. 170 (2006) Nr. 3, S. 677–709.
- [Wen-2008] Wenzel, S.; Weiß, W.; Collisi-Böhmer, S.; Pitsch, H.; Rose, O.: Qualitätskriterien für die Simulation in Produktion und Logistik – Planung und Durchführung von Simulationsstudien. Springer-Verlag, Berlin, Heidelberg, 2008.
- [Wur-2008] Wurman, P. R.; D'Andrea, R.; Mountz, M.: Coordinating Hundreds of Cooperative, Autonomous Vehicles in Warehouses. In: Ai Magazine, Jg. 29 (2008) Nr. 1, S. 9–20.

- [Zou-2018] Zou, B.; Xu, X.; Gong, Y. Y.; De Koster, R. M. B. M.: Evaluating battery charging and swapping strategies in a robotic mobile fulfillment system. In: *European Journal of Operational Research*, Jg. 267 (2018) Nr. 2, S. 733–753.

Durch den Autor betreute studentische Arbeiten

Im Rahmen der durch diese Dissertation aufgegriffenen Forschungsthematik wurden nachfolgende Studienarbeiten erstellt. Die Anfertigung der Arbeiten wurde durch den Autor wissenschaftlich und inhaltlich angeleitet.

Kurzbeleg	Titel	Autor	Art	Abgabe
	Entwicklung und Implementierung eines Routingalgorithmus für Shuttle-Fahrzeuge	Brunner, Andreas	Semesterarbeit	2015
	Development and Implementation of Control Strategies for Shuttle-Systems with more than one Shuttle per Aisle	Pradines, Germain	Masterarbeit	2015
	Entwicklung und Beurteilung von Auftragsvergabestrategien für Shuttle-Systeme	Hafner, Yannic	Semesterarbeit	2016
	Analyse und Bewertung unterschiedlicher Ausprägungen von Shuttle-Systemen	Plapp, Simon Valentin	Bachelorarbeit	2016
	Analyse und Klassifizierung von Routing-Algorithmen für Fahrerlose Transportfahrzeuge	Hölzl, Stefan	Bachelorarbeit	2016
	Entwicklung und Evaluierung eines Modellierungs-konzepts für die Simulation von Shuttle-Systemen	Thomas, Robin	Semesterarbeit	2016
	Entwicklung und Evaluierung von Deadlock-Handling Strategien für Shuttle-Systeme	Kempinski, Błażej Piotr	Semesterarbeit	2016
[fmi-2018]	Entwicklung und simulationsbasierte Evaluierung von Notfallstrategien für Kommissionierroboter	Stigler, Ludwig	Masterarbeit	2018
	Simulationsbasierte Durchsatzanalyse eines innovativen Kommissioniersystems	Meister, Maximilian	Masterarbeit	2018
	Analytische Berechnungsverfahren für die Bestimmung der notwendigen Anzahl an Fahrzeugen in Fahrerlosen Transportsystemen	Knauer, Sophia	Semesterarbeit	2018

Abbildungsverzeichnis

Hauptteil

Abbildung 1-1: Aufbau der Arbeit.	3
Abbildung 2-1: Einordnung von FTS in die Systematik der Fördertechnik [VDI-2510].	5
Abbildung 2-2: Kommissionierroboter TORU der Firma Magazino [Quelle: Magazino GmbH].	9
Abbildung 2-3: Unterfahr-FTF L600 der Firma Grenzebach [Quelle: Grenzebach Maschinenbau GmbH].	10
Abbildung 2-4: Klassifizierung von Shuttle-Systemen nach den Bewegungsachsen der Fahrzeuge.	13
Abbildung 2-5: Gassenwechselndes Shuttle-Fahrzeug OSR Shuttle™ Evo der Firma Knapp [Quelle: KNAPP AG].	13
Abbildung 2-6: Ablauf der ereignisorientierten Simulation.	17
Abbildung 3-1: Unterschiedliche Fahrkurstopologien.	20
Abbildung 3-2: Ausführung eines Transportauftrags.	22
Abbildung 3-3: Klassifizierung von Routing-Algorithmen.	25
Abbildung 3-4: Deadlock zwischen zwei Fahrzeugen.	26
Abbildung 3-5: Deadlock durch einen Kreisschluss von vier Fahrzeugen.	27
Abbildung 3-6: Einflussfaktoren auf die notwendige Flottengröße.	31
Abbildung 3-7: Beispielhafte Durchsatzverläufe: Der Durchsatz steigt zunächst nahezu linear an, flacht dann ab und stagniert (graue Kurve) oder sinkt dramatisch (blaue Kurve).	32
Abbildung 4-1: Aufbau der Methodik.	44
Abbildung 5-1: Modellierung eines Layouts mit unidirektionalen Lagergassen und bidirektionalen Quergassen durch entsprechende Belegung der Knotenattribute [Lie-2017a].	48
Abbildung 5-2: Wechsel der Ausrichtung von Fahrzeugen beider Kategorien [Lie-2017a].	51
Abbildung 5-3: Geschwindigkeits-Zeit-Verlauf bei einer Trapezfahrt (links) und Dreiecksfahrt (rechts).	54
Abbildung 5-4: Funktion $s(t, l)$ und zugehörige Umkehrfunktion $t(s, l)$ für eine Trapezfahrt über die Gesamtstrecke l .	56
Abbildung 5-5: Funktion $s(t, l)$ und zugehörige Umkehrfunktion $t(s, l)$ für eine Dreiecksfahrt über die Gesamtstrecke l .	57

Abbildung 5-6: Fallunterscheidung: Im ersten Fall (links) erreicht das Fahrzeug die maximale Geschwindigkeit, bevor die Verzögerung eingeleitet werden muss, im zweiten Fall (rechts) hingegen nicht.	59
Abbildung 5-7: Zeitdauer bis zum Einleiten der Verzögerung für unterschiedliche Fälle.	60
Abbildung 5-8: Zeitstempel und zugehörige qualitative Geschwindigkeits-Zeit-Verläufe für das Anfahren eines Knotens i (links) und das Durchfahren eines Knotens i (rechts).	61
Abbildung 6-1: Reservierte und freie Zeitfenster auf einem Knoten i [Lie-2017a].	63
Abbildung 6-2: Haltepositionen für den Zugriff auf unterschiedliche Lagerplätze und einen Pickplatz.	65
Abbildung 6-3: Um auf den Lagerplatz 2 zuzugreifen, wird nur der Knoten i belegt. Es wird vereinfachend angenommen, dass der Lagerplatz 2 von der zentrierten Ankunftsposition erreicht werden kann. Das Verfahren auf einem Knoten wird nicht betrachtet.	66
Abbildung 6-4: Nach dem Erreichen der Halteposition für den Zugriff auf den Lagerplatz 9 folgt eine Fahrbewegung, bis die Ankunftsposition auf dem Knoten j erreicht ist.	67
Abbildung 6-5: Unterschiedliche Mengen der Zeitfenster.	68
Abbildung 6-6: Zeitfensterbasiertes Routing (übergeordneter Prozess).	70
Abbildung 6-7: Ungerichteter Graph.	73
Abbildung 6-8: Routingprozess zum Erreichen mehrerer Ziele.	74
Abbildung 6-9: Ausdehnung linearer Fahrbewegungen über alle Nachbarknoten (gelb).	75
Abbildung 6-10: Segment.	76
Abbildung 6-11: Segment mit zeitlichen Puffern (hellrot).	77
Abbildung 6-12: Bestimmung des Beginns der Reservierung.	78
Abbildung 6-13: Der Knoten k ist aus dem gewählten Zeitfenster $f_{i,l}$ durch zwei unterschiedliche Segmente zu erreichen.	79
Abbildung 6-14: Erstellung eines Segmentes.	79
Abbildung 6-15: Ausgangslage für die Erstellung eines Segmentes.	80
Abbildung 6-16: Test auf Erreichbarkeit eines freien Zeitfensters $f_{k,p}$.	81
Abbildung 6-17: Das freie Zeitfenster $f_{k,p}$ ist anfahrbar. Das Segment kann verlängert werden.	83
Abbildung 6-18: Das freie Zeitfenster $f_{k,p}$ ist nicht anfahrbar, jedoch durchfahrbar. Das Segment kann verlängert werden.	85
Abbildung 6-19: Segment, bei welchem das nächste Ziel erreicht wird.	86

Abbildung 6-20: Segment mit der Dreiecksform.	88
Abbildung 6-21: Test auf Erreichbarkeit eines freien Zeitfensters fk, p bei der Dreiecksform (bzw. Blockform).	89
Abbildung 6-22: Segment mit Blockform.	90
Abbildung 6-23: Die Sequenznummer $S_n - 1$ erreicht den Sequenzpunkt aus der entgegengesetzten Richtung.	92
Abbildung 6-24: In dem erreichbaren Zeitfenster fk, p kann das Ziel nicht erreicht werden, da die Sequenz verletzt ist.	92
Abbildung 6-25: Das später erreichbare Zeitfenster fk, q hingegen ermöglicht das Erreichen des nächsten Ziels.	92
Abbildung 6-26: Die Sequenznummer $S_n - 1$ erreicht den Sequenzpunkt aus der gleichen Richtung.	93
Abbildung 6-27: In dem erreichbaren Zeitfenster fk, p kann das Ziel abermals nicht erreicht werden, da die Sequenz verletzt ist.	93
Abbildung 6-28: Durch eine Ausweichbewegung auf den Nachbarknoten ist das freie Zeitfenster fq, k erreichbar, in welchem das nächste Ziel erreicht werden kann.	93
Abbildung 6-29: Das Einhalten der Reihenfolge garantiert Deadlockfreiheit.	95
Abbildung 6-30: Eine verfrühte Abfahrt verhindert die korrekte Ausführung eines geplanten Segmentes.	95
Abbildung 6-31: Ein FTF befährt einen Knoten zu früh.	96
Abbildung 6-32: Ausführung einer berechneten Route.	97
Abbildung 6-33: Beanspruchung von Knoten.	98
Abbildung 6-34: Ein geplantes Segment wird verkürzt.	99
Abbildung 6-35: Zugriffe auf die Datenstruktur der Zeitfenster.	100
Abbildung 6-36: FTF anhalten und Systemneustart vorbereiten.	102
Abbildung 6-37: Neustart angehaltener Fahrzeuge.	103
Abbildung 6-38: Das rote und das blaue FTF müssen neu geroutet werden.	104
Abbildung 6-39: Ein gestörtes FTF (grün) verhindert das Routing eines angehaltenen FTF (rot), wodurch ein weiteres FTF (blau) blockiert wird.	105
Abbildung 6-40: Entstörtes FTF neustarten.	105
Abbildung 6-41: Auflösung eines Deadlocks.	106
Abbildung 6-42: Durch die Störung entsteht ein Deadlock, welcher aufgelöst wird.	107
Abbildung 6-43: Fahrt abbrechen.	108
Abbildung 6-44: FTF umleiten.	109
Abbildung 7-1: Schematischer Aufbau des Simulationsmodells.	111

Abbildung 7-2: Simulationsmodell mit den fünf Modulen und dem Lagerbereich, welcher aus einer Lagerebene besteht.	118
Abbildung 7-3: Implementierung des Layoutmoduls.	119
Abbildung 7-4: Implementierung des Parametermoduls.	120
Abbildung 7-5: Implementierung des Routingmoduls.	121
Abbildung 7-6: Implementierung des Auftragmoduls.	122
Abbildung 7-7: Implementierung des Auswertungsmoduls.	122
Abbildung 7-8: Implementierung eines Knotens.	123
Abbildung 7-9: Implementierte Methoden zur Ausführung von berechneten Routen.	125
Abbildung 7-10: Beispielhafter Layoutgenerator.	126
Abbildung 7-11: Visualisierung des Routingprozesses.	127
Abbildung 7-12: Visualisierung der Route und der beanspruchten Knoten bei der Ausführung.	128
Abbildung 7-13: Visualisierung der reservierten Zeitfenster bei den unterschiedlichen Reservierungsformen Treppe, Dreieck und Block.	128
Abbildung 7-14: Visualisierung mehrerer Routen für unterschiedliche Fahrzeuge.	129
Abbildung 8-1: Kommissionierlager mit Unterfahr-FTF.	132
Abbildung 8-2: Zusammensetzung der unterschiedlichen Spiele, beginnend mit der Aufnahme eines einzulagernden Regals in der Pickzone.	133
Abbildung 8-3: Screenshot aus dem Simulationsmodell (um 90 Grad gedreht): Abgebildet ist der gesamte Lagerbereich einschließlich Pickzone (blau) und Leergutzone (grün).	134
Abbildung 8-4: Untersuchte Layoutvarianten.	135
Abbildung 8-5: Durchsatzverläufe für die betrachteten Layoutvarianten.	137
Abbildung 8-6: Zeitanteile der FTF für das Layout 2.	138
Abbildung 8-7: Heatmap für die Anzahl der Knotenbelegungen für das Layout 2.	139
Abbildung 8-8: Durchsatzverläufe für die betrachteten Layoutvarianten einschließlich Varianten mit Quergassen.	139
Abbildung 8-9: Durchsatzverläufe für unterschiedliche Strategien für den Umgang mit Störungen für das Layout 2 mit Quergassen.	140
Abbildung 8-10: Durchsatzverlust beim Anwenden der Strategie Umleiten gegenüber dem Idealbetrieb.	141
Abbildung 8-11: Shuttle-System der Variante 4.	142
Abbildung 8-12: Jeder Lift versorgt einen eigenen Sequenzstrom.	143

Abbildung 8-13: Modellierung eines Liftes.	144
Abbildung 8-14: Screenshot aus dem Simulationsmodell: Abgebildet ist ein Teil einer Lagerebene einschließlich der Anbindung der Lifte.	145
Abbildung 8-15: Durchsatzverläufe für eine chaotische Auslagerung sowie eine Auslagerung in Sequenz.	147
Abbildung 8-16: Durchsatzverlust durch eine Auslagerung in Sequenz gegenüber einer chaotischen Auslagerung.	147
Abbildung 9-1: Vier Fahrzeuge müssen zum gleichen Zeitpunkt geroutet werden: ungünstige Reihenfolge beim Routing.	152
Abbildung 9-2: Vier Fahrzeuge müssen zum gleichen Zeitpunkt geroutet werden: optimale Reihenfolge beim Routing.	153
Abbildung 10-1: Kombiniertes Lagersystem [Quelle: Lehrstuhl fm].	156
Abbildung 10-2: Regalwand, entlang welcher sich die Roboter vertikal und horizontal bewegen um die Ein- und Auslagerungen durchzuführen.	157
Abbildung 10-3: Wird die Bewegung der Roboter um die dritte Dimension erweitert, hat jeder Knoten bis zu sechs Nachfolger.	158
Abbildung 10-4: Sperrbereich für die Evakuierung eines gestörten Fahrzeuges.	159

Anhang

Abbildung A1-1: Spiel eines Fahrzeuglifts.	A1-1
--------------------------------------------	------

Tabellenverzeichnis

Hauptteil

Tabelle 3-1:	Arbeiten zur simulationsbasierten Durchsatzanalyse.	39
Tabelle 5-1:	Adjazenzlisten für die Knoten $K_{1,4}$ und $K_{1,5}$ aus dem Layout in Abbildung 5-1 ohne Berücksichtigung der Ausrichtung.	49
Tabelle 5-2:	Adjazenzlisten für die Knoten $K_{1,4}$ und $K_{1,5}$ aus Abbildung 5-1 mit Berücksichtigung der Ausrichtung.	50
Tabelle 6-1:	Beispielhafte Liste mit Zielen für einen Routingprozess.	65
Tabelle 6-2:	Initiales Segment.	80
Tabelle 6-3:	Um das anfahrbare Zeitfenster $f_{k,p}$ erweitertes Segment.	83
Tabelle 6-4:	Um das durchfahrbare Zeitfenster $f_{k,p}$ erweiterte Segment.	85
Tabelle 7-1:	An ein Fahrzeug übergebene Route.	124
Tabelle 8-1:	Modellparameter Unterfahr-FTF.	136
Tabelle 8-2:	Mittlere und maximale Breite der Konfidenzintervalle.	140
Tabelle 8-3:	Modellparameter Shuttle-Lager.	146
Tabelle 9-1:	Überprüfung der Anforderungen.	151

Anhang

Tabelle A1-1:	Matrix der möglichen Ebenenwechsel.	A-3
---------------	-------------------------------------	-----

Anhang A

A.1 Spielzeitberechnung der Lifte

Um die Grenzleistung λ der Lifte zu bestimmen, wird das in der VDI-Richtlinie 2692 beschriebene Verfahren für die Bestimmung der mittleren Spielzeit für einen Fahrzeuglift t_{Lift} angewendet, wobei sämtliche Positionier-, Schalt- und Kontrollzeiten vernachlässigt werden [VDI-2696]. Wie im Simulationsmodell wird vorausgesetzt, dass alle Lagerebenen mit der gleichen Wahrscheinlichkeit angefahren werden. Das Spiel eines Liftes setzt sich aus den in Abbildung A1-1 dargestellten Zeitanteilen zusammen.

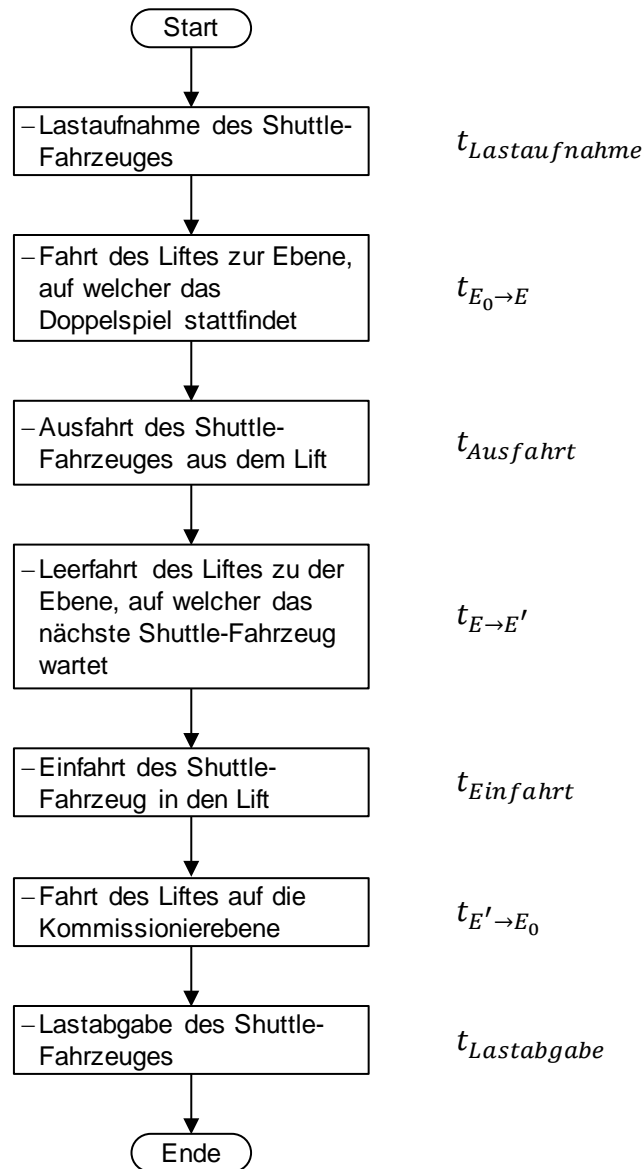


Abbildung A1-1: Spiel eines Fahrzeuglifts.

Das Spiel beginnt mit der Lastaufnahme einer Lagereinheit durch das Shuttle-Fahrzeug, welches sich im Lift auf der Kommissionierebene befindet. Anschließend folgt die Fahrt im Lift auf die Lagerebene E , auf welcher das Doppelspiel stattfindet. Nachdem das Shuttle-Fahrzeug den Lift verlassen hat, folgt eine Leerfahrt des Liftes zu der Lagerebene E' , auf welcher das nächste Shuttle-Fahrzeug wartet. Nach der Ankunft fährt das Shuttle in den Lift und es folgt eine Liftfahrt auf die Kommissionierebene, auf welcher die Lastabgabe durch das Shuttle erfolgt.

Die Zeitdauern für die Lastaufnahme und -abgabe eines Shuttle-Fahrzeuges im Lift sind Parameter des Simulationsmodells:

$$t_{\text{Lastaufnahme}} = t_{\text{Lastabgabe}} = 4s \quad (\text{A1-1})$$

Für die Einfahrt in den Lift, muss das Shuttle im Modell eine Strecke der Länge $1m$ zurücklegen. Mit den entsprechenden Werten für die Beschleunigung, maximale Geschwindigkeit und Verzögerung ergibt sich die Dauer für die Einfahrt in den Lift zu:

$$t_{\text{Einfahrt}} = 2s \quad (\text{A1-2})$$

Für die Bestimmung der Ausfahrtsdauer wird eine Fahrt über eine unendlich lange Distanz herangezogen, sodass Verzögerungsvorgänge nicht betrachtet werden. Das Shuttle-Fahrzeug hat den Lift im Modell nach einer Fahrstrecke s von $0,9m$ vollständig verlassen (siehe Formel (5-10)):

$$t_{\text{Ausfahrt}} = t(s, \infty) = 1,4s \quad (\text{A1-3})$$

Für die Bestimmung der verbleibenden Zeitdauern werden zunächst folgende Variablen definiert, um die durchschnittlichen Liftfahrzeiten zu bestimmen.

- n : Anzahl der Lagerebenen ($n = 10$).
- m : Anzahl der zu überwindenden Lagerebenen.
- ω_m : Wahrscheinlichkeit für einen Ebenenwechsel über m Ebenen.
- $t_{\text{Lift}}(m)$: Fahrzeit, die ein Lift benötigt, um m Lagerebenen zu überwinden

Für die Fahrt von der Kommissionierebene auf die Lagerebene E sowie von der Lagerebene E' auf die Kommissionierebene gilt:

$$\omega_m = \frac{1}{n} \text{ für } m = 1, \dots, n \quad (\text{A1-4})$$

$$t_{E_0 \rightarrow E} = t_{E' \rightarrow E_0} = \sum_{m=1}^n \omega_m t_{Lift}(m) = \frac{1}{n} \sum_{m=1}^n t_{Lift}(m) \quad (\text{A1-5})$$

$$= \frac{1}{10} (0,5 + 1 + 1,5 + 2 + 2,5 + 3 + 3,5 + 4 + 4,5 + 5)s = 2,75s$$

Für die Berechnung der Dauer der Leerfahrt des Liftes ist zu überlegen, wie häufig ein Ebenenwechsel über m Ebenen erfolgt. Die Anzahl der Ebenen, welche zwischen der Lagerebene E , auf welcher das Shuttle-Fahrzeug den Lift verlässt, sowie der Lagerebene E' , auf welcher das nächste Shuttle-Fahrzeug wartet, zu überwinden sind, können durch abzählen bestimmt und der Tabelle A1- 1 entnommen werden.

Tabelle A1- 1: Matrix der möglichen Ebenenwechsel.

m		Ebene E'									
		1	2	3	4	5	6	7	8	9	10
Ebene E	1	0	1	2	3	4	5	6	7	8	9
	2	1	0	1	2	3	4	5	6	7	8
	3	2	1	0	1	2	3	4	5	6	7
	4	3	2	1	0	1	2	3	4	5	6
	5	4	3	2	1	0	1	2	3	4	5
	6	5	4	3	2	1	0	1	2	3	4
	7	6	5	4	3	2	1	0	1	2	3
	8	7	6	5	4	3	2	1	0	1	2
	9	8	7	6	5	4	3	2	1	0	1
	10	9	8	7	6	5	4	3	2	1	0

Für die Wahrscheinlichkeit eines Ebenenwechsels über m Ebenen ergibt sich der folgende Zusammenhang:

$$\omega_m = \frac{2(n-m)}{n^2}, m = 1, \dots, n-m \quad (\text{A1-6})$$

Für die Dauer der Leerfahrt gilt somit im Allgemeinen:

$$t_{E \rightarrow E'} = \sum_{m=1}^{n-1} \omega_m t_{Lift}(m) = \sum_{m=1}^{n-1} \frac{2(n-m)}{n^2} t_{Lift}(m) \quad (\text{A1-7})$$

Und für das betrachtete Lagersystem:

$$t_{E \rightarrow E'} = \frac{18}{100} * 0,5s + \frac{14}{100} * 1s + \frac{12}{100} * 1,5s + \frac{10}{100} * 2s + \frac{8}{100} * 2,5s$$

(A1-8)

$$+ \frac{6}{100} * 3s + \frac{4}{100} * 3,5s + \frac{2}{100} * 4s = 1,65s$$

Nun lässt sich die mittlere Spielzeit bestimmen:

$$t_{Lift} = 4s + 2,75s + 1,4s + 1,65s + 2s + 2,75s + 4s = 18,55s$$

(A1-9)

Da im betrachteten Lager drei Lifte vorhanden sind, ergibt sich der folgende Grenzdurchsatz:

$$\lambda = 3 * \frac{3600 \frac{s}{h}}{18,55s} = 582,21 \frac{1}{h}$$

(A1-10)