

Lösungen für Anwendungsfälle der Fahrzeugeinsatzplanung im öffentlichen Personennahverkehr

DISSERTATION

zur Erlangung des akademischen Grades
doctor rerum politicarum (dr. rer. pol.)
im Fach Wirtschaftsinformatik

eingereicht an der
Fakultät für Wirtschaftswissenschaften der
Universität Paderborn
Paderborn

von Dipl.-Wirt.-Inf. Stefan Bunte

Dekan der Fakultät für Wirtschaftswissenschaften:
Prof. Dr. Peter F. E. Sloane

Gutachter:

1. Jun.-Prof. Dr. Natalia Kliewer
2. Prof. Dr. Ludwig Nastansky

Paderborn, im März 2009

Danksagungen

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Decision Support & Operations Research Lab (DS&OR Lab) der Universität Paderborn. Ich möchte mich an dieser Stelle bei allen bedanken, die zum Gelingen dieser Arbeit beigetragen haben.

Zuerst gilt mein besonderer Dank meinen wissenschaftlichen Betreuern Jun.-Prof. Dr. Natalia Kliewer und Prof. Dr. Leena Suhl, die mir die Tätigkeit als wissenschaftlicher Mitarbeiter ermöglicht haben und durch ihre wissenschaftlichen und sozialen Kompetenzen eine konstruktive, offene und freundliche Arbeitsatmosphäre geschaffen haben.

Ich möchte mich bei allen Kollegen am Lehrstuhl für die gute Zusammenarbeit bedanken. Aufgrund der außerordentlich kompetenten und freundschaftlichen Unterstützung hat mir die Arbeit am DS&OR Lab viel Freude gemacht. Insbesondere danke ich meinen Bürokollegen Philipp Christophel, Viktor Dück, Ingmar Steinzen und Jörg Wiese für die angenehme Zeit und die interessanten Diskussionen.

Des Weiteren möchte ich allen danken, die zugehört, kritisiert, hinterfragt oder mich in anderer Form unterstützt haben.

Mein besonderer Dank gilt meinen Freunden und meiner Familie, die meiner Arbeit viel Verständnis und Geduld entgegengebracht haben. Insbesondere danke ich meinen Eltern dafür, dass sie mir mein Studium ermöglicht haben und mich immer bedingungslos unterstützt haben.

Schließlich gilt mein wichtigster Dank Nicole, die mich immer begleitet hat und mit der ich alle meine Freuden, Sorgen und Wünsche teilen konnte.

Stefan Bunte
Paderborn, März 2009

Inhaltsverzeichnis

1	Einleitung	1
2	Umlaufplanung im öffentlichen Personennahverkehr	5
2.1	Fahrzeugeinsatzplanung im Planungsprozess des ÖPNV	5
2.2	Anwendungsfälle der Umlaufplanung	9
2.2.1	Operative Planung	10
2.2.2	Taktische Planung	12
2.2.3	Strategische Planung	13
2.2.4	Integrierte Planung	13
2.3	Problemstellungen	14
2.3.1	Fahrzeugumlaufplanung	14
2.3.2	Fahrzeugumlaufplanung mit einem Depot	16
2.3.3	Umlaufplanung mit mehreren Depots und Fahrzeugtypen	17
2.3.4	Umlaufplanung mit Fahrzeugtypen	18
2.3.5	Umlaufplanung mit Zeitfenstern	20
2.3.6	Umlaufplanung mit Routenrestriktionen	21
2.3.7	Weitere praxisrelevante Anforderungen	22
3	Grundlegende Optimierungstechniken	23
3.1	Mathematische Optimierung	23
3.2	Column Generation	28
3.3	Lagrange Relaxation	31
3.4	Meta-Heuristiken	33
3.4.1	Lokale Suchverfahren	34
3.4.2	Evolutionäre Algorithmen	37
4	Umlaufplanungsprobleme: Stand der Forschung	39
4.1	Busumlaufplanung mit einem Depot	40
4.2	Busumlaufplanung mit mehreren Depots	47
4.2.1	Einfache Flussmodelle	47
4.2.2	Mehrgüter Modelle	51
4.2.3	Set Partitioning Modelle	54

4.2.4	Modellunabhängige Ansätze	56
4.2.5	Eigenschaften gelöster Instanzen	57
4.3	Busumlaufplanung mit mehreren Depots und Fahrzeugtypgruppen	62
4.4	Busumlaufplanung mit Zeitfenstern	67
4.5	Busumlaufplanung mit Routenrestriktionen	71
4.6	Planungssysteme mit Komponenten zur Busumlaufoptimierung . .	74
4.7	Veröffentlichungen mit Übersichten und Klassifizierungen	77
5	Handlungsbedarf und Zielsetzung	79
6	Probleminstanzen und Eigenschaften	83
6.1	Ursprung der Instanzen	84
6.1.1	Reale Instanzen	84
6.1.2	Künstliche Instanzen	87
6.2	Modelleigenschaften und Kennzahlen	87
6.2.1	Qualität der LP-Relaxation	88
6.2.2	Kennzahl zur Modellauswahl	89
6.2.3	Eigenschaften der Mehrgüter Flussmodelle	102
6.2.4	Abschätzung der Komplexität	105
7	Lösungsmethodik für Busumlaufplanungsprobleme	111
7.1	Heuristischer Lösungsansatz	113
7.2	Bestimmung unterer Schranken	119
7.2.1	Untere Schranken durch Modellaggregation	120
7.2.2	Untere Schranken durch Lagrange Relaxationen	121
7.2.3	Warmstart der Simplexmethoden	127
7.3	Lösungen durch Kantengenerierung	129
7.3.1	Initialisierung	133
7.3.2	Master Problem	135
7.3.3	Pricing Problem	136
7.3.4	Ganzzahlige Lösungen	139
7.4	Systemintegration und Preprocessing	141
7.5	Numerische Ergebnisse	145
7.5.1	Optimierung mit Standardoptimierungssoftware	146
7.5.2	Initialisierungsphase	150
7.5.3	Verbesserungsphase	152
7.5.4	Zusammenfassung und Bewertung	161
8	Optimierungsmethoden für die Umlaufplanung mit Zeitfenstern	165
8.1	Operative Umlaufplanung mit Zeitfenstern	165

8.1.1	Organisatorische Aspekte und Anforderungen	165
8.1.2	Modellierung	167
8.1.3	Lösungsmethoden	168
8.1.4	Numerische Ergebnisse	172
8.2	Taktische Fahrplananalyse mit Zeitfenstern	175
8.2.1	Sortierheuristiken	175
8.2.2	Fix&Optimize Verfahren	178
8.2.3	Adaption von VRP-Methoden	180
8.2.4	Numerische Ergebnisse	183
9	Zusammenfassung und Ausblick	187
	Abbildungsverzeichnis	194
	Tabellenverzeichnis	195
	Algorithmenverzeichnis	197
	Literaturverzeichnis	199

1 Einleitung

In einer globalisierten Welt steigt stetig die Nachfrage nach Mobilität von Waren und Menschen. Durch eine begrenzte Kapazität des Verkehrsaufkommens kommt dem öffentlichen Verkehr (ÖV), unter dem alle allgemein zugängliche Mobilitäts- und Verkehrsdienstleistungen (z.B. Güterverkehr, Personenverkehr, etc.) gefasst sind, eine wachsende Bedeutung zu. Vor allem der öffentliche Personennahverkehr (ÖPNV) spielt eine zentrale gesellschaftliche wie auch wirtschaftliche Rolle. Zu ihm gehören alle Dienstleistungen im Stadt- und Regionalverkehr, die im Durchschnitt eine Fahrtstrecke von 50 km oder eine Fahrtdauer von einer Stunde nicht überschreiten (vgl. [Destatis, 2008a]). Insgesamt nutzen 68% der deutschen privaten Haushalte den ÖPNV und beanspruchten ihn im Jahr 2007 ca. 10,5 Mrd. mal (vgl. [Destatis, 2008b]).

Neben seiner gesellschaftlichen Relevanz stellt der ÖPNV auch für die Volkswirtschaft einen wichtigen Faktor dar. So erwirtschafteten die von 2650 Verkehrsunternehmen mit ca. 250.000 Beschäftigten im Jahr 2007 Beförderungseinnahmen von über 10,5 Mrd. Euro (vgl. [Destatis, 2008b]). In 2590 dieser Unternehmen werden Busse zur Erbringung des Angebots verwendet. Insgesamt sichert dies allein in Deutschland direkt oder indirekt ca. 750.000 Arbeitsplätze (Stand 2007, [BMVBS, 2007]).

Seit der Strukturreform des öffentlichen Verkehrs in Deutschland und der fortschreitenden Privatisierung der Verkehrsunternehmen unterliegen die angebotenen Dienstleistungen im ÖPNV einem stärkeren Wettbewerb. Der ÖPNV ist in Deutschland so organisiert, dass die Erstellung des Angebots sowie die Sicherstellung der Finanzierung in der Verantwortung der Länder und Kommunen ist. Auf der Basis des erstellten Angebots werden Linien oder Bündel von Linien öffentlich ausgeschrieben. Aufgrund der zum Teil nicht gegebenen Wirtschaftlichkeit einzelner Linien leisten die Kommunen und Länder Ausgleichszahlungen oder Subventionen. Verkehrsunternehmen in privater oder öffentlicher Hand können Angebote für die Bedienung der ausgeschriebenen Linien abgeben. Der günstigste Anbieter erhält den Auftrag. Zur Ermöglichung eines kostengünstigen Angebots ist der effiziente Einsatz von Fahrzeugen und Personal von wesentlicher Bedeutung. Die Kosten für den Fahrzeugeinsatz entstehen zum einen durch die Bereitstellung von Fahrzeugen, die neben dem Kauf oder der Miete auch Wartungskosten beinhalten.

tet. Zum anderen fallen für die Bedienung der Linien betriebsabhängige Kosten (z.B. für Kraftstoff) an.

Eine im Auftrag des Deutschen Verkehrsforums erstellte Studie, die auf der Einschätzung von Experten aus Wirtschaft und Wissenschaft basiert, stellt die erwarteten Entwicklungen des ÖPNV bis zum Jahre 2015 vor. Demnach ist ein zunehmender Kostendruck für die Verkehrsunternehmen zu erwarten, da die Förderung mit Geldern aus der öffentlichen Hand dem aktuellen Trend folgend weiter abnehmen wird. Allerdings werden auf der Seite der Verkehrsunternehmen Potentiale zur Kostensenkung von 11,4 % gesehen, ohne dass der Umfang der angebotenen Dienstleistungen reduziert werden muss. Die größten Potentiale werden unter anderem in der *Verbesserung der betrieblichen Effizienz* und der Nutzung von Synergien durch *Kooperationen und Fusionen* gesehen. Die Effizienz kann laut Studie neben einer automatisierten Abrechnung vor allem durch Effizienzsteigerungen bei vorhandenen Technologien zur Fahrzeugeinsatzplanung und Disposition gesteigert werden (vgl. [Lasch et al., 2005]). Für die Unterstützung der Einsatzplanung von Verkehrsunternehmen werden mathematische Optimierungsmethoden eingesetzt, die seit vielen Jahren in der Wissenschaft erforscht werden und in Planungssoftware für den ÖPNV integriert sind (vgl. [Wren, 2003]).

Durch den Zusammenschluss oder Verkauf von Unternehmen werden laut Studie wenige große Verkehrsbetriebe entstehen, so dass neben der effizienteren Planung auch weitere Optimierungspotentiale durch Freiheitsgrade in der Planung größerer Betriebe entstehen. Daher kommt der Optimierung des Fahrzeugeinsatzes und der Sicherstellung der Realisierung geplanter Kostenersparnisse eine wesentliche Rolle für die nachhaltige Wettbewerbsfähigkeit von Verkehrsunternehmen zu (vgl. [Lasch et al., 2005]).

Die Fahrzeugeinsatzplanung erfolgt durch die Erstellung von Umläufen für Fahrzeuge, so dass ein gegebener Fahrplan mit Linienfahrten bedient wird. Dabei spielt in der Praxis eine Reihe von zusätzlichen Aspekten eine Rolle, die weitere Einsparpotentiale ermöglichen (z.B. mögliche Leerfahrten zwischen Haltestellen) oder zusätzliche Nebenbedingungen abdecken (z.B. Einschränkung der möglichen Fahrzeugtypen für Linienfahrten).

Zur Bestimmung optimaler Umläufe existiert eine Vielzahl von Ansätzen aus der Forschung. Die Mehrzahl dieser Methoden ist allerdings durch die theoretische Komplexität der Problemstellung motiviert, so dass wichtige und in der Praxis relevante Erweiterungen außer Acht gelassen werden. Darüber hinaus validieren viele Autoren ihre Optimierungsansätze durch Tests auf künstlich generierten Instanzen, von denen unklar bleibt in welchem Grad sie mit realen Problemstellungen vergleichbar sind.

Die Problemstellung der Umlaufplanung wird in der Literatur allein unter

der Zielsetzung einer operativen Einsatzplanung von Fahrzeugen betrachtet. Allerdings lassen sich darüber hinaus weitere Anwendungsfälle identifizieren, mit denen der Planungsprozess im ÖPNV (z.B. bei strategischen Entscheidungen) zusätzlich unterstützt werden kann. Jeder Anwendungsfall erfordert dabei eine spezielle Ausrichtung der Lösungsmethoden hinsichtlich individueller Anforderungen.

Die Zielsetzung der vorliegenden Arbeit besteht in der **Entwicklung von Optimierungsmethoden für die Umlaufplanung**, die an den realen Anforderungen der Planungsaufgaben im ÖPNV ausgerichtet sind. Dies beinhaltet neben der Berücksichtigung möglichst vieler praxisrelevanter Restriktionen auch die Analyse realer Problemdata und ihrer Eigenschaften hinsichtlich spezifischer Eigenschaften. Für die Entwicklung dieser Methoden kann auf dem Modellierungsansatz von Kliewer (vgl. [Kliewer, 2005]) aufgebaut werden, der aufgrund einer deutlichen Verringerung der zu betrachtenden Modellgrößen eine geeignete Grundlage für die Entwicklung effizienter Methoden darstellt.

Das zweite Ziel dieser Arbeit ist die erstmalige **Identifikation und Beschreibung weiterer Anwendungsfälle** zur Unterstützung des Planungsprozesses im ÖPNV durch den Einsatz von Umlaufoptimierungsmethoden. Durch die Unterstützung der individuellen Planungsanforderungen können die Planer effektiver unterstützt werden, der Einsatz von Optimierungsmethoden im Rahmen des Planungsprozesses erweitert werden und bessere Planungen über den gesamten Planungsprozess hinweg Kosteneinsparungen ermöglichen.

Vor dem Hintergrund zu erwartender größerer Problemstellungen ist das dritte Ziel die **Unterstützung bei der Planung komplexer Fragestellungen**. Neben einer kurzen Laufzeit der Optimierung gehört dazu auch die schnelle Bereitstellung von Informationen zur Laufzeit, um ein Black-Box-Verhalten zu vermeiden, bei dem ein Planer eine nicht abschätzbare Zeit auf eine optimale Lösung wartet. Zu diesen Informationen gehören neben der Abschätzung der Optimierungsdauer auch die Bereitstellung von guten, aber nicht bewiesenen optimalen, Umlaufplänen und eine Einschätzung der Lösungsqualität durch untere Schranken, so dass Planer parallel zu Optimierungen mit Lösungsvorschlägen arbeiten können.

Die Arbeit gliedert sich in neun Kapitel. In *Kapitel 2* wird die Planung des Fahrzeugeinsatzes im ÖPNV beschrieben. Dazu wird der gesamte Planungsprozess dargestellt und erläutert, welche Planungsphasen durch eine Umlaufplanungsoptimierung unterstützt werden können. Die Umlaufplanung kann eine Reihe von relevanten Aspekten beinhalten, die je nach Planungssituation auftreten. Die unterschiedlichen Anwendungsfälle mit Anforderungen werden erläutert und die formalen Problemstellungen klassifiziert.

Für die im Laufe der Arbeit folgenden methodischen Beschreibungen sind einige Kenntnisse des Operations Research erforderlich. *Kapitel 3* erläutert die notwendigen Inhalte und verweist auf weiterführende Literatur.

Kapitel 4 diskutiert für die zuvor abgegrenzten Problemstellungen den aktuellen Stand der Technik. Neben der Beschreibung und Einordnung unterschiedlicher Modelle und Lösungsmethoden werden eingesetzte Planungssysteme vorgestellt.

Auf der Basis der Anforderungen aus Kapitel 2 und dem Stand der Technik aus Kapitel 4 wird in *Kapitel 5* der Forschungsbedarf abgeleitet und die Zielsetzung sowie das Vorgehen zur Erreichung dieser Ziele konkretisiert.

In *Kapitel 6* werden die Eigenschaften unterschiedlicher realer und künstlich generierter Problemdaten für die Umlaufplanung analysiert. Es werden die Auswirkungen dieser Eigenschaften auf die Struktur des mathematischen Modells und das Lösungsverhalten gezeigt sowie Kennzahlen entwickelt, um die Eignung von alternativen Modellen zu motivieren und die Komplexität von Probleminstanzen abzuschätzen.

In *Kapitel 7* wird eine Methode zur Lösung von Umlaufplanungsproblemen unter besonderer Berücksichtigung der praxisrelevanten Anwendungsfälle vorgeschlagen. Die Methode ist durch einen modularen Aufbau und ein iteratives Vorgehen sowohl für die Anwendung für operative als auch für strategische Fragestellungen geeignet. Anhand von umfassenden numerischen Untersuchungen wird die Methode validiert und gezeigt, dass auch Problemstellungen von Großstädten gelöst werden können.

Eine Erweiterung des Umlaufplanungsproblems sieht einen möglichen Eingriff in die Abfahrtszeiten eines gegebenen Fahrplans vor. In *Kapitel 8* werden neue Methoden vorgestellt, die diese Anforderung unter Berücksichtigung der relevanten Anwendungsfälle ermöglichen.

Kapitel 9 fasst die Ergebnisse der Arbeit zusammen und gibt einen Ausblick auf weitere mögliche Forschungsaktivitäten in dem Bereich der Umlaufplanung.

2 Umlaufplanung im öffentlichen Personennahverkehr

Die Planungsaufgabe der Erstellung von Umläufen für Fahrzeuge stellt ein wichtiges und komplexes Entscheidungsproblem für Verkehrsunternehmen des öffentlichen Personennahverkehrs (ÖPNV) dar. Um die Entscheidungen innerhalb der Planungsprozesse des ÖPNV zu erläutern, wird in Abschnitt 2.1 der für diese Arbeit relevante Prozess sowie die jeweiligen Aufgaben und Freiheitsgrade beschrieben. Durch die stark heterogenen Rahmenbedingungen unterschiedlicher Städte und Verkehrsbetriebe ergeben sich für die Umlaufplanung unterschiedliche Restriktionen und Zielsetzungen. Neben diesen bestehen unterschiedliche Szenarien der Anwendung von Lösungsmethoden für Umlaufplanungsprobleme. Die Einsatzgebiete und ihre spezifischen Anforderungen an Lösungsmethoden zur Einsatzplanung werden in Abschnitt 2.2 beschrieben. In Abschnitt 2.3 werden die formalen Anforderungen beschrieben, die in der Umlaufplanung betrachtet werden.

2.1 Fahrzeugeinsatzplanung im Planungsprozess des ÖPNV

In der Planung von Verkehrsleistungen des ÖPNV müssen eine Vielzahl von strategischen, taktischen sowie operativen Fragestellungen beantwortet werden, die unterschiedliche Ziele und Planungszeiträume betreffen. Durch die enorme Komplexität der gesamten Planungsaufgabe und der zum Teil zeitlich unterschiedlichen Planungsfrequenz der Probleme wird der Planungsprozess typischerweise in unterschiedliche Teilprobleme zerlegt, die in sequentieller Abfolge gelöst werden. Eine Optimierung über den gesamten Prozess kann daher in der Praxis nicht durchgeführt werden. Der im Folgenden beschriebene Prozess orientiert sich u.a. an den Ausführungen von [Desaulniers und Hickman, 2006]. Abbildung 2.1 zeigt eine Übersicht über die unterschiedlichen Planungsschritte.

In der *Netzplanung* werden strategische Entscheidungen für den Ausbau der Netztopologie – das Netzdesign – getroffen. Auf der Basis einer Nachfragestudie,

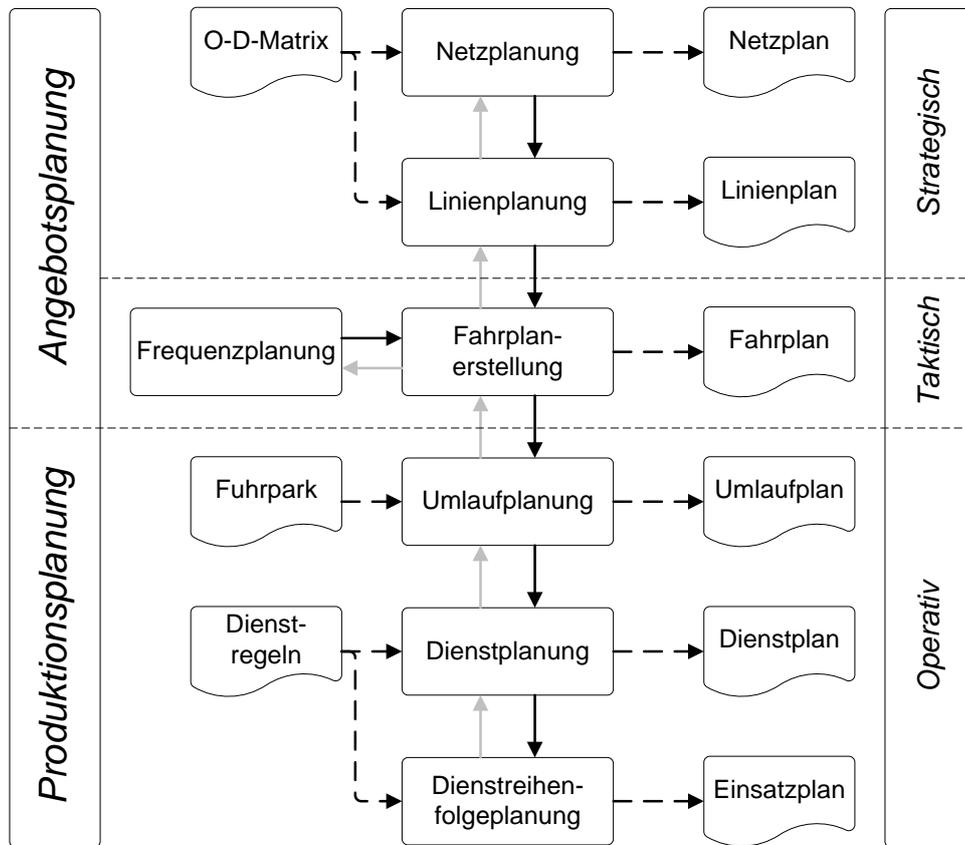


Abbildung 2.1: Sequentieller Planungsprozess im ÖPNV

deren Ergebnis die *Origin-Destination-Matrix (O-D-Matrix)* ist, werden vermutete Passagierströme geplant. Die O-D-Matrix beinhaltet eine quantitative Größe über die erwartete Nachfrage zwischen zwei Orten. Typischerweise kann ein Ort in diesem Sinn ein Verkehrsknotenpunkt oder das Zentrum einer Region (z.T. Stadtteil) sein. Ergebnis ist der *Netzplan*, der neben der O-D-Matrix als Grundlage für die *Linienplanung* dient. Ziel ist es die Linien für das Verkehrsnetz zu bestimmen. Linien sind definiert über ihren Routenverlauf (Linienführung) sowie der Frequenz in welcher sie bedient werden sollen (vgl. [Borndörfer et al., 2004]). Sowohl die Netzplanung als auch die Linienplanung gehören zu den strategischen Entscheidungsproblemen, deren Ergebnis im Normalfall über viele Jahre Bestand hat. Als Ziel haben die strategischen Fragestellungen die Maximierung des Serviceangebots und damit einen Aspekt der Kundenzufriedenheit. Neben der Ermöglichung einer nachgefragten Route werden weitere Aspekte wie die Anzahl von Umsteigevorgängen oder die Routenlänge betrachtet. Je mehr Ressourcen eingesetzt werden, um zum Beispiel zusätzliche Servicefahrten anzubieten oder



	Stunde 5		6		7-19		20		21-22		23
Hauptbahnhof	25	55	25	53	23	53	23	30			20
Westerntor	27	57	27	55	25	55	25	32			22
Neuhäuser Tor	28	58	28	57	27	57	27	33			23
Michaelstraße	29	59	29	58	28	58	28	34			24
Maspermplatz	30	00	30	00	30	00	30	35			25
Bischofsteich	31	01	31	01	31	01	31	36			26
Tegelweg	32	02	32	02	32	02	32	37			27
Nordbahnhof	34	04	34	04	34	04	34	38			28
Schützenplatz Nord	35	05	35	05	35	05	35	39			29
Ausbesserungswerk	36	06	36	06	36	06	36	40			30
Pirolweg	37	07	37	07	37	07	37	41			31
Haustenbecker Straße	38	08	38	08	38	08	38	42			32
Schleswiger Weg	39	09	39	09	39	09	39	43			33
TUV	40	10	40	10	40	10	40	43			33
Friesenweg	41	11	41	11	41	11	41	44			34
An der Talle	42	12	42	12	42	12	42	45			35
Ingolstädter Weg	43	13	43	13	43	13	43	46			36
Württembergischer Weg	44	14	44	14	44	14	44	47			37
Bonifatiusweg	45	15	45	15	45	15	45	48			38

Abbildung 2.2: Fahrplan einer Linie der PaderSprinter (Paderborn)

größere zeitliche Puffer zur Sicherung von Linienanschlüssen einzurichten, desto höher sind die erwarteten Kosten für den Betrieb des Verkehrsnetzes. Aus diesem Grund werden bereits in der strategischen Planung die erwarteten Kosten für den Betrieb eines Liniennetzes grob abgeschätzt und berücksichtigt. Die Abschätzung dieser Kosten erfolgt über einfache analytische Methoden. Zum Beispiel wird die Anzahl der angebotenen Servicefahrten als Maß für die erwarteten Kosten verwendet. Die Linienplanung kann somit als Maximierung des Serviceangebots unter Berücksichtigung einer Kostenrestriktion oder als bikriterielles Optimierungsproblem interpretiert werden. Ergebnis der Linienplanung und strategische Grundlage für die weiteren Planungsschritte ist der *Linienplan*.

Die anschließende Phase der *Fahrplanerstellung* gehört zu den taktischen Aufgaben, die je nach Verkehrsbetrieb in einer Frequenz von wenigen Wochen bis zu einigen Monaten durchgeführt werden. Auf der Basis der definierten Linien wird eine Fahrtenmenge erstellt, in der jede Fahrt einer Linie zugeordnet ist. Abbildung 2.2 zeigt einen Fahrplan für eine Linie. Jede Linie verläuft über eine Abfolge von Haltepunkten. Da das Fahrzeug und Personal nicht in den Zwischenstationen gewechselt werden, werden für die Servicefahrten nur die Endhaltestellen betrachtet. Jede Spalte des dargestellten Fahrplans entspricht daher einer Servicefahrt,

die von einer Endhaltestelle (z.B. Hauptbahnhof) über eine Menge von Zwischenstationen zu der anderen Endhaltestelle der Linie (z.B. Bonifatiusweg) führt. Für jede Servicefahrt sind die Abfahrts- und Ankunftszeiten im Fahrplan festgelegt. Die erste Spalte der Abbildung 2.2 entspricht daher einer Servicefahrt beginnend ab Hauptbahnhof um 5:25 Uhr und ankommend an der Haltestelle Bonifatiusweg um 5:45 Uhr.

Ein Fahrplan kann darüber hinaus weitere, nicht getaktete Fahrten enthalten. Dies sind zum Beispiel Verstärkungsfahrten zur Erhöhung der Kapazität oder Sonderfahrten für Schulbusse. Die im Rahmen der strategischen Planung getroffenen Entscheidungen über die Frequenzen der Linien können von den prognostizierten Nachfrageströmen abweichen, da diese auf Stichproben und Erfahrungswerten basieren. Aus diesem Grund wird das Angebot evaluiert, indem die erwartete Nachfrage mit der realen Nachfrage verglichen wird, und die in der Linienplanung getroffenen Entscheidungen angepasst. Da der Linienverlauf in der Regel nicht verändert werden soll, wird in der *Frequenzplanung* die Frequenz der Linien überarbeitet, um auf dieser Grundlage einen neuen Fahrplan zu erstellen.

Mit der Fahrplanerstellung ist die strategisch-taktische Phase des Planungsprozesses abgeschlossen. Diese Phase konzentriert sich, unter Berücksichtigung von globalen Kostenaspekten, auf die Bereitstellung eines möglichst hohen Servicegrads (Angebotsplanung). In der folgenden operativen Planung werden die Ressourcen, die für die Realisierung des definierten Angebots benötigt werden, vorgeplant (Produktionsplanung).

Die beschriebenen strategischen und taktischen Planungsaufgaben liegen in der kommunalen Hand, die nach Abschluss der Planungen Linien zur Ausschreibung freigibt, auf die sich Verkehrsunternehmen bewerben können. Die Ausschreibung erfolgt in einzelnen Linien oder Linienbündeln, die durch eine Mischung finanziell attraktiver und weniger attraktiver Linien eine Vergabe des gesamten Netzes sicherstellen soll. Ziel der Verkehrsunternehmen ist es, die erworbenen Linien mit einem möglichst kostenminimalen Ressourcenaufwand zu bedienen.

In der *Umlaufplanung* werden die Fahrzeuge auf der Basis eines zu planen oder bereits bestehenden Fuhrparks sowie unterschiedlicher Depotstandorte verplant. Auf weitere Nebenbedingungen und Zielsetzungen der Umlaufplanung wird detaillierter in Abschnitt 2.3 eingegangen. Ergebnis der Umlaufplanung ist ein Umlaufplan mit der Zuweisung von Fahrzeugen zu Fahrten sowie den zu fahrenden Routen. Nach der Einplanung der Fahrzeuge wird das Fahrpersonal ein geplant. Da die Dienstplanung im Gegensatz zur Umlaufplanung einer Vielzahl von gesetzlichen und organisatorischen Regeln unterliegt, ist die Planungsaufgabe erheblich komplexer und wird in zwei Schritten gelöst: Im ersten Schritt, der *Dienstplanung*, werden anonyme Dienste gebildet, die den gesetzlichen Regularien

entsprechen, und den Fahrplan vollständig abdecken. Den ermittelten Diensten werden daraufhin in der *Dienstreihenfolgeplanung* konkrete Fahrer zugewiesen.

Die operativen Phasen der Ressourceneinsatzplanung werden klassisch in der beschriebenen Reihenfolge durchgeführt, allerdings ist auch eine integrierte Betrachtung oder eine abweichende Reihenfolge der Planungsschritte, insbesondere für die Ausnutzung der Optimierungspotentiale, möglich (vgl. [Gintner, 2008]).

2.2 Anwendungsfälle der Umlaufplanung

Nachdem im vorherigen Abschnitt die Planungsschritte des ÖPNV und die Einbettung der Umlaufplanung in diesen Prozess beschrieben wurden, sollen im Folgenden die möglichen Anwendungsfälle der Umlaufplanung innerhalb dieses Prozesses vorgestellt werden. Diese Anwendungsfälle stellen zum einen eine unterschiedliche Planungssituation innerhalb des gesamten Planungsprozesses dar, die die Problemdimensionen und die erforderliche Lösungsqualität beeinflussen. Zum Anderen werden weitere Bedingungen an die Lösungsmethoden gestellt, wie eine maximal zulässige Laufzeit zur Lösung oder der für den Planer mögliche Grad an Lösungsinformationen zur Laufzeit. Insgesamt werden sechs Anwendungsfälle unterschieden, deren Zielsetzung in der folgenden Übersicht gegeben ist:

1. *Operative Planung ohne Zeitfenster*: Erstellung operativer Umlaufpläne zu einem festen Fahrplan
2. *Operative Planung mit Zeitfenstern*: Erstellung von operativen Umlaufplänen mit abweichenden Fahrtzeiten
3. *Taktische Planung ohne Zeitfenster*: Bestimmung von geeigneten Linienbündeln zur Ausschreibung
4. *Taktische Planung mit Zeitfenstern*: Analyse der Fahrplanpotentiale zur Einsparung von Fahrzeugen
5. *Strategische Planung*: Abschätzung von Ressourcenkosten für strategische Szenarien
6. *Integrierte Planung*: Gleichzeitige Lösung von Umlauf- und Dienstplanung

Im Folgenden sollen die identifizierten Anwendungsfälle vorgestellt werden. Die aufgeführten Anforderungen und Annahmen wurden – soweit nicht anders angegeben – in einer im Zuge dieser Arbeit durchgeführten Praxiskooperation mit einem Hersteller für Umlaufplanungssoftware und den Erfahrungen mit Verkehrsunternehmen ermittelt.

2.2.1 Operative Planung

Im Zuge der operativen Planung werden Fahrzeuge kostenminimal zur Bedienung eines festgelegten Fahrplans eingeplant. Dieses stellt den klassischen Anwendungsfall der Umlaufplanung dar, bei dem der Fokus eindeutig auf der Erzielung einer optimalen oder möglichst guten Lösung des Umlaufplanungsproblems liegt, um alle zulässigen Freiheitsgrade zur Minimierung der geplanten Kosten auszunutzen. Die Umlaufplanung stellt – im Sinne der operativen Planung – zumeist keinen zeitkritischen Prozess dar, da die Fahrpläne im Normalfall für einen längeren Zeitraum (z.B. mehrere Monate) mit gewissem Vorlauf erstellt werden. Der Planer muss allerdings die durch die Lösung des Problems entstehenden Umlaufpläne manuell anpassen oder nachbearbeiten, da nicht sämtliche Anforderungen in der modellierten Problemstellung berücksichtigt werden können oder weitere nicht formalisierbare Einflüsse relevant sind (z.B. Expertenwissen über Sonderbehandlungen von Fahrzeugen/Linien). Daher sollte eine Methode zur Lösung von (operativen) Umlaufplanungsproblemen den Fokus auf die Erreichung einer möglichst optimalen Lösung haben. Die Laufzeit der Lösungsmethode sollte jedoch im Bereich von maximal zehn Stunden liegen, damit der Planer die Optimierung ggf. über Nacht starten kann und am darauf folgenden Tag den resultierenden Umlaufplan zur Verfügung hat. In anderen Fällen wäre aus Sicht eines Planers eine Lösungsmethode wünschenswert, die bereits zur Laufzeit der Optimierung Informationen sowie mögliche (heuristische) Lösungen ausgibt. Dadurch wird der Planer in die Lage versetzt, bereits parallel zur Optimierung mögliche Lösungsvorschläge in Betracht zu ziehen, während das System die Suche nach besseren Lösungen fortsetzt. In beiden Fällen sollte ein Black-Box-Verhalten vermieden werden, indem möglichst zu Beginn des Verfahrens Informationen zur Abschätzung der Lösung sowie eine näherungsweise Dauer der Laufzeit ausgegeben wird und während der Optimierung der Status der Lösungsmethode ausgegeben wird. Darüber hinaus sollte bei Abbruch der Optimierung durch den Planer eine möglichst gute gültige Lösung verfügbar sein. Da sich die konkreten Anforderungen der Planer bezüglich der Informationsfülle stark unterscheiden können, besteht daher die Anforderung einer Lösungsmethode, die flexibel auf die konkreten Anforderungen des Planers angepasst werden kann.

Das operative Umlaufplanungsproblem entspricht je nach Struktur des Verkehrsunternehmens und des zu planenden Fahrplans einem Problem mit einem Depot, mit mehreren Depots und/oder mit mehreren Fahrzeugtypgruppen. In Einzelfällen können zudem Anforderungen an den Aufbau von Umläufen vorhanden sein. Dies ist zum Beispiel eine Beschränkung der Anzahl unterschiedlicher Linien, die in einem Umlauf bedient werden, um einen Plan weniger anfällig für

die Auswirkungen von Störungen einzelner Linien zu machen und dadurch die Robustheit der Planung zu erhöhen. In der Regel spielen die Routenrestriktionen in der operativen Umlaufplanung eine untergeordnete Rolle, so dass eine Betrachtung als weiche Restriktion (nach Optimierung der geplanten Kosten) ausreichend ist. Der vorgestellte Anwendungsfall sei im Folgenden als *Operative Planung ohne Zeitfenster* bezeichnet.

Die Berücksichtigung von Zeitfenstern für Servicefahrten spielt in der operativen Planung eine besondere Rolle. Da die Fahrplan- bzw. Frequenzplanung bereits in der taktischen Planung des ÖPNV durchgeführt wird und dadurch der Fahrplan zum Zeitpunkt der Umlaufplanung bereits feststeht, können in der operativen Planung in der Regel keine Eingriffe in den Fahrplan mehr getätigt werden. Um dennoch die Potentiale, die durch eine Verschiebung von Fahrten innerhalb des Fahrplans ermöglicht werden, zu nutzen, können sehr kleine Zeitfenster für die im Fahrplan ausgezeichneten Taktfahrten erlaubt werden (z.B. ± 2 Minuten). Eine Ausnahme stellen Sonderfahrten wie Schul- oder Verstärkungsfahrten dar, die nicht im veröffentlichten Fahrplan ausgezeichnet sind. Für diese Fahrten kann daher eine größere Verschiebung erlaubt werden (z.B. ± 10 Minuten) (vgl. z.B. [Bokinge und Hasselström, 1980]). Ein mögliches Vorgehen der Verkehrsunternehmen ist es, die Freiheitsgrade durch Zeitfenster ohne Änderung der Fahrteninformationen im veröffentlichten Fahrplan auszunutzen. Dies wird von den Fahrgästen als Verspätung oder eine zu frühe Bedienung wahrgenommen, was zu einer Senkung der Kundenzufriedenheit führen kann. Da eine verfrühte Bedienung dazu führen kann, dass ein Fahrgast den Bus nicht mehr erreicht und auf den nächsten Bus warten muss, werden zum Teil Zeitfenster erlaubt, die lediglich eine Verspätung der Servicefahrten erlauben (z.B. $+ 2$ Minuten). Alternativ werden die geänderten Abfahrts- und Ankunftszeiten im Fahrplan veröffentlicht. Dabei kann die Anforderung bestehen, dass die Servicefahrten einer Linie nur gemeinsam verschoben werden können, so dass die Taktung der Fahrten erhalten bleibt. Um die im Fahrplan vorgenommenen Änderungen gering zu halten, besteht zudem die Anforderung an die Lösungsmethoden, dass Fahrten ausschließlich dann verschoben werden, wenn dadurch eine Einsparung eines Fahrzeugs ermöglicht wird. Es sollen also keine Fahrten verschoben werden, wenn ausschließlich operative Kosten gespart werden können. Bezüglich der akzeptierten Optimierungsdauer und dem Informationsgrad der Methode zur Laufzeit gelten die gleichen Anforderungen wie bei der operativen Planung ohne Zeitfenster. Dieser Anwendungsfall sei als *Operative Planung mit Zeitfenstern* bezeichnet.

Der Anwendungsfall der operativen Planung ohne und mit Zeitfenstern ist ausführlich in [Kliewer, 2005] beschrieben.

2.2.2 Taktische Planung

Im Rahmen der taktischen Planung spielt die Lösung von Umlaufplanungsproblemen eine Rolle, wenn nach dem operativen Betrieb eines Fahrplans die auf Nachfrageschätzungen basierende Frequenzplanung revidiert und erneut durchgeführt wird (s. Abschnitt 2.1). Für die Erstellung eines angepassten Fahrplans können Erkenntnisse über das Einsparungspotential in der Umlaufplanungsphase berücksichtigt werden, die der Planer bei der Anpassung berücksichtigen kann. Die Potentiale können durch What-If-Analysen aufgezeigt werden. Durch dieses Vorgehen ist es möglich, innerhalb der taktischen Planung die operative Zielsetzung (vor allem eine Reduzierung der benötigten Fahrzeuganzahl) teilweise zu berücksichtigen und die beiden Planungsebenen zu integrieren. Die oben genannten What-If-Analysen können mit unterschiedlich großen Zeitfenstern (z.B. ± 5 bis ± 20 Minuten) für alle Servicefahrten durchgeführt werden, um mögliche Potentiale zu identifizieren. Da dieser Anwendungsfall eine Planungsunterstützung der Frequenzplanung darstellt, ist die akzeptierte Laufzeit für die Erstellung einer Lösung stark abhängig von der Planungssituation des Planers. Des Weiteren führt die Berücksichtigung von großen Zeitfenstern in der Umlaufplanung zu sehr komplexen Optimierungsmodellen, so dass eine optimale Lösung für reale Problemstellungen nicht oder nur mit sehr großem Laufzeitaufwand möglich ist (vgl. [Bunte et al., 2005]). Daher sollte der Fokus bei den auf diese Problemstellung ausgerichteten Lösungsmethoden auf einer möglichen parametrisierten Skalierbarkeit von Lösungsqualität und Laufzeit sein. Um die Frequenzplanung zu unterstützen sollte neben einer möglichen Skalierung auch die Generierung von Lösungen zur Laufzeit möglich sein. Dadurch kann der Planer neu gefundene Potentiale in der Fahrplanerstellung direkt in seine Planungen integrieren. Dieser Anwendungsfall sei im Folgenden als *Taktische Planung mit Zeitfenstern* bezeichnet.

Neben dem Einsatz in der Frequenzplanung besteht die Möglichkeit die Ausschreibung der Linien bzw. Linienbündel durch Methoden der Umlaufplanung zu unterstützen. So können What-If Analysen möglicher Zusammenstellungen zeigen, welche Linien sich ergänzen und welche Ressourcenkosten für diese Bündel zu erwarten sind. Zusätzlich kann eine Verkehrsunternehmen Analysen über eine potentielle Erweiterung der bestehenden Linien durchführen. So kann die Kalkulation für die Abgabe eines Angebots unterstützt werden und das Verkehrsunternehmen eine fundierte Entscheidung treffen. Wie bei der taktischen Planung mit Zeitfenstern sollte eine Optimierungsmethode skalierbar sein, da sich die akzeptierte Laufzeit je nach Situation unterscheidet. Dieser Anwendungsfall sei im Folgenden als *Taktische Planung ohne Zeitfenster* bezeichnet.

2.2.3 Strategische Planung

In der strategischen Planung werden, wie in Abschnitt 2.1 gezeigt, bereits bei der Bestimmung des Netzplans die erwarteten Ressourcenkosten des Plans abgeschätzt und berücksichtigt, indem die nachfolgenden Planungsstufen rudimentär gelöst werden. Dabei werden für unterschiedliche Netzpläne Szenarien durchgeführt. Die Kosten für den Fahrzeugeinsatz werden innerhalb dieser Szenarien durch einfache Methoden abgeschätzt. Aktuelle Entwicklungen von Produkten zur strategischen Planung zeigen eine zunehmend detaillierte Integration der operativen Aspekte. [Friedrich und Nökel, 2003] beschreiben die Betrachtung von einfachen Umlaufplanungsaspekten innerhalb der strategischen Planungssoftware VISUM (vgl. [Friedrich und Mott, 1998]) und verweisen auf die mögliche Integration spezieller Umlaufplanungsverfahren, die unabhängig von den strategischen Lösungsverfahren in Planungssoftware eingebettet werden können. Methoden zur Optimierung von Umlaufplanungsproblemen können daher auch für Abschätzungen von operativen Kosten in der strategischen Planung dienen (vgl. [Friedrich und Nökel, 2004]). Aufgrund der in dieser Stufe vorhandenen unscharfen Rahmenbedingungen ist es ausreichend, wenn zur Abschätzung ein ungefährender Kostenaufwand berechnet werden kann (z.B. Kostenbereich durch obere und untere Schranke). In der strategischen Planung des Angebots arbeitet ein Planer in der Regel intensiv mit einem Planungssystem und muss eine Vielzahl unterschiedlicher Szenarien zur Einschätzung möglicher Konsequenzen, die mit einem Angebot einhergehen, berechnen. Daher muss eine Lösungsmethode zur Kostenabschätzung des Fahrzeugressourceneinsatzes in kurzer Zeit Lösungen bereitstellen können. Im Gegensatz zur operativen Planung ohne Zeitfenster kann nicht für jedes Szenario mehrere Stunden auf eine Lösung gewartet werden. Dieser Anwendungsfall sei im Folgenden als *Strategische Planung* bezeichnet.

2.2.4 Integrierte Planung

In der integrierten Umlauf- und Dienstplanung werden die nach dem klassischen Vorgehen getrennten Planungsschritte zur Ressourceneinsatzplanung gemeinsam betrachtet. Statt der Berechnung des kostenminimalen Umlaufplans kann auf Kostenoptimalität teilweise verzichtet werden, sofern es zugunsten der effizienteren Dienstplanung sinnvoll ist. Dies geschieht in einem integrierten Ansatz, bei dem beide Aufgaben in einem Optimierungsproblem gelöst werden. Aufgrund der größeren Komplexität einer gemeinsamen Betrachtung können mit spezialisierten Methoden Problemstellungen mit bis zu 653 Servicefahrten und 4 Depots (heuristisch) gelöst werden (vgl. [Gintner, 2008]).

In den aktuellen Lösungsansätzen zur integrierten Planung werden die Modelle in einem iterativen Vorgehen gelöst, bei der der Lösung von Umlaufplanungsproblemen als Unterprobleme eine entscheidende Rolle zukommt (vgl. [Gintner et al., 2006], [Steinzen, 2008]). Daher können neue Methoden für die Umlaufplanung gleichzeitig zur Steigerung der Lösungseffizienz von integrierten Problemen eingesetzt werden. Dieser Anwendungsfall sei im Folgenden als *Integrierte Planung* bezeichnet.

2.3 Problemstellungen

Nach der Vorstellung der möglichen Anwendungsfälle der Umlaufplanung, werden im folgenden Abschnitt die Problemstellungen der Umlaufplanung beschrieben und formal definiert. Die Aufstellung der unterschiedlichen Problemstellungen basiert zum einen auf der vorhandenen wissenschaftlichen Literatur zur Busumlaufplanung (s. Kapitel 4). Des Weiteren sind durch eine im Rahmen dieser Arbeit durchgeführte Kooperation mit einem Anbieter für ÖPNV-Planungssoftware die Anforderungen und Hinweise von Planern und Anwendern mit eingeflossen. Die folgende Aufstellung spiegelt daher insbesondere den Bedarf der unterschiedlichen praxisrelevanten Problemstellungen wider. Eine Übersicht über weitere Detailanforderungen ist in [Kliwer, 2005] gegeben.

2.3.1 Fahrzeugumlaufplanung

Das Ziel des *Fahrzeugumlaufplanungsproblems* (engl: *vehicle scheduling problem*) (VSP) ist die kostenminimale Planung von Fahrzeugen, um einen definierten Fahrplan bedienen zu können. Das VSP umfasst alle Anforderungen, die als Grundlage für die in den nachfolgenden Abschnitten beschriebenen Erweiterungen dienen, und stellt somit das grundlegende Problem der Umlaufplanung dar.

Voraussetzung für die Planung ist ein *Fahrplan*, der eine Menge von den zu bedienenden Fahrten (*Servicefahrten*) enthält. Eine Servicefahrt entspricht einer zu einer festgelegten Zeit zu fahrenden Linienroute zwischen zwei *Endhaltestellen*. In der Umlaufplanung werden die Fahrten auf der Granularitätsebene vollständiger Linienfahrten betrachtet, da während der Bedienung einer Linie ein Fahrzeugwechsel ausgeschlossen ist. Jede Servicefahrt ist definiert durch eine *Abfahrtshaltestelle* sowie eine *Abfahrtszeit*, die Ort und Zeit für den Beginn der Servicefahrt definieren. Entsprechend definiert die *Ankunftshaltestelle* in Verbindung mit der *Ankunftszeit* das Ende der Servicefahrt.

Der Planungshorizont der in der Umlaufplanung betrachteten Fahrtenmenge entspricht in der Regel einem Tag, da zu einem Zeitpunkt des Tages (typischer-

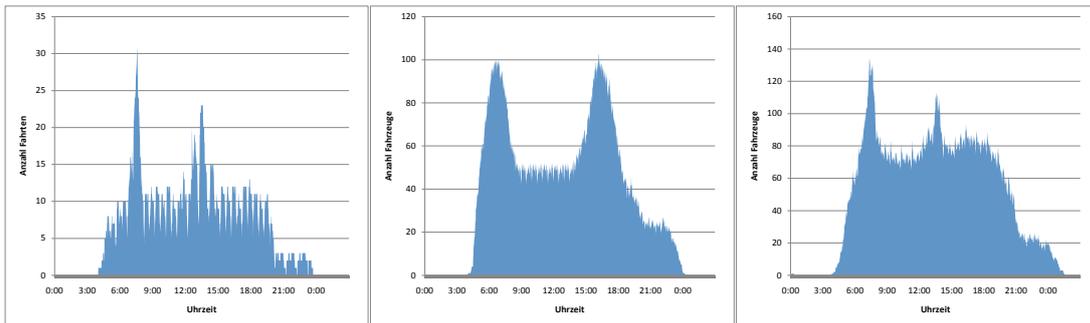


Abbildung 2.3: Auslastungskurven für reale Fahrpläne

weise in den frühen Morgenstunden) keine Servicefahrten zu bedienen sind und dadurch die einzelnen Fahrpläne unabhängig voneinander betrachtet werden können. In der Regel sind reale Fahrpläne von der Eigenschaft geprägt, dass in Zeiten von hoher Nachfrage durch Schul- und Berufspendler die weitaus meisten Fahrten zu bedienen sind. Die Auslastungskurve, die die Anzahl der Fahrten über die Zeit anzeigt, weist durch diesen Effekt in vielen Fällen zwei deutliche Spitzen auf. Exemplarisch sind in Abbildung 2.3 Beispiele für Auslastungskurven von realen Fahrplänen gezeigt.

Ergebnis der Umlaufplanung ist ein *Umlaufplan*, der eine Menge von *Umläufen* enthält. Ein Umlauf entspricht einer zulässigen Sequenz von Fahrten, die von einem Fahrzeug bedient werden können. Die primäre Zielsetzung der Umlaufplanung ist die Minimierung der geplanten Kosten zur regulären Bedienung des Fahrplans. Es werden zwei unterschiedliche Kostenarten unterschieden: die *operativen Kosten* entsprechen dem betriebsbedingten Aufwand einer zu fahrenden Strecke (z.B. Kraftstoff). Sie werden abhängig von der Länge und/oder Dauer der zu fahrenden Wege berechnet. Häufig werden auch Wartezeiten, die an Endhaltestellen außerhalb eines Depots anfallen, in die operativen Kosten mit einbezogen. Darüber hinaus fallen *fixe Kosten* (auch: *Fixkosten*) für die Verwendung eines Fahrzeugs an. Sie stellen Abschreibungskosten für den Kauf eines Fahrzeugs und den einsatzunabhängigen Erhalt des Fahrzeugs dar (z.B. Wartung). In der Praxis werden die Gesamtkosten mit einer Kombination aus operativen und fixen Kosten bewertet. Da der Kauf bzw. das Vorhalten eines Fahrzeugs für ein Verkehrsunternehmen einen beträchtlichen Kostenaufwand bedeutet, sind die fixen Kosten in der Regel erheblich höher als die operativen Kosten. Daraus resultierend werden im Rahmen der Umlaufplanung ohne Beschränkung der Allgemeinheit primär die fixen Kosten (Anzahl der Fahrzeuge) minimiert und sekundär die operativen Kosten.

2.3.2 Fahrzeugumlaufplanung mit einem Depot

Das *Fahrzeugumlaufplanungsproblem mit einem Depot* (engl: *single depot vehicle scheduling problem*) (SD-VSP) stellt die Erweiterung des Problems um die Berücksichtigung eines *Depots* dar, in dem die Fahrzeuge über Nacht stationiert sind. Dieses entspricht im SD-VSP einem Standort (z.B. Stellplatz oder Garage), von dem aus die Fahrzeuge zur Abfahrtshaltestelle der ersten Fahrt ausrücken und zu dem sie nach Abschluss des Umlaufs zurückkehren. Die durch diese Wege bedingten zusätzlichen Leerfahrten werden *Ausrückfahrten* bzw. *Einrückfahrten* oder allgemein *Depotfahrten* genannt. In der Umlaufplanung mit einem Depot wird von einer homogenen Fahrzeugflotte, in der jedes Fahrzeug für die Bedienung aller Servicefahrten des Fahrplans ausgelegt ist, ausgegangen.

Die Komplexität der Umlaufplanung ist vor allem durch die Möglichkeit gegeben, dass Fahrzeuge auch zwischen der Bedienung von zwei Servicefahrten eine Leerfahrt (*Umsetzfahrt*) machen können, um die Endhaltestelle zu wechseln. Diese Anforderung wird auch als *Deadheading* bezeichnet. Durch eine Umsetzfahrt entstehen auf der einen Seite höhere operative Kosten, andererseits können die Fahrzeuge effizienter eingesetzt werden, so dass tendenziell insgesamt weniger Fahrzeuge zur Bedienung des Fahrplans benötigt werden und dadurch fixe Kosten (und operative Kosten durch weniger Aus-/Einrückfahrten) verringert werden können.

Die oben beschriebene Problemstellung des SD-VSP wird im Folgenden formal definiert: Sei S die Menge aller Endhaltestellen und T die Menge aller Servicefahrten des zu bedienenden Fahrplans. Der Beginn einer Servicefahrt $t \in T$ ist definiert durch die Abfahrtshaltestelle $g_t \in S$ und die Abfahrtszeit e_t . Das Ende ist durch die Ankunftshaltestelle $h_t \in S$ und die Ankunftszeit f_t gegeben. Entsprechend seien die Haltestellen und Zeitpunkte für Umsetz- und Depotfahrten definiert. Sei d_{ij} die Dauer einer Umsetzfahrt zwischen den Endhaltestellen $i, j \in S$. Die Umsetzfahrt ist gültig bzw. die Servicefahrten sind zueinander kompatibel ($i \alpha j$), wenn gilt:

$$f_i + d_{i,j} \leq e_j, \quad \text{für } i \neq j \quad (2.1)$$

$$f_i \leq e_j, \quad \text{für } i = j \quad (2.2)$$

Andernfalls sind die Fahrten nicht-kompatibel ($i \not\alpha j$) und können nicht von einem Fahrzeug bedient werden. Ein gültiger, nicht-leerer Umlauf u aus einem Umlaufplan U ($u \in U$) mit $|U|$ Umläufen ist definiert als eine Sequenz von Fahrten mit dem regulären Ausdruck (vgl. [Friedl, 2003])

$$A S (S | (U S))^* E$$

wobei für alle Servicefahrten $i, j \in u$ gilt: $(i \alpha j)$ oder $(j \alpha i)$. Der Ausdruck gibt die gültige Kombination der Ausrückfahrten (A), Servicefahrten (S), Umsetzfahrten (U) und Einrückfahrten (E) an.

Die Anzahl der im Depot vorhandenen Fahrzeuge ist limitiert durch die Kapazität w . Seien c_u^{seq} die operativen Kosten eines Umlaufs $u \in U$ und c^{veh} die fixen Kosten für ein Fahrzeug. Die geplanten Kosten eines Umlaufplans U sind für das SD-VSP:

$$c^{total} = \underbrace{\sum_{u \in U} c_u^{seq}}_{\text{operative Kosten}} + \underbrace{\sum_{u \in U} c^{veh}}_{\text{fixe Kosten}}$$

Das Umlaufplanungsproblem (SD-VSP) kann daher definiert werden als *das Finden einer Zuweisung von Fahrten zu Fahrzeugen, so dass*

- jede Fahrt des Fahrplans bedient wird,
- jedes Fahrzeug eine gültige Route fährt und
- die – gewichteten operativen und fixen – Kosten minimiert werden.

Das SD-VSP ist von seiner Komplexität her als einfach zu bezeichnen, da es in polynomieller Zeit optimal gelöst werden kann. Allerdings werden in der Praxis neben den oben beschriebenen Anforderungen z.T. weitere Bedingungen an die Umlaufplanung gestellt, die Erweiterungen des SD-VSP darstellen. Diese werden in den folgenden Abschnitten beschrieben und treten in der Praxis einzeln oder in Kombination unterschiedlicher Erweiterungen auf.

2.3.3 Umlaufplanung mit mehreren Depots und Fahrzeugtypen

Viele Verkehrsunternehmen verfügen über mehrere Depotstandorte, die individuelle Kapazitäten für Fahrzeuge haben. Diese können durch eine unzureichende Kapazität in einem einzelnen Depotstandort begründet sein. Ein weiterer Motivationsgrund zur Unterhaltung mehrerer Depotstandorte liegt aber auch darin, dass durch die Ermöglichung kürzerer Ein- und Ausrückfahrten eine Senkung der operativen Kosten erreicht werden kann. Da die Umlaufplanung in der Regel für einen Tag durchgeführt wird, muss sichergestellt werden, dass die Belegung der Depots am Ende des Tages der Ausgangssituation entspricht. Das heisst, dass ein Fahrzeug am Ende seines Umlaufs in das gleiche Depot zurückkehren muss, von dem aus es gestartet ist. Dadurch wird sichergestellt, dass die Fahrzeugbelegung

am Ende des Tages der Ausgangssituation entspricht, so dass die Planung für einen oder mehrere Tage wiederverwendet werden kann. Die hier beschriebene Problemstellung geht von bereits bestehenden Depotstandorten aus.

Das um die oben beschriebenen Anforderungen erweiterte SD-VSP ist als *Mehrdepot-Umlaufplanungsproblem* (engl.: *multiple depot vehicle scheduling problem*) (MD-VSP) bekannt. Die in Abschnitt 2.3.1 aufgestellte formale Definition des SD-VSP kann durch folgende Erweiterungen als MD-VSP definiert werden: Sei D die Menge aller Depots und w_d mit $d \in D$ die individuellen Depotkapazitäten. Die geplanten operativen Kosten für einen Umlauf sind abhängig von dem Standort des Depots, da sich längere Aus- und Einrückfahrten ergeben können. Seien U_d alle Umläufe des Depots $d \in D$. Daraus ergeben sich die zu minimierenden geplanten Kosten für das MD-VSP:

$$c^{total} = \sum_{d \in D} \sum_{u \in U_d} c_{u,d}^{seq} + \sum_{u \in U} c^{veh}$$

Das MD-VSP gehört für $|D| \geq 2$ zu der Problemklasse der NP-schweren Problemstellungen, Die theoretische Komplexität ist in [Bertossi et al., 1987] gezeigt.

2.3.4 Umlaufplanung mit Fahrzeugtypen

Neben der Zugehörigkeit zu einem Depot können sich die Fahrzeuge auch in ihren Eigenschaften – wie Geschwindigkeit, Größe oder Kosten – unterscheiden. Eine festgelegte Kombination von Eigenschaften eines Fahrzeugs wird im Folgenden *Fahrzeugtyp* genannt. Für den Fall, dass sich die Fahrzeuge in ihren Typen unterscheiden, aber jede Servicefahrt von einem beliebigen Fahrzeug(typen) bedient werden darf, müssen die Umläufe einem Depot und einem Fahrzeugtypen zugewiesen werden. Aufgrund der nicht eingeschränkten Bedienung aller Fahrten kann jede Kombination aus Depot und Fahrzeugtyp als ein Depot angesehen werden, dessen Kapazität die Kapazität des Fahrzeugtypen in dem entsprechenden Depot ist. Zur Vereinfachung der Erklärungen wird für die weiteren Ausführungen dieser Arbeit der Begriff „Depot“ als Kombination aus Fahrzeugtyp und Depotstandort verwendet.

Für den Fall, dass sämtliche Fahrplanfahrten mit einem beliebigen Fahrzeugtypen bedient werden können, kann das Problem als MD-VSP gelöst werden, bei dem die typabhängigen Kosten einbezogen werden. Die operativen und fixen Kosten sind abhängig von dem verwendeten Fahrzeugtyp, da zum Beispiel der Verbrauch und die Investitionskosten aufgrund unterschiedlicher Bauweise variieren können. Aufgrund der Möglichkeit einer gleichartigen Betrachtung wird das MD-VSP mit Fahrzeugtypen und ohne Einschränkung der Bedienung für Ser-

vicefahrten für den weiteren Verlauf dieser Arbeit unter der Problemstellung des MD-VSP berücksichtigt.

Neben der Berücksichtigung unterschiedlicher Depotstandorte und Fahrzeugtypen existieren allerdings häufig weitere Einschränkungen für Servicefahrten, die es nur einer Teilmenge von Fahrzeugtypen erlaubt eine Fahrt zu bedienen. Zum Beispiel können auf Linien, die an Krankenhäusern oder Altenheimen entlang führen, nur Fahrzeugtypen mit einer behindertengerechten Bauweise (z.B. Niederflurbusse) verwendet werden. Häufig führen auch streckenbedingte Engpässe, wie Tunnel oder Engstellen der Straße, dazu, dass einige Fahrzeugtypen (z.B. Gelenkbusse) nicht für die Bedienung einer Servicefahrt geeignet sind. Die Einschränkung der Fahrzeugtypen ist nicht ausschließlich von der Route einer Linie abhängig, sondern kann auch von den Bedienzeiten abhängen. So kann es z.B. auf einer Linie, die tagsüber ein hohes Fahrgastaufkommen hat (z.B. durch Geschäftszeiten), in den Nachtstunden zu einer Veränderung der Fahrzeugtypeinschränkungen kommen (vgl. [Grötschel et al., 1997]).

Eine Fahrt kann in der Problemstellung der *Umlaufplanung mit Fahrzeugtypgruppen* (engl: *multiple vehicle type group vehicle scheduling problem*) (MVTG-VSP) ausschließlich von einem Fahrzeug einer *Fahrzeugtypgruppe* bedient werden, wobei eine Fahrzeugtypgruppe eine Teilmenge aller Fahrzeugtypen ist. Die maximal verfügbaren Fahrzeuge (Kapazität) sind je Depotstandort und Fahrzeugtyp gegeben. Aus diesem Grund können die Fahrten in der Problemstellung des MVTG-VSP nur von einer Untermenge aller Depots – im Sinne einer Kombination von Depotstandort und Fahrzeugtyp – bedient werden.

Seien U_d die Menge der Umläufe aus Depot $d \in D$. Formal kann das in Abschnitt 2.3.3 beschriebene MD-VSP zum MVTG-VSP erweitert werden, indem für jede Servicefahrt $t \in T$ die Menge der zulässigen Depots $V_t \subseteq D$ definiert wird. Als zusätzliche Nebenbedingung ist für die Gültigkeit eines Umlaufs, neben der Kompatibilität der Fahrten, die Einhaltung der Fahrzeugtypgruppen erforderlich. Das heisst, für alle Servicefahrten eines Umlaufs müssen die die Fahrzeugtypgruppen mindestens einen gemeinsamen Fahrzeugtypen enthalten:

$$\bigcap_{t \in U} V_t \neq \emptyset, \quad \forall U \in U_d, \quad \forall d \in D$$

Dieser Umstand ist für die Konzeption von Methoden zur Lösung des MVTG-VSP wichtig, da nicht jede Sequenz aus kompatiblen Fahrten gültig ist. Auch die theoretische Komplexität des MVTG-VSP ist durch Betrachtung unterschiedlicher Fahrzeugtypen im allgemeinen Fall NP-schwer (vgl. [Lenstra und Kan, 1981]).

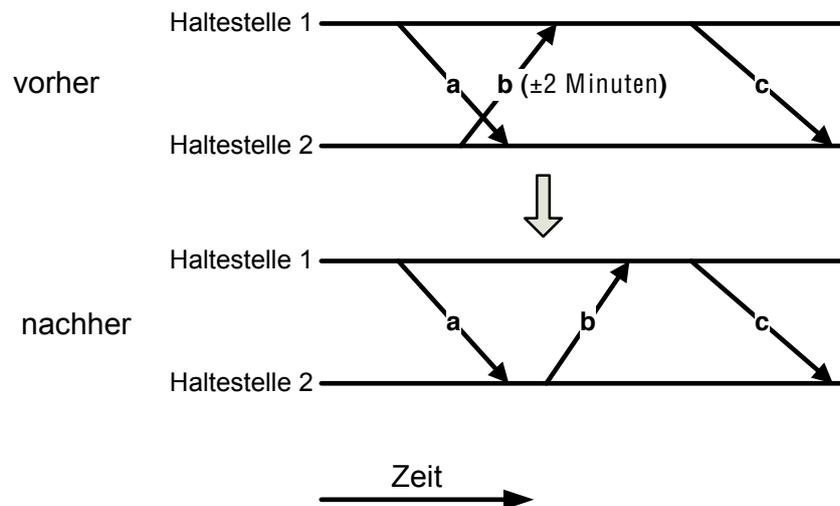


Abbildung 2.4: Reduzierung der benötigten Fahrzeuge durch Zeitfenster

2.3.5 Umlaufplanung mit Zeitfenstern

Wie in Abschnitt 2.1 beschrieben wurde, wird die Umlaufplanung auf der Basis eines bestehenden Fahrplans durchgeführt und ist somit eine von der Fahrplanerstellung unabhängige Planungsaufgabe. Bei der Lösung der *Umlaufplanung mit Zeitfenstern* (engl: *vehicle scheduling problem with time windows*) (VSP-TW) sollen durch die zeitliche Verschiebung der Servicefahrten in einem individuell begrenzten Intervall weitere Einsparpotentiale ermöglicht werden. Durch die Veränderung der Abfahrts- und Ankunftszeiten einer Fahrt können – unter Beibehaltung der Fahrtdauer – neue Anschlüsse ermöglicht werden. Diese können sowohl über eine direkte Anschlussmöglichkeit an einer Endhaltestelle als auch durch zusätzliche Kompatibilitäten von Fahrten mittels Umsetzfahrten erreicht werden. Der Umfang der möglichen zeitlichen Verschiebung von Servicefahrten wird auch *Zeitfenster* genannt. Die durch Zeitfenster ermöglichten Anschlüsse können neben der Reduzierung der operativen Kosten auch zu einer Einsparung von Fahrzeugen und resultierend zu geringeren fixen Kosten führen. Abbildung 2.4 zeigt schematisch die durch Zeitfenster mögliche Reduzierung benötigter Fahrzeuge: die obere Abbildung stellt einen Fahrplan mit drei Servicefahrten (a, b und c) als Pfeile über Zeit (horizontal) und Ort (vertikal) dar. Der ursprüngliche Fahrplan kann durch die zeitliche Überschneidung von Fahrten a und b nicht von einem Fahrzeug bedient werden. Bei Zulassung einer zeitlichen Verschiebung (z.B. um zwei Minuten für Fahrt b) können alle Fahrten, wie in der unteren Abbildung dargestellt, durch ein Fahrzeug abgedeckt werden. Obwohl die Verwendung

von Zeitfenstern in der Umlaufplanung hohe Potentiale für die Kosteneinsparung bedeuten, sind die Anwendungsszenarien durch die Auswirkungen auf den Originalfahrplan in der Praxis stark eingeschränkt. Mögliche Anwendungsfälle werden in Abschnitt 2.2 näher erläutert.

Um die in den vorhergehenden Abschnitten beschriebenen Umlaufplanungsprobleme mit Zeitfenstern für Servicefahrten zu erweitern, seien q_t bzw. p_t die Angaben der möglichen zeitlichen Verschiebung von Fahrt $t \in T$ nach vorne bzw. nach hinten. Die für Servicefahrt t mögliche Abfahrtszeit e_t^{tw} kann durch eine relative Verschiebung innerhalb des individuell vorgegebenen Intervalls liegen:

$$e_t^{tw} \in [e_t - q_t; e_t + p_t] \quad \forall t \in T$$

Die Ankunftszeit der Servicefahrt ist aufgrund der als konstant angenommenen Fahrtdauer gegeben durch $f_t^{tw} = f_t - (e_t - e_t^{tw})$. Zwei Servicefahrten i und j sind dementsprechend kompatibel ($i \alpha j$), wenn gilt: $f_i^{tw} + d_{ij} \leq e_j^{tw}$.

Das VSP-TW gehört der Problemklasse der NP-vollständigen Probleme an, da es durch Betrachtung von einem Fahrzeug und einem Depot auf das *Handlungsreisendenproblem* (engl: *traveling salesman problem*) (TSP) mit Zeitfenstern reduziert werden kann (vgl. [Savelsbergh, 1985]).

2.3.6 Umlaufplanung mit Routenrestriktionen

Unter der *Umlaufplanung mit Routenrestriktionen* (engl: *vehicle scheduling problem with route constraints*) (VSP-RC) werden allgemein Problemstellungen verstanden, bei denen das VSP weiteren Restriktionen unterliegt, die eine Begrenzung der umlaufabhängigen Kapazität darstellen. Dies kann zum Beispiel eine zeitliche Restriktion sein, die die Umlaufdauer begrenzt oder eine maximale Wegstrecke, die durch eine begrenzte Reichweite der Fahrzeuge begründet ist. Eine weitere Umlaufrestriktion (auch Ressource genannt) kann die Anzahl unterschiedlicher Linien oder Linienwechsel eines Umlaufs darstellen, die aufgrund von technischen oder organisatorischen Rahmenbedingungen begrenzt sein soll. In der Praxis werden diese Routenrestriktionen häufig als weiche Restriktionen betrachtet und als eine, den Kosten untergeordnete, Zielgröße betrachtet. Im VSP-RC werden sie allerdings als harte Restriktion einbezogen und können daher im Vergleich zum VSP ohne Routenrestriktionen zu einer Erhöhung der geplanten Kosten führen.

Das VSP-RC gehört zu der Problemklasse der NP-schweren Probleme (vgl. [Lenstra und Kan, 1981]).

2.3.7 Weitere praxisrelevante Anforderungen

Neben den beschriebenen Problemstellungen werden an die Erstellung von Umlaufplänen weitere Anforderungen gestellt wie z.B. die Berücksichtigung von Vorbereitungs- und Wendezeiten für Fahrten, eine Rückkehr ins Depot zur Ermöglichung von Pausenzeiten, tageszeitabhängige Verbindungsangaben oder die Berücksichtigung einer Mindestausnutzung von Depotkapazitäten. Eine Vielzahl dieser Anforderungen ist in [Kliwer, 2005] beschrieben und soll an dieser Stelle nicht weiter diskutiert werden, da sie in der Modellierung berücksichtigt werden können und keine Anpassung der Lösungsmethoden erfordern. Die in den nachfolgenden Kapiteln vorgestellten Methoden können diese Erweiterungen wie in [Kliwer, 2005] gezeigt berücksichtigen.

3 Grundlegende Optimierungstechniken

In den nachfolgenden Kapiteln werden Modelle und Verfahren für die Busumlaufplanung vorgestellt. Um die Verständlichkeit der Ausführungen sicherzustellen, sollen in diesem Kapitel die notwendigen theoretischen Grundlagen erläutert werden. Für eine weiterführende Diskussion der Details wird an den entsprechenden Stellen auf geeignete Literatur verwiesen.

3.1 Mathematische Optimierung

Seit über 50 Jahren werden Methoden der mathematischen Optimierung (*engl.: mathematical programming*) für die Lösung realer Planungsprobleme verwendet (vgl. [Bixby, 2002]). Dazu wird die reale Problemstellung in der Form eines mathematischen Modells formuliert. Das Modell besteht aus einer *Zielfunktion* (*engl.: objective function*), in der gewichtete *Entscheidungsvariablen* (*engl.: decision variables*) maximiert oder minimiert werden sowie zusätzlicher *Restriktionen* (*engl.: constraints*), die bei der Bestimmung der Variablenausprägungen zu berücksichtigen sind (vgl. [Suhl und Mellouli, 2006]).

Lineare Optimierung Im Falle ausschließlich linearer Restriktionen und einer linearen Zielfunktion nennt man ein mathematisches Modell *lineares Optimierungsproblem* (*engl.: linear program*) (*LP*) (vgl. [Chvatal, 1983]). Die Entscheidungsvariablen können reelle Werte annehmen. Für die Lösung von LPs existieren effiziente Algorithmen, die eine exakt optimale Lösung für das Problem finden können. Dies sind zum Beispiel Innere-Punkte-Verfahren (vgl. [Karmarkar, 1984]), die einen polynomiellen Laufzeitaufwand haben, oder Simplex-Methoden (vgl. [Dantzig et al., 1955]). Für gegebene Vektoren der Entscheidungsvariablen x und der Kostenfaktoren c sowie gegebener Restriktionen $Ax = b$ lautet ein lineares Optimierungsproblem mit Minimierung der Zielfunktion (vgl. [Wolsey, 1998]):

$$\min cx, \quad s.t. Ax = b, \quad x \geq 0 \tag{3.1}$$

(Gemischt)-ganzzahlige Optimierung Im Falle der zusätzlichen Bedingung, dass alle (bzw. ein Teil der) Entscheidungsvariablen ausschließlich ganzzahlige Wertausprägungen annehmen dürfen, handelt es sich um ein *ganzzahliges Optimierungsproblem* (engl: *integer program*) (*IP*) (bzw. *gemischt-ganzzahliges Optimierungsproblem* (engl: *mixed integer program*) (*MIP*)). Die Formulierung des LPs aus (3.1) als ganzzahliges Optimierungsproblem lautet:

$$\min cx, \quad s.t. Ax = b, \quad x \in \mathbb{R} \quad (3.2)$$

Im Gegensatz zu den LPs gehören diese Modelle zu der Klasse der \mathcal{NP} -harten Probleme (vgl. [Wolsey, 1998]). Da für diese Klasse keine Algorithmen bekannt sind, die in polynomieller Zeit eine optimale Lösung finden, basieren typische Verfahren auf der impliziten Enumerierung möglicher Ausprägungen der Entscheidungsvariablen. Bei diesen wird zur Laufzeit durch Ausnutzung von unteren Schranken eine Vielzahl an Lösungen ausgeschlossen. Zu diesen Verfahren gehören LP-basierte Branch & Bound (B&B) oder Branch & Cut Verfahren (B&C), die auch in kommerziellen Optimierungs-Bibliotheken eingesetzt werden (vgl. z.B. [Suhl, 2000], [ILOG, 2007]). Hier wird nach Berechnung der LP-Relaxation – also der Lösung des MIPs unter Vernachlässigung der Ganzzahligkeitsbedingungen – die Lösung auf fraktionelle Variablenwerte untersucht. Im B&B-Algorithmus wird das Problem auf der fraktionellen Lösung einer Entscheidungsvariable in zwei Unterprobleme geteilt. Im B&C werden dem Modell zusätzliche Restriktionen hinzugefügt, die die Formulierung verstärken sollen. Die Lösung einer LP-Relaxation, bei der die Variablen fraktionelle Werte annehmen dürfen, liefert eine untere Schranke für die Lösung des MIPs (vgl. [Nemhauser und Wolsey, 1999]).

Total unimodulare Matrizen Eine besondere Eigenschaft einiger MIPs kann die Lösung des ganzzahligen Problems vereinfachen: sei A die $(n \times m)$ - Matrix, die die Koeffizienten der n Entscheidungsvariablen für alle m Restriktionen eines MIPs enthält. Besitzt diese Koeffizientenmatrix A die Eigenschaft der totalen Unimodularität, so besteht eine optimale Lösung der LP-Relaxation ausschließlich aus ganzzahligen Werten und ist somit auch eine optimale Lösung für das MIP. Das MIP kann also in polynomieller Zeit optimal gelöst werden.

Eine Matrix A ist total unimodular, wenn jede quadratische Teilmatrix von A eine Determinante von 1, -1 oder 0 hat. Für den Nachweis zur totalen Unimodularität von Matrix A mit Koeffizienten a_{ij} gilt als hinreichende Bedingung (vgl. [Wolsey, 1998]):

- alle Koeffizienten der Matrix sind aus $\{-1, 0, 1\}$,
- jede Spalte der Matrix enthält maximal zwei Koeffizienten ungleich 0,

- es existiert eine Aufteilung (M_1, M_2) der Zeilen von A , so dass jede Spalte j , die zwei Koeffizienten ungleich 0 enthält, folgende Gleichung erfüllt:

$$\sum_{i \in M_1} a_{ij} - \sum_{i \in M_2} a_{ij} = 0.$$

Ausgewählte Optimierungsprobleme

Zum besseren Verständnis der in den nachfolgenden Kapiteln vorgestellten Ansätze sollen an dieser Stelle einige, für die Lösung von Busumlaufplanungsproblemen relevante, Optimierungsprobleme vorgestellt werden. Da viele mathematischen Modelle zur Optimierung der Umlaufplanung die Berücksichtigung von ganzzahligen Variablen erfordern, wird insbesondere auf diese Anforderung eingegangen.

Umladeproblem Das *Umladeproblem* (engl: *minimum cost flow problem*) (MCF) gehört zu der Klasse der Netzwerkprobleme. Bei diesen basiert die Formulierung der Problemstellung auf einem Netzwerk. Anwendung finden sich zum Beispiel häufig bei Problemstellungen aus den Bereichen Logistik und Transport.

Sei $G = (N, A)$ ein gerichtetes Netzwerk mit den Mengen der Knoten N und Kanten A ($A \subseteq N \times N$). Jeder gerichteten Kante $(i, j) \in A$ ist ein Kostenwert c_{ij} pro Flusseinheit zugewiesen sowie eine minimale (maximale) Kapazität l_{ij} (u_{ij}). In einer gültigen Lösung darf die Flussgröße diese Kapazität nicht unterschreiten (überschreiten). Zu jedem Knoten $j \in N$ ist ein Bedarf b_j an Flusseinheiten gegeben. Ein positiver Bedarf entspricht einem Angebot von Flusseinheiten, ein negativer einer Nachfrage. Die Knoten werden dementsprechend in folgender Weise benannt:

$$b_j \begin{cases} < 0 : \text{Bedarfsknoten,} \\ = 0 : \text{Umladeknoten,} \\ > 0 : \text{Angebotsknoten.} \end{cases} \quad \forall j \in N$$

Gesucht ist im MCF eine kostenminimale Flusslösung, die alle Knotenbedarfe erfüllt und deren Kantenflüsse innerhalb der individuellen Kapazitäten liegen. In der mathematischen Formulierung werden die Kantenflüsse als Entscheidungsvariablen abgebildet:

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (3.3)$$

$$s.t. \sum_{i:(i,j) \in A} x_{ij} - \sum_{i:(j,i) \in A} x_{ji} = b_j \quad \forall j \in N \quad (3.4)$$

$$l_{ij} \leq x_{ij} \leq u_{ij} \quad \forall (i, j) \in A \quad (3.5)$$

In der Zielfunktion (3.3) werden die gesamten Flusskosten minimiert. Restriktionen (3.4) stellen die Erfüllung der Bedarfe sicher, indem die Summe aller eingehenden Flüsse abzüglich der Summe aller ausgehenden Flüsse eines Knotens dem Bedarf entsprechen müssen. (3.5) garantiert die Einhaltung der individuellen Kantenkapazitäten.

Unter der Voraussetzung, dass die Bedarfe für alle Knoten ($b_i \in \mathbb{N}, \forall i \in N$) und alle Kantenkapazitäten ganzzahlig sind ($l_{ij} \in \mathbb{N} \wedge u_{ij} \in \mathbb{N}, \forall (i, j) \in A$), nehmen die Werte einer optimalen Lösung auch ganzzahlige Werte an, da die Koeffizientenmatrix des Optimierungsproblems *total unimodular* ist (vgl. [Wolsey, 1998]). Das MCF kann daher mit Methoden für die Optimierung von LPs in polynomieller Zeit gelöst werden. Darüber hinaus kann die Netzwerk-Struktur des MCF in spezialisierten Algorithmen ausgenutzt werden, um die Lösungsgeschwindigkeit gegenüber allgemeinen LP-Methoden zu verbessern. Beispiele für Algorithmen dieser Art sind Netzwerk-Simplex-Methoden oder Primal-Duale Methoden (vgl. [Ahuja et al., 1993]).

Transportproblem Das *Transportproblem* (engl: *transportation problem*) (TP) ist ein Spezialfall des MCF. Die Knotenmenge N besteht aus einer Menge von Angebotsknoten N_1 mit Angeboten $a_i \forall i \in N_1$ sowie einer Menge von Bedarfsknoten N_2 mit Bedarfen $b_j \forall j \in N_2$. Jeder Knoten des Netzwerks gehört zu einer der beiden Mengen, so dass keine Umladeknoten vorhanden sind. Es existieren keine Kanten zwischen zwei Angebotsknoten (bzw. zwei Bedarfsknoten). Somit ist $G = (N, A)$ mit $N = N_1 \cup N_2$ ein bipartiter Graph mit $\mathcal{O}(|N_1| \cdot |N_2|)$ Kanten und das TP kann wie folgt formuliert werden:

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (3.6)$$

$$s.t. \sum_{j:(i,j) \in A} x_{ij} = a_i \quad \forall i \in N_1 \quad (3.7)$$

$$\sum_{i:(i,j) \in A} x_{ij} = b_j \quad \forall j \in N_2 \quad (3.8)$$

Die Zielfunktion (3.6) minimiert die Summe der Flusskosten. Restriktionen (3.7) bzw. (3.8) stellen die Erfüllung der Angebote bzw. Bedarfe sicher. Bedarfe und Angebote sind in dieser Darstellung bewusst getrennt, um die zugrunde liegende Struktur eines bipartiten Graphen besser zu verdeutlichen. Diese Struktur kann in spezialisierten Methoden (z.B. der MODI-Methode) zur Verbesserung der Lösungsgeschwindigkeit genutzt werden (vgl. [Domschke und Drexl, 2005]).

Mehrgüter Flussproblem Im Gegensatz zu den oben vorgestellten Problemen, in denen homogene Typen von Flusseinheiten betrachtet werden, wird im *Mehrgüter Flussproblem* (engl: *multicommodity flow problem*) (MCCF) der Transport von unterschiedlichen Gütern innerhalb eines Netzwerks betrachtet. Sei K die Menge dieser unterschiedlichen Güter, dann ist das MCCF gegeben durch:

$$\min \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k x_{ij}^k \quad (3.9)$$

$$s.t. \sum_{i:(i,j) \in A} x_{ij}^k - \sum_{i:(j,i) \in A} x_{ji}^k = b_j^k \quad \forall j \in N, \forall k \in K \quad (3.10)$$

$$\sum_{k \in K} x_{ij}^k \leq U_{ij} \quad \forall (i,j) \in A \quad (3.11)$$

$$l_{ij}^k \leq x_{ij}^k \leq u_{ij}^k \quad \forall (i,j) \in A, \forall k \in K \quad (3.12)$$

Die Gesamtkosten über alle Güter werden in der Zielfunktion (3.9) minimiert. Restriktionen (3.10) stellen die Erfüllung der Knotenbedarfe für jedes Gut $k \in K$ sicher. Neben den minimalen und maximalen Kapazitäten einer Kante pro Gut (3.12) sind die Güterflüsse über *Bündelrestriktionen* (engl: *bundle constraints*) (3.11) miteinander verbunden. Diese stellen sicher, dass die Summe aller Güterflüsse einer Kante (i,j) eine globale Gesamtkapazität von U_{ij} nicht überschreitet. Die Koeffizientenmatrix eines MCCF ist nicht total unimodular, sobald zwei oder mehr Güter betrachtet werden.

Für den Fall, dass eine ganzzahlige Lösung erforderlich ist, kann das Modell um folgende Restriktionen erweitert werden:

$$x_{ij} \in \mathbb{N} \quad \forall (i,j) \in A \quad (3.13)$$

Das resultierende *Ganzzahlige Mehrgüter Flussproblem* (engl: *integer multicommodity flow problem*) (IMCCF) ist aufgrund der Bündelrestriktionen und der damit nicht mehr total unimodularen Koeffizientenmatrix NP-vollständig (vgl. [Garrey und Johnson, 1979]).

Im Falle von unabhängigen Güterflüssen kann für jedes Gut $k \in K$ ein MCF formuliert werden und die entstehenden Unterprobleme jeweils durch Verfahren mit polynomieller Laufzeit gelöst werden.

Set Partitioning Problem Im *Set Partitioning Problem* (SPP) wird eine Menge von Entscheidungsmöglichkeiten N betrachtet, von der jedes Element eine Teilmenge aller Eigenschaften M abdeckt. Eine gültige Lösung enthält eine Teilmenge gewählter Entscheidungsmöglichkeiten, so dass durch sie jede Eigenschaft

genau einmal abgedeckt ist. Ziel der Optimierung ist das Finden einer gültigen Lösung mit minimalen Kosten $c_j \forall j \in N$. In der mathematischen Formulierung repräsentieren die Entscheidungsvariablen die unterschiedlichen Möglichkeiten N :

$$\min \sum_{j \in N} c_j x_j \quad (3.14)$$

$$s.t. \sum_{j \in N} a_{ij} x_j = 1 \quad \forall m \in M \quad (3.15)$$

$$x_j \in \{0, 1\} \quad \forall j \in N \quad (3.16)$$

Die Zielfunktion (3.14) minimiert die Gesamtkosten. Alternativ können Gesamtkosten durch individuelle Kostenfaktoren c_j minimiert werden. In (3.15) wird sichergestellt, dass jede Eigenschaft $i \in M$ genau einmal abgedeckt wird. Die diskrete Modellierung einer Entscheidungswahl ist durch die Definition als (0,1)-Variablen in (3.16) modelliert. Das SPP gehört zu der Klasse der NP-harten Probleme (vgl. [Garey und Johnson, 1979]).

3.2 Column Generation

Das *Column Generation Verfahren* ist eine Methode zur Lösung von linearen Problemen mit einer großen Anzahl an Entscheidungsvariablen. Motiviert ist die Methode durch das Prinzip der Dantzig-Wolfe-Dekomposition, durch die zu einem Modell eine äquivalente Formulierung mit einer großen Anzahl an Variablen, aber weniger Restriktionen, gebildet werden kann (vgl. [Dantzig und Wolfe, 1960]). Sei P ein beliebiges lineares Problem mit einer großen Anzahl von Variablen und sei P gegeben durch:

$$\min \sum_{j \in N} c_j x_j \quad (3.17)$$

$$s.t. \sum_{j \in N} a_{ij} x_j = b_i \quad \forall i \in M \quad (3.18)$$

$$x_j \geq 0 \quad \forall j \in N \quad (3.19)$$

Anstatt das vollständige LP zu lösen wird im Column Generation Verfahren eine Sequenz von Unterproblemen (*restricted master problems*) (RMPs) gelöst. Diese beinhalten jeweils nur eine (in der Regel vergleichsweise kleine) Untermenge der Variablen von P . Nach jeder Lösung eines RMPs wird im *pricing problem* (PP) über die Aufnahme einer die Lösung potentiell verbessernden Variable in das

Algorithmus 3.1 : Column Generation Verfahren

- (Schritt 1) **Initialisierung**
 Wähle initiale Untermenge von Variablen $N^0 \subset N$
 $t = 0$
- (Schritt 2) **Lösung des restricted master problem**
 Löse $RMP(N^t)$ und speichere duale Werte π^t
- (Schritt 3) **Lösung des pricing problem**
 Löse $PP(\pi^t)$ zur Identifikation einer aufzunehmenden
 Variable n' mit $\bar{c}_{n'} < 0$
 Wenn $\bar{c}_x \geq 0 \ \forall x \notin N^t$, dann Abbruch
- (Schritt 4) **Aktualisierung des restricted master problem**
 $N^{t+1} = N^t \cup n'$
 $t = t + 1$
 Gehe zu Schritt 2
-

RMP entschieden. Der generische Ablauf eines Column Generation Verfahrens ist in Algorithmus 3.1 dargestellt und wird im Folgenden näher erläutert.

In Schritt 1 wird das erste RMP so initialisiert, dass das Problem lösbar und beschränkt ist. Dies kann zum Beispiel durch die Identifikation geeigneter Variablen durch eine Heuristik für das Gesamtproblem geschehen. Alternativ kann durch das Einfügen von künstlichen Variablen mit sehr hohen (dominierenden) Kosten die Lösbarkeit sichergestellt werden. Sollten nach Beendigung des Column Generation Verfahrens künstliche Variablen in der Lösung sein, so ist das Gesamtproblem nicht lösbar, da – die Dominanz der künstlichen Variablenkosten vorausgesetzt – keine Lösung ohne künstliche Variablen existiert. Durch die Lösung des RMPs in Schritt 2 werden die dualen Werte (π) der Restriktionen berechnet. Da eine Untermenge des Gesamtproblems betrachtet wird, stellt der Zielfunktionswert des RMPs zum Zeitpunkt t ($Z(RMP^t)$) eine obere Schranke für den optimalen Zielfunktionswert des Gesamtproblems dar: $Z(RMP^t) \geq Z(P)$. In Schritt 3 wird nach einer Variable n' gesucht, die nicht in der Variablenmenge des aktuellen RMPs vorhanden ist ($n' \notin N^t$) und negative reduzierte Kosten $\bar{c}_{n'}$ hat. Die reduzierten Kosten können mit Hilfe der dualen Werte aus Schritt 2 für jede Variable wie folgt berechnet werden:

$$\bar{c}_{n'} = c_{n'} - \sum_{i \in M} a_{ij} \pi_i$$

Es gibt eine Vielzahl unterschiedlicher Strategien zur Auswahl einer Variable mit negativen reduzierten Kosten. Häufige Anwendung findet das *Dantzig Pricing*, bei

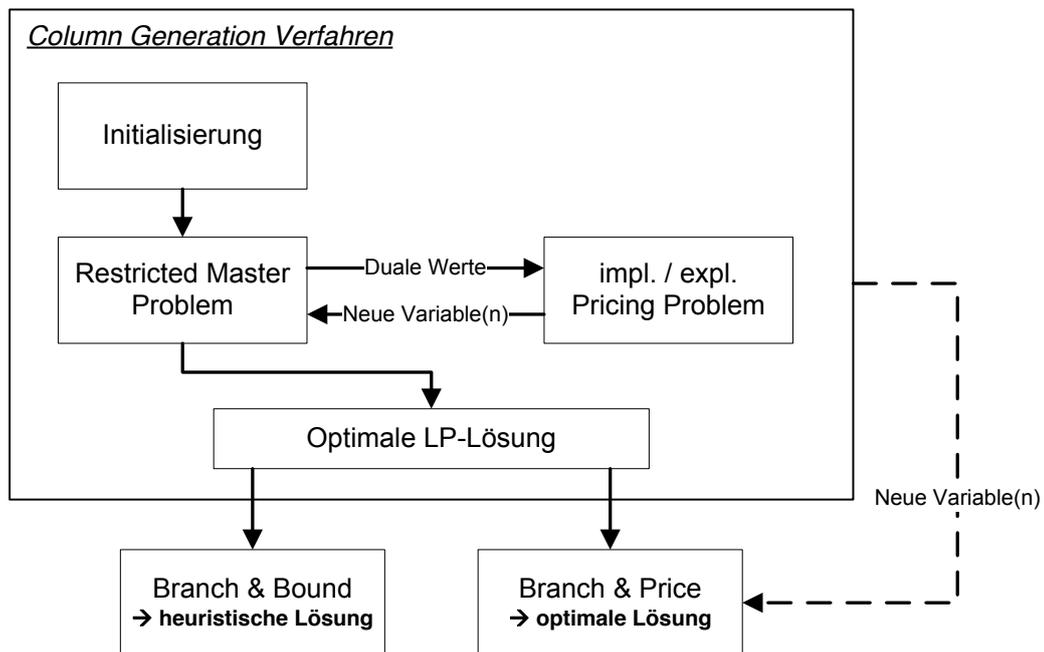


Abbildung 3.1: Column Generation für MIPs

dem die Variable mit den geringsten reduzierten Kosten ausgewählt wird. Sollte im PP keine Variable mit negativen reduzierten Kosten gefunden werden, so stellt die Lösung des aktuellen RMPs eine optimale Lösung für P dar. Andernfalls wird die identifizierte Variable n' dem aktuellen RMP hinzugefügt und die nächste Iteration durchgeführt (Schritt 4).

Da das Hinzufügen von nur einer Variable je Iteration zu einer hohen Anzahl von Iterationen des Algorithmus führt, können zur Beschleunigung mehrere Variablen im PP identifiziert und hinzugefügt werden (*multiple pricing*). Eine weitere Strategie zur Beschleunigung des Algorithmus ist die Eliminierung von Variablen mit hohen reduzierten Kosten aus den RMPs. Dadurch kann die Modellgröße der RMPs verkleinert werden und durch eine kürzere Lösungszeit der RMPs die Lösungsdauer des gesamten Algorithmus reduziert werden. Es existiert eine Vielzahl von weiteren Beschleunigungstechniken für das Column Generation Verfahren, die in [Desaulniers et al., 2002] ausführlich beschrieben werden.

Der Ablauf des Column Generation Verfahrens ist in Abbildung 3.1 zusammenfassend dargestellt. Nach der Initialisierung werden im Wechsel das RMP und das PP gelöst. Es existieren Modelle, die aufgrund ihrer sehr großen Anzahl von Variablen nicht im Speicher gehalten werden können. In diesen Fällen ist eine implizite Suche nach verbessernden Variablen im PP notwendig (*implizites Column Generation*). Wenn dagegen alle Variablen im Speicher gehalten werden

können und somit alle reduzierten Kosten berechnet werden können, wird das Verfahren *explizites Column Generation* genannt.

Die theoretische Problemkomplexität des Column Generation Verfahrens ist abhängig von der Methode zur Lösung des Pricing Problems, da das RMP in der Komplexitätsklasse \mathcal{P} ist (vgl. [Desrosiers und Lübbecke, 2005]).

In der Literatur gibt es eine Vielzahl von Beschreibungen der unterschiedlichen Anwendungen sowie weiterer Anpassungen des Column Generation Verfahrens. Übersichten sind in [Desaulniers et al., 2005] und [Desrosiers und Lübbecke, 2005] zu finden. Wenn das Problem P ein MIP ist und somit ganzzahlige Variablen enthält, kann das Problem zur Lösung des initialen LPs im B&B Verfahren verwendet werden. Für die im letzten RMP enthaltenen Variablen wird eine MIP-Lösung bestimmt. Da die LP-Lösung des Column Generation Verfahrens zwar optimal bezüglich kontinuierlicher Variablen ist, aber in einer optimalen MIP-Lösung auch Variablen enthalten sein können, die im letzten RMP nicht vorhanden waren, gibt dieses Vorgehen keine Garantie für eine optimale MIP-Lösung. Um diese unter Verwendung des Column Generation Verfahrens zu bestimmen, haben [Barnhart et al., 1998] das Branch & Price (B&P) Verfahren entwickelt, bei dem im Suchbaum des B&B mögliche, verbessernde Variablen nachgeneriert werden (s. Abb. 3.1).

3.3 Lagrange Relaxation

Methoden zur Lösung von MIPs können durch untere Schranken beschleunigt werden (vgl. [Wolsey, 1998]). Ein gängiges Verfahren ist die Bestimmung durch Relaxation der Ganzzahligkeitsbedingungen (LP-Relaxation). Obwohl Methoden zur Lösung von LPs mit polynomieller Laufzeit existieren, sind die Verfahren für viele Problemstellungen ineffizient, die zum Beispiel eine große Anzahl an Entscheidungsvariablen enthalten und/oder die aufgrund komplexer Restriktionen schwierig zu lösen sind.

Eine weitere Methode zur Bestimmung unterer Schranken ist die Lagrange Relaxation, bei der statt der Relaxation der Ganzzahligkeitsbedingungen einige oder alle Restriktionen des Modells relaxiert werden. In vielen praxisrelevanten Problemstellungen kann eine Untermenge der Restriktionen identifiziert werden, die das Problem schwierig zu lösen macht, so dass nach Relaxation dieser Restriktionen ein einfacheres – bestenfalls mit einer unimodularen Koeffizientenmatrix und damit in polynomieller Zeit zu lösendes – MIP entsteht (vgl. [Beasley, 1993]). Sei P das zu lösende MIP:

$$P : \quad \min \sum_{j \in N} c_j x_j \quad (3.20)$$

$$s.t. \sum_{j \in N} a_{ij} x_j = d_i \quad \forall i \in M_1 \quad (3.21)$$

$$\sum_{j \in N} b_{ij} x_j = e_i \quad \forall i \in M_2 \quad (3.22)$$

$$x_j \in \mathbb{N} \quad \forall j \in N \quad (3.23)$$

Sei $M = M_1 \cup M_2$ die Menge der Restriktionen und $M_1 \cap M_2 = \emptyset$. Sei weiterhin M_1 die Menge der „schwierigen“ Nebenbedingungen. Durch Relaxation der Restriktionen M_1 und der Einführung eines Vektor λ ist das *Lagrange Problem* $LR(\lambda)$ zu P wie folgt definiert:

$$LR(\lambda) : \quad \min \sum_{j \in N} c_j x_j + \sum_{i \in M_1} \lambda_i (d_i - \sum_{j \in N} a_{ij} x_j) \quad (3.24)$$

$$s.t. \sum_{j \in N} b_{ij} x_j = e_i \quad \forall i \in M_2 \quad (3.25)$$

$$x_j \in \mathbb{N} \quad \forall j \in N \quad (3.26)$$

Die Zielfunktion ist für jede relaxierte Restriktion $i \in M_1$ um einen Strafwert $\lambda_i(d_i - \sum_{j \in N} a_{ij} x_j)$ ergänzt. Sollte die Restriktion in einer Lösung erfüllt sein, so beträgt der Strafwert 0. Andernfalls ist die Höhe der Bestrafung abhängig von Abweichung von Restriktion und dem Straffaktor λ_i . Die in Vektor λ enthaltenen Koeffizienten werden Lagrange Multiplikatoren genannt. Die Lösung von $LR(\lambda)$ ist eine untere Schranke für das ursprüngliche Problem P .

Um eine möglichst gute untere Schranke zu bekommen, müssen die Lagrange Multiplikatoren λ geeignet gewählt werden. Daher sollte das Lagrange Problem $LR(\lambda)$ – und damit die untere Schranke – maximiert werden. Das resultierende Problem wird als das *Lagrange Dual Problem* (LDP) bezeichnet:

$$LDP : \quad \max LR(\lambda) \quad (3.27)$$

Die Qualität der bestimmten unteren Schranke ist abhängig von dem definierten Lagrange Problem: hat $LR(\lambda)$ eine unimodulare Koeffizientenmatrix und somit die Ganzzahligkeitseigenschaft, so ist die optimale Lösung des LDP gleich der LP-Relaxation von P . Wenn das Lagrange Problem diese Eigenschaft nicht erfüllt, so ist die untere Schranke besser oder gleich der LP-Relaxation von P :

$$\begin{aligned} LR(\lambda) \text{ unimodular} &\rightarrow LP^* = LDP^* \\ LR(\lambda) \text{ nicht unimodular} &\rightarrow LP^* \leq LDP^* \end{aligned}$$

Zur Bestimmung guter oder optimaler Lagrange Multiplikatoren kann zum Beispiel das Subgradientenverfahren verwendet werden (vgl. [Held et al., 1974]). Die durch dieses Vorgehen bestimmten Multiplikatoren können auch als duale Werte innerhalb des Column Generation Verfahrens verwendet werden (vgl. [Huisman et al., 2005]). Für weitere Details zur Lagrange Relaxation sei auf [Geoffrion, 1974] verwiesen.

3.4 Meta-Heuristiken

In der Praxis auftretende Planungsprobleme enthalten häufig eine ausgeprägte kombinatorische Komplexität und/oder schwierige Nebenbedingungen. Viele praxisrelevante Problemstellungen lassen sich aus diesem Grund mit exakten Methoden der mathematischen Optimierung nicht oder nur mit einem hohen zeitlichen Aufwand optimal lösen. Daher spielen Heuristiken zur Bestimmung von gültigen Lösungen eine wichtige Rolle. Neben spezialisierten Verfahren, die auf die Lösung einer bestimmten Problemstellung ausgerichtet sind (z.B. [Lin und Kernighan, 1973] zur Lösung des *traveling salesman problem* (TSP)) existiert die Methodenklasse der *Meta-Heuristiken*. Diese Verfahren stellen ein problemunabhängiges und abstraktes Prinzip zur heuristischen Lösung von komplexen Optimierungsproblemen dar, das (theoretisch) auf alle Optimierungsprobleme anwendbar ist (vgl. [Michalewicz und Fogel, 2000]). Viele Meta-Heuristiken basieren auf der Analogie zu einem in der Natur auftretenden Phänomen und gestalten die Informationsauswertung und Steuerung der Lösungssuche durch die Adaption dieses natürlichen Verhaltens. Meta-Heuristiken basieren nach [Michalewicz und Fogel, 2000] auf den Konzepten der

- **Darstellung** des Problems (*engl: representation*),
- dem **Ziel** der Optimierung (*engl: objective*) und
- der **Bewertungsfunktion** (*engl: evaluation function*).

Die Darstellung eines Problems definiert die Art und Größe des Lösungsraums für die Meta-Heuristik. Das Ziel wird durch einen eindeutigen Ausdruck der bestmöglichen Lösung definiert. Die Bewertungsfunktion bestimmt zu jedem möglichen Punkt im Lösungsraum einen kardinalen Wert, der die Qualität der Lösung misst. Neben der Bewertung der Güte kann die Funktion auch Bestrafungs- oder Steuerungsfunktionen enthalten, die mit der Qualitätsfunktion gewichtet werden. Anhand dieser Konzepte kann eine Meta-Heuristik die *Suche* nach guten Lösungen steuern.

Durch Kombination der Prinzipien verschiedener Meta-Heuristiken können weitere Verfahren (hybride Meta-Heuristiken) entwickelt werden. Weiterführende Übersichten und Erläuterungen sind in [Michalewicz und Fogel, 2000] und [Blum und Roli, 2003] zu finden. Die im Rahmen dieser Arbeit verwendeten Meta-Heuristiken sollen im Folgenden grundlegend beschrieben werden.

3.4.1 Lokale Suchverfahren

Die im Folgenden beschriebenen Meta-Heuristiken gehören der Klasse der lokalen Suchverfahren an, die den Ansatz verfolgen, dass ein Lösungsraum mit einer endlichen Anzahl von Lösungen durch eine Sequenz von Lösungen intelligent durchsucht werden kann. Diese Sequenz basiert auf der Definition der *Nachbarschaft* $N(x)$ einer Lösung x , die eine Menge von Lösungen mit ähnlichen Eigenschaften enthält. Durch die Suche in der Nachbarschaft einer aktuellen Lösung nach einer besseren Lösung können iterativ bessere Lösungen gefunden werden. Dieser Vorgang wird auch *Intensivierung* der Suche genannt und findet im *Hill-climber Algorithmus* Anwendung (vgl. Algorithmus 3.2).

Algorithmus 3.2 : Hill-climber Algorithmus

(Schritt 1) **Initialisierung**

$$t = 0$$

Wähle zufällige Lösung x_t

(Schritt 2) **Nachbarschaftslösung bestimmen**

$$t = t + 1$$

$$x_t = \min_{x \in N(x_{t-1})} f(x)$$

(Schritt 3) **Lösung aktualisieren**

Wenn $f(x_t) < f(x_{t-1})$, dann gehe zu Schritt 2

Der Algorithmus bestimmt jeweils die Nachbarschaft der aktuellen Lösung (Schritt 2) und startet die nächste Iteration auf der Basis der durch Auswertung der Bewertungsfunktion $f(x)$ besten gefundenen Lösung. Sobald keine bessere Lösung gefunden werden kann, endet der Algorithmus. Die so bestimmte Lösung wird *lokales Optimum* genannt, da sie innerhalb ihrer Nachbarschaft die beste Lösung darstellt.

Um der Stagnation in lokalen Optima zu entgehen, bedarf es einer Strategie, mit der durch die Intensivierung nicht betrachtete Regionen des Lösungsraums in die Suche einbezogen werden können. Diese Erweiterung wird *Diversifizierung*

genannt. Ein Beispiel für eine Diversifizierungsstrategie ist der Neustart des Hill-climber Algorithmus in einem zufälligen Punkt des Lösungsraums, sobald ein lokales Optimum erreicht ist.

Die richtige Balance von Intensivierung und Diversifizierung stellt einen kritischen Erfolgsfaktor für lokale Suchverfahren dar. Die im Folgenden vorgestellten Meta-Heuristiken bieten unterschiedliche Strategien für eine geeignete Intensivierung und Diversifizierung der lokalen Suche an (vgl. [Michalewicz und Fogel, 2000]).

Variable Nachbarschaftssuche Die *Variable Nachbarschaftssuche* (engl: *variable neighborhood search* (VNS)) stellt eine einfache und effektive Form der lokalen Suche dar (vgl. [Hansen und Mladenović, 2001]). Sie bedient sich dem Vorgehen der Hill-climber Methode, verwendet allerdings eine Menge von unterschiedlichen Nachbarschaftsdefinitionen N_k . Die Methode betrachtet eine andere Nachbarschaft sobald ein lokales Optimum nachgewiesen werden kann (vgl. [Mladenović und Hansen, 1997]). Die Variable Nachbarschaftssuche ist in Algorithmus 3.3 dargestellt.

Algorithmus 3.3 : Variable Nachbarschaftssuche

- (Schritt 1) **Initialisierung**
 $t = 0$
 $k = 1$
 Wähle zufällige Lösung x_t
- (Schritt 2) **Nachbarschaftslösung bestimmen**
 $t = t + 1$

$$x_t = \min_{x \in N_k(x_{t-1})} f(x)$$
- (Schritt 3) **Lösung aktualisieren**
 Wenn $f(x_t) < f(x_{t-1})$, dann gehe zu Schritt 2
 Wenn $k \neq |N|$, dann $k = k + 1$ und gehe zu Schritt 2
-

Nach dem zum Bergsteiger Algorithmus analogen Vorgehen in den Schritten 1 und 2 wird die nächste definierte Nachbarschaft verwendet, sobald der Algorithmus in einem lokalen Optimum angelangt ist (Schritt 3). Alternativ zu dem dargestellten Vorgehen kann die Nachbarschaftsauswahl auch so häufig fortlaufen, bis eine Lösung erreicht ist, für die nach Durchsuchung aller Nachbarschaften keine bessere Lösung gefunden werden kann.

Tabu Suche Die *Tabu Suche* wurde von Fred Glover (vgl. [Glover, 1986]) entwickelt. Als Strategie zur Diversifizierung wird die Auswahl der Lösungssequenz abhängig von der Nachbarschaft und einer Historie des Suchvorgangs durchgeführt. Züge bzw. Attribute, die in einem Schritt getätigt bzw. geändert werden, werden für eine bestimmte Dauer *tabu* gesetzt und dürfen bei der Suche nach einer neuen Lösung nicht beachtet werden. Neben diesem Prinzip der *Neuheit* (engl: *recency*) kann als Kriterium auch die *Häufigkeit* (engl: *frequency*) eines getätigten Zuges beachtet werden. Dadurch können bisher nicht getätigte Züge häufiger getätigten Zügen vorgezogen werden, so dass unbekannte Bereiche des Lösungsraums durchsucht werden. Neben dem strikten Verbot von *tabu*-gesetzten Lösungen, können Aspirationskriterien definiert werden, die das Verbot der Auswahl einer *tabu*-gesetzten Lösung übergehen können, sofern ein deutlicher Vorteil dieser Auswahl ersichtlich ist. Der Ablauf der Tabu Suche ist in Algorithmus 3.4 dargestellt.

Algorithmus 3.4 : Tabu Suche

- (Schritt 1) **Initialisierung**
 $t = 0$
 Initialisiere Historie H
 Wähle zufällige Lösung x_t
 $x_{best} = x_t$
- (Schritt 2) **Nachbarschaftslösung bestimmen**
 $t = t + 1$
 $x_t = \min_{x \in N(H, x_{t-1})} f(x)$
 Wenn Kriterium erreicht, dann Abbruch
- (Schritt 3) **Lösung aktualisieren**
 Aktualisiere H
 Wenn $f(x_t) < f(x_{best})$, dann $x_{best} = x_t$
 gehe zu Schritt 2
-

Nach der Initialisierung der Startlösung und der Historie (Schritt 1) wird die beste Lösung der Nachbarschaft $N(H, x)$ einer Lösung x unter Berücksichtigung der *Tabu*-Informationen aus H ausgewählt. Diese Lösung wird unabhängig von der Lösungsqualität übernommen, so dass auch schlechtere Lösungen als die aktuelle übernommen werden. Deshalb wird in x_{best} die global beste gefundene Lösung gespeichert (Schritt 3). Eine Übersicht über weitere Literatur sowie Anpassungen der Tabu Suche ist in [Glover und Laguna, 1993] zu finden.

3.4.2 Evolutionäre Algorithmen

In den *evolutionären Algorithmen* wird der Lösungsraum nicht auf der Basis von Nachbarschaftslösungen durchsucht. Stattdessen wird die Suche durch die Identifizierung und Nutzung guter Eigenschaften von Lösungen gestaltet. Die Darstellung findet in Form einer Genstruktur statt. Jede Lösung ist durch ein *Chromosom* fester Länge definiert. Es enthält *Gene*, die jeweils eine Eigenschaft der Lösung festlegen. Die Ausprägung eines Gens wird als *Allel* bezeichnet, die Position innerhalb des Chromosoms als *Locus*. Da in einem Chromosom die Eigenschaften einer Lösung codiert sind, werden die Codierung (*Genotyp*) und ihre Interpretation (*Phenotyp*) unterschieden. Zum Beispiel kann ein Element des Wertebereichs einer Funktion die Zahl „5“ sein (Phenotyp), die innerhalb eines genetischen Algorithmus in ihrer binären Darstellung als „101“ codiert ist (Genotyp).

Die *evolutionären Algorithmen* adaptieren die Prinzipien der natürlichen Evolution. Sie betrachten daher nicht einzelne Lösungen, sondern eine Menge (*Population*) von Lösungen. Die Bewertung eines Individuums (= Lösung) wird daher auch als *Fitness* bezeichnet. Eine zu einem bestimmten Zeitpunkt bestehende Population wird *Generation* genannt. Analog zu der Evolutionstheorie von [Darwin, 1859] entwickelt sie sich in drei Schritten zu einer neuen Generation weiter. Algorithmus 3.5 zeigt den Ablauf eines evolutionären Algorithmus. Die Reihenfolge der Schritte 2 bis 4 ist variabel und kann daher in anderen Darstellungen abweichen.

Algorithmus 3.5 : Evolutionärer Algorithmus

(Schritt 1)	Initialisierung $t = 0$ Initialisiere Population P_t
(Schritt 2)	Bewertung Bewerte P_t
(Schritt 3)	Selektion $t = t + 1$ selektiere $P_t \subseteq P_{t-1}$
(Schritt 4)	Variation Variiere P_t Wenn Kriterium erreicht, dann Abbruch, sonst gehe zu Schritt 2

Eine initiale Population wird in Schritt 1 aufgebaut. Diese Population sollte aus Individuen mit stark unterschiedlichen Eigenschaften gebildet werden, um eine ausreichende Diversifizierung zu ermöglichen. Nach der *Bewertung* der Fitness der Individuen (Schritt 2) werden in der *Selektion* (Schritt 3) nach dem Prinzip des „Survival of the Fittest“ Individuen mit guten Eigenschaften ausgewählt. Durch Kreuzung der ausgewählten Individuen werden in der *Variation* neue Individuen auf der Basis der identifizierten guten Eigenschaften generiert (Schritt 4). Durch die zufällige Einstreuung von Unregelmäßigkeiten in der Kreuzung (*Mutation*) werden die Eigenschaften und somit die Suche zusätzlich diversifiziert.

Zum theoretischen Hintergrund und Beweis der Eigenschaftenerhaltung über die Generationen sei auf [Holland, 1975] verwiesen. Eine Übersicht über Erweiterungen und Anwendungsbeispiele der evolutionären Algorithmen ist in [Reeves, 1993] zu finden.

4 Umlaufplanungsprobleme: Stand der Forschung

Die Planung von Busumläufen stellt durch ihre hohe Praxisrelevanz bereits seit ca. 40 Jahren ein vielfach betrachtetes Thema in der internationalen Forschung dar. In diesem Kapitel soll der aktuelle Stand der Forschung beschrieben und ein Überblick über die existierenden Modelle und Verfahren für die unterschiedlichen Problemstellungen gegeben werden.

Das Kapitel ist nach den unterschiedlichen Problemstellungen unterteilt, die in Kapitel 2 beschrieben wurden. Die vorliegende Arbeit thematisiert die allgemeinste Variante der Problemstellungen unter Kombination aller Erweiterungen, das *multiple depot multiple vehicle type group vehicle scheduling problem with time windows and route constraints* (MD-MVTG-VSP-TW-RC). Die im Folgenden vorgestellten Ansätze stellen daher jeweils einen Sonderfall dieser Problemstellung dar. Aufgrund der chronologischen Entwicklung der Forschung und der untereinander gegebenen Abhängigkeit vieler Ansätze wird der Stand der Forschung beginnend mit dem theoretisch einfachsten und am längsten erforschten Problem begonnen.

In Abschnitt 4.1 werden die Ansätze zur Lösung des Umlaufplanungsproblem mit einem Depot (SD-VSP) vorgestellt. Nachdem in 4.2 die Ansätze für die Berücksichtigung von mehreren Depots (MD-VSP) vorgestellt werden, fokussiert Abschnitt 4.3 die Berücksichtigung von Fahrzeugtypgruppen (MVTG-VSP). Die bereits genannten Problemstellungen können jeweils um die Berücksichtigung von Zeitfenstern erweitert werden (VSP-TW), was in Abschnitt 4.4 vorgestellt wird. In 4.5 werden die bestehenden Ansätze für die Berücksichtigung von allgemeinen oder speziellen Umlaufrestriktionen (VSP-RC) betrachtet. Abschnitt 4.6 präsentiert eine Übersicht über Planungssysteme, in die die vorgestellten Forschungsergebnisse eingeflossen sind, sowie praktische Erfahrungen mit den Systemen. In 4.7 werden Veröffentlichungen vorgestellt, die das VSP innerhalb einer Übersicht oder Klassifizierung behandeln.

Die Abfolge der vorgestellten Ansätze innerhalb der Abschnitte richtet sich grundsätzlich an der chronologischen Reihenfolge der Veröffentlichungen aus, wobei Inhalte gleicher Strömungen zum besseren Verständnis zusammengefasst wer-

den. Die mathematischen Formulierungen orientieren sich zur besseren Vergleichbarkeit nicht an den Ausführungen der jeweiligen Autoren, sondern werden einheitlich in der in Kapitel 2 vorgestellten Notation präsentiert.

4.1 Busumlaufplanung mit einem Depot

Das SD-VSP ist ein in polynomieller Laufzeit lösbares Problem. Da es – wie später gezeigt wird – allerdings in vielfältiger Weise als Unterproblem zur Lösung erweiterter Problemstellungen verwendet wird, werden bis in die heutige Zeit spezialisierte Ansätze zur Lösung dieser Problemstellung entwickelt. In diesem Abschnitt werden die unterschiedlichen Ansätze zur Lösung des SD-VSP vorgestellt.

Dekompositionsmodell Der erste Ansatz zur (exakten) Lösung des SD-VSP wurde in [Saha, 1970] vorgestellt und kann als *Dekompositionsmodell* (engl: *minimum decomposition model*) bezeichnet werden. In dieser Veröffentlichung wird das VSP allerdings ohne Berücksichtigung möglicher Umsetzfahrten, Depots sowie operativer Kosten, so dass eine vereinfachte Problemstellung vorliegt.

Die Servicefahrten werden als *teilsortierte Menge* (engl: *partial ordered set*) definiert. Das heißt, dass eine Sortierung hinsichtlich der Abfahrtszeit der Servicefahrten vorgenommen wird ($e_i < e_j \rightarrow i < j, \forall i, j \in T$) und die Ankunftszeiten nicht bei der Sortierung berücksichtigt werden ($f_i < f_j \not\rightarrow i < j$). Unter Verwendung des *Dilworth Theorems* für teilsortierte Mengen (vgl. [Dilworth, 1950]) zeigt Saha, dass die größte Menge von paarweise inkompatiblen Fahrten gleich der minimalen Anzahl von Umläufen ist. Das resultierende Maximierungsmodell zur Bestimmung dieser Menge enthält eine Entscheidungsvariable x_{ij} für jede Fahrtenkombination $i, j \in T$, wobei die Kosten definiert sind durch $c_{ij} = 1$, wenn $f_i \leq e_j$, sonst $c_{ij} = -\infty$:

$$\max \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \quad (4.1)$$

$$s.t. \sum_j x_{ij} \leq 1 \quad \forall i \in T \quad (4.2)$$

$$\sum_i x_{ij} \leq 1 \quad \forall j \in T \quad (4.3)$$

$$x_{ij} \geq 0 \quad \forall i, j \in T \quad (4.4)$$

Das Modell wird durch eine Anpassung des Labelling Algorithmus von Dantzig und Hoffmann (vgl. [Dantzig und Hoffman, 1956]) gelöst.

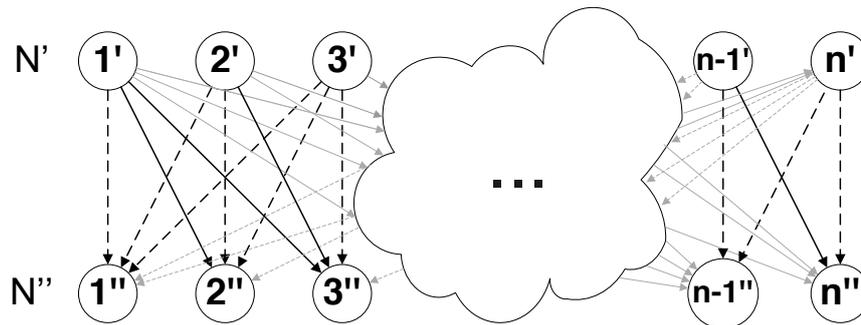


Abbildung 4.1: Zuweisungsmodell für das SD-VSP

Bodin und Rosenfield erweitern den Ansatz, indem sie mögliche Umsetzfahrten betrachten (Deadheading) und zusätzlich mögliche Depotfahrten so berücksichtigen, dass ein Fahrzeug auch während eines Umlaufs ggf. ins Depot zurückkehren kann (vgl. [Bodin und Rosenfield, 1976]). Da das Modell zu komplex für eine exakte Optimierung ist, präsentieren sie eine Heuristik mit zwei Phasen. In der ersten Phase werden nur Kompatibilitäten zwischen zeitlich nah beieinander liegenden Fahrten (z.B. maximal 20 Minuten zwischen Bedienzeiten) berücksichtigt und das Problem gelöst. Die Umläufe der Lösung werden fixiert und als Fahrten für eine Optimierung mit allen möglichen Kompatibilitäten in der zweiten Phase verwendet.

Das Dekompositionsmodell ist der erste Ansatz zur Lösung des SD-VSP, kann allerdings weder operative Kosten noch eine Kapazität für die Fahrzeuganzahl in die Optimierung mit einbeziehen.

Zuweisungsmodell In [Orloff, 1976] wird das SD-VSP als ein *Set Covering Problem* formuliert, in dem für jeden möglichen Umlauf eine Entscheidungsvariable definiert wird und über die Menge aller Umläufe die kostenminimale Menge gesucht wird, die alle Fahrten mindestens einmal abdeckt. Dieser Ansatz war allerdings mit der zu der Zeit vorhandenen Technik nicht lösbar, da die große Zahl von Variablen nicht im Speicher gehalten werden konnte. Deshalb formuliert Orloff das SD-VSP alternativ als ein *Zuweisungsmodell* (engl.: *assignment model*), bei dem jede Servicefahrt $t \in T$ durch einen Ankunfts-knoten $t' \in N'$ und einen Startknoten $t'' \in N''$ repräsentiert wird. Die möglichen Verbindungen zwischen kompatiblen Fahrtenpaaren (i,j) werden jeweils durch eine Kante von i' nach j'' modelliert und haben die operativen Kosten der Umsetzfahrten. Diese Kosten können zudem mögliche Wartekosten für Haltestellen außerhalb des Depots beinhalten. Alle nicht gültigen Verbindungen werden durch eine Kante repräsentiert, deren Kosten die Verwendung eines weiteren Fahrzeugs sowie die

operativen Kosten für Depotfahrten beinhaltet. Somit ist das zugrunde liegende Netzwerkmodell ein vollständiger bipartiter Graph, dessen Struktur in Abbildung 4.1 verdeutlicht wird. Zur Lösung des SD-VSP wird ein Optimierungsproblem auf diesem Graphen definiert, das bei Minimierung der Kantenkosten die Bedienung einer Servicefahrt durch Zuweisungsrestriktionen sicherstellt:

$$\min \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \quad (4.5)$$

$$s.t. \sum_j x_{ij} = 1 \quad \forall i \in N' \quad (4.6)$$

$$\sum_i x_{ij} = 1 \quad \forall j \in N'' \quad (4.7)$$

$$x_{ij} \geq 0 \quad \forall i \in N', \forall j \in N'' \quad (4.8)$$

Im Gegensatz zum Dekompositionsmodell können im Zuweisungsmodell operative Kosten berücksichtigt werden. Allerdings kann eine feste oder maximale Anzahl von vorhandenen Fahrzeugen nicht modelliert werden.

Transportmodell Gavish und Shlifer betrachten eine Vielzahl von Problemen, die alle auf Zuweisungsproblemen basieren, und präsentieren das *Transportmodell* (engl: *transportation model*), in dem die zuvor vorgestellte Struktur des Zuweisungsmodells für das SD-VSP aufgenommen und erweitert wird (vgl. [Gavish und Shlifer, 1979]). In dem bipartiten Graphen werden ausschließlich die Kanten zwischen kompatiblen Fahrten eingefügt, die die operativen Kosten der Verbindungen beinhalten. Zusätzlich werden zwei Knoten zur Modellierung der Depotankünfte ($n + 1'$) und Depotabfahrten ($n + 1''$) modelliert. Vom Knoten für die Depotabfahrt $n + 1'$ modellieren Kanten zu den Abfahrtsknoten der Servicefahrten mögliche Depotausrückfahrten. Entsprechend führen Kanten von den Ankunfts-knoten der Fahrten zum Depotankunftsknoten. Neben den operativen Kosten für die Depotfahrten enthalten diese Kanten auch die fixen Kosten – jeweils den halben Teil – für ein Fahrzeug. Die Autoren formulieren ein Transportproblem mit einem Angebot von einer Einheit auf allen Abfahrtsknoten und einem Bedarf von einer Einheit auf allen Ankunfts-knoten der Servicefahrten. Die Abfahrtsknoten bzw. Ankunfts-knoten des Depots haben ein Angebot bzw. einen Bedarf in der Höhe der Anzahl an verfügbaren Fahrzeugen. Für den Fall, dass keine festgelegte Fahrzeuganzahl, sondern eine obere Schranke oder keine Begrenzung vorliegt, ist eine Kante mit Kosten von 0 zwischen den beiden Depotknoten modelliert. Das Modell kann daher mit einem großen Depotangebot M definiert werden, da nicht

benötigte Flusseinheiten keine zusätzlichen Kosten verursachen und damit die Lösbarkeit der Problemstellung gewährleistet werden kann. Die Modellformulierung wird auch als *Quasi-Zuweisungsproblem* (engl: *Quasi Assignment Problem*) bezeichnet, da das mathematische Modell ein Zuweisungsproblem ist, dass um lediglich zwei Restriktionen (für die Depotknoten) erweitert ist:

$$\min \sum_{i=1}^{n+1} \sum_{j=1}^{n+1} c_{ij} x_{ij} \quad (4.9)$$

$$\text{s.t.} \sum_{j=1}^{n+1} x_{ij} = 1 \quad \forall i \in N \quad (4.10)$$

$$\sum_{i=1}^{n+1} x_{ij} = 1 \quad \forall j \in N \quad (4.11)$$

$$\sum_{j=1}^{n+1} x_{n+1,j} = M \quad (4.12)$$

$$\sum_{i=1}^{n+1} x_{i,n+1} = M \quad (4.13)$$

$$x_{ij} \geq 0 \quad \forall i, j \in N \quad (4.14)$$

Die besondere Struktur des Modells wurde in [Paixao und Branco, 1987] ausgenutzt, um die *ungarische Methode* (engl. *Hungarian Method*) (vgl. [Kuhn, 1955]) zur Lösung des SD-VSP anzupassen.

In [Paixao und Branco, 1990] wird das Modell in der Hinsicht erweitert, dass im Falle einer durch eine zu geringe Fahrzeugkapazität ungünstigen Problemstellung möglichst viele Fahrten bedient werden. Dazu wird für alle Servicefahrten eine Kante vom Abfahrtsknoten zum Ankunfts-knoten eingefügt, die dominierend hohe Kosten hat und somit nur im Falle eines unlösbaren Modells gewählt wird. Die in einer Lösung gewählten Kanten repräsentieren somit Fahrten, die mit dem berechneten Umlaufplan nicht bedient werden können. Der um diese Kanten erweiterte Graph des Transportmodells ist in Abbildung 4.2 dargestellt.

In [Song und Zhou, 1990] wird ein erweiterter *Iterative Kürzeste-Wege-Algorithmus* (engl: *Successive-Shortest-Path-Algorithm*) vorgestellt, der allgemein zur Lösung des Quasi-Zuweisungsproblems und speziell zur Lösung des SD-VSP verwendet werden kann. Es werden keine Ergebnisse präsentiert, aber es wird angemerkt, dass der Grundalgorithmus im Gegensatz zu der in [Paixao und Branco, 1987] erweiterten ungarischen Methode besser für die Lösung des Zuweisungsmodells geeignet ist und die Autoren nach Anpassung des Verfahrens sehr positive

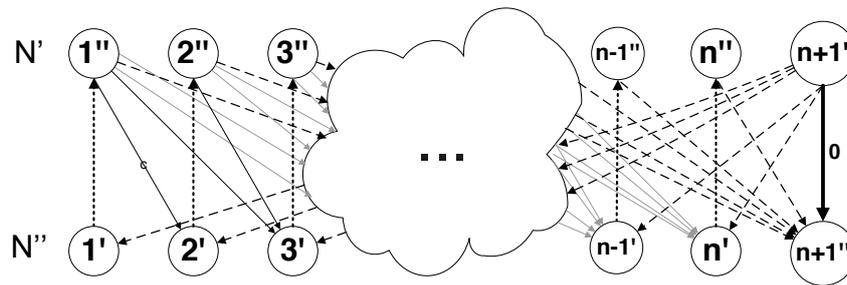


Abbildung 4.2: Transportmodell für das SD-VSP

Resultate bekommen haben („... and (we) obtain very positive results“).

Silva et al. berücksichtigen innerhalb des Transportmodells weitere praxisrelevante Problemstellungen (z.B. eine Präferenz von Verbindungen) und stellen ein iteratives Verfahren vor, in dem ein initiales Modell mit ausschließlich „kurzen“¹ Kanten gelöst wird und anschließend alle nicht berücksichtigten Kanten mit negativen reduzierten Kosten hinzugefügt werden (vgl. [Silva et al., 1999]). Die Transportprobleme werden mit dem *Out-of-Kilter-Verfahren* gelöst (vgl. [Fulkerson, 1961]) bis in einer Iteration keine Kante mit negativen reduzierten Kosten mehr gefunden wird.

Freling et al. präsentieren weitere Verfahren zur Lösung des Zuweisungsmodells (vgl. [Freling et al., 2001]). Sie präsentieren unter anderem Auktionsalgorithmen und weitere Verfahren, die zur Optimierung – ähnlich wie in [Silva et al., 1999] – auf die iterative Lösung von kleineren Modellen zurückgreifen.

Netzwerkflussmodell In [Bodin et al., 1983] wird ein *Netzwerkflussmodell* (engl: *network flow problem*) für das SD-VSP präsentiert. Der Ansatz ist von einer frühen Modellierung des Tankerplanungsproblems als Flussproblem motiviert (vgl. [Dantzig und Fulkerson, 1954]). Jede Servicefahrt ist durch zwei Knoten – für Abfahrt und Ankunft – repräsentiert, die durch eine Fahrtkante, die die operativen Kosten der Servicefahrt enthält, verbunden sind. Für alle kompatiblen Fahrtenpaare existieren gerichtete Kanten zwischen dem Abfahrtsknoten der früheren und dem Ankunftsknoten der späteren Fahrt. Zwei Depotknoten sind durch Kanten mit operativen Kosten für Aus- und Einrückfahrten mit den entsprechenden Knoten der Servicefahrten verbunden. Die Fixkosten für Fahrzeuge werden durch eine gerichtete Kante zwischen dem Ankunftsknoten und Abfahrtsknoten des Depots modelliert (s. Abbildung 4.3).

¹Als „kurze“ Kanten werden allgemein Verbindungskanten bezeichnet, die zwei Servicefahrten so verbinden, dass an einer Haltestelle keine lange unproduktive Wartezeit entsteht. Die

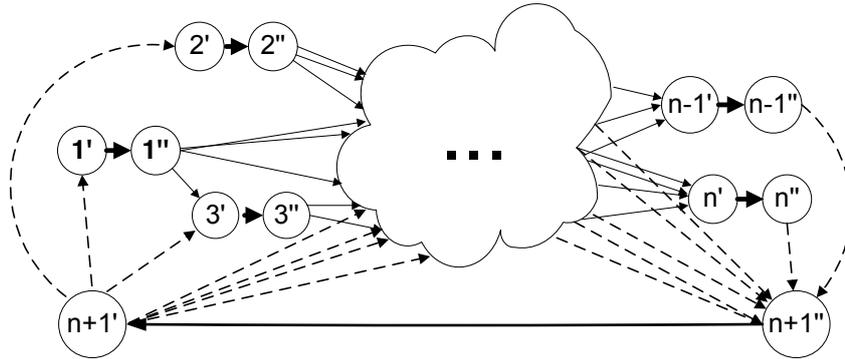


Abbildung 4.3: Netzwerkflussmodell für das SD-VSP

Ein Weg vom Abfahrtsknoten $(n + 1')$ zum Ankunfts-knoten $(n + 1'')$ des Depots entspricht einem gültigen Umlauf. Das SD-VSP wird durch Formulierung eines *Umladeproblems* (s. Abschnitt 3.1) gelöst, das ausschließlich Umladeknoten enthält. Damit sichergestellt ist, dass alle Servicefahrten bedient werden, sind die unteren und oberen Schranken der Servicefahrt-Kanten gleich 1. Die obere Schranke für die zurückführende Kante zwischen den beiden Depotknoten ist gleich der maximal verfügbaren Anzahl von Fahrzeugen. Das resultierende Modell lautet:

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (4.15)$$

$$s.t. \sum_{i:(i,j) \in A} x_{ij} - \sum_{i:(j,i) \in A} x_{ji} = 0 \quad \forall j \in N \quad (4.16)$$

$$1 \leq x_{ij} \leq 1 \quad \forall (i, j) \in T \quad (4.17)$$

$$x_{ij} \geq 0 \quad \forall (i, j) \in A \quad (4.18)$$

In [Löbel, 1996] wird ein angepasster *Netzwerk-Simplex Algorithmus* (engl: *network simplex algorithm*) (vgl. [Dantzig, 1951]) für das Netzwerkflussmodell beschrieben, der auf die Lösung von sehr großen Instanzen des SD-VSP ausgelegt ist.

Modellunabhängige Heuristiken Die folgenden Ansätze basieren nicht auf einer der zuvor vorgestellten Modellierungen des SD-VSPs, sondern stellen alternative heuristische Lösungsmethoden dar.

genaue Definition ist je nach Autor unterschiedlich.

Wren beschreibt eine heuristische Methode, bei der in einer ersten Phase so viele Fahrzeuge eingesetzt werden, wie die maximale Fahrtenauslastung des Fahrplans aufzeigt (vgl. [Wren, 1972]). Die Fahrten werden sukzessiv einem Fahrzeug zugeteilt, solange der Umlauf des Fahrzeugs nicht ungültig wird. Diese Lösung beinhaltet in der Regel viele nicht bediente Fahrten, die vorläufig dem ersten Fahrzeug zugeteilt werden. In der zweiten Phase wird ein Verbesserungsverfahren angewendet, bei dem ungültige Verbindungen innerhalb eines Umlaufs in der Kostenfunktion bestraft werden. Auf der Basis dieser Kostenfunktion wird ein 2-opt Verfahren angewendet, bei dem jeweils zwei Verbindungen darauf geprüft werden, ob durch einen Tausch der durch sie verbundenen Umlaufteile eine bessere Lösung entstehen kann. Dieses Tauschverfahren wird solange durchgeführt bis keine zwei Verbindungen mehr existieren, die durch einen Tausch eine Verbesserung der Lösung bringen würden. Diese Lösung wird 2-optimal genannt. Wenn die Lösung noch ungültige Umläufe enthält, wird in einer dritten Phase so lange ein Fahrzeug hinzugefügt und die zweite Phase wiederholt, bis die Lösung gültig ist.

In [Rattadilok und Wren, 2006] wird dieser Ansatz dadurch erweitert, dass die Prüfungen in Phase 2 nicht statisch durch den 2-opt durchgeführt werden, sondern auch mehr Verbindungen in die Prüfung mit einbezogen werden können und die Anzahl durch einen dynamischen Parameter λ gesteuert wird. Diese λ -opt-Heuristik ist von der Suchmethodik her vergleichbar mit dem Algorithmus aus [Lin und Kernighan, 1973] für das TSP.

In [Stern und Ceder, 1981] und [Ceder und Stern, 1981] wird die allgemein fehlende Akzeptanz von Optimierungsmethoden in der Umlaufplanung beschrieben, da nicht alle praxisrelevanten Nebenbedingungen betrachtet werden und große Probleme nicht gelöst werden können. Konkret berichten sie von diesem Verhalten bei dem Einsatz der in [Gavish et al., 1978] vorgeschlagenen Lösungsmethode. Die Autoren schlagen daher eine graphisch-analytische Methode als Entscheidungsunterstützung vor. Sie soll die Grundlage einer Interaktion zwischen System und Planer sein und eine schnelle Darstellung der Auswirkungen von Änderungen ermöglichen. Der Methodik liegt der *Deficit Function Approach* aus [Gertsbach und Gurevich, 1977] zugrunde. Der Ansatz ermöglicht die Berechnung und Analyse von unteren Schranken für die benötigte Fahrzeuganzahl über den Tagesverlauf und verwendet dazu das *Fleet Size (Lower Bound) Theorem* aus [Bartlett, 1957]. Dieses Theorem gibt eine untere Schranke für die benötigte Fahrzeugkapazität an einer Endhaltestelle zu einem Zeitpunkt an (vgl. [Salzborn, 1972]). Das Verfahren berücksichtigt die Betrachtung von Verbindungsmöglichkeiten heuristisch oder durch den Benutzer. In [Stern und Ceder, 1983] wird die Berechnung der unteren Schranken unter Einbeziehung der Verbindungsmöglichkeiten verbessert.

Veröffentlichung	Fahrtenanzahl	Laufzeit (Min.)
[Saha, 1970]	319	30
[Gavish und Shlifer, 1979]	1407	114
[Paixao und Branco, 1987]	1300	4
[Paixao und Branco, 1990]	412	2
[Silva et al., 1999]	2732	- ¹
[Freling et al., 2001]	1500	1
[Löbel, 1996]	4225	5

¹ keine Angaben

Tabelle 4.1: Eigenschaften gelöster SD-VSP Probleminstanzen

Ceder erweitert den Ansatz um eine Entscheidungsunterstützung zur Kosteneinsparung durch die Verschiebung von Fahrten (vgl. [Ceder, 2004]).

Tabelle 4.1 zeigt eine Übersicht über die Problemgröße von SD-VSP Instanzen, die – sofern angegeben – von den unterschiedlichen Ansätzen gelöst werden konnten.

4.2 Busumlaufplanung mit mehreren Depots

Das Umlaufplanungsproblem mit mehreren Depots (MD-VSP) gehört zu den schwer zu lösenden Problemstellungen (s. Abschnitt 2.3.3). Daher gibt es eine Vielzahl von heuristischen Ansätzen zur Lösung dieses Problems. Allerdings existieren auch eine Reihe von exakten Modellen, die zur Lösung verwendet werden. Da viele – sowohl exakte als auch heuristische – Verfahren auf den Ideen unterschiedlicher Modellierungsansätze basieren, werden die drei grundsätzlich unterschiedlichen Modellarten und die existierenden Lösungsmethoden in den Abschnitten 4.2.1 bis 4.2.3 vorgestellt und in Abschnitt 4.2.4 die modellunabhängigen Heuristiken diskutiert. Da das MD-VSP einen Spezialfall des MVTG-VSP darstellt, werden die Ansätze aus der Literatur, die beide Problemstellungen betrachten nur kurz erwähnt und ausführlicher in Abschnitt 4.3 erläutert.

4.2.1 Einfache Flussmodelle

Die einfachen Flussmodelle modellieren das MD-VSP auf der Basis des Netzwerkflussmodells für das SD-VSP – das heisst sämtliche Fahrten werden als Fluss eines Gutes in einem Netzwerk betrachtet. Für die Berücksichtigung von mehreren Depots im gleichen Netzwerk, müssen die zusätzlichen Nebenbedingungen des

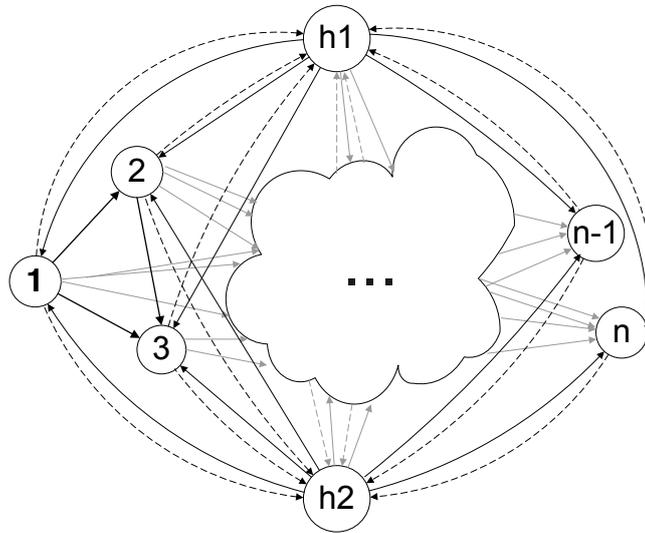


Abbildung 4.4: Einfaches Flussmodell für das MD-VSP

MD-VSP daher explizit in der mathematischen Formulierung beachtet werden.

Jede Fahrt sowie jedes Depot wird als Knoten dargestellt. Kompatible Fahrten werden durch Verbindungskanten miteinander verbunden. Zusätzlich modellieren Kanten zwischen jedem Depot und jeder Fahrt mögliche Aus- und Einrückfahrten. Der resultierende Graph $G = (A, N)$ ist daher definiert durch die Knotenmenge $N = T \cup D$ und die Kantenmenge $A = \Psi \cup (D \times T) \cup (T \times D)$ mit den Kanten Ψ zwischen kompatiblen Fahrten ($\Psi = \bigcup_{i \neq j} (i, j), \forall i, j \in T$). Abbildung 4.4 veranschaulicht den Graphen mit zwei Depots.

Auf der Basis des Graphen G existieren zwei unterschiedliche Formulierungen des MD-VSP, die sich in der Berücksichtigung der Nebenbedingungen für die Rückkehr der Fahrzeuge in das gleiche Depot unterscheiden und im Folgenden vorgestellt werden.

Flussmodell mit Teiltour Eliminierung Carpaneto et al. präsentieren mit der folgenden Formulierung das erste exakte Verfahren zur Lösung des MD-VSP (vgl. [Carpaneto et al., 1989]). In der Zielfunktion werden die Gesamtkosten über alle Kanten minimiert und durch Nebenbedingungen wird die Flusserhaltung im Netzwerk sowie die einmalige Bedienung jeder Servicefahrt sichergestellt. Um die Rückkehr eines Fahrzeugs in sein Startdepot zu modellieren werden Restriktionen zur Eliminierung von Teiltouren eingeführt. Diese verbieten sämtliche Pfade im Netzwerk, die mehr als ein Depot beinhalten bzw. nicht an einem Depot starten und enden (\mathcal{FP}). Da die Anzahl der möglichen ungültigen Pfade sehr groß ist,

kann das Modell nicht direkt gelöst werden. Stattdessen wird in [Carpaneto et al., 1989] ein Branch & Bound Verfahren vorgeschlagen, bei dem Unterprobleme ohne die Restriktionen zum Verbot der Subtouren gelöst werden. Im Falle einer nicht gültigen Lösung des Unterproblems werden im weiteren Verlauf die Restriktionen eingefügt, die die ungültige Lösung verbieten. Neben dem Verfahren stellen die Autoren einen Instanzengenerator vor, der auf Basis möglichst realistischer Annahmen reale VSP-Instanzen nachbilden soll. Der Generator wird in der Literatur zur Erstellung von Instanzen häufig verwendet und soll daher an dieser Stelle erwähnt werden (s. Kapitel 6 für eine Einschätzung der auf diese Weise künstlich generierten Instanzen).

Fischetti et al. erweitern das Verfahren aus [Carpaneto et al., 1989], indem sie das Branch & Bound-Verfahren zur Laufzeit durch stärkere Restriktionen zur Subtour Eliminierung ergänzen (vgl. [Fischetti et al., 1999]). Für dieses Branch & Cut Verfahren präsentieren die Autoren einen Separationsalgorithmus zur Ableitung der verstärkenden Restriktionen sowie weitere Beschleunigungstechniken. In [Fischetti et al., 2001b] werden weitere valide Ungleichungen für den Einsatz innerhalb des Branch & Cut Verfahrens vorgestellt. In Verbindung mit der später vorgestellten Heuristik aus [Dell'Amico et al., 1993] ermöglichen diese die Lösung größerer Problemstellungen.

Die drei Ansätze verwenden leicht unterschiedliche Modelle, verfolgen allerdings alle das Prinzip der Subtour Eliminierung. Zum Beispiel werden in [Carpaneto et al., 1989] die Depotkapazitäten durch die Modellierung von Knoten und zugehörigen Kanten für jedes mögliche Fahrzeug berücksichtigt. In [Fischetti et al., 1999] und [Fischetti et al., 2001b] wird die Fahrzeuganzahl pro Depot durch weitere Restriktionen begrenzt. Zum besseren Verständnis soll an dieser Stelle das Modell aus [Fischetti et al., 2001b] gezeigt werden:

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (4.19)$$

$$s.t. \sum_{i:(i,j) \in A} x_{ij} = \sum_{i:(j,i) \in A} x_{ji} \quad \forall j \in N \quad (4.20)$$

$$\sum_{j:(i,j) \in A} x_{ij} = 1 \quad \forall i \in T \quad (4.21)$$

$$\sum_{(i,j) \in P} x_{ij} \leq |P| - 1 \quad \forall P \in \mathcal{FP} \quad (4.22)$$

$$\sum_{j:(i,j) \in A} x_{ij} \leq w_d \quad \forall d \in D \quad (4.23)$$

$$x_{ij} \in \mathbb{N} \quad \forall (i, j) \in A \quad (4.24)$$

In der Zielfunktion (4.19) werden die Kosten über alle Kanten minimiert. Restriktionen (4.20) stellen den Erhalt des Flusses im Netzwerk sicher und (4.21) sichern die einmalige Bedienung jeder Servicefahrt. In (4.22) werden alle ungültigen Pfade verboten, indem der Fluss über alle Kanten dieses Pfads beschränkt wird. In (4.23) werden die Depotkapazitäten berücksichtigt und (4.24) definiert die ganzzahligen Variablen auf den Kanten.

Flussmodell mit Zuweisungsvariablen Mesquita und Paixao präsentieren einen alternativen Ansatz auf der Basis des einfachen Netzwerkflussgraphen (vgl. [Mesquita und Paixao, 1992]). Die Gültigkeit der Umläufe wird im Gegensatz zu den Ansätzen mit Subtour Eliminierung durch zusätzliche Variablen y_{id} erreicht, die angeben, ob eine Servicefahrt $i \in T$ von einem Fahrzeug aus Depot $d \in D$ bedient wird:

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (4.25)$$

$$s.t. \sum_{j:(i,j) \in A} x_{ij} = 1 \quad \forall i \in T \quad (4.26)$$

$$\sum_{i:(i,j) \in A} x_{ij} = 1 \quad \forall j \in T \quad (4.27)$$

$$x_{dj} - y_{jd} \leq 0 \quad \forall j \in T, \forall d \in D \quad (4.28)$$

$$x_{id} - y_{id} \leq 0 \quad \forall i \in T, \forall d \in D \quad (4.29)$$

$$y_{id} + x_{ij} - y_{jd} \leq 1 \quad \forall i, j \in T, \forall d \in D \quad (4.30)$$

$$\sum_{d \in D} y_{id} = 1 \quad \forall i \in T \quad (4.31)$$

$$\sum_{j:(d,j) \in A} x_{dj} \leq w_d \quad \forall d \in D \quad (4.32)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (4.33)$$

$$y_{id} \in \{0, 1\} \quad \forall i \in T, \forall d \in D \quad (4.34)$$

Wie in den anderen Modellen werden auch hier (ausschließlich) die Kosten der Flussvariablen minimiert (4.25). Restriktionen (4.26) und (4.27) stellen die Bedienung jeder Fahrt dadurch sicher, dass bei jedem Fahrtnoten genau eine Flusseinheit ankommt bzw. abgeht. In den Restriktionen (4.28) bis (4.30) werden die Variablengruppen miteinander verbunden. Dadurch kann in (4.31) sichergestellt werden, dass jede Fahrt genau von einem Depot bedient wird. Restriktionen (4.32) modellieren die Depotkapazitäten und (4.33) bzw. (4.34) definieren die Fahrt- bzw. Depot-Variablen.

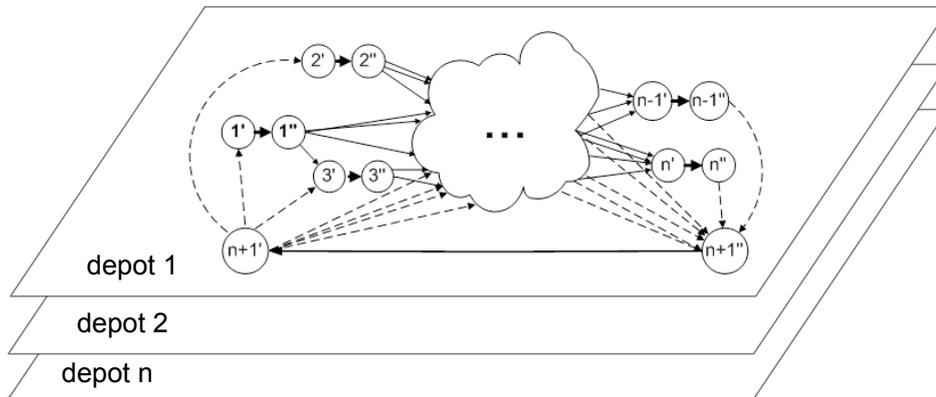


Abbildung 4.5: Mehrgüter Flussmodell für das MD-VSP

In [Mesquita und Paixao, 1992] wird das Modell durch Umformulierung und Relaxation der Bedingungen (4.28) bis (4.30) gelöst. Dadurch entstehen zwei unabhängige Unterprobleme für die jeweilige Gruppe von Variablen, die leichter zu lösen sind. Die Autoren verwenden auf der Basis dieser Relaxation eine Subgradientenmethode zur Berechnung von unteren Schranken und präsentieren ein Verfahren zur Ableitung einer gültigen Lösung, das Fahrketten von ungültigen Umläufen vertauscht und solange rekombiniert bis die Lösung gültig ist.

4.2.2 Mehrgüter Modelle

Die im Folgenden vorgestellten *Mehrgüter Modelle* (engl: *multicommodity models*) zur Lösung des MD-VSP unterscheiden bereits in der Netzwerkmodellierung explizit die unterschiedlichen Depots. Es können innerhalb dieser Modellierungsart das *Mehrgüter Flussmodell* und das *Mehrgüter Matching Modell* unterschieden werden.

Mehrgüter Flussmodell Das *Mehrgüter Flussmodell* (engl: *multicommodity flow problem*) erweitert das Netzwerkflussmodell (s. Abschnitt 4.1), indem für jedes Depot ein Netzwerk – im Folgenden Schicht oder Netzwerkschicht genannt – verwendet wird. In dem entstehenden Multigraphen ist der Fluss innerhalb einer Netzwerkschicht unabhängig von Flüssen in anderen Schichten. Als einzige Verbindung müssen die Kanten für die Servicefahrten so verknüpft werden, dass jede Fahrt über alle Depots genau einmal bedient wird. Abbildung 4.5 illustriert den Schichtaufbau des Mehrgüter Flussmodells. Sei $N^d \subset N$ die Menge der Knoten aus Netzwerkschicht $d \in D$. Bodin et al. formulieren ein mathematisches Modell

für das oben beschriebene Netzwerk ohne ein theoretisch exaktes Verfahren zu nennen (vgl. [Bodin et al., 1983]):

$$\min \sum_{(i,j)^d \in A} c_{ij}^d x_{ij}^d \quad (4.35)$$

$$s.t. \sum_{i:(i,j)^d \in A} x_{ij}^d - \sum_{i:(j,i)^d \in A} x_{ji}^d = 0 \quad \forall j \in N^d, \forall d \in D \quad (4.36)$$

$$\sum_{d \in D} x_t^d = 1 \quad \forall t \in T \quad (4.37)$$

$$x_d \leq w_d \quad \forall d \in D \quad (4.38)$$

$$x_{ij}^d \in \mathbb{N} \quad \forall (i,j)^d \in A \quad (4.39)$$

Die Entscheidungsvariablen x_{ij}^d sind fortan definiert als Flussgröße zwischen i und j in der Netzwerkschicht von Depot d . Unter Minimierung der Gesamtkosten (4.35) werden für alle Knoten des Netzwerks lokale Bedingungen zur Flussserhaltung definiert (4.36). In Restriktion (4.37) werden die in unterschiedlichen Schichten vorhandenen Kanten einer Servicefahrt über so genannte *Überdeckungsrestriktionen* (engl: *cover constraints*) verbunden und die einmalige Bedienung jeder Servicefahrt sichergestellt. Die Depotkapazitäten werden durch eine obere Schranke auf der jeweiligen Rückflusskante eines Depots formuliert (4.38). Die Flussvariablen werden in (4.39) als ganzzahlige Variablen definiert. An dieser Stelle soll angemerkt werden, dass je nach konkreter Modellierungsart eine Vielzahl der Variablen als (0,1)- oder kontinuierliche Variablen definiert werden kann.

Lamatsch formuliert auf der Basis des SD-VSPs Modells aus [Desrosiers et al., 1982] ein Mehrgüter Flussproblem und präsentiert die erste theoretisch exakte Lösungsmethode (vgl. [Lamatsch, 1992]). Durch Umformulierung und Lagrange-Relaxation von Restriktionen entsteht ein einfaches Unterproblem, so dass nicht jedes Fahrzeug in seinem Startdepot ankommen muss. Durch Anpassung der dualen Werte für die relaxierten Restriktionen über ein Subgradientverfahren kann die Lösung dieser Relaxation verbessert werden. Der Autor präsentiert darüber hinaus einen Algorithmus zur Herstellung der primalen Gültigkeit, um ähnlich wie in [Mesquita und Paixao, 1992] heuristische Lösungen abzuleiten.

Forbes et al. verwenden eine ähnliche Lagrange-Relaxation, um das MD-VSP in einem 3 Phasen-Algorithmus zu lösen (vgl. [Forbes et al., 1994]). Dieser sowie die Ansätze aus [Löbel, 1997] und [Kliwer, 2005] sollen allerdings aufgrund der Betrachtung der erweiterten Problemstellung MVTG-VSP erst in Abschnitt 4.3 näher erläutert werden.

In [Banihashemi und Haghani, 2000] werden die Servicefahrten so in drei Mengen (Morgens-, Mittags- und Abendfahrten) geteilt, dass es für jede Kombination

aus Morgen- und Abendfahrt günstiger ist, über das Depot zu fahren und dort zu pausieren, anstatt eine direkte Verbindungsfahrt durchzuführen und eine lange Wartezeit in einer Haltestelle zu verursachen. Durch diese Modellierung kann das Netzwerk um bis zu 40% aller Verbindungskanten reduziert werden. Das Optimierungsproblem kann aufgrund der geringeren Anzahl von Variablen durch Standardoptimierungssoftware gelöst werden.

Mehrgüter Matchingmodell Das *Mehrgüter Matchingmodell* (engl: *multicommodity matching model*) ist im Unterschied zum Mehrgüter Flussproblem auf einem bipartitem Multigraphen definiert. Für jedes Depot ist ein bipartiter Graph definiert, bei dem jede Fahrt durch zwei Knoten für Ankunft und Abfahrt modelliert ist. Der Graph ist wie beim Zuweisungsmodell für das SD-VSP vollständig, wobei die Kosten der Kanten abhängig von der Kompatibilität der jeweiligen Fahrten sind (s. Abbildung 4.1). Zusätzlich repräsentieren zwei Knoten sowie Kanten zu und von allen Fahrten das Depot des Teilgraphen. Für nicht kompatible Fahrten enthalten die Kosten auf der Verbindungskante die Fixkosten für die Verwendung eines weiteren Fahrzeugs.

Bertossi et al. stellen das Modell in Verbindung mit einer Formulierung auf, bei der die Zuweisung von einer Fahrt $i \in T$ zu einem Depot $d \in D$ über eine weitere Gruppe von Variablen y_i^d geregelt wird (vgl. [Bertossi et al., 1987]):

$$\min \sum_{(i,j)^d \in A} c_{ij}^d x_{ij}^d \quad (4.40)$$

$$s.t. \sum_{j:(i,j)^d \in A} x_{ij}^d = y_i^d \quad \forall i \in T, \forall d \in D \quad (4.41)$$

$$\sum_{i:(i,j)^d \in A} x_{ij}^d = y_j^d \quad \forall j \in T, \forall d \in D \quad (4.42)$$

$$\sum_{d \in D} y_t^d = 1 \quad \forall t \in T \quad (4.43)$$

$$\sum_{(i,j)^d \in A | i \neq j} x_{ij}^d \leq w_d \quad \forall d \in D \quad (4.44)$$

$$x_{ij}^d \in \{0, 1\} \quad \forall (i,j)^d \in A \quad (4.45)$$

In der Zielfunktion (4.40) werden die Gesamtkosten minimiert. Durch Restriktionen (4.41) bzw. (4.42) werden jeder Fahrt ein Nachfolger bzw. ein Vorgänger sowie die Zugehörigkeit zu einem Depot zugeteilt. In (4.43) wird sichergestellt, dass jede Fahrt genau von einem Depot bedient wird. Die Obergrenze für verwendete Fahrzeuge eines Depots wird in Restriktion (4.44) durch die Beschränkung

der Flusssumme aller Verbindungskanten zwischen nicht kompatiblen Fahrten sichergestellt.

Die Autoren stellen eine Heuristik vor, die auf Basis dieser Formulierung Restriktionen (4.43) relaxiert und das resultierende Lagrange Maximierungsproblem durch ein Subgradientenverfahren löst. Eine gültige Lösung wird durch die Verwendung einer Greedy-Methode auf einer Lösung der Lagrange Relaxation erstellt.

Mesquita und Paixao lösen das Mehrgüter Matchingmodell mit einem Branch & Bound Verfahren und zeigen die unterschiedlichen Qualitäten der Lösungen von LP-Relaxationen verschiedener Modellierungsansätze (vgl. [Mesquita und Paixao, 1999]).

Huisman et al. verwenden eine ähnliche Formulierung zur Lösung des MD-VSP als Unterproblem einer dynamischen Optimierung (vgl. [Huisman et al., 2001] und [Huisman et al., 2004]). Die Autoren verwenden Standardoptimierungssoftware zur Lösung der Modelle.

4.2.3 Set Partitioning Modelle

Die zuvor vorgestellten Ansätze für das MD-VSP haben gemeinsam, dass innerhalb des Modells Entscheidungsvariablen für einzelne Servicefahrten verwendet werden und dadurch die Servicefahrten explizit als Aktivitäten berücksichtigt werden. Ein alternativer Ansatz soll in diesem Abschnitt vorgestellt werden, der vollständige Umläufe als Entscheidungsgrundlage betrachtet. Im Folgenden werden diese Ansätze auch als Fahrt- oder Umlauf-basierte Modelle unterschieden.

Bereits in [Orloff, 1976] ist die Formulierung des MD-VSPs als Set Covering Problem (SCP) angesprochen, aber nicht weiter ausgeführt. Erst Ribeiro und Soumis formulieren das MD-VSP als *Set Partitioning Modell* (SPP), das sie durch Dantzig-Wolfe-Dekomposition vom Mehrgüter Flussmodell ableiten (vgl. [Ribeiro und Soumis, 1994]). Die Autoren zeigen, dass die optimalen Zielfunktionswerte der LP-Relaxationen nicht schlechter sind als die LP-Relaxationen der einfachen Flussmodelle (s. Abschnitt 5). Im Set Partitioning Modell wird für jeden Umlauf r aus der Menge aller gültigen Umläufe Ω^d des Depots $d \in D$ eine Entscheidungsvariable definiert, deren Matrixkoeffizienten a_{ir} angeben, ob Servicefahrt i von Umlauf r bedient wird ($a_{ir} = 1$, sonst $a_{ir} = 0$).

$$\min \sum_{d \in D} \sum_{r \in \Omega^d} c_r^d x_r^d \quad (4.46)$$

$$s.t. \sum_{d \in D} \sum_{r \in \Omega^d} a_{ir} x_r = 1 \quad \forall i \in T, \forall d \in D \quad (4.47)$$

$$\sum_{r \in \Omega^d} x_r \leq w_d \quad \forall d \in D \quad (4.48)$$

$$x_r \in \{0, 1\} \quad \forall r \in \Omega^d, \forall d \in D \quad (4.49)$$

Die Zielfunktion (4.46) minimiert die Kosten der gewählten Umläufe über alle Depots. In (4.47) ist sichergestellt, dass jede Fahrt genau durch einen gewählten Umlauf abgedeckt wird. Die Depotkapazitäten werden in Restriktion (4.48) beachtet, indem die Anzahl der gewählten Umläufe pro Depot beschränkt werden. In (4.49) wird für jeden möglichen Umlauf je Depot eine Entscheidungsvariable definiert.

Da die Formulierung bereits für kleine Problemstellungen eine sehr große Anzahl von Variablen beinhaltet, wird die LP-Relaxation des SPPs in [Ribeiro und Soumis, 1994] mittels implizitem Column Generation gelöst, bei dem das Unterproblem zur Bestimmung der aufzunehmenden Variablen ein Kürzeste Wege Problem ist. Sie verwenden dafür die generische Column Generation Software GENCOL (vgl. [Sanso et al., 1990]).

In [Bianco et al., 1994] wird ein alternativer Weg zur Lösung des SPPs vorgeschlagen, bei dem im ersten Schritt die optimale Fahrzeuganzahl durch Lösung eines SD-VSP mit einem künstlichen Depot berechnet wird. Es wird ein Theorem präsentiert, dass auf der Basis der reduzierten Kosten und einer unteren Schranke Variablen aus der zu betrachtenden Menge aller gültigen Umläufe ausschließen kann. Dieses Theorem kann ausgenutzt werden, indem das duale Modell heuristisch gelöst wird und die dadurch entstehende untere Schranke zur Ausschließung von Variablen verwendet werden kann. Die Autoren stellen acht verschiedene Heuristiken vor und verwenden diese in einem iterativen Lösungsprozess des dualen Modells. Die Anzahl der Variablen kann durch dieses Verfahren für die betrachteten Problemstellungen so weit reduziert werden, dass das Modell mit den verbleibenden Variablen mit einem LP-basierten B&B Verfahren gelöst werden kann.

Hadjar et al. erweitern den GENCOL-Algorithmus aus [Ribeiro und Soumis, 1994] durch Strategien zur Fixierung von Variablen und zur Definition und Stärkung von validen Ungleichungen innerhalb eines Branch & Cut Verfahrens (vgl. [Hadjar et al., 2006]).

In [Oukil et al., 2004] und [Oukil et al., 2007] wird ein stabilisiertes Column Generation Verfahren beschrieben. Bei diesem werden die dualen Werte des Master Problems in der Hinsicht stabilisiert, dass sie durch geeignete Strafkosten für Abweichungen nicht von Iteration zu Iteration extrem unterschiedliche Werte annehmen, sondern stabil bleiben. Die Aussagequalität der dualen Werte und damit die Konvergenz des Column Generation kann durch diese Erweiterung vor allem für stark degenerierte Problemstellungen verbessert werden.

4.2.4 Modellunabhängige Ansätze

Neben den in den vorherigen Abschnitten beschriebenen Modellen und Verfahren für das MD-VSP existiert eine Reihe von heuristischen Lösungsansätzen, die unabhängig von den Modellierungsarten sind.

Das erste heuristische Verfahren zur Lösung des MD-VSPs wurde in [Gavish et al., 1978] vorgestellt. Es wird eine 2-Phasen-Heuristik präsentiert, bei der in der ersten Phase anonyme Umläufe gebildet werden, indem das SD-VSP mittels eines angepassten Netzwerk-Simplex-Algorithmus für ein künstliches Depot gelöst wird. In der zweiten Phase wird ein Zuweisungsproblem gelöst, welches – basierend auf den unterschiedlichen Kosten von Aus-/Einrückfahrten und ggf. unterschiedlichen Fahrzeugkosten – jeden berechneten Umlauf einem Depot zuteilt. Diese Heuristik wird auch als *Schedule-First-Cluster-Second-Heuristik (SFCS-Heuristik)* bezeichnet. In [Hoffstadt, 1981] wird das gleiche Vorgehen präsentiert, wobei das SD-VSP in der ersten Phase als Zuweisungsmodell gelöst wird.

Bodin et al. stellen eine Greedy-Heuristik vor, bei der die Servicefahrten nach ihren Abfahrtszeiten sortiert werden und nacheinander in bestehende Umläufe oder einen neuen Umlauf einsortiert werden, so dass die Zusatzkosten je Fahrt minimal sind (vgl. [Bodin et al., 1983]). Außerdem zeigen sie eine 2-Phasen Heuristik, die zuerst jede Fahrt zu einem Depot zuteilt und anschließend jedes Depot als SD-VSP optimiert. Die Methodik – auch als *Cluster-First-Schedule-Second-Heuristik (CFSS-Heuristik)* bezeichnet – stellt das natürliche manuelle Vorgehen vieler Verkehrsunternehmen zu dieser Zeit dar, wird aber von den Autoren das erste Mal formalisiert. Als weitere Methode erweitern sie den Ansatz aus [Wren, 1972] für das SD-VSP, indem jeweils zwei Umläufe gewählt werden und diese bei einem zufälligen Zeitpunkt getrennt und rekombiniert werden. Es werden allerdings keine numerischen Ergebnisse präsentiert.

El-Azm verwendet die SFCS-Heuristik als Startheuristik und verbessert die Lösung durch ein 2-opt-Verfahren (vgl. [El-Azm, 1985]). Auch in [Daduna und Mojsilovic, 1988] wird die SFCS-Heuristik zur Lösung des MD-VSPs verwendet. Die Methode berücksichtigt individuelle Kosten für mögliche Verbindungen und

wird innerhalb des Systems HOT II eingesetzt (s. Abschnitt 4.6).

In [Rousseau et al., 1990] wird eine CFSS-Heuristik vorgeschlagen, damit in realen Planungssituationen operative Restriktionen von Fahrzeugen in Verbindung mit unterschiedlichen Depots besser berücksichtigt werden können. Die SD-VSP in der zweiten Phase werden durch eine Netzwerkflussmodellierung gelöst, bei der die Fahrten und Haltestellen in einem Time-Space-Netzwerk modelliert werden, allerdings keine Aggregation der abgebildeten Aktivitäten im Netzwerk vorgenommen wird (vgl. [Desrosiers et al., 1982]).

Dell’Amico et al. präsentieren eine 3-Phasen Heuristik, bei der in der ersten Phase das MD-VSP als einfaches Flussmodell mit Teiltour Eliminierung verwendet wird (vgl. [Dell’Amico et al., 1993]). Unter Relaxation dieser Restriktionen zur Teiltour Eliminierung wird das Problem gelöst, so dass ein Umlaufplan entsteht, bei dem die Umläufe in unterschiedlichen Depots starten und enden können. Die Lösung der Relaxation bestimmt die optimale Anzahl von Fahrzeugen, so dass in der zweiten Phase die nicht verwendeten Kanten analysiert werden können. Wenn die reduzierten Kosten einer Kante zeigen, dass sie die Lösung um Kosten für ein weiteres Fahrzeug steigern würde, kann diese als „verboten“ markiert werden. In Phase 3 wird von einem Depot, dessen Fahrzeugkapazität noch nicht ausgeschöpft ist, ein kürzester Weg durch das Netzwerk gesucht, so dass weder „verbotene“ Kanten verwendet noch andere Depotknoten besucht werden. Das Verfahren übernimmt bereits gültige Umläufe aus Phase 2 und versucht iterativ die Lösungen der restlichen Umläufe zu verbessern.

Pepin et al. zeigen die ersten Ansätze, bei denen Meta-Heuristiken zur Lösung des MD-VSP verwendet werden (vgl. [Pepin et al., 2006]). Die Autoren entwickeln eine *Large Neighborhood Search Methode (LNS)*, in der in jeder Iteration ein Teil der gültigen Lösung reoptimiert wird, sowie eine *Tabu Suche* (s. Abschnitt 3.4.1) und vergleichen die Methoden mit bestehenden Heuristiken, die auf LP-Modellen der Mehrgüter Flussmodellierung basieren.

4.2.5 Eigenschaften gelöster Instanzen

Viele der vorgestellten Lösungsansätze aus der Literatur werden anhand von künstlich generierten Instanzen validiert. Diese werden zu einem Großteil mit Hilfe des in [Carpaneto et al., 1989] präsentierten Verfahrens erstellt. Gerade die Veröffentlichungen der letzten Jahre zeigen, dass sich diese Methode zu einem Quasi-Standard für die Betrachtung von künstlichen Probleminstanzen entwickelt hat. So werden die durch diese Vorgehensweise erstellen Daten zum Beispiel in [Dell’Amico et al., 1993], [Forbes et al., 1994], [Ribeiro und Soumis, 1994], [Bianco et al., 1994], [Fischetti et al., 2001b], [Hadjar et al., 2006], [Pepin

et al., 2006] und [Oukil et al., 2007] verwendet. Die Methode generiert Fahrpläne sowie weitere benötigte Informationen zur Definition von Umlaufplanungsproblemen mit mehreren Depots (MD-VSP). Eine Betrachtung von Erweiterungen der Problemstellung ist in der Veröffentlichung nicht vorgesehen. Das Ziel der Autoren war es, den Generator so zu konzipieren, dass reale Planungssituationen von Verkehrsunternehmen des öffentlichen Verkehrs simuliert werden. Für eine Analyse der Unterschiede zu realen Fahrplänen sei an dieser Stelle auf Abschnitt 6.2.2 verwiesen.

Im Folgenden wird das Verfahren zur Erstellung einer Probleminstanz beschrieben. Dabei wird die in Abschnitt 2.3 eingeführte formale Definition für die Problembeschreibung verwendet. Eingangsgrößen für die Generierung einer Probleminstanz sind die Anzahl der zu erstellenden Servicefahrten $|T|$ sowie die Anzahl der Depots $|D|$. Die relevanten Orte werden auf einer quadratischen Topologie mit einer Seitenlänge von 60 Entfernungseinheiten bestimmt. Für die Bildung der Endhaltestellen S von Servicefahrten wird die Anzahl durch einen zufälligen Wert zwischen $\frac{|T|}{3}$ und $\frac{|T|}{2}$ bestimmt. Jeder Endhaltestelle wird eine zufällig bestimmte Koordinate innerhalb des 60×60 - Quadrats zugewiesen. Für die Auswahl der Depotstandorte schlagen die Autoren zwei unterschiedliche Methoden vor. Im ersten Fall werden sie auf die gleiche Weise wie die Standorte der Endhaltestellen bestimmt. Die so generierten werden als Instanzen der Klasse A betrachtet. Alternativ (Klasse B) werden zwei Depots an den äußersten Positionen des Quadrats platziert und ein drittes Depot zufällig gewählt. Die Fahrzeugkapazitäten der Depots werden zufällig zwischen $3 + \frac{|T|}{3|D|}$ und $3 + \frac{|T|}{2|D|}$ bestimmt. Durch diese Wahl können Instanzen generiert werden, bei denen die Depotkapazitäten zu gering sind, so dass keine gültige Lösung existiert. Dieser Fall ist allerdings unwahrscheinlich.

Die Entfernung zwischen zwei Orten i und $j \in D \cup S$ wird durch die euklidische Distanz zwischen den Koordinaten von i und j bestimmt. Die Dauer einer Verbindungsfahrt zwischen diesen Punkten entspricht ebenfalls der euklidischen Distanz in Minuten. Daher entspricht eine Entfernungseinheit der Strecke, die ein Fahrzeug innerhalb von einer Minute zurücklegen kann. Die Dauer ist unabhängig von der Art der Fahrt (Service-, Umsetz- oder Depotfahrt). Die Definition der Verbindungsmatrix kann auf diese Weise für alle Kombinationen von Endhaltestellen bestimmt werden.

Für die Generierung der Servicefahrten werden zwei unterschiedliche Fahrrentypen betrachtet. Auf der einen Seite sollen „lange“ Fahrten gebildet werden, die den Service von außerstädtischen Linien oder eine Kette von innerstädtischen Servicefahrten darstellen soll. Diese langen Fahrten haben die gleiche Haltestelle als Abfahrts- und Ankunftshaltestelle. Diese wird zufällig aus der Menge aller

Haltestellen bestimmt. Die Abfahrtszeit für eine lange Fahrt wird zufällig zwischen 5:00 Uhr und 20:00 Uhr bestimmt. Die Fahrtdauer beträgt zwischen 3 und 5 Stunden, so dass die späteste mögliche Ankunftszeit um 01:00 Uhr des folgenden Tages wäre. Neben den langen Fahrten werden auch „kurze“ Fahrten gebildet. Diese sollen zusätzliche innerstädtische Fahrten repräsentieren, die in den Stoßzeiten (Berufs-, Schul- und Pendlerverkehr) vermehrt auftreten. Für diese Fahrten werden Abfahrts- und Ankunftshaltestellen unabhängig voneinander zufällig gewählt. Die Abfahrtszeit ist zwischen 7:00 Uhr und 18:00 Uhr, wobei die Stoßzeiten eine höhere Auswahlwahrscheinlichkeit haben. Die Abfahrt findet mit einer Wahrscheinlichkeit von 15% zwischen 7:00 Uhr und 8:00 Uhr statt. Genau so wird die Abfahrtszeit zu 15% zwischen 17:00 Uhr und 18:00 Uhr gewählt. Zu 70% findet die Abfahrt zwischen 8:00 Uhr und 17:00 Uhr statt, was einer Wahrscheinlichkeit von 7,8% je Stunde entspricht. Die Fahrtdauer beträgt unabhängig von dieser Auswahl die Verbindungszeit zwischen den beiden Haltestellen und einer zufälligen Auswahl zwischen 5 und 40 Minuten. Für jede zu generierende Servicefahrt besteht eine Wahrscheinlichkeit von 60%, dass sie als lange Fahrt modelliert wird, andernfalls als kurze Fahrt.

Die Fixkosten für die Verwendung eines Fahrzeugs werden mit 10.000 Kosteneinheiten modelliert. Die operativen Kosten einer Depotfahrt werden mit dem zehnfachen der Verbindungszeit modelliert. Bei einer Verbindungsfahrt wird zusätzlich zu diesen Kosten noch das zweifache der Wartezeit kalkuliert. Die Servicefahrten werden nicht mit Kosten versehen. Um dieses Kostenmodell in eine lineare Abhängigkeit von Entfernungs- und zeitabhängigen Daten zu überführen, können die Kosten pro Distanzeinheit auf 8 Kosteneinheiten gesetzt werden und die Kosten für eine Zeiteinheit außerhalb des Depots auf 2 Kosteneinheiten. Durch diese Modellierung werden auch die Servicefahrten mit Kosten versehen, allerdings stellen diese Zusatzkosten lediglich eine Verschiebung der Zielfunktion dar, da in jeder gültigen Lösung für die generierte Probleminstanz jede Fahrt bedient sein muss.

Das Verfahren zur Generierung der künstlichen Probleminstanzen ist in Algorithmus 4.1 dargestellt. Die Koordinaten eines Ortes i werden durch (x_i, y_i) dargestellt, wobei x_i die Ausprägung der horizontalen Dimension und y_i die vertikale Dimension des Quadrates darstellt. Die zeitlichen Angaben sind als Minutenanzahl nach 0:00 Uhr des betrachteten Tages gegeben. Der Operator $i \in [a; b]$ bedeutet in der Darstellung, dass für die Variable i ein zufälliger, diskreter Wert aus der Gleichverteilung zwischen a und b bestimmt wird. Analog definiert $i \in [a; b]$ die Bestimmung eines kontinuierlichen Wertes. Die Kostendefinitionen sind nicht explizit im Algorithmus gezeigt, da sie keine variable Größe für die Generierung einer Instanz darstellen. Der Algorithmus bildet Depotstandorte nach Klasse A.

Algorithmus 4.1 : Instanzengenerator nach [Carpaneto et al., 1989]

(Schritt 1) **Endhaltstellen und Depots festlegen**

wähle $|S| \in \left[\frac{|T|}{3}; \frac{|T|}{2} \right]$

for $i = 1, \dots, |S|$ **do**

$x_i \in [0; 60]$
 $y_i \in [0; 60]$
 $S = S \cup [x_i, y_i]$

for $j = 1, \dots, |D|$ **do**

$x_j \in (0; 60)$
 $y_j \in (0; 60)$
 $D = D \cup (x_j, y_j)$

(Schritt 2) **Verbindungsmatrix aufbauen**

for $i = 1, \dots, |S|; j = 1, \dots, |S|$ **do**

$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$

(Schritt 3) **Servicefahrten generieren**

for $i = 1, \dots, |T|$ **do**

$z' \bar{\in} [0; 1]$

if $z' \leq 0,4$ **then**

$g_i \in S$

$h_i \in S$

$z'' \bar{\in} [0; 1]$

if $z'' \leq 0,15$ **then**

$e_i \in [420; 480]$

else if $z'' \leq 0,70$ **then**

$e_i \in [480; 1020]$

else

$e_i \in [1020; 1080]$

$f_i \in [e_i + d_{g_i, h_i} + 5; e_i + d_{g_i, h_i} + 40]$

else

$g_i \in S$

$h_i = g_i$

$e_i \in [300; 1200]$

$f_i \in [e_i + 180; e_i + 300]$

(Schritt 4) **Depotkapazitäten bestimmen**

for $i = 1, \dots, |D|$ **do**

$w_i \in \left[3 + \frac{|T|}{3|D|}; 3 + \frac{|T|}{2|D|} \right]$

Veröffentlichung	$ D $	$ T $	Z	G	I
[Gavish et al., 1978]	-	1200	-	-	A
[Bertossi et al., 1987]	3	50	-	opt	A
[Carpaneto et al., 1989]	2	70	-	opt	C
[Carpaneto et al., 1989]	3	60	-	opt	C
[Mesquita und Paixao, 1992]	3	200	26	1,50	A
[Lamatsch, 1992]	2	215	1128	3,00	R
[Dell'Amico et al., 1993]	10	1000	9	1,0	C
[Forbes et al., 1994]	3	200	329	opt	C
[Ribeiro und Soumis, 1994]	6	300	56	0,001	C
[Bianco et al., 1994]	6	250	24	0,2	C
[Fischetti et al., 1999]	5	463	19	opt	R
[Mesquita und Paixao, 1999]	4	352	11	opt	R
[Fischetti et al., 2001b]	5	300	52	0,03	C
[Fischetti et al., 2001b]	5	580	166	0,02	R
[Haghani und Banihashemi, 2002]	4	500	1810	opt	R
[Hadjar et al., 2006]	4	500	230	opt	C
[Pepin et al., 2006]	8	1500	52	0.8	C
[Oukil et al., 2007]	3	500	-	-	C

Tabelle 4.2: Eigenschaften gelöster MD-VSP Probleminstanzen

Neben den durch diesen Algorithmus generierten künstlichen Instanzen wurden in mehreren Veröffentlichungen auch reale Problemstellungen betrachtet. Eine Übersicht über die Größe der Problemstellungen, die durch die in den vorhergegangenen Abschnitten vorgestellten Verfahren gelöst werden konnten, ist in Tabelle 4.2 gegeben. In den meisten Veröffentlichungen ist eine Vielzahl von unterschiedlichen Problemgrößen gelöst. Die angegebenen Problemgrößen sind die jeweils größten gelösten Instanzen einer Veröffentlichung. Die hier angegebenen Werte sind stark abhängig von der Wahl der Parameter und der genauen Modellierung unterschiedlicher Annahmen und Kosten. Deshalb stellt diese Übersicht keine Bewertung der Verfahren dar, sondern soll einen Überblick über die betrachteten Modellgrößen wiedergeben. Die Anzahl der Servicefahrten sowie die Anzahl der betrachteten Depots stellt die wichtigste Einflussgröße für die Einschätzung der Komplexität einer Instanz dar. Darüber hinaus spielt die Struktur der Probleminstanzen eine wichtige Rolle, da z.B. die Anzahl der möglichen Verbindungen zwischen Fahrten stark von ihr abhängig ist.

In den Spalten $|D|$ und $|T|$ ist die Anzahl der Depots und Fahrten gegeben.

Spalte Z gibt die Laufzeit in Minuten an, die benötigt wurde, um die Instanz bis zu einem in Spalte G angegebenen Gap zu optimieren. Der Gap definiert den Abstand der besten gültigen Lösung UB zur besten bekannten unteren Schranke LB in Prozent ($G = \frac{UB-LB}{LB} * 100$) bzw. „opt“, wenn die Lösung bewiesen optimal ist. Spalte I gibt die Art der verwendeten Instanzen an (C für die künstlich generierten Instanzen auf der Basis des Generators aus [Carpaneto et al., 1989], R für reale Instanzen und A für andere Instanzen unbekannter Herkunft). Im Falle eines „-“ wurden von den Autoren keine Angaben gemacht.

Die Tabelle zeigt, dass in den aufgeführten Veröffentlichungen, die ausschließlich das MD-VSP betrachten, reale und künstlich generierte Instanzen mit bis zu 4 Depots und 500 Fahrten optimal gelöst wurden. Unter Verwendung von heuristischen Lösungsmethoden wurden Instanzen bis zu 8 Depots und 1500 Fahrten mit einem Fehler von unter 1% gelöst.

4.3 Busumlaufplanung mit mehreren Depots und Fahrzeugtypgruppen

Das Umlaufplanungsproblem mit Fahrzeugtypgruppen (MVTG-VSP) stellt eine übergeordnete Problemklasse dar, von der das zuvor vorgestellte MD-VSP ein Spezialfall ist. Aus diesem Grund sind sämtliche in diesem Abschnitt beschriebenen Verfahren auch zur Lösung des MD-VSPs geeignet. Aufgrund der besonderen Anforderung, dass eine Fahrt nur von einer Untermenge aller Depots (im Sinne der Kombination von Depotstandort und Fahrzeugtyp) bedient werden darf (s. Abschnitt 2.3), bedarf es spezieller Modelle und Verfahren zur Lösung. Die Lösungsansätze werden in chronologischer Reihenfolge dargestellt.

Das in [Bodin et al., 1983] vorgestellte Greedy-Verfahren zur Lösung des MD-VSP kann auch Fahrzeugtypgruppen berücksichtigen, da bei der regelbasierten Einteilung von Fahrten in Umläufe ungültige Fahrzeugtypen ausgeschlossen werden können. Auch die CFSS-Heuristik kann für das MVTG-VSP erweitert werden, indem in der ersten Phase die Zuteilung von Fahrten zu Depots unter Berücksichtigung der Fahrzeugtypgruppen durchgeführt werden kann. Im Gegensatz zur CFSS-Heuristik ist die SFCS-Heuristik nicht übertragbar, da die in der ersten Phase erstellten Umläufe hinsichtlich der erlaubten Fahrzeugtypen ungültig sein können. Bodin et al. erweitern das in [Wren, 1972] vorgestellte 2-opt Verfahren um die Berücksichtigung von Fahrzeugtypgruppen, indem eine Überprüfung gültiger Depotzugehörigkeit in den Algorithmus eingefügt wird.

In [El-Azm, 1985] wird die für das MD-VSP vorgestellte Methode an die Problemstellung angepasst, indem die Fahrten hinsichtlich ihres Freiheitsgrades der

Zuordnungsmöglichkeiten unterschieden werden. Zuerst werden für einen Fahrzeugtyp Umläufe mit den exklusiv von ihm zu bedienenden Servicefahrten erstellt. Anschließend werden diesen Umläufen möglichst viele Fahrten hinzugefügt, die von dem Typ bedient werden dürfen und kein weiteres Fahrzeug erfordern. Dieses Vorgehen wird für alle Fahrzeugtypen durchgeführt. Sollten Servicefahrten anschließend noch nicht zu einem Umlauf zugewiesen sein, werden für diese mittels einer Greedy-Heuristik weitere Umläufe gebildet.

In [Ceder, 1993] wird eine spezielle Definition (hierarchisches Modell) der Fahrzeugtypgruppen verwendet, indem die Regel angenommen wird, dass mit Sortierung der Depots nach $d \in \{1, \dots, |D|\}$ gilt: $d \in V_t \rightarrow (d+1) \in V_t, \quad \forall t \in T$. Unter dieser Annahme wird eine iterative Heuristik präsentiert, die auf der graphischen Analyse der Deficit Function basiert (s. Abschnitt 4.1).

Forbes et al. lösen das MVTG-VSP mit Hilfe des Mehrgüter Flussmodells (vgl. [Forbes et al., 1994]). Da die unterschiedlichen Depots in den unabhängigen Netzwerkschichten modelliert werden, können die Zugehörigkeiten von Fahrten zu Fahrzeugtypen direkt im Netzwerkaufbau berücksichtigt werden. Die Schichten haben daher für das MVTG-VSP unterschiedliche Knoten- und Kantenmengen, aber das mathematische Modell bleibt bestehen. Die Autoren verwenden eine Lagrange-Relaxation Restriktionen zur Erhaltung des Netzwerkflusses in den Depotknoten, um das MD-VSP in einem 3 Phasen-Algorithmus zu lösen: in der ersten Phase wird die angesprochene Relaxation gelöst. In der zweiten Phase wird die LP-Relaxation des Gesamtmodells gelöst, indem die Lösung aus Phase 1 als Startbasis für den dualen Simplex-Algorithmus übergeben wird. In der dritten Phase wird auf der Basis der optimalen LP-Lösung ein LP-basiertes B&B Verfahren durchgeführt.

In [Costa et al., 1995] wird die Erweiterung der drei unterschiedlichen Modellierungsansätze aus Abschnitt 4.2 um die Berücksichtigung von Fahrzeugtypgruppen diskutiert. Wie bereits in [Forbes et al., 1994] beschrieben, kann das Mehrgüter Flussmodell die Fahrzeugtypgruppen im Netzwerkaufbau berücksichtigen. Das einfache Flussmodell mit Zuweisungsvariablen kann durch eine geeignete Anpassung der mathematischen Formulierung zur Lösung des MVTG-VSP verwendet werden. Die Autoren präsentieren eine Erweiterung des Set Partitioning Modell, für das die implizite Enumeration der möglichen Umläufe angepasst werden muss. Es werden die Laufzeiten und Eigenschaften von drei Lösungsmethoden für die Modelle verglichen, von denen ein Set Partitioning Ansatz (vgl. [Ribeiro und Soumis, 1994]) die besten Ergebnisse zeigt. Allerdings verwenden die Autoren sehr kleine Instanzen mit weniger als 100 Servicefahrten, so dass keine Aussagen zum Lösungsverhalten bei größeren Problemstellungen möglich ist.

Löbel löst das MVTG-VSP mit einem Column Generation Verfahren auf der

Formulierung des Mehrgüter Flussproblems, da für große Problemstellungen das Mehrgüter Flussmodell zu viele Variablen hat, um sie im Speicher zu halten oder das Modell zu lösen (vgl. [Löbel, 1998]). Durch die starke Degeneration großer MVTG-VSP-Modelle müssen sehr viele Spalten generiert werden, um die LP-Lösung im RMP zu verbessern. Da das Verfahren durch diese Schwierigkeit nur sehr langsam konvergieren würde, präsentiert der Autor eine spezielle Pricing-Methode, die er Lagrangean Pricing nennt: zur Bestimmung der zu generierenden Spalten werden Lagrange-Relaxationen mit den reduzierten Kosten einer LP-Lösung auf den Kanten gelöst und alle in einer Lösung vorhandenen Kanten generiert. Hierzu verwendet der Autor die Relaxation der Depotverhaltensbedingungen sowie die Relaxation der Überdeckungsrestriktionen. Das jeweilige Lagrange-Maximierungsproblem wird mit einer in [Kokott und Löbel, 1996] beschriebenen Subgradientenmethode gelöst. Zur Bestimmung der initialen Kantenmenge für das RMP werden zwei unterschiedliche Eröffnungsheuristiken vorgestellt. Die erste ist eine CFSS-Heuristik, bei der die Servicefahrten in der ersten Phase dem nächstgelegenen Depot zugeteilt werden. Die zweite Heuristik ist eine SFCS-Heuristik, bei der die in Phase 1 entstehenden Umläufe als Abschätzung für die benötigte Fahrzeuganzahl dienen und in der zweiten Phase möglichen Depots zugeteilt werden können. Nachgelagert werden je Depot alle ihm zugewiesenen Fahrten als SD-VSP optimiert. Für den Fall der Verletzung von Depotkapazitäten werden so lange einige in der Lösung verwendete Kanten im Netzwerk verboten und der Prozess erneut durchgeführt bis die Lösung gültig ist (vgl. [Löbel, 1997]). Der Column Generation Prozess wird so lange mit dem Lagrangean Pricing durchgeführt bis der Algorithmus konvergiert. Anschließend wird durch Einsatz des Dantzig Pricing die Optimalität der Lösung bewiesen. Zur Lösung der RMP verwendet dieser Ansatz den primalen Simplex-Algorithmus von CPLEX und führt kein LP-Preprocessing durch, um in jeder Iteration von der optimalen Basis der vorherigen Iteration starten zu können (vgl. auch [Löbel, 1999]).

In [Huisman et al., 2004] wird das Mehrgüter Matchingmodell zur Lösung von MVTG-VSP innerhalb eines Ansatzes zur dynamischen Umlaufplanung verwendet. Zur Lösung der mathematischen Modelle wird CPLEX verwendet.

Kliwer verwendet die mathematische Formulierung des Mehrgüter Flussmodells und stellt eine alternative Modellierung des dazu verwendeten Netzwerks auf (vgl. [Kliwer, 2005]). Sie nutzt dabei die Eigenschaft des Fahrplans, dass die Definition der Kompatibilität von Fahrten transitiv ist. Das heisst für drei Servicefahrten $i, j, k \in T$ gilt:

$$(i \alpha j) \wedge (j \alpha k) \rightarrow (i \alpha k) \quad \forall i, j, k \in T$$

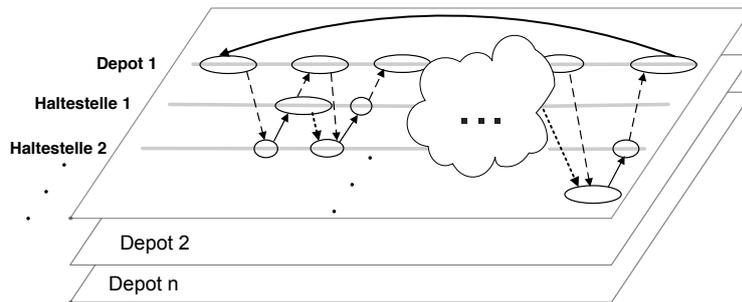


Abbildung 4.6: Time-Space-Netzwerk für das MVTG-VSP

Diese implizite Definition der Kompatibilität von Servicefahrten verwenden die Autoren, um alle Umlaufmöglichkeiten in einem aggregierten Time-Space-Netzwerk zu modellieren. In diesem werden die Haltestellen als Zeitschienen (Timelines) betrachtet, zwischen denen Servicefahrten getätigt werden müssen und Umsetz- oder Ein-/Ausrückfahrten stattfinden können. Durch zusätzliche Wartekanten innerhalb der Zeitlinien kann die oben vorgestellte Transitivität sichergestellt werden. Somit ist die maximale Anzahl der für jede Servicefahrt zu betrachtenden Umsetzungsmöglichkeiten durch die Anzahl der Endhaltestellen $|S|$ und Servicefahrten $|T|$ ($\mathcal{O}(|T| \cdot |S|)$ statt der quadratischen Anzahl der Servicefahrten $\mathcal{O}(|T| \cdot |T|)$) nach oben begrenzt. Dies führt auf realen Instanzen zu einer Einsparung von 97% - 99% der in den anderen Modellen notwendigen Verbindungskanten ohne die optimale Lösung des MVTG-VSPs zu verlieren. Die Knoten des Netzwerks können aggregiert werden, wodurch die Anzahl der benötigten Restriktionen zur Flussershaltung reduziert werden kann. Der Aufbau eines Time-Space-Netzwerks ist in Abbildung 4.6 illustriert.

Die mathematische Formulierung ändert sich nur in der Definition der Variablen, die nun allgemein ganzzahlig sein müssen ($x_{ij}^d \in \mathbb{N}$), da durch die Aggregation der Fluss von mehr als einer Flusseinheit auf einer Kante (außer einer Servicefahrt-Kante) möglich ist. Die Autoren lösen das vollständige mathematische Modell mit Hilfe eines MIP-Solvers (MOPS oder CPLEX), da durch das aggregierte Netzwerk die Anzahl der Variablen ausreichend reduziert werden konnten. Die Lösung entspricht einer Flusslösung, die mehrere Flusseinheiten pro Kante enthalten kann. Aus diesem Grund werden in einem nachgelagerten Schritt die Umläufe als Wege einzelner Flusseinheiten aus den Flüssen extrahiert. Da ein aggregierter Netzwerkknoten mehrere eingehende Flusseinheiten enthalten kann, können durch unterschiedliche Zuordnung der eingehenden zu den ausgehenden Flusseinheiten unterschiedliche Umlaufpläne mit gleichen Gesamtkosten gebildet werden. Dies bietet den Vorteil, dass nach der Optimierung eine Vielzahl von

Veröffentlichung	$\varnothing D$	$ T $	Z	G	I
[El-Azm, 1985]	-	400	190	-	-
[Forbes et al., 1994]	1,31	600	90	opt	R
[Löbel, 1998]	1,56	24906	591	- ¹	R
[Huisman et al., 2004]	1,71	1104	2	opt	R
[Kliwer, 2005]	6,5	11062	513 ²	opt	R
[Gintner et al., 2005]	6,5	11062	303	0,004 ³	R

¹ Optimale Fahrzeuganzahl, aber keine Aussage zum Gap der operativen Kosten

² Laufzeit zur Lösung der LP-Relaxation

³ Gap der operativen Kosten

Tabelle 4.3: Eigenschaften gelöster MVTG-VSP Probleminstanzen

kostenminimalen Lösungen zur Auswahl steht, aus denen in der Dekomposition eine oder mehrere geeignete auszuwählen sind. Die Autoren präsentieren unterschiedliche Strategien zur Auswahl, die je nach Zielsetzung und Einsatzgebiet unterschiedlich sind (vgl. auch [Kliwer et al., 2002] und [Kliwer et al., 2006b]).

Gintner et al. präsentieren einen heuristischen Ansatz, der auf der aggregierten Time-Space-Netzwerk Formulierung aus [Kliwer et al., 2006b] basiert und ein zweistufiges Verfahren zur Lösung sehr großer MVTG-VSPs darstellt (vgl. [Gintner et al., 2005]). In der ersten Phase werden Unterprobleme erstellt, die alle Servicefahrten beinhalten, aber deren Komplexität durch die Betrachtung von einem oder nur einigen Depots/Fahrzeugtypen erheblich verringert ist. Diese Problemstellungen werden in der ersten Phase unabhängig voneinander gelöst und es werden mit unterschiedlichen Dekompositionsstrategien Umläufe erstellt. Anschließend werden die Fahrtenverbindungen, die in allen Umlaufplänen auftreten, fixiert und die so entstehenden Fahrketten als eine Fahrt für die Lösung des MVTG-VSPs mit allen Depots/Fahrzeugtypen betrachtet. Durch die Anzahl der in Phase 1 zu erstellenden Umlaufpläne kann das Verhältnis von Lösungsqualität zu Laufzeit geregelt werden, da mit einer großen Anzahl von erstellten Umlaufplänen tendenziell weniger Verbindungen fixiert werden können, aber die Lösung näher am Optimum liegt und umgekehrt. In der Regel sind die Lösungen sehr nah am Optimum, so dass sie aus praktischer Sicht als quasi-optimal betrachtet werden können.

Die in der Literatur präsentierten Ergebnisse zur Lösung von MVTG-VSPs unterscheiden sich stark in Bezug auf Struktur und Problemgröße der Problemstellungen, da jeweils unterschiedliche reale Daten für die numerischen Ergeb-

nisse verwendet wurden. Im Gegensatz zum MD-VSP ist die Anzahl der Depots kein aussagekräftiger Parameter zur Einschätzung der Komplexität, da nicht jede Fahrt von jedem Depot bedient werden kann. Vielmehr gibt die Kennzahl der *durchschnittlichen Gruppengröße* $\varnothing D$ den Grad der kombinatorischen Komplexität in der Dimension der Depotzuteilung an. Konkret gibt die Kennzahl an, von wie vielen unterschiedlichen Depot-Fahrzeugtyp-Kombinationen eine Servicefahrt durchschnittlich bedient werden kann. Tabelle 4.3 gibt eine Übersicht über die größten in den Veröffentlichungen betrachteten Problemstellungen, sofern dazu Angaben gemacht wurden (s. Tabelle 4.2).

4.4 Busumlaufplanung mit Zeitfenstern

Das Umlaufplanungsproblem mit der Berücksichtigung von Zeitfenstern für Servicefahrten ist – wie in Abschnitt 2.3.5 vorgestellt – eine hochgradig praxisrelevante Problemstellung. Aus diesem Grund reichen die Anfänge der Optimierungsansätze für das VSP-TW bis in die frühen 1970er Jahre zurück. In diesem Abschnitt werden zuerst die Ansätze für das VSP-TW mit einem Depot und anschließend die kombinierte Betrachtung von weiteren Problemstellungen aufgezeigt. Die Ansätze werden innerhalb ihrer Kategorie in chronologischer Reihenfolge präsentiert.

VSP-TW mit einem Depot Martin-Löf schlägt eine Flussformulierung auf der Basis eines rudimentären Time-Space-Netzwerks vor (vgl. [Martin-Löf, 1970]). Dabei beachtet der Autor weder die Möglichkeit von Umsetzfahrten noch von Depotfahrten. Die Zeitfenster für Fahrten werden durch die Einteilung der Zeitlinien einer Haltestelle in diskrete Zeitpunkte betrachtet. Es wird ein Flussproblem formuliert, das für jeden möglichen (diskreten) Zeitpunkt die Flusserhaltung in jeder Haltestelle sicherstellt. Zusätzlich werden Überdeckungsrestriktionen eingeführt, die gewährleisten, dass für jede Fahrt ein Zeitpunkt gewählt wird, an dem sie innerhalb ihres Zeitfensters abfährt. Ein Branch & Bound Algorithmus wird zur Lösung des Modells vorgestellt.

In [Levin, 1971] wird die Berücksichtigung von diskreten Zeitfenstern für das *Flugzeugplanungsproblem* (engl: *aircraft rotation problem*) (ohne Deadheading und Depots) modelliert, indem in ein Netzwerkflussmodell für jede mögliche Ausprägung des Zeitfensters eines Fluges Kanten-Duplikate mit den entsprechenden Abflugszeiten eingefügt werden. Die originale Flugkante sowie ihre Duplikate werden durch zusätzliche Überdeckungsrestriktionen miteinander verbunden und es wird sichergestellt, dass genau eine in einer gültigen Lösung gewählt wird. Die-

ser Ansatz ist nicht explizit zur Lösung von Busumläufen veröffentlicht worden, allerdings kann das Prinzip dieser Modellierung für die Busumlaufplanung mit Zeitfenstern übernommen werden.

Orloff formuliert ein Set Covering Problem zur Lösung des VSP-TW und schlägt valide Ungleichungen zur Verstärkung der LP-Relaxation vor (vgl. [Orloff, 1976]). Da das Modell zu viele Spalten enthalten würde, um es im Speicher halten zu können, verwerfen die Autoren den Ansatz und schlagen stattdessen eine Divide & Conquer Heuristik vor, bei der jeweils zwei Servicefahrten gewählt werden und deren Anschluss fixiert wird. Dieses Vorgehen wird mit bereits fixierten Reihen von Servicefahrten weitergeführt bis keine gültigen Anschlüsse mehr gefunden werden. Anschließend wird das für das TSP entwickelte 3-opt Verfahren aus [Lin und Kernighan, 1973] auf die Verbindungen angewendet.

In [Bokinge und Hasselström, 1980] wird eine Heuristik vorgestellt, die zuerst ein SD-VSP löst, bei dem die Fahrten bei ihrer durch die Zeitfenster zugelassenen spätest möglichen Abfahrt beginnen und bei ihrer frühest möglichen Ankunft enden. Die Lösung stellt eine untere Schranke für das VSP-TW dar. Beginnend mit den Fahrten mit den kleinsten Zeitfensterintervallen werden die Abfahrtszeiten iterativ so fixiert, dass möglichst wenig Fahrten gleichzeitig stattfinden. Das durch die fixierten Abfahrtszeiten definierte SD-VSP wird optimiert und der Prozess wird iterativ bis zur Konvergenz des Algorithmus fortgesetzt. Die Autoren verwenden zur Lösung des SD-VSP eine Netzwerkflussmodellierung mit einem Time-Space-Netzwerk, bei dem die Depot-Knoten innerhalb des Netzwerks für Zeitscheiben von 30 Minuten eingefügt werden. Das resultierende Netzwerkflussproblem wird mit dem primalen Netzwerk-Simplex Algorithmus PNET optimiert (vgl. [Glover et al., 1974]).

Desrosiers et al. formulieren das VSP-TW als Set Partitioning Problem, wobei die Menge der gültigen Umläufe durch alle durch die Verwendung der Zeitfenster ermöglichten Umläufe erweitert wird (vgl. [Desrosiers et al., 1984]). Sie lösen die LP-Relaxation mit einem Column Generation Verfahren, bei dem als Unterproblem ein *Kürzeste Wege Problem mit Zeitfenstern* (engl: *shortest path problem with time windows*) zu lösen ist. Die Autoren verwenden den Ford-Bellman Algorithmus zur Lösung des Unterproblems und die Simplex-Methode für das Masterproblem. Eine optimale Lösung wird auf Basis der LP-Relaxation mit einem Branch & Cut Algorithmus bestimmt.

In [El-Azm, 1985] wird eine untere Schranke für das VSP-TW durch Verkürzung der Fahrten wie in [Bokinge und Hasselström, 1980] bestimmt. Auf dieser Relaxation wird im Anschluss ein Branch & Bound Algorithmus ausgeführt, der für jede ungültige Verbindung zwei Unterprobleme bildet, in der Verwendung dieser Verbindung erzwungen bzw. verboten wird. Dadurch bedarf es in der Lösung

der Unterprobleme keiner weiteren Restriktionen, sondern es müssen ausschließlich SD-VSP gelöst werden.

In [Desrosiers et al., 1985] wird das VSP-TW als SD-VSP mit zusätzlichen Variablen und Restriktionen für die Abfahrtszeiten der Servicefahrten modelliert. Auf der Relaxation der zusätzlichen Nebenbedingungen wird ein Branch & Bound Verfahren ausgeführt, für das verschiedene Branching-Strategien aufgezeigt werden.

Ferland und Fortin beschreiben eine heuristische Methode zur Lösung des VSP-TW sowie einer erweiterten Problemstellung, in der Gruppen von Fahrten zusätzlich innerhalb eines Zeitfensters verschoben werden können (vgl. [Ferland und Fortin, 1989]). Diese Verschiebung muss allerdings um den gleichen Betrag für alle Fahrten einer Gruppe gelten. Durch diese zusätzliche Verschiebung von Fahrtengruppen sollen externe Einflüsse (z.B. Schulschlüsse) modelliert werden. Zur Lösung eines initialen SD-VSPs (ohne Zeitfenster) verwenden die Autoren eine Netzwerk-Simplex Methode auf dem Netzwerkflussmodell. Die reduzierten Kosten in der optimalen Lösung werden analysiert und vielversprechende Fahrten werden auf eine andere Zeit fixiert. Das Wechselspiel von Optimierung, Analyse und Fixierung der Abfahrtszeiten wird iterativ fortgesetzt bis das Verfahren konvergiert.

Daduna et al. betrachten das VSP-TW mit Fokus auf die Sichtweise eines Benutzers (vgl. [Daduna et al., 1993] und [Daduna und Völker, 1997]). Sie lösen das SD-VSP mit Hilfe des Zuweisungsmodells und ermitteln somit die optimale Anzahl der benötigten Fahrzeuge ohne Zeitfenster. Anschließend werden neue Verbindungsmöglichkeiten innerhalb des Modells zugelassen (zu höheren Kosten), die durch die Verwendung von Zeitfenstern erlaubt sind. Das Modell wird mit der zusätzlichen Restriktion, dass nur die um eins verringerte optimale Anzahl an Fahrzeugen erlaubt ist, gelöst. Dadurch erreichen die Autoren die Einsparung eines Fahrzeugs mit möglichst wenigen Änderungen im Fahrplan, da die operativen Kosten für die durch Zeitfenster hergestellten Verbindungen höher sind. Die so ermittelten Fahrtenverschiebungen, die zu der Einsparung eines Fahrzeugs führen würden, werden dem Benutzer präsentiert und zu jeder in der Lösung verschobenen Fahrt wird abgefragt, ob die Verschiebung zulässig ist.

In [Ceder, 2001] wird ein Ansatz zur integrierten Fahrplanerstellung und Umlaufplanung präsentiert. Das Verfahren verwendet die Deficit Function um nach der Erstellung eines Fahrplans Umläufe zu bilden. In einem iterativen Prozess werden der Fahrplan und der Umlaufplan angepasst bis eine für den Benutzer zufriedenstellende Abwägung zwischen Kosten und Servicegrad erreicht ist.

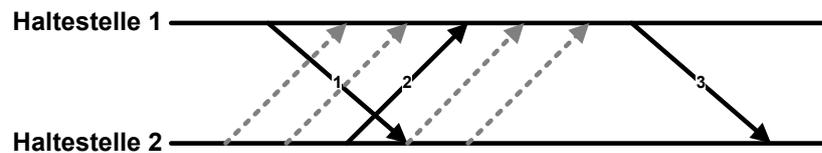


Abbildung 4.7: Modellierung diskreter Zeitfenster nach [Kliwer, 2005]

VSP-TW mit mehreren Depots Mingozi et al. erweitern den SPP-Ansatz aus [Bianco et al., 1994] um die Berücksichtigung von Zeitfenstern für Fahrten, indem sie das Pricing Problem als *Kürzeste Wege Problem mit Zeitfenstern* (engl.: *shortest path problem with time windows*) lösen (vgl. [Mingozi et al., 1995]).

In [Desaulniers et al., 1998] wird das VSP-TW mit mehreren Depots und nicht-linearen Kostenstrukturen (z.B. exponentielle Kostenfunktion für Wartekosten) betrachtet. Die Autoren formulieren ein nicht-lineares Mehrgüter Netzwerkflussproblem und formen es in ein SPP-Modell um. Als Lösungsverfahren wird ein Column Generation Verfahren verwendet, bei dem die Zeitfenster ähnlich wie in [Desrosiers et al., 1984] berücksichtigt werden.

Hadjar und Soumis erweitern das Column Generation Verfahren für das SPP Modell, indem das Unterproblem mit einem spezialisierten Algorithmus der dynamischen Programmierung gelöst wird (vgl. [Hadjar und Soumis, 2005]). Außerdem werden Beschleunigungstechniken präsentiert, die durch die Fixierung von Servicefahrten auf feste Abfahrtszeiten oder Löschung von Verschiebemöglichkeiten zu einer Verringerung der kombinatorischen Komplexität führen.

VSP-TW mit Fahrzeugtypgruppen Ferland und Michelon betrachten das VSP-TW mit Fahrzeugtypgruppen und erweitern unterschiedliche Heuristiken und exakte Verfahren für das MVTG-VSP aus der Literatur um die Berücksichtigung von Zeitfenstern für Servicefahrten (vgl. [Ferland und Michelon, 1988]). Sie zeigen die Erweiterung für die Heuristiken aus [Levin, 1971], [Orloff, 1976] und [Bodin et al., 1983] sowie für die exakten Ansätze aus [Desrosiers et al., 1984] und [Desrosiers et al., 1985], führen allerdings keine numerischen Tests zur Evaluierung der unterschiedlichen Ansätze durch.

Kliwer erweitert das Modell zur Lösung von MVTG-VSP um die Berücksichtigung von Zeitfenstern (vgl. [Kliwer, 2005]). In das aggregierte Time-Space-Netzwerk werden für alle diskreten Ausprägungen der jeweiligen Zeitfenster Duplikate der Fahrkanten eingefügt (s. Abbildung 4.7). Die Kanten einer Fahrt werden über alle Zeitfensterausprägungen und alle Depots in die Überdeckungsrestriktionen integriert, um die einmalige Bedienung jeder Servicefahrt sicher-

zustellen. Die Komplexität der durch die weiteren Kanten erzeugten Verbindungsmöglichkeiten wird zum einen durch die Aggregation des TSN und zum anderen durch ein Verfahren zur Identifizierung und Löschung redundanter Kanten verringert. Zur Lösung von großen Problemen mit der Berücksichtigung von Zeitfenstern präsentieren die Autoren zwei Verfahren, um kritische Fahrten zu identifizieren, die für eine Einsparung von Fahrzeugen am vielversprechendsten sind. Unter ausschließlicher Berücksichtigung von Zeitfenstern für die identifizierten Fahrten kann das Problem heuristisch gelöst werden. Die Identifikationsverfahren basieren zum einen auf der Analyse einer Optimierung mit verkürzten Fahrten und zum anderen auf der Analyse der Fahrtenauslastung über den Tagesverlauf.

4.5 Busumlaufplanung mit Routenrestriktionen

Die Lösung von VSP unter Berücksichtigung von weiteren Restriktionen auf gesamten Umläufen (VSP-RC) wird im Falle von verschiedenen praktischen Rahmenbedingungen relevant. So werden die Anforderungen wie die Beschränkung der Umlaufdauer, eine maximale Anzahl von Linienwechseln oder eine durch Treibstoff beschränkte Länge der Fahrtstrecke allgemein zu Routenrestriktionen zusammengefasst. Die Ansätze zur Lösung dieser allgemeinen oder einer speziellen Problemstellung werden in diesem Abschnitt vorgestellt, wobei die Erweiterung des Ein- und Mehrdepotproblems getrennt behandelt wird.

SD-VSP mit Routenrestriktionen Bodin et al. präsentieren eine Erweiterung der in Abschnitt 4.2.4 vorgestellten Greedy-Heuristik um die Betrachtung beliebiger Routenrestriktionen, indem bei der Zuweisung einer Fahrt zu einer Route neben dem übrigen Regelwerk auch die Verletzung einer Routenrestriktion ausgeschlossen wird (vgl. [Bodin et al., 1983]). Darüber hinaus stellen die Autoren ein mathematisches Modell zur Lösung des VSP-RCs auf, indem sie jede Fahrt als einen Knoten definieren und für mögliche Paare von kompatiblen Servicefahrten ($i, j \in T$ mit $i \alpha j$) Kanten mit den operativen Fahrzeugkosten einfügen. Die Einhaltung der Routenrestriktionen wird über Rückkanten sichergestellt. Diese werden als entgegengesetzte Kanten (j, i) mit den Fixkosten für die zusätzliche Verwendung eines weiteren Fahrzeugs eingefügt, wenn die Bedienung beider Fahrten keine Routenrestriktion verletzt. Das mathematische Modell stellt die Flusserhaltung sowie die Bedienung jeder Fahrt sicher und enthält zusätzliche Restriktionen, die gewährleisten, dass jeder gültige Umlauf maximal eine Rückkante enthält. Diese Restriktionen sichern die Einhaltung der Routenrestriktionen.

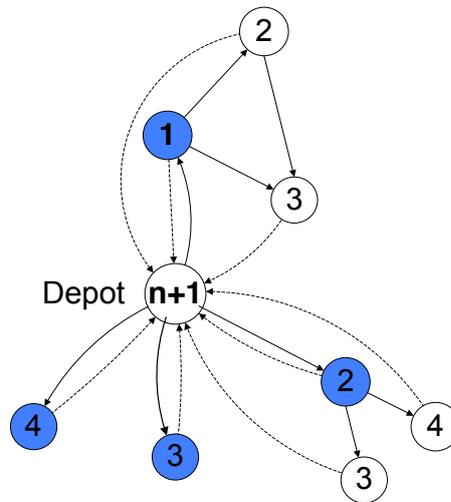


Abbildung 4.8: Multilevel Modell für das VSP-RC

In [Bodin et al., 1985] wird ein Verfahren beschrieben, in dem beliebige Routenrestriktionen durch eine konstruktive Heuristik berücksichtigt werden. Die Servicefahrten werden so in Gruppen aufgeteilt, dass jede mögliche Zusammenstellung von Fahrten in einer Gruppe keine Routenrestriktion verletzt. Diese Fahrtengruppen werden jeweils als SD-VSP gelöst. Anschließend wird durch die Zusammenlegung mehrerer Umläufe versucht die Lösung unter Einhaltung der Routenrestriktionen zu verbessern.

Freling und Paixao präsentieren zwei Modellierungsansätze für das VSP-RC (vgl. [Freling und Paixao, 1995]). Das erste Modell basiert auf einem speziellen Graphen, der einen Depotknoten und daran angeschlossen einen Fahrtnknoten für jede Servicefahrt enthält. Diese Fahrtnknoten repräsentieren den Beginn eines Umlaufs mit der jeweiligen Fahrt. Angeschlossen an den Fahrtnknoten der Servicefahrt i ist jeweils ein Teilgraph mit allen Fahrten (und den entsprechenden kompatiblen Kanten), die in einem mit i beginnenden Umlauf enthalten sein können ohne eine Routenrestriktion zu verletzen. Von jedem Knoten führt eine Rückkante zum Depotknoten zurück, so dass eine Route vom Depotknoten über einen oder mehrere Fahrtnknoten zurück zum Depotknoten einen gültigen Umlauf darstellt. Ein Beispiel für einen solchen Graphen – auch Multilevel Modell genannt – mit $|T| = 4$ ist in Abbildung 4.8 dargestellt. In dem Beispiel ist ein Umlauf beginnend mit Fahrt 1 möglich, der auch Fahrten 2 und 3 bedient. Durch eine Routenrestriktion ist die zusätzliche Bedienung von Fahrt 4 nicht möglich. Beginnend mit Fahrt 2 ist die Bedienung von Fahrt 4 allerdings möglich, da die Routenrestriktion nicht verletzt ist.

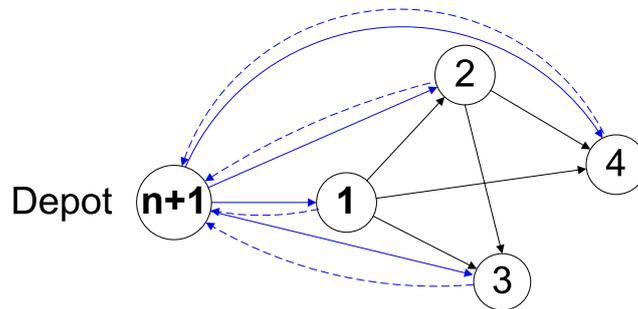


Abbildung 4.9: Rückkanten Modell für das VSP-RC

Der zweite Ansatz aus [Freling und Paixao, 1995] nimmt die Idee der Rückkanten auf und modelliert das VSP-RC als Netzwerk mit einem Depotknoten, zu dem Rückkanten von allen Fahrtknoten führen. Im Gegensatz zum Multilevel Modell muss die Berücksichtigung der Routenrestriktionen explizit durch zusätzliche Restriktionen in der mathematischen Formulierung sichergestellt werden. Der Graph des Modells mit Rückkanten ist für das Beispiel aus Abbildung 4.8 in Abbildung 4.9 illustriert.

MD-VSP mit Routenrestriktionen In [Mingozi et al., 1995] wird der Ansatz aus [Bianco et al., 1994] erweitert, indem die Routenrestriktionen innerhalb des Set Partitioning Modells berücksichtigt werden. Die Einschränkung der Routenmöglichkeiten führt innerhalb der Column Generation Methode zur Anpassung des Pricing Problems, in dem neue Variablen gesucht werden. Statt einen kürzesten Pfad zu suchen, muss ein *Ressourcen-beschränktes Kürzeste Wege Problem* (engl: *resource constrained shortest path problem*) gelöst werden, bei dem jede Routenrestriktion über eine Pfad-Ressource abgebildet wird.

Haghani und Banihashemi erweitern ihr in Abschnitt 4.2.2 vorgestelltes Modell zur Lösung des MD-VSPs um Nebenbedingungen, die jeden möglichen Umlauf, der eine Routenrestriktion verletzt, durch eine Restriktion explizit ausschließen (vgl. [Haghani und Banihashemi, 2002]). Da dieses Modell nicht praktikabel zu lösen ist, lösen sie das MD-VSP iterativ und fügen jeweils nur die verletzten Routenrestriktionen ein. Die Autoren stellen fest, dass der Gap zwischen optimalen Lösungen des MD-VSPs und des VSP-RTCs sehr klein ist. Deshalb schlagen sie eine Heuristik vor, bei der innerhalb des Verfahrens die bereits gültigen Umläufe fixiert werden und aus ungültigen Umläufen der größtmögliche gültige Teil fixiert wird. Eine alternative Heuristik löst statt der MIP-Formulierung die LP-Relaxation des Modells und fixiert nicht fraktionale Variablen in der optimalen LP-Lösung.

In [Haghani et al., 2003] wird für das Modell eine Heuristik vorgeschlagen, die jede Fahrt einem Depot zuweist und anschließend mehrere VSP-RC mit jeweils einem Depot löst. Als Zuweisung der Fahrten zu Depots schlagen die Autoren eine Entscheidung des Planers oder eine Bestimmung durch Lösen eines MD-VSPs (ohne Routenrestriktionen) vor.

Kliwer et al. diskutieren die Beachtung von Routenrestriktionen für die Modellierung des MVTG-VSPs mit einem aggregierten Time-Space-Netzwerk (vgl. [Kliwer et al., 2004]). Für die Betrachtung der Routenbedingungen – in diesem Fall die Reduktion der Anzahl an Linienwechseln oder der Anzahl unterschiedlicher Linien pro Fahrzeugumlauf – als weiche Bedingung werden unterschiedliche Methoden zur Dekomposition der kostenoptimalen Flusslösung vorgestellt. Neben regelbasierten Entscheidungsstrategien, die alle Netzwerkknoten unabhängig voneinander betrachteten, werden auch globale Dekompositionsalgorithmen präsentiert. Zum Beispiel kann durch die Lösung eines Set Partitioning Problems zur Dekomposition des Flusses in die Umläufe die bestmögliche Erfüllung von Routenrestriktionen bei optimalen Kosten gefunden werden. Zur weiteren Betrachtung der Anzahl der Linienwechsel pro Umlauf als Routenrestriktion präsentieren die Autoren eine Modellerweiterung des aggregierten Time-Space-Netzwerks. Durch das Einfügen von so genannten Linienkanten, die von einer Ankunft einer Servicefahrt zu der nächsten Fahrt der gleichen Linie führen und negative Kosten haben, wird die Verwendung von gleichen Linien innerhalb eines Umlaufs belohnt. Dadurch kann die Linienheterogenität bei leichter Anhebung der geplanten Kosten weiter reduziert werden.

In [Vanitchakornpong et al., 2008] wird eine lokale Suchmethode vorgestellt, die auf einer Matrix mit den Dimensionen der Servicefahrten und der verfügbaren Fahrzeuge basiert. In jeder Iteration des Verfahrens wird für eine zufällig bestimmte Fahrt die mögliche Einordnung in die Umläufe anderer Fahrzeuge untersucht und bewertet. Neben den fixen und operativen Kosten verwendet die Methode eine Gewichtung unterschiedlicher harter und weicher Restriktionen zur Berechnung der Kosten. Das Verfahren kann daher beliebig komplexe Nebenbedingungen für VSP berücksichtigen (z.B. Routenrestriktionen).

4.6 Planungssysteme mit Komponenten zur Busumlaufoptimierung

Die in den vorherigen Abschnitten beschriebenen Veröffentlichungen haben einen Fokus auf die Modellierung und Lösung von Busumlaufplanungsproblemen als wissenschaftliche Fragestellung. Allerdings gibt es auch aus der Sichtweise des

praktischen Einsatzes von Optimierungsverfahren zur Busumlaufplanung eine Reihe von Veröffentlichungen. Diese lassen sich in der Regel einem konkreten Entscheidungsunterstützungssystem zur Planung von Busumläufen zuordnen. Deshalb sollen im Folgenden die wichtigsten Softwaresysteme zur Busumlaufplanung vorgestellt werden. Eine Vielzahl der beschriebenen Systeme beinhaltet neben der Komponente zur Lösung von Umlaufplanungsproblemen auch weitere Module und Funktionen. Da es in dieser Arbeit um die Lösung von VSP geht, wird allerdings ausschließlich auf die Umlaufplanungskomponenten der Systeme eingegangen. Die Systeme werden in alphabetischer Reihenfolge präsentiert.

AUTOBUS / OPTIBUS Das System AUTOBUS ist ein graphisches System, das durch einen hohen Grad an Interaktion zwischen Mensch und Maschine versucht, den Planer bei seinen Entscheidungen zu unterstützen. Die VSP-Komponente wird in [Ceder und Stern, 1985] beschrieben und verwendet das Deficit Function Verfahren aus [Ceder und Stern, 1981]. Ende der 1980er Jahre ist aus dem AUTOBUS-System ein erweitertes Planungssystem mit dem Namen OPTIBUS hervorgegangen. In [Ceder et al., 1990] wird die erweiterte Funktionalität hinsichtlich der Erstellung von Fahr-, Umlauf- und Dienstplänen beschrieben. Das Verfahren für die Umlaufplanung bleibt methodisch unverändert.

BERTA Das in [Löbel und Strubbe, 1995] vorgestellte Verfahren auf der Basis des Mehrgüter Flussmodells wird als Optimierungsmodul für VSP innerhalb des Planungssystems BERTA der Berliner Verkehrsbetriebe (BVG) eingesetzt. Die Autoren beschreiben darüber hinaus praxisrelevante Anforderungen und Zielsetzungen, die von einem VSP-System gefordert werden.

DIVA Das „Dialoggesteuerte Verkehrsmanagement- und Auskunftssystem (DIVA)“ umfasst die Unterstützung einer Vielzahl von Planungsentscheidungen für Verkehrsbetriebe. Innerhalb der Komponente zur Planung der Busumläufe wird das Mehrgüter Flussmodell aus [Löbel, 1997] verwendet.

HASTUS Das Planungssystem HASTUS beinhaltet eine Umlaufoptimierung mit automatisierter Planung von Fahrern (vgl. [Rousseau und Blais, 1985]). Das System verwendete zur Lösung der VSP anfangs die in [Scott, 1985] beschriebene Methode, die später durch den in [Blais und Rousseau, 1990] beschriebenen Set Partitioning Ansatz des Moduls MINIBUS ersetzt wird. Die Erweiterung der verwendeten Methoden durch die Möglichkeit einer Betrachtung von Linienwechseln pro Block, Linienbündeln, Mehrdepotplanung und Zeitfenstern für Fahrten wird in [Hamer und Seguin, 1992] vorgestellt (vgl. auch [Hanisch, 1999]).

HOT Das System HOT wurde für die Fahrzeugeinsatzplanung bei der Hamburger Hochbahn entwickelt (vgl. [Hoffstadt, 1981]). Der Nachfolger HOT II wird in [Daduna und Mojsilovic, 1988] beschrieben und umfasst eine interaktive Heuristik zur Berücksichtigung von Zeitfenstern für Servicefahrten (vgl. auch [Daduna et al., 1993]).

INTERPLAN / MOBILE-PLAN Das in [Kliwer, 2005] vorgestellte Mehrgüter Flussmodell auf der Basis eines aggregierten Time-Space-Netzwerks wurde in die operative Planungssoftware INTERPLAN der Firma PTV AG integriert. Anfang 2008 ist das Produkt von der INIT AG übernommen worden und wird seitdem innerhalb des Produkts MOBILE-PLAN weitergeführt.

MICROBUS Das Softwaremodul MICROBUS (vgl. [Bertram und Winckler, 1990]) beinhaltet eine Optimierungskomponente auf der Basis des Mehrgüter Flussmodells aus [Löbel, 1997] und wird innerhalb des professionellen Planungssystems für den öffentlichen Personennahverkehr IVU.suite eingesetzt.

RUCUS Das vom US-amerikanischen Amt für Verkehr beauftragte Planungssystem für Umlauf- und Dienstplanung RUCUS führte laut [Bennington und Rebibo, 1975] aufgrund fehlender Interaktivität und unverständlicher Sprache zu wenig Akzeptanz innerhalb der Verkehrsunternehmen. Aus diesem Grund wurde der Nachfolger RUCUS II entwickelt (vgl. [Luedtke, 1985]), der unter Verwendung einer modellunabhängigen Heuristik eine weite Verbreitung innerhalb der USA erreichte.

UCOST In [Bodin et al., 1978] wird das Planungssystem UCOST beschrieben, mit dem eine Linien- und Umlaufplanung sowie eine Abschätzung der benötigten Fahrer durchgeführt werden kann. Die Autoren gehen vor allem auf die Handhabung unterschiedlicher Kosten ein. Als Optimierungsverfahren zur Lösung von VSP wird die in [Bodin und Rosenfield, 1976] beschriebene Heuristik verwendet.

VAMPIRES / TASC / BUSMAN / BOOST Das in England entwickelte System VAMPIRES verwendet laut [Smith und Wren, 1981] den Lösungsansatz aus [Wren, 1972]. Das System kann unterschiedliche Fahrzeugtypen beachten und schlägt dem Benutzer im Falle ungültiger Verbindungen die Verschiebung von Servicefahrten vor. In [Smith und Wren, 1981] wird das zunächst unabhängige System TASC zur Planung von Fahr- und Umlaufplänen beschrieben. Da es für den Planer sehr einfach zu bedienen ist, findet es eine erheblich bessere Akzeptanz, obwohl der Planungsalgorithmus eine – aus methodischer Sicht einfache –

Konstruktionsheuristik ist. Die beiden Systeme werden in dem Entscheidungsunterstützungssystem BUSMAN zusammengeführt (vgl. [Williamson, 1985]). Die Heuristik aus TASC wird als Analysewerkzeug zur Durchführung von Szenarien verwendet und der Algorithmus aus VAMPIRES wird zu MINI-VAMPIRES umbenannt und in das Planungssystem integriert, so dass die Daten aus anderen Planungsphasen direkt für die Busumlaufoptimierung verfügbar sind. Das System BUSMAN II beinhaltet Weiterentwicklungen, die allerdings nicht die Optimierungsmethode betreffen (vgl. [Wren und Chamberlain, 1990] und [Chamberlain und Wren, 1992]). In [Kwan und Rahin, 1999] wird die Überführung des Systems und der Algorithmen von der prozeduralen in die objektorientierte Programmierung dokumentiert und das resultierende System BOOST beschrieben (vgl. auch [Wren und Kwan, 1999]).

Volvo Traffic Planning Package Der in [Bokinge und Hasselström, 1980] vorgestellte Algorithmus zur Lösung von Umlaufplanungsproblemen mit Zeitfenstern für Servicefahrten kommt innerhalb des Systems „Volvo Traffic Planning Package“ zum Einsatz.

4.7 Veröffentlichungen mit Übersichten und Klassifizierungen

Mehrere Autoren haben in der Vergangenheit Übersichten oder Klassifizierungen von Umlaufplanungsthemen mit unterschiedlichen Schwerpunkten veröffentlicht.

In [Magnanti, 1981] und [Desrosiers et al., 1995] werden Umlaufplanungsprobleme als ein Anwendungsfall von allgemeinen Problemen im Bereich Transport betrachtet.

Wren gibt eine Zusammenfassung von Verfahren zur Lösung des Umlaufplanungsproblems ohne bzw. mit einem Depot (vgl. [Wren, 1981]). Der Autor geht vor allem auf den Stand der Technik hinsichtlich praktischer Erfahrungen ein.

Lenstra und Kan geben einen Überblick über NP-harte Problemstellungen aus dem Bereich Transport und diskutieren das Laufzeitverhalten verschiedener Lösungsansätze (vgl. [Lenstra und Kan, 1981]).

In [Bodin und Golden, 1981] wird eine Taxonomie präsentiert, anhand derer allgemeine Problemstellungen aus den Bereichen Routing, Scheduling sowie Routing and Scheduling kategorisiert werden können. Die Autoren betrachten das Umlaufplanungsproblem als ein Beispiel für ein Scheduling Problem. In [Bodin et al., 1983] ist eine ausführliche Klassifikation von Routing, Scheduling sowie

Routing and Scheduling Problemen gegeben. Es werden die zu damaliger Zeit bekannten Modelle und Lösungsmethoden vorgestellt.

In [Carraresi und Gallo, 1984] werden verschiedene Modelle für Umlauf- und Dienstplanungsprobleme aufgezeigt. Die Autoren stellen die Formulierungen vor und diskutieren Zusammenhänge von Zuweisungs- und Matchingmodellen.

Daduna und Paixao geben eine Übersicht über typische Anforderungen, Restriktionen und Zielsetzungen von Umlaufplanungsproblemen (vgl. [Daduna und Paixao, 1995]). Sie unterscheiden das MD-VSP und erweiterte Problemstellungen und schildern Erfahrungen mit dem Einsatz von Planungssystemen.

In [Mesquita und Paixao, 1999] wird eine Übersicht über die verschiedenen Formulierungen des MD-VSPs sowie die Qualität der jeweiligen LP-Relaxationen gezeigt.

Wren gibt eine ausführliche Übersicht über die Fortschritte in den Bereichen der Umlauf- und Dienstplanung (vgl. [Wren, 2003]). Er geht vor allem auf die englischen Systeme und Anforderungen ein und schildert Erfahrungsberichte und Einschätzungen von Anwendern.

In [Bunte und Kliewer, 2006] wird eine Literaturübersicht über die Modellierungsansätze für Umlaufplanungsprobleme präsentiert. Die Modelle werden anhand der unterschiedlichen formalen Problemstellungen klassifiziert und beschrieben.

5 Handlungsbedarf und Zielsetzung

Wie in Kapitel 4 gezeigt wurde, kommt der Erforschung und dem Einsatz von Optimierungsverfahren zur Unterstützung der Busumlaufplanung im ÖPNV bereits seit über 40 Jahren eine große Bedeutung zu. Gerade durch die Arbeiten der letzten Jahre konnten die Modelle und Methoden zur Lösung von VSP deutlich verbessert werden. Trotz der Fortschritte in diesen Bereichen gestaltet sich der Einsatz der Optimierungsverfahren in die Praxis häufig schwierig. Auf der Basis der identifizierten Anforderungen an Umlaufplanungsmethoden (s. Kapitel 2) und dem Stand der Technik (s. Kapitel 4) wird im Folgenden der Handlungsbedarf abgeleitet und die Zielsetzung dieser Arbeit aufgestellt.

Handlungsbedarf

Die Erforschung von Lösungsansätzen zur Optimierung von Umlaufplanungsproblemen richtet sich zu einem Großteil an der Problemstellung mit mehreren Depots (MD-VSP) aus, da dieses bereits eine schwere theoretische Komplexität hat und daher für die Validierung theoretisch motivierter Methoden ausreicht (s. Abschnitt 4.2). Aus diesem Grund wird in einer Vielzahl von Veröffentlichungen mathematischer Optimierungsmethoden nur diese Problemstellung berücksichtigt. Die in der Praxis relevanten Erweiterungen, wie zum Beispiel die Betrachtung von Fahrzeugtypgruppen, wird in diesen Ansätzen nicht betrachtet (s. Abschnitt 2.3). Zudem werden die meisten Lösungsansätze an Probleminstanzen validiert, die künstlich generiert wurden oder deren Ursprung unbekannt ist. Dabei bleibt unklar, ob qualitative Unterschiede zwischen realen und künstlichen Instanzen existieren (s. Abschnitt 4.2.5). Es besteht daher der Handlungsbedarf der **Ausrichtung der Forschung auf reale Problemstellungen**, um die Eignung von Optimierungsmethoden für den Einsatz in Verkehrsunternehmen sicherzustellen.

Das Umlaufplanungsproblem wird in der Literatur unter dem Anwendungsfall der operativen Fahrzeugeinsatzplanung oder unabhängig von dem Anwendungsfall als theoretische Problemstellung betrachtet. Im Rahmen des Planungsprozesses im ÖPNV existieren allerdings eine Reihe von weiteren Anwendungsfällen für den Einsatz von Optimierungsmethoden der Umlaufplanung. Diese Anwendungsfälle können sowohl andere Planungsphasen (z.B. die strategische Planung)

als auch erweiterte Zielsetzungen in der operativen Planung betreffen (z.B. die integrierte Dienstplanung). Abhängig von einem konkreten Anwendungsfall werden andere Anforderungen an Optimierungsmethoden gestellt, wie zum Beispiel die maximale Laufzeit oder die Berücksichtigung zusätzlicher Restriktionen (s. Abschnitt 2.2). Um die Planung im ÖPNV durch Optimierungsmethoden bestmöglich zu unterstützen, bedarf es daher neben der Berücksichtigung aller praxisrelevanten Problemerkweiterungen auch einer **Ausrichtung auf die unterschiedlichen Anwendungsfälle** der Umlaufplanung.

Exakte Optimierungsmethoden zur Lösung von Umlaufplanungsproblemen wurden unter der Zielsetzung entwickelt, mit möglichst kurzer Laufzeit eine optimale Lösung zu finden. Die Dauer einer Optimierung ist bei diesen Methoden allerdings nicht im voraus bekannt, so dass Planer eine nicht vorhersehbare Zeitspanne auf Ergebnisse warten oder die Optimierung ohne Ergebnis abbrechen müssen. Für einen effektiven Einsatz von Optimierungsmethoden in der Fahrzeugeinsatzplanung besteht daher der Bedarf dieses **Black-Box-Verhalten** zu vermeiden und den Planer durch die Ausgabe zusätzlicher Informationen zur Laufzeit, wie die erwartete numerische Komplexität oder vorläufige Lösungen, zu unterstützen.

Wie in einer Studie zum „ÖPNV-Markt der Zukunft“ gezeigt wird (vgl. [Lasch et al., 2005]), kann aufgrund der aktuellen und in der Zukunft erwarteten Entwicklungen im Bereich des ÖPNV mit einer zunehmenden Größe der zu planenden Fahrpläne gerechnet werden, da aufgrund von erwarteten Übernahmen und Fusionen von Verkehrsunternehmen oder gemeinsamer Planung kooperierender Unternehmen größere Liniennetze geplant werden können (s. Kapitel 1). Auf der anderen Seite können bereits heute einige große Probleminstanzen auch durch die fortschrittlichsten Lösungsansätze nicht optimal gelöst werden. Vor dem Hintergrund dieser Entwicklungen werden **effiziente Optimierungsmethoden** benötigt, die die Lösung realer Problemstellungen der Umlaufplanung ermöglichen.

Es besteht daher der Handlungsbedarf, einen Lösungsansatz zur Optimierung von Umlaufplanungsproblemen zu entwickeln, der die **Kombination aller aufgeführten Anforderungen** erfüllt.

Ziele

Zur Erfüllung des im vorangegangenen Abschnitt beschriebenen Handlungsbedarfs werden im Folgenden die Ziele dieser Arbeit aufgeführt:

1. Um eine Ausrichtung der Forschung auf die Lösung realer Problemstellungen zu ermöglichen, ist ein Ziel dieser Arbeit die **Analyse realer und künstlich generierter Probleminstanzen** hinsichtlich unterschiedlicher

Eigenschaften und die Klärung der Frage, ob qualitative Unterschiede in der Beschaffenheit der Instanzen bestehen. Auf Basis der gefundenen Eigenschaften soll eine Einschätzung gegeben werden, welche Modellierung für die Lösung realer Probleminstanzen geeignet ist.

2. Um den aufgeführten Handlungsbedarf für den Einsatz von Optimierungsmethoden zur Fahrzeugeinsatzplanung im ÖPNV erfüllen zu können, soll im Rahmen dieser Arbeit die **Konzeption und Entwicklung von Lösungsmethoden** durchgeführt werden, die innerhalb eines Lösungsframeworks zusammengefasst werden. Die Methoden sollen die unterschiedlichen **Kombinationen der praxisrelevanten Problemstellungen effizient lösen** können.
3. Unter Verwendung dieses Lösungsframeworks sollen **Strategien zur Berücksichtigung der unterschiedlichen Anwendungsfälle** innerhalb des Planungsprozesses des ÖPNV entwickelt werden. Diese Strategien sollen durch die Kombination und Parametrisierung geeigneter Methoden oder die Verwendung spezialisierter Methoden die Anforderungen aller Anwendungsfälle erfüllen.
4. Die Lösungsstrategien sollen die **Vermeidung eines Black-Box-Verhaltens** bei dem Einsatz der Optimierungsmethoden sicherstellen. Dazu soll eine Abschätzung der erwarteten Optimierungslaufzeit möglich sein und die Option bestehen, dass dem Planer bereits zur Laufzeit der Optimierung sowie bei vorzeitigem Abbruch möglichst mehrere gültige Lösungen bereitgestellt werden.

Umsetzung

Die Ausführungen zur Erreichung der genannten Ziele werden in drei Kapitel aufgeteilt. In Kapitel 6 werden reale und künstlich generierte Instanzen beschrieben und hinsichtlich ihrer Eigenschaften analysiert. Die Auswirkungen dieser Eigenschaften auf gängige Modellierungsarten werden gezeigt und Kennzahlen zur Messung der Eigenschaften entwickelt, so dass die Wahl einer Modellierung für die Lösung realer Probleme begründet werden kann. Zur Abschätzung der zu erwartenden Lösungszeit wird eine Kennzahl entwickelt und anhand der vorliegenden realen Instanzen validiert.

In den folgenden Kapiteln 7 und 8 wird die Konzeption neuer Methoden zur Lösung von Umlaufplanungsproblemen mit Ausrichtung auf die Anwendungsfälle vorgestellt. Die Methoden werden die spezifischen Eigenschaften realer Instanzen

ausnutzen, um eine möglichst effiziente Optimierung zu ermöglichen. Darüber hinaus wird durch eine flexible Möglichkeit der Parametrisierung seitens des Planners der Grad der Informationen zur Laufzeit bestimmbar sein. Für die Lösung möglicher LP- und MIP-Unterprobleme wird auf den Einsatz von Standardoptimierungssoftware zurückgegriffen, sofern kein alternatives Lösungsverfahren zu einer signifikant besseren Laufzeit führt. Dadurch können der Argumentation aus [Kliwer, 2005] folgend die fortschreitenden Entwicklungen der Standardoptimierungssoftware zur nachhaltigen Effizienzsteigerung der in dieser Arbeit entwickelten Methoden führen. Die Verfahren werden anhand ausführlicher numerischer Tests auf realen Probleminstanzen validiert und sofern möglich mit alternativen Verfahren verglichen.

6 Probleminstanzen und Eigenschaften

Nach der in Kapitel 4 vorgenommenen Vorstellung der Modellierungs- und Lösungsansätze für die unterschiedlichen Problemstellungen der Busumlaufplanung, widmet sich das folgende Kapitel der Diskussion unterschiedlicher Probleminstanzen und ihrer Eigenschaften.

Wie in Tabellen 4.2 und 4.3 gezeigt wurde, verwenden die in der Literatur vorgestellten Ansätze unterschiedliche Instanzen zur Validierung einer vorgestellten Methode. Es werden reale und künstlich generierte Instanzen unterschieden. Bei der Verwendung von realen Instanzen werden die Daten in der Regel im Rahmen einer Kooperation mit einem Verkehrsunternehmen oder einem Anbieter für Planungssoftware erhoben. Aufgrund von rechtlichen und/oder organisatorischen Restriktionen werden diese Instanzen in der Regel nicht veröffentlicht, so dass eine standardisierte Menge realer Probleminstanzen nicht verfügbar ist. Daneben existiert eine Reihe von Veröffentlichungen, die ihren Ansatz anhand von künstlich generierten Instanzen validieren. Das von Carpaneto et al. vorgestellte Verfahren zur Generierung künstlicher Instanzen für das MD-VSP wird in den meisten dieser Arbeiten verwendet und gilt somit als Quasi-Standard für die Betrachtung künstlich generierter VSP-Problemstellungen (vgl. [Carpaneto et al., 1989]).

Da in den Veröffentlichungen aus der Literatur die Eigenschaften der Instanzen und ihre Auswirkungen auf die Wahl eines geeigneten Modellierungs- und Lösungsansatzes nicht oder nur wenig thematisiert sind, wird in diesem Kapitel eine Analyse künstlicher und realer Instanzen durchgeführt. Der Ursprung der im Rahmen dieser Arbeit betrachteten Instanzen wird in Abschnitt 6.1 beschrieben. Aufbauend auf der Analyse der Instanzen werden in Abschnitt 6.2 Kennzahlen entwickelt, die neben offensichtlichen Eigenschaften der Instanzen (z.B. Fahrtenanzahl) auch weitere Struktureigenschaften berücksichtigen. Auf der Basis dieser Kennzahlen werden die Eigenschaften der unterschiedlichen Modellierungsansätze aufgezeigt und eine Abschätzung der numerischen Komplexität und der damit verbundenen Optimierungslaufzeit diskutiert.

6.1 Ursprung der Instanzen

Im Folgenden werden die Probleminstanzen vorgestellt, die im weiteren Verlauf dieser Arbeit betrachtet werden. In Abschnitt 6.1.1 werden zuerst die realen Instanzen diskutiert, die aus bestehenden Veröffentlichungen verfügbar sind oder im Rahmen einer Kooperation mit einem Hersteller für Planungssoftware erhoben wurden. Anschließend wird in Abschnitt 6.1.2 der Algorithmus zur Generierung von MD-VSP Instanzen vorgestellt. Beide Abschnitte enthalten zudem eine Definition einer für die weitere Diskussion relevanten Instanzenmenge.

6.1.1 Reale Instanzen

In einer Vielzahl von Veröffentlichungen zu Lösungsansätzen für VSP testen die jeweiligen Autoren die Verfahren auf Probleminstanzen, die aus realen Daten gewonnen wurden. Soweit dem Autor dieser Arbeit bekannt ist, sind diese Instanzen bis auf eine Ausnahme nicht öffentlich zugänglich oder für die Forschung verfügbar. Aus diesem Grunde wurden im Rahmen dieser Arbeit verschiedene Forschungseinrichtungen bzw. Forscher angefragt, um eine mögliche Verfügbarkeit der Daten zu erfragen. Aufgrund von rechtlichen und/oder organisatorischen Gründen (z.B. sensible Kosten- und Planungsinformationen) konnten die Instanzen allerdings nicht zur Verfügung gestellt bzw. für eine Veröffentlichung verwendet werden.

Eine Ausnahme bildet die Veröffentlichung von [Fischetti et al., 2001b], in der die Autoren neben künstlich generierten Instanzen auch reale Instanzen eines italienischen regionalen Verkehrsunternehmens für numerische Tests verwendet haben. Diese Instanzen wurden im Internet zur Verfügung gestellt (vgl. [Fischetti et al., 2001a]). Die Instanzen enthalten Angaben zu den Fahrten sowie die Kosten für eine Verbindung von zwei Servicefahrten. Die Daten beinhalten allerdings keine zeitlichen Angaben der Service- oder Verbindungsfahrten. Da diese Informationen eine Voraussetzung für die in den folgenden Abschnitten betrachteten Modelle darstellen, werden die Instanzen im Verlauf dieser Arbeit nicht weiter betrachtet.

Durch eine Kooperation mit einem Hersteller für operative Planungssoftware für den ÖPNV stehen zur Validierung der, in dieser Arbeit vorgestellten, Konzepte reale Planungsdaten von mehreren großen deutschen Städten zur Verfügung. Die Daten wurden zum Teil bereits in [Kliwer, 2005] verwendet, allerdings ist die Instanzenmenge um einige Problemstellungen erweitert worden. In Tabelle 6.1 sind die für diese Arbeit verfügbaren realen Umlaufplanungsinstanzen mit ihren charakteristischen Kennzahlen aufgeführt. Die Instanzen sind aufsteigend

Name	Alias ¹	$ S $	$ T $	#Lin.	#Dep.	#Typ.	#Gr.	$\varnothing D$	Kap.
real_1	monheim	34	424	1	1	1	1	1,0	–
real_2	–	44	426	9	1	1	1	1,0	–
real_3	–	207	867	58	2	3	3	2,0	–
real_4	–	88	1296	38	1	3	3	1,3	–
real_5	mun28	70	1808	36	13	1	1	13,0	–
real_6	hal	21	2047	19	2	3	3	3,2	✓
real_7	mun18	96	2394	47	12	1	1	12,0	–
real_8	mun14	44	2452	23	2	1	1	2,0	–
real_9	–	67	2633	17	3	1	1	3,0	–
real_10	mun15	73	2825	45	2	1	1	2,0	–
real_11	–	209	3067	64	1	5	5	1,0	–
real_12	–	351	9124	65	4	15	11	11,2	✓
real_13	mun2	140	10710	74	3	12	6	7,1	✓
real_14	–	156	10710	74	19	12	6	30,6	✓
real_15	mun1-kap	161	11062	75	19	12	12	6,5	✓

¹ Ggf. entsprechender Name aus [Kliwer, 2005]

Tabelle 6.1: Reale Probleminstanzen

nach der Anzahl der Servicefahrten sortiert. Die Spalte *Alias* gibt den in [Kliwer, 2005] verwendeten Namen der Instanz an, sofern sie in der Arbeit betrachtet wurde. Allerdings können sich die Kennzahlen und Komplexitäten der Instanzen abweichend, da für die in dieser Arbeit untersuchten Instanzen eine nicht vollständig gegebene Verbindungsmatrix nach der Dreiecksregel vervollständigt wurde. Dadurch ist die Anzahl der möglichen Verbindungen zwischen Endhaltestellen im Vergleich deutlich gestiegen (s. Tabelle 6.1). Die Spalten $|S|$ und $|T|$ geben die Anzahl der Endhaltestellen und Servicefahrten an. Spalte *#Lin.* nennt die Anzahl der im Fahrplan vorhandenen Linien. Spalten *#Dep.* und *#Typ.* geben die Anzahl der betrachteten Depotstandorte und unterschiedlichen Fahrzeugtypen an. In *#Gr.* wird die Anzahl der Fahrzeugtypgruppen genannt. Die Spalte $\varnothing D$ nennt die durchschnittliche Gruppengröße wie in Abschnitt 4.3 definiert. Spalte *Kap.* gibt an, ob eine Instanz Daten für Kapazitäten enthält (✓) oder beliebig viele Fahrzeuge in jedem Depot verfügbar sind (–).

Da die realen Instanzen zum Teil unterschiedliche Fahrzeugtypen und Fahrzeugtypgruppen enthalten und somit zu der Problemkategorie der MVTG-VSPs angehören, lassen sie sich nicht, oder nur schlecht, mit den im folgenden Abschnitt vorgestellten künstlich generierten Instanzen vergleichen, da diese der Problem-

Name	$ S $	$ T $	#Lin.	#Dep.	#Typ.	#Gr.	$\varnothing D$	Kap.
konv_1	34	424	1	1	1	1	1,0	–
konv_2	44	426	9	1	1	1	1,0	–
konv_3	207	867	58	2	1	1	2,0	–
konv_4	88	1296	38	1	1	1	1,0	–
konv_5	70	1808	36	13	1	1	13,0	–
konv_6	21	2047	19	2	1	1	2,0	✓
konv_7	96	2394	47	12	1	1	12,0	–
konv_8	44	2452	23	2	1	1	2,0	–
konv_9	67	2633	17	3	1	1	3,0	–
konv_10	73	2825	45	2	1	1	2,0	–
konv_11	209	3067	64	1	1	1	1,0	–
konv_12	351	9124	65	4	1	1	4,0	✓
konv_13	140	10710	74	3	1	1	3,0	✓
konv_14	156	10710	74	19	1	1	19,0	✓
konv_15	161	11062	75	19	1	1	19,0	✓

Tabelle 6.2: Konvertierte Probleminstanzen

klasse der MD-VSPs angehören. Um die beiden Instanzenmengen in der folgenden Diskussion miteinander vergleichen zu können, wurden die realen Instanzen zu MD-VSP konvertiert. Dazu wurden zum einen die Fahrzeugtypen und Fahrzeugtypgruppen zu jeweils einem Typ und einer Gruppe zusammengefasst. Zum anderen wurde eine Verwendung des Fahrzeugtyps aus jedem Depot ermöglicht. Im Vergleich zu den ursprünglichen MVTG-VSP Instanzen treten bezüglich der Komplexitätsveränderung daher zwei Effekte auf:

1. Durch die Zusammenfassung der Fahrzeugtypen reduziert sich die Zahl der Depot-Fahrzeugtyp-Kombinationen, wodurch die durchschnittliche Gruppengröße verringert und das Problem daher weniger komplex wird.
2. Durch das Entfernen der Fahrzeugtypgruppen kann jede Servicefahrt von jedem Depot bedient werden. Je nach der Struktur der Fahrzeugtypgruppen kann daher die durchschnittliche Gruppengröße und damit die kombinatorische Komplexität steigen.

Welcher Effekt bei der Konvertierung einer Instanz größer ist, ist daher von der Struktur der Fahrzeugtypen und Fahrzeugtypgruppen abhängig und von Instanz zu Instanz unterschiedlich. Zum Beispiel ist die durchschnittliche Gruppengröße

von *konv_14* kleiner als in der realen Instanz, aber von *konv_15* größer. Die resultierenden Instanzen sind analog zu Tabelle 6.1 in Tabelle 6.2 dargestellt.

Sowohl die realen wie auch die konvertierten Probleminstanzen sind mit einer Spezifikation im Internet veröffentlicht (vgl. [Kramkowski und Bunte, 2008]). Die Namen, Entfernungsangaben und Kosten wurden dazu aus Datenschutzgründen verfälscht, so dass die Angaben keinen realistischen Werten entsprechen, aber das Verhältnis erhalten bleibt.

6.1.2 Künstliche Instanzen

Die in der aktuellen Literatur verwendeten künstlich generierten Instanzen werden zu einem Großteil mit Hilfe des in [Carpaneto et al., 1989] präsentierten Verfahrens erstellt. Die Methode generiert Fahrpläne sowie weitere benötigte Informationen zur Definition von Umlaufplanungsproblemen mit mehreren Depots (MD-VSP) und ist in Abschnitt 4.2.5 beschrieben.

Für die folgenden Diskussionen wurde der beschriebene Algorithmus zur Generierung von künstlichen MD-VSP Instanzen verwendet, um für die in Abschnitt 6.1.1 beschriebenen realen Instanzen vergleichbare Probleme zu generieren. Für die Generierung wurden die gleichen Größen für Fahrten- und Depotanzahl verwendet. Um die Auswirkungen von zufälligen Extremwerten innerhalb der Generierung auszugleichen, wurden für jede Originalinstanz 10 künstliche Instanzen erzeugt. Alle Angaben zu den künstlich generierten Instanzen stellen daher Mittelwerte über diese dar. Die Instanzen zu einer Originalinstanz *real_x* werden im Folgenden *carp_x* bezeichnet.

6.2 Modelleigenschaften und Kennzahlen

Zur Modellierung des MD-VSP und MVTG-VSP existieren – wie in Kapitel 4 gezeigt wurde – unterschiedliche Ansätze. In der Literatur werden die einfachen Flussmodelle, die Mehrgüter Flussmodelle und die Set Partitioning Modelle unterschieden. Auch die Ansätze zur Lösung der erweiterten Problemstellungen (z.B. VSP-TW und VSP-RC) basieren zu einem Großteil auf diesen Modellen. Daher werden in diesem Abschnitt die Vor- und Nachteile der Modellierungen hinsichtlich der in Abschnitt 6.1 beschriebenen Probleminstanzen diskutiert. Auf der Basis dieser Diskussion soll die Auswahl der in den weiteren Kapiteln verwendeten Modellierung motiviert werden. Dazu sollen Charakteristika der unterschiedlichen Probleminstanzen analysiert werden, um neben der Auswahl der Modellierung auch relevante Kennzahlen zum Beispiel zur Abschätzung der Lösungskomplexität einer Instanz zu entwickeln.

6.2.1 Qualität der LP-Relaxation

Die effizientesten Ansätze zur Lösung von VSP haben gemeinsam, dass sie unabhängig vom Modell eine LP-basierte Lösungsmethode verwenden. Das heisst, dass durch initiale Lösung der LP-Relaxation eine untere Schranke sowie eine – bis auf die Ganzzahligkeit der Variablen – gültige Lösung bestimmt wird und anschließend mit einem Branch & Bound oder Branch & Cut Verfahren eine optimale Lösung gesucht wird. Daher ist die Qualität der LP-Relaxation – im Sinne einer möglichst hohen unteren Schranke – von entscheidender Bedeutung für einen LP-basierten Lösungsansatz. Neben der Berechnung einer initialen LP-Relaxation werden auch innerhalb des Branch & Bound Verfahrens LP-Relaxationen für das „Abschneiden“ von Teilen des Suchbaums berechnet (s. Abschnitt 3.1). Da, wie später gezeigt wird, die Lösung der LP-Relaxation von realen Busumlaufplanungsproblemen mit hohem Aufwand verbunden ist, kommt der Qualität der Relaxation eine besondere Bedeutung zu, um einen hohen Aufwand für die Berechnungen vieler LP-Relaxationen innerhalb des Branch & Bound zu vermeiden. Je größer die untere Schranke der LP-Relaxation ist, desto mehr Teilbäume können im B&B abgeschnitten werden.

[Mesquita und Paixao, 1999] betrachten die LP-Relaxationen der unterschiedlichen Flussmodelle und beweisen, dass der Wert der LP-Relaxation des einfachen Flussmodells mit Teiltour Eliminierung kleiner (oder gleich) der des Flussmodells mit Zuweisungsvariablen ist. Darüber hinaus zeigen die Autoren, dass diese wieder kleiner gleich der LP-Relaxation der Mehrgüter Flussmodelle ist. Zwar kann die Relaxation durch Hinzufügen von gültigen Ungleichungen verbessert werden, aber die große Anzahl von benötigten Restriktionen führt zu einer steigenden Komplexität. Diese ist mit einem zusätzlichen Zeitaufwand zur Lösung der LP-Relaxation verbunden, was dazu führt, dass Verfahren, die diesen Ansatz verfolgen, gerade für große Problemstellungen ineffizient werden. Dieses Verhalten haben bereits die Autoren des zuletzt veröffentlichten Lösungsansatzes auf der Basis eines einfachen Flussmodells angemerkt: *„As expected, when several depots are present the performance of the method deteriorates due to the very large number of cuts that need to be generated.“* (vgl. [Fischetti et al., 1999]). Aufgrund dieser Eigenschaft ist die Modellierung als einfaches Flussmodell nicht für die Lösung von großen realen Umlaufplanungsproblemen geeignet und wird für die weitere Betrachtung ausgeschlossen.

In [Ribeiro und Soumis, 1994] wird gezeigt, dass die LP-Relaxationen des Mehrgüter Flussmodells und des Set Partitioning Modells die gleiche untere Schranke bestimmen. Daher kann anhand der Qualität der LP-Relaxationen keine Aussage getroffen werden, welche Modellierungsart einen Vorteil für die Lösung

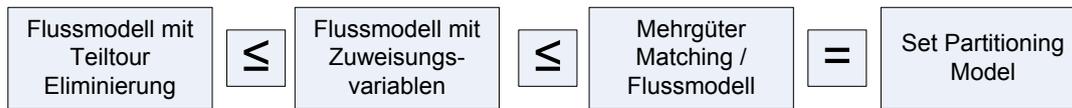


Abbildung 6.1: Qualität der Schranken durch unterschiedliche LP-Relaxationen

von Busumlaufplanungsproblemen darstellt. Abbildung 6.1 stellt eine Übersicht über die Qualitäten der unterschiedlichen LP-Relaxationen dar.

6.2.2 Kennzahl zur Modellauswahl

Im Folgenden sollen das Mehrgüter Flussmodell und das Set Partitioning Modell hinsichtlich ihres Verhaltens für reale Problemstellungen untersucht werden. Dazu wird das Verhalten der Modelle hinsichtlich der Anzahl der Modellvariablen sowie der Modelldegeneration für reale und künstliche Probleminstanzen betrachtet. Die Komplexität eines mathematischen Modells ist abhängig von der Dimension der Problematrix. Im Allgemeinen ist, bei gleicher Problemstruktur und gleicher Anzahl von Restriktionen, ein mathematisches Modell umso schwieriger zu lösen, je mehr Variablen es hat. Die Lösungsschwierigkeit zeigt sich in einer erhöhten Lösungszeit von LP-Verfahren. Daher wird im folgenden eine Abschätzung über die Anzahl von Variablen je nach Modellierungsart aufgestellt. In den folgenden Untersuchungen werden ohne Beschränkung der Allgemeinheit MD-VSP als Problemstellungen betrachtet.

Variablenanzahl

Die Anzahl der benötigten Variablen ist abhängig von der Struktur des Fahrplans (z.B. Länge und zeitliche Verteilung der Servicefahrten), da die Anzahl möglicher Umsetzfahrten von der Anzahl möglicher Anschlüsse abhängig ist. Um die Ausführungen zu verdeutlichen werden zwei Beispielfahrpläne betrachtet, die für ein Depot und fünf Servicefahrten eine jeweils deutliche Ausprägung der Struktur annehmen. Es soll angenommen werden, dass die Dauer einer Umsetzfahrt die halbe Fahrdauer beträgt. Der in Tabelle 6.3 dargestellte Fahrplan enthält Servicefahrten mit einer relativ langen Fahrdauer (je 2 Stunden), die zwischen zwei Haltestellen mit einer Taktung von einer Stunde geplant sind. Durch die Fahrdauer und der Dauer für eine Umsetzfahrt ist es nicht möglich zwei Servicefahrten innerhalb eines Umlaufs zu bedienen. Es gibt also keine mögliche Umsatzfahrt und für die Bedienung jeder Fahrt reicht die Verwendung eines Fahrzeugs aus. Der in Tabelle 6.4 dargestellte Fahrplan enthält die gleichen Abfahrts-

Fahrt	Abfahrtsort	Ankunftsart	Abfahrtszeit	Ankunftszeit
1	1	2	07:45	09:45
2	2	1	08:15	10:15
3	1	2	08:45	10:45
4	2	1	09:15	11:15
5	1	2	09:45	11:45

Tabelle 6.3: Beispiel 1 für einen Fahrplan

Fahrt	Abfahrtsort	Ankunftsart	Abfahrtszeit	Ankunftszeit
1	1	2	07:45	08:05
2	2	1	08:15	08:35
3	1	2	08:45	09:05
4	2	1	09:15	09:35
5	1	2	09:45	10:05

Tabelle 6.4: Beispiel 2 für einen Fahrplan

zeiten für die Fahrten wie in Beispiel 1. Jedoch beträgt die Fahrtendauer nur noch 20 Minuten, so dass mit einer Umsetzdauer von 10 Minuten ein Anschluss zwischen allen Fahrten möglich ist. Es sind daher 10 Umsetzungsmöglichkeiten (zwischen allen Fahrtenpaaren) gegeben und alle Servicefahrten des Fahrplans können durch ein Fahrzeug bedient werden. In Abbildung 6.2 sind die Unterschiede zwischen den Beispielen als Gantt-Diagramm und Treppenfunktion der gleichzeitig stattfindenden Fahrten graphisch veranschaulicht.

Die Anzahl der möglichen Umsetzfahrten ist also nicht anhand der Information über die Anzahl von Servicefahrten und der Endhaltestellen im Fahrplan bestimmbar. Die Anzahl der in einer mathematischen Modellierung benötigten Variablen N kann daher auch nicht exakt bestimmt werden. Deshalb sollen im Folgenden Abschätzungen für den Fall der mindestens benötigten (*best case*) und maximal benötigten (*worst case*) Variablen gegeben werden.

Im SPP ist im Falle des *best case*, bei dem keine Verbindungsmöglichkeiten zwischen den Fahrten existieren, für jede Fahrt und jedes Depot ein Umlauf möglich. Unter der Annahme, dass keine Umläufe betrachtet werden, die von anderen dominiert werden, gilt: $|N|_{best} = |D| \cdot |T|$. Bei der Abschätzung des *worst case* müssen alle Umläufe betrachtet werden, die bei der Möglichkeit von allen Verbindungen zwischen Fahrten möglich sind. Für die Bildung eines Umlaufs mit i abgedeckten Fahrten gibt es $\binom{|T|}{i}$ Möglichkeiten. Da im Set Partitioning Modell Gleich-

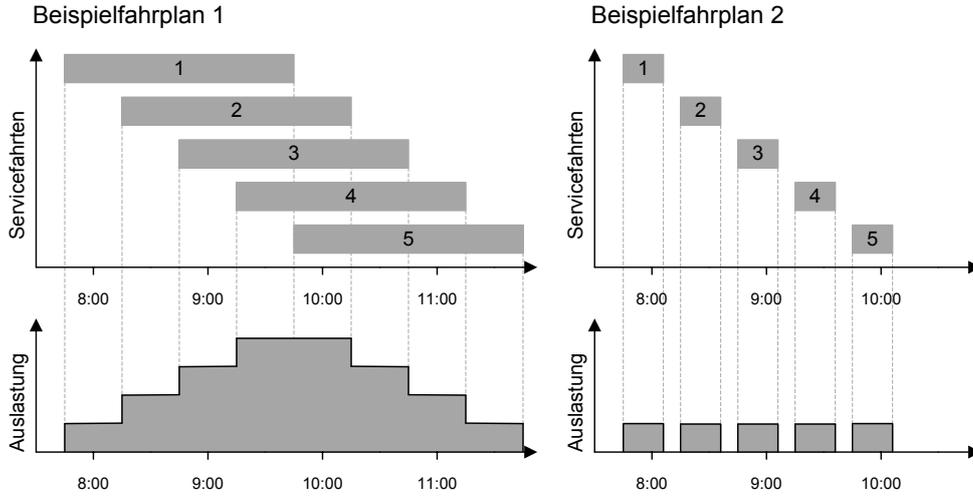


Abbildung 6.2: Vergleich der Auslastung beispielhafter Fahrpläne

heitsbedingungen für die Abdeckung der Fahrten verwendet werden, müssen alle gültigen Umläufe gebildet werden. Diese sind für jedes Depot zu bilden. Daher beträgt die Anzahl der Variablen im *worst case*: $|N|_{worst} = |D| \cdot \left(\sum_{i=1}^{|T|} \binom{|T|}{i} \right)$.

In der mathematischen Formulierung des (nicht aggregierten) Mehrgüter Flussmodells entspricht jede Kante im verwendeten Netzwerkmodell einer Variable. Im *best case* (ohne Anschlussmöglichkeiten) existieren in jeder Netzwerkschicht eine Kante für jede Fahrt, zwei Kanten für mögliche Depotfahrten und eine Rückkante ($|N|_{best} = |D| \cdot (3 \cdot |T| + 1)$). Im Falle von Anschlussmöglichkeiten zwischen allen Fahrten, werden zusätzlich zu den oben angesprochenen Kanten nach jeder Fahrt Verbindungskanten zu allen nachfolgenden Fahrten verwendet. Diese Kanten werden in allen Schichten des Netzwerks eingefügt, so dass die Gesamtanzahl der Kanten $|N|_{worst} = |D| \cdot \left(3 \cdot |T| + 1 + \sum_{i=1}^{|T|} (|T| - i) \right)$ beträgt. Zusammenfassend beträgt die Abschätzung der Variablenanzahl der Modelle:

$$\text{Set Partitioning Modell: } |N|_{best} = |D| \cdot |T| \quad (6.1)$$

$$|N|_{worst} = |D| \cdot \left(\sum_{i=1}^{|T|} \binom{|T|}{i} \right) \quad (6.2)$$

$$\text{Mehrgüter Flussmodell: } |N|_{best} = |D| \cdot (3 \cdot |T| + 1) \quad (6.3)$$

$$|N|_{worst} = |D| \cdot \left(3 \cdot |T| + 1 + \sum_{i=1}^{|T|} (|T| - i) \right) \quad (6.4)$$

$ T = t$	Mehrgüter		SPP	
	$ N $ ($\Phi = 1$)	$ N $ ($\Phi = t^{-1}$)	$ N $ ($\Phi = 1$)	$ N $ ($\Phi = t^{-1}$)
1	4	4	1	1
2	7	8	2	3
3	10	13	3	7
4	13	19	4	15
5	16	26	5	31
6	19	34	6	63
7	22	43	7	127
8	25	53	8	255
9	28	64	9	511
10	31	76	10	1.023
15	46	151	15	32.767
20	61	251	20	1.048.575

Tabelle 6.5: Analyse der Anzahl der Modellvariablen für ein Depot

Für die Einschätzung der Modellgröße wird die Kennzahl der *Fahrplandichte* Φ definiert, die eine Kennzahl für den Grad der strukturellen Komplexitätsausprägung darstellen soll. Wie in den oben aufgestellten Beispielen erkennbar ist, tritt der *best case* in dem Fall ein, dass alle Fahrten von einem Fahrzeug bedient werden können. Entsprechend ist der *worst case* bei einer benötigten Fahrzeuganzahl von $|T|$. Sei $u^*(F)$ die optimale Anzahl der benötigten Fahrzeuge zur Bedienung des Fahrplans F . Sei die Fahrplandichte Φ definiert als die Anzahl der benötigten Fahrzeuge im Verhältnis zu der Anzahl an Servicefahrten: $\Phi_F = \frac{u^*(F)}{|T|} \in [\frac{1}{|T|}; 1]$. Für die oben vorgestellten Beispiele gilt daher: $\Phi_{\text{Beispiel1}} = 1$ und $\Phi_{\text{Beispiel2}} = 0, 2$. Bei der Betrachtung der Variablenanzahl gilt für beide Modellierungsarten, dass die Anzahl der Variablen durch eine größere Anzahl von Verbindungsmöglichkeiten bei gleicher Fahrtenanzahl ansteigt. Jedoch ist die Zunahmerate im SPP erheblich größer im Vergleich zum Mehrgüter Flussmodell. Die Fahrplandichte gibt daher eine Tendenz an, wie viele Verbindungen innerhalb des Fahrplans möglich sind. Für $\Phi = 1$ ist die Variablenanzahl gleich der des *best case* und für kleinere Werte nähert sich die Anzahl der Variablen dem *worst case* an. Da die Anzahl der Depots in alle Berechnungen als gleicher Faktor eingeht, ist die Anzahl der Variablen beider Modelle in Tabelle 6.5 für einen Fahrplan mit bis zu 20 Servicefahrten und einem Depot dargestellt. Die Spalten enthalten jeweils die Variablenanzahl im *best case* ($\Phi = 1$) und *worst case* ($\Phi = t^{-1}$).

Durch den erheblich stärkeren Anstieg der Variablenanzahl im SPP im Vergleich zum Mehrgüter Flussmodell kann an dieser Stelle festgehalten werden, dass für tendenziell niedrige Werte der Fahrplandichte die Mehrgüter Flussmodellierung besser zur Lösung von VSP geeignet ist.

Degeneration

Die zuvor definierte Fahrplandichte hat neben der Auswirkung auf die Anzahl von Modellvariablen auch Einfluss auf den Grad der Degeneration der LP-Relaxationen. Eine Basislösung eines LP mit m Restriktionen besteht aus m Variablen. Wenn eine Basis bzw. Basislösung Variablen enthält, deren Wert gleich null ist, so ist die Lösung nicht eindeutig, da diese Basisvariablen durch beliebige andere Variablen ersetzt werden können, ohne dass sich der Zielfunktionswert ändert. Je mehr Variablen in einer Basislösung gleich null sind, desto höher ist der Grad dieser Degeneration. Je stärker ein Modell degeneriert ist, desto schwieriger ist es für LP-Methoden wie der primalen oder dualen Simplexmethode dieses Modell zu lösen, da ggf. viele Iterationen durchgeführt werden müssen bis sich der Zielfunktionswert verändert (vgl. [Chvatal, 1983]). Bei der Betrachtung des Degenerationsgrades der Set Partitioning und Mehrgüter Flussmodelle ist festzustellen, dass beide Formulierungen indirekt von der Fahrplandichte abhängig sind. Der Grad der Degeneration entspricht jeweils der Anzahl m der Restriktionen abzüglich der Anzahl der Variablen, die ungleich null sind.

Das SPP enthält eine Überdeckungsrestriktion für jede Servicefahrt sowie eine Kapazitätsrestriktion für jedes Depot ($m = |T| + |D|$). In einer Lösung ist für jeden gewählten Umlauf eine Variable ungleich null. Somit ergibt sich der Grad der Degeneration durch $|T| + |D| - v^*$ Basisvariablen, die ungleich null sind. Je größer die Fahrplandichte ist, desto mehr Umläufe befinden sich in einer optimalen Lösung und desto geringer ist der Grad der Degeneration.

Das Mehrgüter Flussmodell enthält Restriktionen zur Flusserhaltung für jeweils zwei Knoten pro Fahrt sowie zwei Depotknoten. Da jede Netzwerkschicht diese Knoten enthält ergibt sich für die Anzahl der Restriktionen $m = (2 \cdot |T| + 3) \cdot |D|$. In einer Lösung werden pro bedienter Fahrt zwei Variablen aktiviert (die Fahrtkante und eine hinführende Kante). Zusätzlich wird jeder Umlauf mit einer Einrückfahrt abgeschlossen und für alle Depots die Rückkante aktiviert¹. Daher beträgt die Anzahl der in einer Lösung vorhandenen Variablen mit einem Wert größer null: $2 \cdot |T| + v^* + |D|$. Da die Anzahl der Restriktionen mit dem Faktor $|D|$ wächst, die Anzahl der Basisvariablen ungleich null aber nur wenig steigt, ist der Grad der Degeneration von Mehrgüter Flussmodellen sowohl von der Anzahl

¹falls nicht alle Depots verwendet werden, entsprechend weniger

der verwendeten Fahrzeuge, aber vor allem von der Anzahl der Depots abhängig.

Zusammenfassend kann also festgestellt werden, dass die Fahrplandichte im Falle des Mehrgüter Flussmodells keinen primären Einfluss auf den Grad der Degeneration hat. Im SPP hat sie jedoch einen starken Einfluss auf den Grad der Degeneration. Je kleiner die Fahrplandichte, desto stärker ist die Degeneration im SPP. [Oukil et al., 2007] zeigen, dass ein größerer Grad der Degeneration im Set Partitioning Modell zu einer starken Erhöhung der Lösungszeit führt.

Des Weiteren wirkt sich der durch den Wert der Fahrplandichte beschriebene Strukturunterschied auf die Dichte der Koeffizientenmatrix aus. Da ein Umlauf aus einer größeren Anzahl von Servicefahrten besteht, enthält eine Spalte des SPP mehr Koeffizienten, die ungleich null sind. Diese höhere Dichte der Koeffizientenmatrix führt zu einer weiteren Reduzierung der Performanz von Set Partitioning Methoden: „... *set-partitioning solution approaches ... run into trouble when the number of nonzero entries of each set partitioning column ... increases.*“ (vgl. [Fischetti et al., 1999]).

Modellwahl für reale Probleminstanzen

Wie in Kapitel 4 dargestellt wurde, verwenden alle Ansätze, die auf der Set Partitioning Modellierung basieren, die künstlich generierten Probleminstanzen nach [Carpaneto et al., 1989] zur Validierung der Ergebnisse. Auf der anderen Seite kommt das Mehrgüter Flussmodell zu einem Großteil bei der Verwendung von realen Problemstellungen zur Anwendung (s. Tabellen 4.2 und 4.3). Die zuvor aufgestellte Fahrplandichte kann durch die Analyse der realen und künstlichen Probleminstanzen eine Begründung für diese Beobachtung geben.

Abbildung 6.3 zeigt die Ausprägung der Fahrplandichte Φ für die realen, konvertierten und künstlich generierten Probleminstanzen, die in Abschnitt 6.1 vorgestellt wurden. Die Sortierung folgt der Auflistung in den angegebenen Tabellen. Zur Bestimmung der optimalen Fahrzeuganzahl der Instanzen wurde die im nächsten Kapitel vorgestellte Methode verwendet. Für die künstlich generierten Instanzen 11-15 konnten keine optimalen Fahrzeuganzahlen, und damit kein Wert für Φ , bestimmt werden (s. Tabelle 6.6). Der Mittelwert für die realen sowie künstlich generierten Instanzen wird jeweils durch eine horizontale gestrichelte Linie dargestellt. Im Mittel beträgt die Fahrplandichte für die realen Problemstellungen 0,047, für die konvertierten 0,046 und für die künstlich generierten 0,208. Das heißt, dass in den realen Problemstellungen durchschnittlich ca. 19 Fahrten (Φ^{-1}) von einem Umlauf bedient werden können. Dagegen sind die künstlich generierten Fahrpläne so strukturiert, dass im Mittel lediglich weniger als fünf Fahrten pro Umlauf bedient werden können.

Tabelle 6.6 sind die Ergebnisse für die konvertierten und künstlichen Instanzen gegeben.

Die Anzahl der Kanten (= Modellvariablen) sind für die verbindungs-basierte ($|A_{VNB}|$) und aggregierte TSN-basierte ($|A_{TSN}|$) Mehrgüter Flussmodellierung angegeben. Die benötigte Zeit zur Lösung der LP-Relaxation des TSN-basierten Modells ist in Spalte t_{LP} in Sekunden angegeben. Im Fall der Instanzen *carp_14* und *carp_15* konnte das Netzwerkmodell trotz einer speicher-effizienten Implementierung nicht in einem Computer mit 8 Gigabyte Arbeitsspeicher aufgebaut werden, da die Größe der Netzwerke basierend auf dem Verhalten der übrigen Instanzen um 100.000.000 Kanten enthalten würden und zur Speicherung einer einzelnen Information für jede Kante (double-Wert) bereits 762 Megabyte an Speicher benötigt werden.

Die Ergebnisse zeigen, dass im VBN-basierten Mehrgüter Flussmodell auch bei deutlichen Unterschieden in der Fahrplandichte nur wenig mehr Kanten benötigt werden. Im Falle einer SPP-Modellierung würde die Anzahl der Variablen dagegen für die konvertierten Instanzen um ein Vielfaches ansteigen (s. Tabelle 6.5).

Durch Aufbau des Mehrgüter Flussmodells anhand eines aggregierten Time-Space-Netzwerks werden erheblich weniger Kanten benötigt. Die Menge ist abhängig von der Anzahl der Servicefahrten sowie der Endhaltestellen (vgl. [Kliwer, 2005]). Die konvertierten (und realen) Instanzen enthalten im Vergleich zu den künstlich generierten Instanzen weniger Endhaltestellen. Die Anzahl beträgt für die konvertierten Instanzen zwischen 1% und 24% der Fahrtenmenge (s. Tabelle 6.2) und für die künstlich generierten laut Definition zwischen 33% und 50% (s. Algorithmus 4.1). Daher kann die Modellgröße für die konvertierten Instanzen erheblich stärker verringert werden. Die zur Lösung des LP benötigte Laufzeit verändert sich für ein Instanzenpaar überproportional zur Kantenanzahl.

Die Ergebnisse zeigen bereits für die Mehrgüter Flussmodellierung, dass sich die strukturellen Unterschiede der realen und künstlich generierten Instanzen in einer deutlich größeren Variablenanzahl und längerer Laufzeit niederschlagen. Diese Unterschiede würden sich, wie im vorigen Abschnitt gezeigt wurde, in einer erheblich stärkeren Form auf die Komplexität des Set Partitioning Modells auswirken.

Da die Set Partitioning Modelle in der Literatur bereits auf den künstlich generierten Instanzen keine klar dominierende Performanz aufweisen und die realen Problemstellungen wie oben gezeigt eine deutlich höhere Komplexität bezüglich der Kantenanzahl vermuten lassen, ist das Mehrgüter Flussmodell aus Sicht der Effizienz für die Lösung realer Problemstellungen vorzuziehen. Ein Vorteil der Set Partitioning Modellierung liegt allerdings in der Möglichkeit durch die umlauf-basierte Variablendefinition Routenrestriktionen im Modell zu berücksichtigen.

Instanz	Φ	$ A_{VBN} $	$ A_{TSN} $	t_{LP}
konv_1	0,07	67.875	4.482	0
carp_1	0,23	70.346	15.213	1
konv_2	0,08	81.572	2.915	0
carp_2	0,24	70.535	14.270	1
konv_3	0,08	720.373	44.068	5
carp_3	0,22	583.626	117.106	16
konv_4	0,04	782.753	25.595	3
carp_4	0,21	648.846	121.931	11
konv_5	0,03	19.360.518	544.449	- ¹
carp_5	0,21	16.727.293	3.296.395	- ¹
konv_6	0,06	2.758.279	37.978	8
carp_6	0,20	3.282.519	633.831	239
konv_7	0,03	31.058.285	781.180	- ¹
carp_7	0,21	26.984.550	5.305.671	- ¹
konv_8	0,04	5.226.481	101.287	26
carp_8	0,21	4.687.068	913.999	389
konv_9	0,05	8.666.287	133.500	108
carp_9	0,20	8.108.261	1.544.143	1.209
konv_10	0,03	7.133.118	134.122	34
carp_10	0,20	5.160.404	1.083.042	541
konv_11	0,05	4.321.394	85.968	7
carp_11	0,20	3.654.884	678.143	145
konv_12	0,04	153.739.259	3.039.350	- ¹
carp_12	-	127.638.879	21.068.680	-
konv_13	0,03	162.583.722	1.431.591	- ¹
carp_13	-	132.290.980	23.442.512	-
konv_14	0,03	973.765.478	10.260.621	- ¹
carp_14	-	Out of Mem	Out of Mem	-
konv_15	0,03	1.046.908.432	10.818.864	- ¹
carp_15	-	Out of Mem	Out of Mem	-

¹ Optimale Fahrzeuganzahl mit Heuristik aus Kapitel 7 berechnet

Tabelle 6.6: Verhaltensunterschiede von künstlichen und realen Instanzen

Diese Restriktionen sind im Mehrgüter Flussmodell zwar nicht exakt abbildbar, können aber – wie häufig in der Praxis zu beobachten ist – als weiche Restriktionen berücksichtigt werden (vgl. [Kliwer et al., 2004]). Daher wird für die weitere Diskussion sowie für die Modellierungsauswahl der in den folgenden Kapiteln konzipierten Methoden die Mehrgüter Flussmodellierung verwendet.

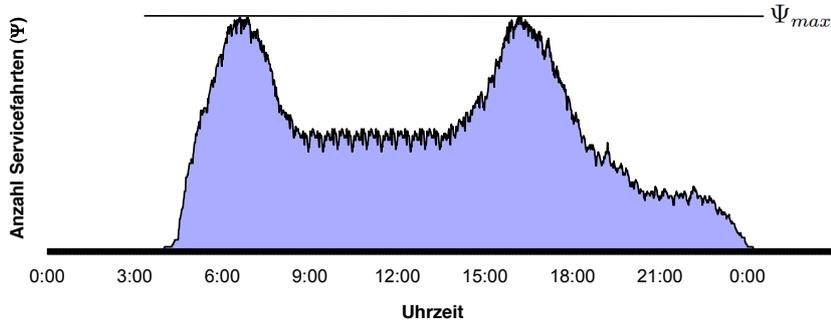
Allgemein kann anhand der vorgestellten Kennzahl der Fahrplandichte Φ eine Abwägung getroffen werden, ob für Instanzen tendenziell ein Set Partitioning Modell oder Mehrgüter Flussmodell Modell vielversprechend ist.

Substitution von Φ

Zur Diskussion der unterschiedlichen Modelleigenschaften für reale und künstlich generierte Umlaufplanungsprobleme wurde im vorangegangenen Abschnitt die Kennzahl der Fahrplandichte Φ verwendet. Diese wird aus der Anzahl von Servicefahrten im Fahrplan $|T|$ sowie der optimalen Fahrzeuganzahl u^* gebildet. Für die Einschätzung einer bisher nicht gelösten oder ggf. nicht lösbaren Probleminstanz kann die Fahrplandichte daher nicht berechnet werden. Aus diesem Grund soll im folgenden Abschnitt diskutiert werden, wie die optimale Anzahl von Fahrzeugen u^* anhand des Fahrplans abgeschätzt und zur Berechnung von Φ substituiert werden kann.

Zur Abschätzung der minimal benötigten Fahrzeuganzahl kann eine untere Schranke anhand des Auslastungsgraphen der Servicefahrten bestimmt werden. Der Auslastungsgraph Ψ zeigt zu diskreten Zeitintervallen die Anzahl der zeitgleich stattfindenden Servicefahrten an. Die Zeitintervalle werden in der Größe der Planungsgenauigkeit gewählt. Dies ist in der Regel eine minutengenaue Planung. Der Auslastungsgraph kann rekursiv definiert werden, indem zu Beginn des Planungszeitraums keine Fahrten stattfinden: $\Psi_0 = 0$. Die Auslastung jedes folgenden Zeitintervalls kann durch Aktualisierung des vorigen Zeitintervalls berechnet werden, indem die zu diesem Zeitpunkt beginnenden Servicefahrten addiert und die endenden Servicefahrten subtrahiert werden. Zum Zeitintervall t beträgt die Auslastung daher: $\Psi_t = \Psi_{t-1} + |\{i \in T \mid e_i = t\}| - |\{i \in T \mid f_i = t\}|$. Eine untere Schranke für die minimal benötigte Fahrzeuganzahl stellt das Maximum des Auslastungsgraphen Ψ_{max} dar, da ein Fahrzeug keine zwei zeitgleichen Fahrten bedienen kann. Abbildung 6.4 zeigt ein Beispiel für einen Auslastungsgraphen und das Maximum des Graphen.

Diese untere Schranke entspricht für den Fall, dass keine Umsetzfahrten möglich sind oder die Dauer aller möglichen Umsetzfahrten null ist, auch der optimalen Fahrzeuganzahl. Mehr Fahrzeuge als das Maximum des Auslastungsgraphen an gibt werden benötigt, wenn zum Beispiel durch notwendige Umsetzfahrten nicht


 Abbildung 6.4: Auslastungsgraph eines Fahrplans mit Auslastungsspitze Ψ_{max}

alle Fahrzeuge zum Zeitpunkt der maximalen Auslastung stattfindenden Servicefahrten verfügbar sind. Für die Abschätzung einer minimal benötigten Fahrzeuganzahl auf der Basis des Auslastungsgraphen spielt die Dauer für Verbindungsfahrten zwischen Haltestellen daher eine wichtige Rolle. Um die erwartete Zeit für eine Umsetzfahrt nach der Servicefahrt zu kalkulieren, kann die durchschnittliche Verbindungszeit zwischen allen Endhaltestellen betrachtet werden: $\varnothing d = \sum_{i \in S} \sum_{j \in S} \frac{1}{|S|^2} d_{ij}$. Da in einem Fahrplan in der Regel nicht alle Endhaltestellen gleich häufig bedient werden, kann dieser Durchschnitt zum Beispiel durch eine wenig besuchte Endhaltestelle, die weit außerhalb liegt, verzerrt werden. Statt einer gleichverteilten Gewichtung sollten die paarweisen Verbindungen daher mit der Häufigkeit einer Abfahrt oder Ankunft gewichtet werden. Die resultierende gewichtete durchschnittliche Verbindungszeit zwischen Haltestellen $\widetilde{\varnothing d}$ kann durch Verwendung der mittleren Gewichte für Haltestellenpaare definiert werden:

$$\widetilde{\varnothing d} = \sum_{i \in S} \sum_{j \in S} \frac{\gamma_i \gamma_j}{2|S|^2} d_{ij} \quad \text{mit} \quad \gamma_i = \frac{|\{t \in T | g_t = i\}| + |\{t \in T | h_t = i\}|}{2|T|}, \quad \forall i \in S$$

Als alternative Abschätzung der minimalen Fahrzeuganzahl kann daher ein angepasster Auslastungsgraph $\widetilde{\Psi}$ definiert werden, in dem jede Servicefahrt um die gewichtete durchschnittliche Verbindungszeit zwischen Haltestellen verlängert wird. Die Aktualisierungsfunktion der diskreten Zeitintervalle lautet:

$$\widetilde{\Psi}_t = \widetilde{\Psi}_{t-1} + |\{i \in T | e_i = t\}| - |\{i \in T | (f_i + \widetilde{\varnothing d}) = t\}|$$

Reale Fahrpläne basieren häufig auf getakteten Linienfahrten, die über den Tag zwischen zwei Endhaltestellen pendeln. Diese Fahrten sind häufig bereits so geplant, dass eine Bedienung beider Fahrten durch das gleiche Fahrzeug sinnvoll

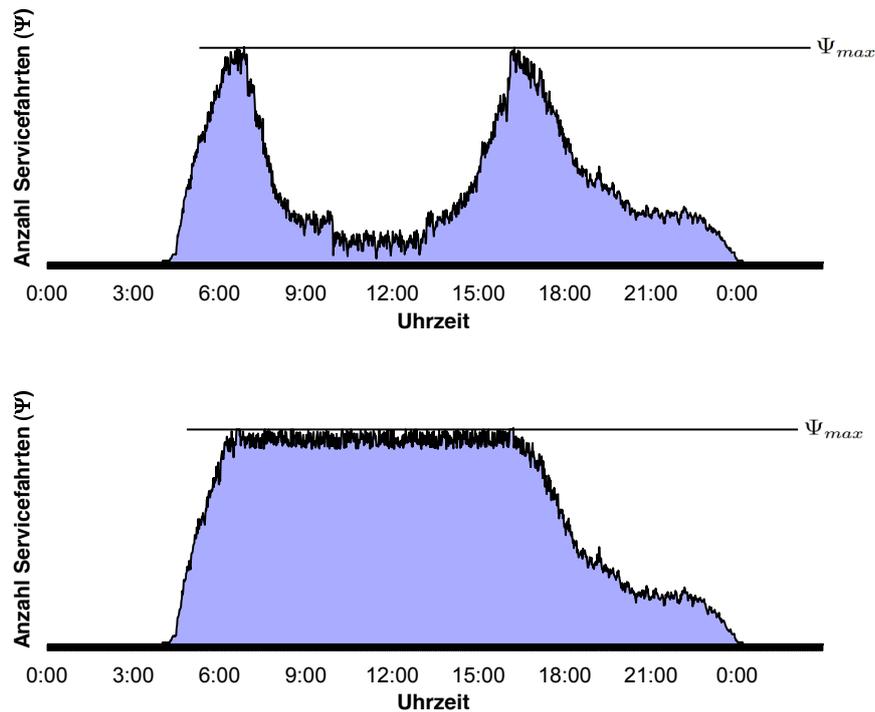


Abbildung 6.5: Unterschiede in Fahrplanstrukturen bei gleicher maximaler Auslastung

und kostenoptimal ist. Die Annahme der mittleren gewichteten Verbindungsdauer kann daher auf realen Fahrplänen zu einer Überschätzung der optimalen Fahrzeuganzahl führen.

Eine alternative Anpassung von Ψ_{max} kann zu einer besseren Abschätzung der optimalen Fahrzeuganzahl führen. Reale Fahrpläne zeigen vielfach erhöhte Ausprägungen in den Stoßzeiten, zu denen der Berufs-, Schul- und Pendlerverkehr stattfindet und vermehrt Unterstützungsfahrten erfolgen. Die Stärke der Ausprägung ist je nach Stadt und Tag unterschiedlich, aber kann einen wesentlichen Einfluss auf die Anzahl der Fahrzeuge haben. Dieser strukturelle Unterschied kann bei gleicher Höhe der maximalen Ausprägung von $\tilde{\Psi}$ zu großen Unterschieden in der Anzahl der benötigten Fahrzeuganzahl führen. Abbildung 6.5 veranschaulicht dieses Verhalten an zwei Beispielen.

Je mehr Servicefahrten zu einem Zeitpunkt kurz vor oder nach der maximalen Ausprägung von Ψ stattfinden, desto größer ist die Wahrscheinlichkeit, dass ein Fahrzeug nicht für die Bedienung einer Fahrt im Maximum zur Verfügung steht, da es eine Umsetzfahrt tätigt. Fahrten, die zu einem Zeitpunkt stattfinden, an dem nur wenige Servicefahrten zeitgleich bedient werden müssen, bergen daher ein geringeres Risiko. Um dieser Struktur Rechnung zu tragen, wird zur

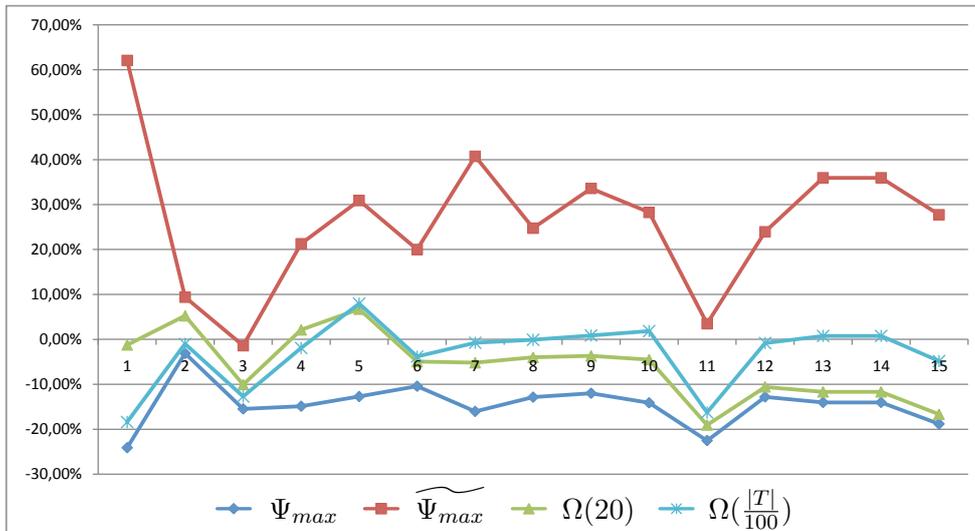


Abbildung 6.6: Vergleich von Abschätzungen der optimalen Fahrzeuganzahl

Abschätzung der benötigten Fahrzeuge ein weiterer Wert einbezogen. Damit das unterschiedliche Risiko der Servicefahrten – je nach Ausprägung des Auslastungsgraphen – beachtet wird, wird die Ausprägung eines Zeitintervalls im Verhältnis zum Maximum quadratisch einbezogen. Über alle Zeitintervalle des Planungszeitraums wird der relative Anteil der aktuell stattfindenden Fahrten zur maximalen Auslastung berechnet und der Mittelwert aller quadratischen Anteile berechnet. Dieser wird, skaliert mit dem Faktor ω , als zusätzlicher Summand mit in die Schätzung der optimalen Fahrzeuganzahl einbezogen. Resultierend soll $\Omega(\omega)$ eine Abschätzung für u^* sein, die die Ausprägung der Fahrten und die relative Fahrtenverteilung mit in die Berechnung einbezieht:

$$\Omega(\omega) = \Psi_{max} + \frac{\omega}{|\Psi|} \sum_{t \in \Psi} \left(\frac{t}{\Psi_{max}} \right)^2$$

Die vorgeschlagenen Abschätzung für die optimale Fahrzeuganzahl können als Abschätzung der Fahrplandichte verwendet werden, indem sie durch die Anzahl der Servicefahrten geteilt werden. Abbildung 6.6 zeigt die Abweichungen der unterschiedlichen Schätzwerte von der optimalen Fahrzeuganzahl u^* . Die Graphik zeigt das erwartete Verhalten, dass Ψ_{max} eine Unterschätzung und $\widetilde{\Psi}_{max}$ eine Überschätzung der optimalen Fahrzeuganzahl und damit für Φ ist. Mit Hilfe der Funktion $\Omega(\omega)$ stellt dagegen eine gute Möglichkeit zur Abschätzung dar. Durch eine Analyse der Parametereinstellung für ω hat sich für die in dieser Arbeit betrachteten Instanzen ein stabiler Wert von 20 ergeben. Eine Beobachtung war, dass bei der Parametereinstellung $\omega = \frac{|T|}{100}$ eine sehr gute Abschätzung möglich ist.

Ein möglicher Grund könnte die tendenzielle Unterschätzung der Fahrzeuganzahl bei vielen Servicefahrten sein, da durch eine größere Menge von Freiheitsgraden bessere Verbindungen möglich sind.

6.2.3 Eigenschaften der Mehrgüter Flussmodelle

Im vorhergehenden Abschnitt wurde gezeigt, dass die Mehrgüter Flussmodellierung für die im Rahmen dieser Arbeit entwickelten Lösungsverfahren gut geeignet ist, da sie für die Lösung von realen VSP im Vergleich zu alternativen Modellen eine effiziente Modellierung darstellt. Innerhalb dieser Modellierung können die verbindungs-basierte und aggregierte Time-Space-Netzwerk-basierte Modellierung unterschieden werden. In diesem Abschnitt werden die Unterschiede und Eigenschaften der Modelle diskutiert, um die Ausrichtung der im Rahmen dieser Arbeit entwickelten Methoden zu motivieren.

Qualitative Unterschiede

Für die folgende Diskussion werden die verbindungs-basierte Modellierung (VBN), in der jede mögliche Verbindung zwischen zwei Fahrten explizit als eine Kante (und damit eine Variable in der mathematischen Formulierung) modelliert wird, und die in [Kliwer et al., 2006b] vorgestellte aggregierte Time-Space-Netzwerk-basierte Modellierung (TSN) unterschieden. Letztere erreicht durch eine aggregierte Betrachtung von Warte- und Umsetzflüssen innerhalb des Netzwerks für reale Probleminstanzen eine Reduzierung der benötigten Kanten zwischen 97% und 99% ohne dass optimale Lösungen des VSP verloren gehen (vgl. [Kliwer, 2005]). Dieser Effekt ist auch in Tabelle 6.6 erkennbar. Eine detailliertere Beschreibung der Modelle ist in Kapitel 4 gegeben.

Neben dem unterschiedlichen Aufbau des mathematischen Modells sollen vorab einige qualitative Unterschiede zwischen VBN und TSN beschrieben werden. Die Flusslösungen der beiden Modelle unterscheiden sich dahingehend, dass im VBN obere Flussschranken von 1 für alle Kanten sind, da jede Verbindungsmöglichkeit explizit als Kante im Netzwerk modelliert ist. Durch diese eindeutige Festlegung der Anschlüsse von Servicefahrten ergibt sich daraus direkt der Aufbau der Umläufe. Im TSN dagegen wird durch die Modellierung von Warte- und Umsetzaktionen keine direkte Verknüpfung von Servicefahrten vorgenommen, sondern eine Flusslösung gebildet, die in der Regel mehrere Warte- und Umsetzaktionen beinhaltet. Ein Umlaufplan wird durch Dekomposition dieser Flusslösung erstellt. Dabei kann durch unterschiedliche Strategien, wie die Fahrten verknüpft werden, eine Vielzahl von Umlaufplänen mit identischen Kosten gebildet wer-

den. Durch diesen zusätzlichen Freiheitsgrad weist das TSN die Vorteile auf, zum einen weitere weiche Restriktionen (z.B. Routenrestriktionen) in der Wahl der Dekompositionsstrategie berücksichtigen zu können und zum anderen mehrere alternative Umlaufpläne zu einer Lösung anbieten zu können.

Durch die Aggregation von Umsetzungsmöglichkeiten im TSN können allerdings im Gegensatz zum VBN keine paarweisen Verbindungen explizit betrachtet werden. So können zum Beispiel keine Kosten für paarweise individuelle Fahrtenverknüpfungen modelliert werden, die von der standardisierten Berechnung auf der Basis von Verbindungsdauer und -länge abweichen. Eine hybride Modellierung, die auf dem TSN aufbaut und für individuelle Verbindungskosten explizite Kanten einfügt, ist allgemein nicht möglich, da im TSN auch über mehr als einen Knoten (z.B. über Umsetzfahrten) Verbindungen möglich sein können, so dass die explizit modellierten Informationen umgangen werden können. Die Modellierung expliziter Verbindungsinformationen wird allerdings von Planern im ÖPNV in der Regel nicht gefordert.

Fraktionalität der linearen Relaxation

Bei der LP-basierten Lösung von VSP auf der Basis von Mehrgüter Flussmodellen wird in einer ersten Phase (LP-Phase) die lineare Relaxation des mathematischen Modells gelöst. Anschließend werden die fraktionalen Variablen der Lösung sukzessiv auf einen ganzzahligen Wert und die LP-Relaxation reoptimiert (IP-Phase). Innerhalb dieses Branch&Bound Verfahrens kann der Suchraum durch die Verbesserung von oberen und unteren Schranken eingegrenzt werden bis eine ganzzahlige Lösung gefunden und die Optimalität bewiesen ist. Die Komplexität von LP-basierten Branch&Bound Verfahren für Optimierungsprobleme zeichnet sich dadurch aus, dass die LP-Phase mit polynomieller Laufzeit gelöst werden kann und die IP-Phase einen exponentiellen Laufzeitaufwand erfordert. Die meisten Ansätze zur Lösung von Optimierungsproblemen konzentrieren methodische Verbesserungen daher auf die IP-Phase, die in der Regel ein Vielfaches der Gesamtlaufzeit ausmacht.

Bei der Lösung von Mehrgüter Flussmodellen für reale VSP zeigt sich das besondere Verhalten, dass die Lösung der LP-Relaxation nur wenige Entscheidungsvariablen aufweist, deren Werte nicht ganzzahlig sind. Dieses Verhalten wurde in der Literatur bereits von mehreren Autoren beobachtet und scheint vor allem bei der Lösung von realen Problemstellungen aufzutreten. [Forbes et al., 1994] beschreiben als erste dieses Verhalten, das sie „near-integrality property“ nennen und im folgenden „Fast-Ganzzahligkeitseigenschaft“ genannt wird. In [Hadjar et al., 2006] wird das Verhalten als besondere Eigenschaft deutscher

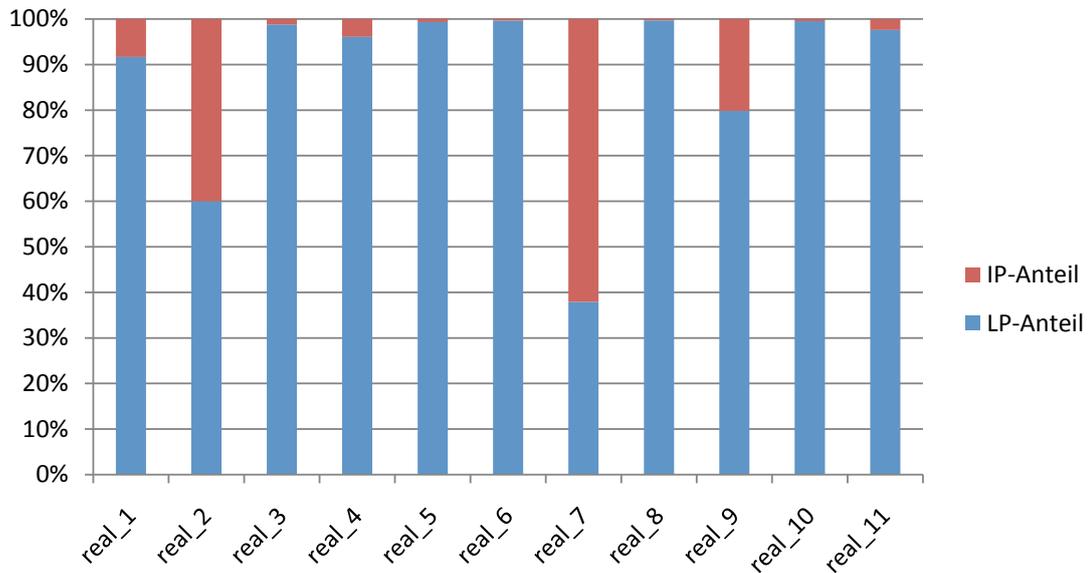


Abbildung 6.7: Vergleich der Laufzeiten für LP- und IP-Phase

Probleminstanzen interpretiert. Allerdings beschreiben auch Veröffentlichungen, die nicht-deutsche reale Instanzen betrachten, wie zum Beispiel [Mesquita und Paixao, 1999] dieses Verhalten. Durch die geringe Anzahl von fraktionellen Variablen in der LP-Lösung, erfordert die IP-Phase in den Veröffentlichungen häufig nur geringen Aufwand. Auf der anderen Seite stellt die Lösung des LP von realen VSP eine besondere Herausforderung dar, da aufgrund der großen Modelldimensionen von vielen Hunderttausend bis Millionen Variablen und der schwierigen Modellstruktur Standardverfahren zur Lösung von LP an ihre Grenzen stoßen. Die Lösung von realen VSP mit Hilfe von LP-basierten Methoden zeigt daher ein im Vergleich zu anderen Optimierungsproblemen umgekehrtes Verhalten.

Das Verhältnis der Laufzeiten von LP- und IP-Phase bei der Lösung von realen Probleminstanzen ist in Abbildung 6.7 dargestellt. In der Mehrzahl der Instanzen überwiegt der Anteil des Aufwands zur Lösung der initialen LP-Relaxation bei weitem dem der IP-Phase.

Mit zunehmender Größe der mathematischen Modelle steigt der Grad der Degeneration bei der Optimierung der LP-Relaxation. Je stärker die Modelle degeneriert sind, desto schwieriger ist es für Simplex-Verfahren ein Modell zu (re-)optimieren, da für viele Iterationen Basislösungen gefunden werden, die den gleichen Zielfunktionswert enthalten. Aus diesem Grund steigt auch der anteilige Laufzeitaufwand der IP-Phase für sehr große Modelle an, da trotz weniger fraktioneller Variablen die Reoptimierung der LP-Probleme im B&B sehr aufwändig

ist.

Da der Fokus dieser Arbeit auf die Unterstützung von realen Anwendungsfällen der Umlaufplanung liegt und die in Abschnitt 6.1.1 vorgestellten sowie andere reale Probleminstanzen die Fast-Ganzzahligkeitseigenschaft aufweisen, ist die im nächsten Kapitel vorgestellte Methodik zur Lösung von VSP auf die effiziente Lösung der LP-Relaxationen ausgerichtet. Sie steht daher im Gegensatz zu den in der Literatur vorgestellten Methoden zur Lösung von künstlich generierten Instanzen, die auf die Beschleunigung der IP-Phase (zum Beispiel durch Cuts oder Branching-Strategien) ausgerichtet sind. Darüber hinaus wird die Fast-Ganzzahligkeitseigenschaft in der im Rahmen dieser Arbeit entwickelten Methodik genutzt, um die Planungsanforderungen hinsichtlich der Ausgabe von Lösungen zur Optimierungslaufzeit zu realisieren. Die Methodik setzt auf die Mehrgüter Flussmodellierung auf der Basis eines aggregierten Time-Space-Netzwerks auf, da, wie gezeigt, die Modellkomplexität erheblich verringert werden kann und die qualitativen Vorteile eine bessere Unterstützung für den Planer ermöglichen. Bevor die Methodik im nächsten Kapitel vorgestellt wird, soll abschließend eine Diskussion der Komplexität von Probleminstanzen vorgenommen werden.

6.2.4 Abschätzung der Komplexität

Die Abschätzung der Komplexität einer realen Umlaufplanungsinstanz ist für die Arbeit des Planers wichtig, damit er das Verhalten einer Optimierungskomponente besser einschätzen kann und die Auswirkungen von Änderungen an Planungsinstanzen versteht. Darüber hinaus kann eine Abschätzung für die automatische Konfiguration von Optimierungsmethoden verwendet werden. Aus diesen Gründen soll im Folgenden eine Kennzahl entwickelt werden, die auf der Basis der in dieser Arbeit vorliegenden Instanzen eine Tendenz für den Laufzeitaufwand geben kann. Aufgrund der vielfachen Einflussgrößen bei der Lösung von Optimierungsproblemen kann die Kennzahl keine Voraussage für den Laufzeitaufwand leisten, sondern soll vielmehr auf der Basis von Problemanalyse und Erfahrungswerten eine Richtlinie darstellen.

Da die Depots des VSP in der Mehrgüter Flussmodellierung jeweils als ein Gut modelliert werden, bilden die Netzwerkschichten für jedes Depot voneinander unabhängige Optimierungsprobleme, die ausschließlich über die Überdeckungsrestriktionen für die Servicefahrten miteinander verbunden sind. Dieses kann als unabhängiger Bereich in der Entscheidungsmatrix dargestellt werden, der für jeden Knoten in der Netzwerkschicht eine Zeile und für jede Kante eine Spalte enthält. Die unterschiedlichen Netzwerkschichten überschneiden sich weder in den

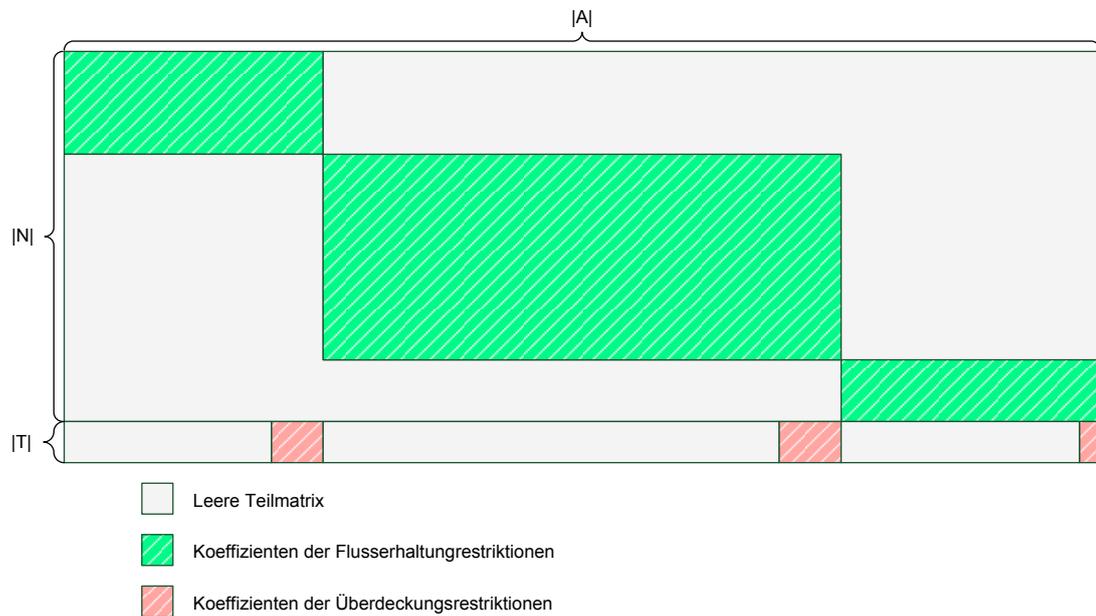


Abbildung 6.8: Struktur der Entscheidungsmatrix des Mehrgüter Flussmodells

Knoten noch in den Kanten, so dass die Gesamtmatrix eine diagonale Struktur von kleineren Netzwerkmodellen aufweist (s. Abbildung 6.8). Jeder Bereich steht dabei für ein Depot und stellt alleine betrachtet ein SD-VSP dar. Zusätzlich zu den Knotenrestriktionen sind in jeder Schicht noch die oberen (und ggf. unteren) Schranken für jede Kante gegeben.

Die Verbindung der einzelnen Matrixbereiche wird über die Überdeckungsrestriktionen für die Sicherstellung der Bedienung aller Servicefahrten erstellt. Diese enthalten für jede Servicefahrt eine Zeile, die im Falle des MD-VSPs in einer Spalte jeder Schicht einen Eintrag enthält. Diese Spalte stellt die Fahrkante der Servicefahrt in der jeweiligen Schicht dar. Im Falle des MVTG-VSPs muss eine Fahrt nicht für jede Schicht einen Eintrag haben, sondern durch die Definition der Fahrzeugtypgruppen nur zu einer Untermenge aller Schichten. Im Falle der Relaxation der Überdeckungsrestriktionen zerfällt das mathematische Modell in die unabhängigen Matrixbereiche der Schichten und kann daher als mehrere kleine SD-VSPs gelöst werden. Diese Eigenschaft wird in Abschnitt 7.2.2 für die Berechnung unterer Schranken und in Abschnitt 7.3.3 zur Bestimmung von lösungsrelevanten Kanten verwendet.

Zur Abschätzung der Komplexität und der zu erwartenden Lösungszeit des LP kommen aufgrund der besonderen Modellstruktur mehrere Einflussfaktoren zum tragen:

- Die Anzahl der Servicefahrten $|T|$ stellt zum einen die entscheidende Ein-

flussgröße für die Variablenanzahl innerhalb einer Netzwerkschicht dar, da für jede Fahrt eine Fahrtkante, Ein- und Ausrückfahrten und Verbindungsfahrten zu potentiellen Anschlüssen modelliert werden. Darüber hinaus bestimmt die Fahrtenmenge die Anzahl der Überdeckungsrestriktionen, die das Modell erst NP-schwer machen.

- Die Anzahl der Kanten in einer Schicht wird mit der Anzahl der Depots bzw. der durchschnittlichen Gruppengröße $\varnothing D$ vervielfacht, da für jede Depot/Fahrzeugtyp-Kombination ein Matrixbereich hinzukommt. Neben dem Einfluss auf die Anzahl der Kanten wird durch $\varnothing D$ die durchschnittliche Dichte einer Überdeckungsrestriktion definiert. Je größer die Anzahl der Möglichkeiten der Zuordnung einer Servicefahrt zu einer Schicht ist, desto schwieriger ist das Modell zu lösen.
- Neben der Anzahl der Servicefahrten und Depots beeinflusst die Fahrplanstruktur die Größe des mathematischen Modells. Je nach Konstellation der Servicefahrten und der Entfernungsmatrix ist die Anzahl der möglichen Verbindungen unterschiedlich. Da die Verbindungskanten auch in der TSN-basierten Modellierung einen Großteil der Netzwerkkanten ausmachen, hat dieser Faktor einen großen Einfluss auf die Variablenanzahl. Beispiele für die Auswirkung unterschiedlicher Konstellationen sind in Tabellen 6.3 und 6.4 gegeben.

Um für die Einschätzung der numerischen Komplexität nicht das Netzwerk aufbauen und die Anzahl der Verbindungen berechnen zu müssen, kann auf die vorgestellte Kennzahl Φ zurückgegriffen werden (s. Abschnitt 6.2.2), da diese über die Einschätzung der benötigten Fahrzeuge gleichzeitig die Tendenz für die Anzahl der möglichen Verbindungen angibt. Je größer die optimale Fahrzeuganzahl ist, desto weniger Fahrten können zur Bedienung innerhalb eines Umlaufs verbunden werden und desto geringer ist tendenziell die Anzahl der zu betrachtenden Kanten.

Auf der Basis der, für diese Arbeit gegebenen, Instanzen wurde der Einfluss der unterschiedlichen Faktoren untersucht und anhand numerischer Tests unterschiedliche Konstellationen evaluiert. Anhand empirischer Untersuchungen konnte festgestellt werden, dass die im Folgenden vorgestellte Kennzahl Λ zur tendenziellen Einschätzung der erwarteten Laufzeit geeignet ist. Sie bildet auf der Basis der in dieser Arbeit betrachteten Instanzen einen Vorschlag für ein Komplexitätsmaß und erhebt keinen Anspruch auf allgemeine Gültigkeit.

Die Kennzahl wird durch Multiplikation von Fahrtenmenge $|T|$, durchschnittlicher Gruppengröße $\varnothing D$ und umgekehrt proportionaler Fahrplanstruktur $\frac{1}{\Phi}$ be-

Instanz	$ T $	$\varnothing D$	Φ	Λ	t_{LP}
real_1	424	1,00	0,07	0	0
real_2	426	1,00	0,08	0	0
real_3	867	2,00	0,08	1	2
real_4	1296	1,30	0,04	2	0
real_5	1808	13,00	0,03	4.292	6.187
real_6	2047	3,21	0,06	21	17
real_7	2394	12,00	0,03	3.614	13.185
real_8	2452	2,00	0,04	12	3
real_9	2633	3,00	0,05	32	108
real_10	2825	2,00	0,03	21	13
real_11	3067	1,00	0,06	1	0
real_12	9124	11,18	0,04	8.615	- ¹
real_13	10710	5,00	0,03	1.212	6.808
real_14	10710	30,57	0,03	276.920	- ¹
real_15	11062	6,47	0,04	2.374	2.000

¹ Optimierung wurde nach 36.000 Sekunden Laufzeit abgebrochen

Tabelle 6.7: Abschätzung der numerischen Komplexität realer Instanzen

rechnet. Der unterschiedlich starke Einfluss der Faktoren wird durch eine quadratische Berücksichtigung der Fahrplanstruktur und einem kubischen Einfluss von der Gruppengröße berücksichtigt. Um die Kennzahl so zu skalieren, dass sie einen Richtwert für die erwartete Lösungszeit auf einem aktuellem Computersystem angibt, wurde der Wert mit dem Faktor 10^{-6} angepasst. Diese Konstante sollte je nach vorhandenen Ressourcen angepasst werden und stellt einen Richtwert dar. Die resultierende Kennzahl für die Einschätzung der numerischen Komplexität einer Instanz lautet daher:

$$\Lambda = \frac{|T| \cdot \varnothing D^3}{\Phi^2 \cdot 10^6}$$

In Tabelle 6.7 ist die Kennzahl in Verbindung mit den relevanten Faktoren und der bei Optimierung der Modelle benötigten Laufzeit angegeben. Spalte t_{LP} gibt die Optimierungsdauer zur Lösung der linearen Relaxation der Instanz in Sekunden an. Zur Lösung wurde das jeweils schnellste LP-Verfahren der Standardoptimierungssoftware CPLEX 11.0.1 verwendet. Die Ergebnisse zeigen, dass wie erwartet zwar keine Vorausbestimmung der Laufzeit möglich ist, aber eine Tendenz und Abschätzung der Laufzeit möglich ist. Da die Einstellung der Einfluss-

faktoren vor allem auf Erfahrungen und Analysen unterschiedlicher Einstellungen zurückgeht, sollte die Kennzahl möglichst anhand weiterer realer Probleminstanzen validiert werden.

Zusammenfassung

Die in den vorangegangenen Abschnitten gegebenen Ausführungen zeigen erstmals eine Analyse realer und künstlich generierter Instanzen von Umlaufplanungsproblemen. Es werden Unterschiede in der Fahrplanstruktur dieser Instanzentypen aufgezeigt und anhand einer Kennzahl messbar gemacht. Mit dem Fokus auf die Lösung realer Umlaufplanungsprobleme werden die unterschiedlichen Modellierungsansätze untersucht und Vorteile der Mehrgüter Flussmodelle gegenüber alternativen Flussmodellen und dem Set Partitioning Modell gezeigt. Unter den Mehrgüter Flussmodellen eignet sich die Modellierung auf der Basis eines aggregierten Time-Space-Netzwerks nach [Kliwer, 2005] in besonderer Weise, da die Anzahl der zu betrachtenden Variablen im mathematischen Modell gegenüber der klassischen Modellierung deutlich verringert wird.

Zur Einschätzung der numerischen Komplexität von Instanzen wird erstmals eine Kennzahl eingeführt, die über die Berücksichtigung der Anzahl von Servicefahrten und Depots hinaus geht und die Struktur des Fahrplans mit einbezieht.

Auf der Basis der aus den gezeigten Analysen gewonnen Erkenntnisse können Lösungsmethoden konzipiert werden, die in besonderer Hinsicht die Eigenschaften der Probleminstanzen berücksichtigen. Daher werden in den folgenden Kapiteln neue Methoden vorgestellt, die durch das Nutzen dieser Erkenntnisse eine effiziente Lösung von Umlaufplanungsproblemen ermöglichen.

7 Lösungsmethodik für Busumlaufplanungsprobleme

Die Lösung von Problemstellungen im Bereich der Busumlaufplanung (s. Kapitel 2) wird seit vielen Jahren in Forschung und Planungssystemen betrachtet (s. Kapitel 4). Trotz großer Fortschritte, die in den letzten Jahren im Bereich der Lösung von VSP erreicht wurden, existieren reale Problemstellungen, die unter Berücksichtigung praxisrelevanter Anforderungen und Anwendungsfälle nicht gelöst werden können (s. Abschnitt 5). Wie in Abschnitt 4.1 gezeigt wurde, stellt die Lösung von VSP mit einem Depot und homogener Fahrzeugflotte (SD-VSP) für aktuelle Optimierungssysteme keine Schwierigkeit dar, da auch Instanzen mit einer sehr großen Anzahl von Fahrten innerhalb von Sekunden oder wenigen Minuten optimal gelöst werden können. Sobald weitere Freiheitsgrade – wie die Berücksichtigung von mehreren Depots, Fahrzeugtypen bzw. Fahrzeugtypgruppen oder Zeitfenstern für Fahrten – hinzukommen, können aktuelle Ansätze zur exakten Lösung von VSP für große Probleminstanzen an ihre Grenzen stoßen.

Im Rahmen dieser Arbeit sind daher neue Lösungsverfahren entwickelt worden, die VSP unter besonderer Berücksichtigung der in Abschnitt 2.2 beschriebenen praxisrelevanten Anwendungsfälle lösen können. Gleichzeitig sind die Verfahren für den jeweiligen Anwendungsfall so effizient, dass auch sehr große reale VSP gelöst werden können.

In diesem Kapitel wird eine neue Methodik zur Lösung von VSP mit mehreren Depots (MD-VSP) und/oder Fahrzeugtypgruppen (MVTG-VSP) vorgestellt. Die Methode ist aufgrund ihrer Flexibilität und Anpassbarkeit für alle in Abschnitt 2.2 beschriebenen Anwendungsfälle der Planung ohne Zeitfenster einsetzbar. Zur einfacheren Darstellung ist für den weiteren Verlauf dieses Kapitels unter dem MD-VSP auch die Betrachtung des MVTG-VSP eingeschlossen, sofern die beiden Problemstellungen nicht explizit unterschieden werden. Methoden für die Berücksichtigung von Zeitfenstern für Fahrten werden in Kapitel 8 präsentiert.

Eine Anforderung der *strategischen Planung* und der *taktischen Planung ohne Zeitfenster* ist es, die Gesamtkosten des erwarteten operativen Fahrzeugeinsatzes in kurzer Zeit abschätzen zu können (s. Abschnitte 2.2.3 und 2.2.2). Diese Abschätzung kann durch die Berechnung einer gültigen Lösung und einer

zulässigen unteren Schranke getätigt werden. Es gilt dabei eine Abwägung zwischen der Laufzeit der Methode und der Lösungsqualität zu beachten. Ein wünschenswertes Ziel zur Planung innerhalb dieser Anwendungsfälle ist eine Methode, die

- eine gültige Lösung mit der optimalen Anzahl der benötigten Fahrzeuge und günstigen operativen Kosten findet,
- eine enge untere Schranke berechnet und
- diese Lösungen innerhalb von Sekunden oder bei großen Probleminstanzen wenigen Minuten berechnet.

Die in diesem Kapitel vorgestellte Methodik berücksichtigt diese Zielsetzungen. Darüber hinaus können die gewonnenen Erkenntnisse dieser Abschätzung genutzt werden, um für die Anwendungsfälle der *operativen Planung ohne Zeitfenster* und der *integrierten Planung* gute oder optimale Lösungen zu finden (s. Abschnitte 2.2.1 und 2.2.4). Dazu schließt sich an eine kurze Initialisierungsphase, in der die Abschätzungen durchgeführt werden, eine Verbesserungsphase an, in der neben der schrittweisen Verbesserung der Lösung auch die weiteren Anforderungen wie die Vermeidung eines Black-Box-Verhaltens berücksichtigt werden. Im Laufe der Verbesserungsphase können dem Planer gültige Lösungen sowie ggf. weitere Informationen gegeben werden. Dadurch kann die Methode so gesteuert werden, dass im Laufe der zur Verfügung stehenden Optimierungszeit eine möglichst gute Lösung gefunden werden kann. Für den Fall einer unbeschränkten Laufzeit konvergiert die Methode gegen eine optimale Lösung. Abbildung 7.1 zeigt schematisch die Phasen und das Vorgehen der Methodik.

Die einzelnen Bausteine der Methodik werden in diesem Kapitel vorgestellt. Dazu werden in den Abschnitten 7.1 und 7.2 zuerst die Verfahren beschrieben, die innerhalb der Initialisierungsphase zum Einsatz kommen. In Abschnitt 7.1 wird eine neue Heuristik vorgestellt, die unter Berücksichtigung der weiteren Anforderungen eine Lösung mit optimaler Anzahl an Fahrzeugen garantieren kann. Abschnitt 7.2 zeigt Verfahren zur Abschätzung einer unteren Schranke für die Problemstellung. In Abschnitt 7.3 ein Kantengenerierungsalgorithmus vorgestellt, der auf der Basis der Lösungen aus der Initialisierungsphase eine schrittweise Verbesserung der gültigen Lösungen ermöglicht.

Durch die Integration weiterer Lösungsmodule kann die Anpassbarkeit und die Effizienz der Methode weiter verbessert werden. Diese Ansätze werden in Abschnitt 7.4 präsentiert. Zur Evaluierung der Methodik werden in Abschnitt 7.5 die Ergebnisse numerischer Tests für die in Abschnitt 6.1 diskutierten realen Pro-

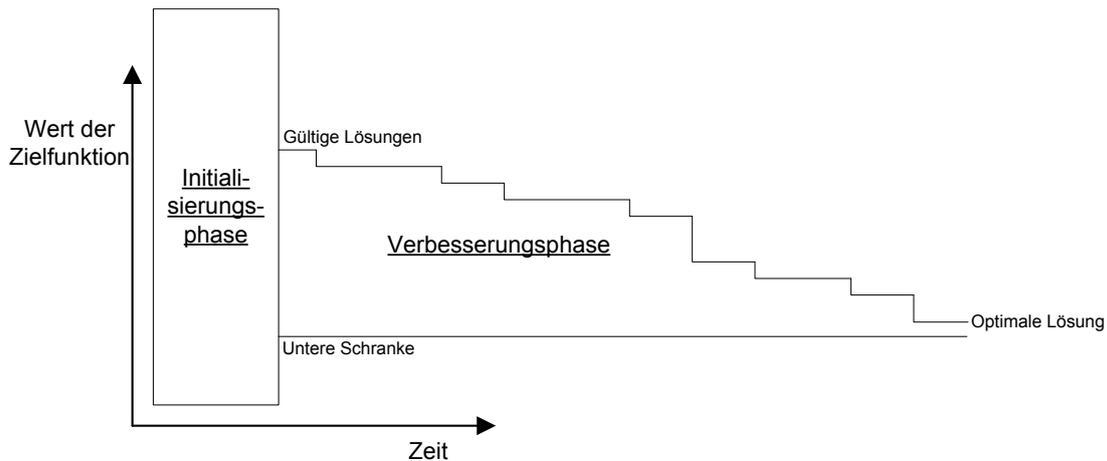


Abbildung 7.1: Vorgehen zur Berücksichtigung der Anforderungen von Anwendungsfällen ohne Zeitfenster

bleminstanzen vorgestellt. Abschließend wird in Abschnitt 7.5.4 eine Evaluierung der Methodik vorgenommen und die Erkenntnisse zusammengefasst.

7.1 Heuristischer Lösungsansatz

Zur Bestimmung einer gültigen Lösung für das MD-VSP sind bereits eine Reihe von Ansätzen veröffentlicht worden (s. Kapitel 4). In die Konzeption des im Folgenden vorgestellten heuristischen Lösungsansatzes können verschiedene Aspekte bestehender Verfahren eingebracht werden, so dass der Ansatz eine Weiterentwicklung des Stands der Technik darstellt.

Die Heuristik baut auf der in [Kliwer, 2005] vorgestellten Fahrketten-Heuristik auf, in der mehrere vereinfachte Unterprobleme gelöst werden, um anhand der resultierenden Umlaufpläne gleiche Ketten von Servicefahrten zu identifizieren. Diese Ketten werden bei der Lösung des Gesamtproblems in einer zweiten Phase fixiert, so dass aufgrund der geringeren numerischen Komplexität in weniger Zeit nahezu optimale Lösungen gefunden werden können (vgl. [Gintner et al., 2005]). Dieses Verfahren wird hinsichtlich des Einsatzes zur schnellen Findung einer gültigen Lösung angepasst und erweitert, so dass

- die Heuristik in möglichst kurzer Zeit eine gültige Lösung findet,
- die Lösung die optimale Anzahl der benötigten Fahrzeuge enthält und
- die Heuristik auf die Lösung von realen Probleminstanzen ausgelegt ist.

Um die benötigte Laufzeit innerhalb der ersten Phase gering zu halten, wird als Spezialfall der Fahrketten-Heuristik nur ein Unterproblem gebildet, das bei weitestgehend Reduktion der numerischen Komplexität die Bestimmung einer optimalen Fahrzeuganzahl sicherstellen kann. Zur Bildung dieses Unterproblems wird erstmals ein automatisiertes Vorgehen gezeigt. Die resultierenden Umläufe werden jeweils als eine Fahrkette fixiert, so dass in der zweiten Phase statt der Lösung eines (NP-schweren) MD-VSPs der Spezialfall betrachtet wird, dass jeder Umlauf genau einem Depot/Fahrzeugtypen zugewiesen werden muss. Das Verfahren wird durch eine dritte Phase erweitert, in der für jedes Depot die Umläufe aufgebrochen werden und ein von den anderen Depots unabhängiges SD-VSP gelöst wird. Hierdurch können die operativen Kosten unter Beibehaltung der optimalen Fahrzeuganzahl weiter gesenkt werden.

Unabhängig vom Einsatz dieser erweiterten Heuristik zur schnellen Bestimmung gültiger Lösungen, kann die Fahrketten-Heuristik für die Reduzierung der numerischen Komplexität bei großen Problemstellungen ergänzend eingesetzt werden, weshalb diese Technik in Abschnitt 7.4 aufgenommen und in die in diesem Kapitel vorgestellte Methodik integriert ist.

Die drei Phasen des im Rahmen dieser Arbeit entwickelten heuristischen Ansatzes werden im Folgenden beschrieben.

Phase 1: Berechnung der optimalen Anzahl von Umläufen

In dieser Phase wird durch Ausnutzung der Komplexitätsstruktur des MD-VSPs ein vereinfachtes Problem betrachtet, das dennoch die Identifizierung der optimalen Umlaufanzahl sicherstellt. Wie in Abschnitt 6.2.3 dargestellt, ist die numerische Komplexität des MD-VSPs stark von der Anzahl der Depots und Fahrzeugtypen abhängig. Daher wird in der ersten Phase der gesamte Fahrplan mit einer eingeschränkten Menge an Depots und Fahrzeugtypen betrachtet, deren automatisierte Bildung im Folgenden gezeigt wird.

Substitution von Fahrzeugtypen Die Fahrzeugtypen können zu einem gewissen Grad substituiert werden, ohne dass die optimale Fahrzeuganzahl verändert wird. Die Substitution erfolgt in drei Schritten:

1. Alle Fahrzeugtypen, die eine identische Kostenstruktur (fixe - und operative Kosten) haben und identischen Fahrzeugtypgruppen zugeteilt sind, können zusammengefasst werden. Die Typen werden durch einen Fahrzeugtypen mit gleichen Kosten und gleicher Fahrzeugtypgruppenzuordnung substituiert. Je Depot entsprechen die Kapazitäten für den Fahrzeugtyp der Summe

aller substituierten Typen. Durch diese Veränderung bleibt der optimale Zielfunktionswert unverändert.

2. Die Fahrzeugtypen mit gleicher Zuordnung zu Fahrzeugtypgruppen und unterschiedlichen Kosten können zusammengefasst werden, wenn alle Kostenwerte eines Typs die Kosten der anderen Typen dominieren. Die Fahrzeugtypen können analog zum ersten Fall durch das Summieren der Kapazitäten durch einen neuen substituiert werden, dessen Kostenstruktur des Typs mit den geringsten Kosten entspricht. Durch dieses Ersetzen kann die optimale Lösung der ursprünglichen Problemstellung verändert werden, da aufgrund der Kapazitäten nicht alle Umläufe durch den günstigsten Typen abgedeckt werden können. Durch „gierige“ Verwendung der jeweils günstigsten Fahrzeugtypen kann nach einer Optimierung eine Lösung konstruiert werden (Verwendung des günstigsten Typs für möglichst viele Umläufe, danach den zweitgünstigsten Typen usw.).
3. Fahrzeugtypen, die identischen Fahrzeugtypgruppen zugeordnet sind, aber unterschiedliche Kostenstrukturen haben, von denen keine die andere dominiert, können durch einen neuen Typen ersetzt werden. Diese Ersetzung kann allerdings nicht mehr den Erhalt einer optimalen Lösung der Problemstellung garantieren. Die optimale Berechnung der benötigten Fahrzeuganzahl ist allerdings durch die gleiche Gruppenzuordnung der substituierten Fahrzeugtypen sichergestellt.

In allen drei Fällen ist es zur Reduzierung von Fahrzeugtypen erforderlich, dass die Typen eine gleiche Zuordnung zu Fahrzeugtypgruppen haben. Andernfalls kann die Berechnung der optimalen Fahrzeuganzahl nicht gewährleistet werden. Für den Fall, dass ein Fahrzeugtyp a weniger Gruppen zugeordnet ist als Typ b , können bei Ersetzung von b durch a die Fahrten nicht abgedeckt werden, die exklusiv von Typ b zu bedienen sind. Im anderen Fall, dass a mehr Gruppen zugeordnet ist als b , können Umläufe entstehen, die nur von a aber nicht von b bedient werden können. Reicht die Kapazität von a nicht aus, um alle ermittelten Umläufe zu bedienen, wäre die Lösung ungültig.

Die möglichen Konstellationen zur Ersetzung von Fahrzeugtypen werden anhand des in Tabelle 7.1 gegebenen Beispiels veranschaulicht. Jede Zeile entspricht der Definition eines Fahrzeugtyps mit den Werten für die Berechnung der fixen und operativen Kosten. Des Weiteren ist die Kapazität des Typs für ein Depot angegeben. Es seien drei Fahrzeugtypgruppen A, B und C gegeben. Jeder Typ ist einer Untermenge dieser Fahrzeugtypgruppen zugeordnet. Anhand des ersten Ersetzungsschritts können Typen a und b zusammengefasst werden. Die

Typ	Fixe Kosten	Operative Kosten		Kapazität	Gruppenzuordnung
	je Fahrzeug	je Km	je Min		
a	5000	10	8	10	A, B
b	5000	10	8	5	A, B
c	6000	10	12	7	A, B
d	6000	8	5	8	A, B
e	5000	10	8	10	A, B, C
f	5000	10	8	10	A

Tabelle 7.1: Beispiel: Definition von Fahrzeugtypgruppen

Gesamtkapazität des ersetzenden Typs a' beträgt 15. Im zweiten Schritt kann auch Typ c durch a' substituiert werden, da bei gleicher Gruppenzuordnung alle Kostenwerte von c die Kostenwerte von a' dominieren. Der resultierende Typ a'' hat eine Kapazität von 22. Der dritte Schritt ersetzt auch Typ d aufgrund der identischen Gruppenzuteilung durch a'' . Da die Fahrzeugtypen e und f eine abweichende Gruppenzuordnung haben, können sie nicht weiter zusammengefasst werden. Nach den drei Schritten der Aggregation sind von ursprünglich 6 Fahrzeugtypen noch 3 zu betrachten. Dabei entspricht die optimale Anzahl an Fahrzeugen der der Originalformulierung.

Substitution von Depots Neben der Reduzierung der zu betrachtenden Fahrzeugtypen in der ersten Phase der hier vorgestellten Heuristik wird auch die Anzahl der Depots verringert. Unter der Annahme einer vollständig gegebenen Verbindungsmatrix können alle Depots durch ein beliebiges Depot ersetzt werden ohne die Optimalität der Fahrzeuganzahl zu verlieren. Dies gilt, da die Verbindungszeiten zwischen den Haltestellen in den betrachteten Modellen unabhängig vom verwendeten Fahrzeugtyp sind und sich dadurch die Gültigkeit einer Sequenz der von einem Fahrzeug bedienbaren Servicefahrten nicht von Depot zu Depot unterscheidet. Lediglich die Ein- und Ausrückfahrten können eine unterschiedliche Dauer haben. Bei der Betrachtung eines Planungszeitraums von einem Tag befinden sich die Fahrzeuge zu Beginn des Tages in den Depots. Daher müssten Fahrzeuge aus einem weiter entfernten Depot lediglich frühzeitiger mit der Ausrückfahrt beginnen bzw. kehren durch eine längere Einrückfahrt erst später in das Depot zurück. Die Unterschiede würden sich nur auf die Anzahl der benötigten Fahrzeuge auswirken, wenn ein Depot so weit von den Haltestellen der Servicefahrten entfernt ist, dass sich die Planungen von zwei Fahrplantage überschneiden. Dieser Fall tritt in den betrachteten Praxisproblemen allerdings

nicht auf. Die operativen Kosten für die Umläufe können sich neben unterschiedlichen Ein- und Ausrückfahrten auch dadurch unterscheiden, dass in einem Fall eine Rückkehr in ein Depot zwischen zwei Servicefahrten für eine Pause möglich ist und für ein anderes Depot aufgrund längerer Depotfahrten nicht möglich ist.

Lösung der reduzierten Problemstellung Durch die Reduzierung der zu betrachtenden Fahrzeugtypen und Depots kann ein VSP mit einem Depot und wenigen Fahrzeugtypen aufgestellt werden, das die gleiche optimale Anzahl von benötigten Fahrzeugen hat wie die ursprüngliche Problemstellung. Die reduzierte Problemstellung wird unter Verwendung der TSN-basierten Modellierung als Mehrgüter Flussformulierung gelöst. Aufgrund der reduzierten numerischen Komplexität kann das Optimierungsmodell mit Optimierungsverfahren aus Standardbibliotheken in kurzer Zeit optimal gelöst werden, obwohl es mit mehreren Fahrzeugtypen noch eine NP-schwere Komplexität haben kann (s. Abschnitt 7.5).

Phase 2: Bildung von gültigen Umläufen mit optimaler Fahrzeuganzahl

Nach der Optimierung der in Phase 1 aufgestellten Problemstellung können die Umläufe aus der Flusslösung dekomponiert werden und entsprechend der Fahrketten-Heuristik aus [Kliwer, 2005] fixiert werden. Wenn neben der Minimierung der geplanten Kosten weitere Zielsetzungen oder weiche Restriktionen bestehen, kann der Freiheitsgrad der Dekomposition genutzt werden, um bezüglich der weiteren Anforderungen günstige Fahrtensequenzen zu generieren.

Die Lösung des MD-VSPs bei fixierten Fahrketten ist in diesem Fall einfach, da durch die Fixierung der Umläufe bei optimaler Fahrzeuganzahl nur noch eine Zuweisung jedes Umlaufs auf ein Depot durchgeführt werden muss. Vereinfacht kann diese zweite Phase daher als Problem der Zuweisung der gebildeten Umläufe Γ zu den Depots D dargestellt werden, wobei die Entscheidungsvariablen x_{ij} nur für gültige Zuweisungen (z.B. beschränkt durch Fahrzeugtypgruppen) definiert sind (s. Abbildung 7.2).

$$\min \sum_{i \in \Gamma} \sum_{j \in D} c_{ij} x_{ij} \quad (7.1)$$

$$s.t. \sum_{j \in D} x_{ij} = 1 \quad \forall i \in \Gamma \quad (7.2)$$

$$\sum_{i \in \Gamma} x_{ij} \leq w_j \quad \forall j \in D \quad (7.3)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (7.4)$$

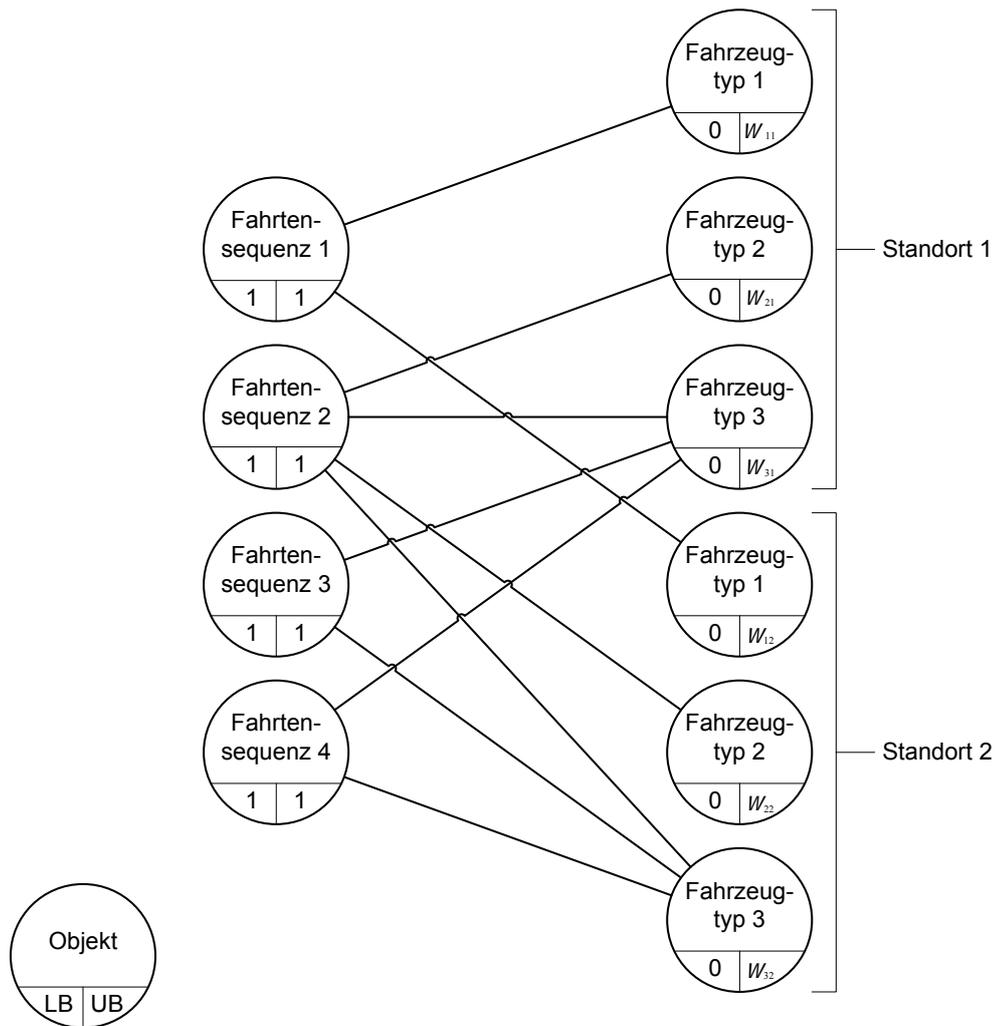


Abbildung 7.2: Beispiel: Möglichkeiten der Zuweisung von Umläufen zu Depots und Fahrzeugtypen

Phase 3: Bildung von Umläufen mit geringen operativen Kosten

Obwohl die in Phase 2 gefundene Lösung bereits eine gültige und bezüglich der Fahrzeuganzahl optimale Lösung darstellt, können die operativen Kosten der Umläufe durch einen weiteren Schritt verbessert werden. Diese Erweiterung der Fahrketten-Heuristik aus [Kliewer, 2005] kann einige Potentiale der in der ersten Phase stark eingeschränkten Freiheitsgrade zur Kostensenkung nutzen. Diese Phase ist daher insbesondere bei dem in diesem Abschnitt gezeigten automatisierten Bildung eines Problems mit geringer numerischer Komplexität sinnvoll.

Alle Depots, denen in Phase 2 mehr als ein Umlauf zugewiesen wurde, können Optimierungspotentiale aufweisen, da die zugewiesenen Fahrtensequenzen auf der Basis einer nicht optimalen Lösung des MD-VSPs in Phase 1 gebildet wurden. Um innerhalb eines Depots günstigere Umläufe zu finden, kann aus den durch die Umläufe bedienten Servicefahrten ein neuer Fahrplan gebildet werden. Dieser Fahrplan kann für jedes Depot als SD-VSP gelöst werden, um innerhalb eines Depots optimale Umläufe zu erhalten. Da die Optimierung für jedes Depot unabhängig durchgeführt werden kann, werden in der dritten Phase mehrere einfache Optimierungsprobleme gelöst. Die resultierende Lösung ist gleich oder besser der Lösung aus Phase 2. Sie stellt für das MD-VSP allerdings noch eine heuristische Lösung dar, da durch die Zuweisung der Fahrten auf Depots die Freiheitsgrade in Phase 3 verringert werden.

7.2 Bestimmung unterer Schranken

Im Falle von Anwendungen, in denen eine Vielzahl von Umlaufplanungsproblemen gelöst werden müssen (wie zum Beispiel bei der Auswertung unterschiedlicher Szenarien oder der Lösung von VSP als Unterproblem), ist es erforderlich zusätzlich zu einer gültigen Lösung eine untere Schranke für die Kosten des Problems berechnen zu können. Nachdem im vorangegangenen Abschnitt ein Verfahren für die schnelle Berechnung einer guten gültigen Lösung vorgestellt wurde, werden in diesem Abschnitt verschiedene Methoden zur Generierung einer unteren Schranke diskutiert.

Wie in Abschnitt 6.2.3 erläutert wurde, haben reale Planungsinstanzen des MD-VSP die besondere Eigenschaft, dass eine Lösung der linearen Relaxation nur sehr wenige oder keine Variablen mit fraktionellen Lösungswerten enthält. Darüber hinaus weicht der optimale Zielfunktionswert für das MD-VSP in der Regel nicht stark vom LP-Optimum ab. Die LP-Relaxation bietet sich daher als gute untere Schranke an. Allerdings ist eine Optimierung der LP-Relaxation bereits so schwierig, dass die Berechnung einen großen Laufzeitaufwand erforder-

dert. Die folgenden Ansätze zur Bestimmung von unteren Schranken verfolgen daher die Strategie die LP-Relaxation zu vereinfachen, um die benötigte Optimierungszeit zu verkürzen und gleichzeitig eine möglichst nahe am optimalen LP-Zielfunktionswert liegende Schranke zu erhalten. In dem in Abschnitt 7.2.1 präsentierten Ansatz wird die Anzahl der Variablen im Modell verringert, indem die Kostenwerte so angepasst werden, dass bei Erhaltung einer gültigen unteren Schranke mehrere Variablen aggregiert werden können. Die zweite Strategie verfolgt die Relaxation von Restriktionen, die das Modell schwierig zu lösen machen. Dazu werden in Abschnitt 7.2.2 geeignete Reformulierungen für das mathematische Modell präsentiert und die Lösung der abgeleiteten Optimierungsprobleme gezeigt. In Abschnitt 7.2.3 wird die Möglichkeit diskutiert die exakte Optimierung eines MD-VSPs mit Hilfe der vorgestellten unteren Schranken zu unterstützen.

7.2.1 Untere Schranken durch Modellaggregation

Um eine schnelle untere Schranke für das MD-VSP berechnen zu können, muss die numerische Komplexität des mathematischen Modells reduziert werden. Wie in Abschnitt 6.2 gezeigt wurde, ist die numerische Komplexität stark abhängig von der Anzahl der betrachteten Depotstandorte und Fahrzeugtypen. In der in Abschnitt 7.1 vorgestellten Heuristik wurden in der ersten Phase diese Mengen aggregiert, so dass bei Sicherstellung einer gültigen und bezüglich der Fahrzeuganzahl optimalen Lösung möglichst wenige Depots betrachtet werden müssen. Die resultierende Reduktion der numerischen Komplexität führt, wie in den Ergebnissen in Abschnitt 7.5 gezeigt wird, zu einer kurzen Laufzeit für die Optimierung. Dieser Ansatz kann auf die Bestimmung einer unteren Schranke übertragen werden. Anstatt die Gültigkeit einer Lösung zu erhalten, muss sichergestellt werden, dass die Vereinfachungen eine gültige untere Schranke garantieren. Eine untere Schranke bleibt bei der Durchführung von Veränderungen an einem Minimierungsproblem gültig, die zu einer Verringerung von Kostenwerten und damit den Gesamtkosten führen.

Um möglichst viele Fahrzeugtypen zusammenzufassen, kann das in Abschnitt 7.1 präsentierte Verfahren zur Substitution von Fahrzeugtypen in folgender Weise angepasst werden. Die erste Phase, in der Fahrzeugtypen mit exakt gleichen Kostenwerten für fixe und operative Kosten durch einen Typ ersetzt werden, kann entsprechend übernommen werden, da durch eine Addition der jeweiligen Kapazitäten dieser Fahrzeugtypen die optimalen Zielfunktionswerte des ursprünglichen und angepassten Modells identisch sind. In der vorgestellten Heuristik werden in dem zweiten und dritten Schritt Fahrzeugtypen mit unterschiedlichen Kostenwerten zusammengefasst. Dieses Vorgehen der Substitution kann für das Verfahren

zur Bestimmung einer unteren Schranke angepasst werden, indem nach der Identifikation einer möglichen Substitution die Kostenwerte des neuen Fahrzeugtypen so definiert werden, dass für alle Kostenarten der jeweils kleinste Wert über alle zu ersetzenden Fahrzeugtypen gewählt wird. Durch diese Bildung der Kostenstrukturen werden alle im Modell abgebildeten Kosten gleich- oder unterschätzt, wodurch sichergestellt ist, dass der Zielfunktionswert des vereinfachten Modells eine gültige untere Schranke für das ursprüngliche Problem darstellt.

Wie in der ersten Phase der vorgestellten Heuristik können alle Depotstandorte zusammengefasst werden. Durch diese Zusammenlegung wird in der Heuristik zwar sichergestellt, dass die in einer Lösung vorhandenen Fahrtensequenzen gültig sind, aber der Zielfunktionswert der Modelllösung stellt in der ersten Phase weder eine gültige untere noch eine obere Schranke der Problemstellung dar. So können bei der Betrachtung eines Depots Kosten für Depotfahrten entstehen, die unter Verwendung eines anderen Depots geringer wären. In dem Verfahren zur Bestimmung einer unteren Schranke müssen deshalb alle Entfernungsdefinitionen zwischen dem Depot und allen anderen Haltestellen angepasst werden. Für jede Verbindungsdefinition zwischen einer Haltestelle i und dem Depot d wird als Entfernung die kürzeste Entfernung aller Verbindungen zwischen i und dem ursprünglichen Depot $j \in D$ verwendet. Entsprechend werden alle Verbindungszeiten angepasst. Wie bei dem Zusammenfassen der Fahrzeugtypen werden alle möglichen Verbindungskosten zwischen einem Depot und einer Start- bzw. Endhaltestelle auf diese Weise richtig eingeschätzt oder unterschätzt.

Durch die konsequente Nicht-Überschätzung aller anfallenden Kosten, kann sichergestellt werden, dass mit der gleichen reduzierten numerischen Komplexität wie in der ersten Phase der in Abschnitt 7.1 vorgestellten Heuristik eine gültige untere Schranke für das ursprüngliche Modell gefunden werden kann. Die Qualität dieser Abschätzung ist abhängig davon wie stark sich die Kostenstrukturen der Fahrzeugtypen und die Verbindungsentfernungen bzw. -dauern der Depots zu Haltestellen unterscheiden. Unter der Annahme, dass reale Instanzen in der Regel nur wenig Varianz in den Kostenstrukturen aufweisen, kann das Verfahren gute untere Schranken finden. Die durch dieses Vorgehen gefundene untere Schranke wird im weiteren Verlauf dieser Arbeit als *aggregierte untere Schranke (AggrLB)* bezeichnet.

7.2.2 Untere Schranken durch Lagrange Relaxationen

Eine alternative Möglichkeit zur Bestimmung von unteren Schranken kann durch die Lagrange Relaxation, in der ausgewählte Restriktionen der Modellformulierung relaxiert werden und als duale Variablen in die Zielfunktion mit einfließen,

erreicht werden. Eine Erklärung dieser Methode ist in Abschnitt 3.3 gegeben. Für das Mehrgüter Flussmodell des MD-VSPs werden im Folgenden zwei mögliche Lagrange Relaxationen betrachtet.

Relaxation der Überdeckungsrestriktionen

Ein übliches Vorgehen bei der Entscheidung welche Restriktionen relaxiert werden ist die Auswahl „schwieriger“ Modellrestriktionen. Für NP-schwere Optimierungsprobleme werden diese (sofern möglich) so gewählt, dass die übrigen Restriktionen ein Optimierungsmodell mit einer unimodularen Entscheidungsmatrix bilden. Dadurch kann sichergestellt werden, dass die Lagrange Probleme effizient gelöst werden und der Zielfunktionswert einer optimalen Lösung des Lagrange Dual Problems dem Wert der LP-Relaxation entspricht (s. Abschnitt 3.3).

Wie in Abschnitt 6.2.3 gezeigt wurde, besteht die Entscheidungsmatrix des Mehrgüter Flussmodells aus mehreren unabhängigen Teilen (eine je Netzwerkschicht $d \in D$), die über die Überdeckungsrestriktionen der Servicefahrten miteinander verbunden sind. Das Modell lautet:

$$MDVSP : \quad \min \sum_{(i,j)^d \in A} c_{ij}^d x_{ij}^d \quad (7.5)$$

$$s.t. \quad \sum_{i:(i,j)^d \in A} x_{ij}^d - \sum_{i:(j,i)^d \in A} x_{ji}^d = 0 \quad \forall j \in N^d, \forall d \in D \quad (7.6)$$

$$\sum_{d \in D} x_t^d = 1 \quad \forall t \in T \quad (7.7)$$

$$x_d \leq w_d \quad \forall d \in D \quad (7.8)$$

$$x_{ij}^d \in \mathbb{N} \quad \forall (i,j)^d \in A \quad (7.9)$$

Ausgehend von der Mehrgüter Flussformulierung (*MDVSP*) können die Überdeckungsrestriktionen (7.7) relaxiert werden. Dieses Vorgehen wird zum Beispiel in [Ahuja et al., 1993] für allgemeine Mehrgüter Flussprobleme gezeigt. Das resultierende Lagrange Problem LR_{cc} wird in der Zielfunktion um die dualen Variablen λ_t erweitert, die die Strafkosten für die Verletzung der relaxierten Restriktionen abbilden. Das Optimierungsproblem hat ohne die Überdeckungsrestriktionen eine unimodulare Matrix (s. Abschnitt 3.1). Die Restriktionen zur Sicherstellung der Ganzzahligkeit der Variablen (7.9) müssen deshalb nicht im Lagrange Problem betrachtet werden, da die Ganzzahligkeit für optimale LP-Lösung gegeben ist. Eine gültige Lösung des Lagrange Problems stellt einen Umlaufplan dar, bei dem nicht alle Fahrten bedient werden und/oder Servicefahrten mehrfach bedient werden. Jede Lösung für einen festen Vektor $\lambda \in \mathbb{R}$ stellt eine gültige untere Schranke

für *MDVSP* dar.

$$LR_{cc}(\lambda) : \quad \min \sum_{(i,j)^d \in A} c_{ij}^d x_{ij}^d + \sum_{t \in T} \lambda_t (1 - \sum_{d \in D} x_t^d) \quad (7.10)$$

$$s.t. \quad \sum_{i:(i,j)^d \in A} x_{ij}^d - \sum_{i:(j,i)^d \in A} x_{ji}^d = 0 \quad \forall j \in T, \forall d \in D \quad (7.11)$$

$$x_d \leq w_d \quad \forall d \in D \quad (7.12)$$

Eine Schwierigkeit der Lagrange Relaxation ist die geeignete Wahl des Vektors λ . Beispielsweise ist die Lösung des Lagrange Problem für $\lambda = 0$ trivial, da keine negativen Kostenwerte existieren und die Lösung mit $x_{ij}^d = 0$ ($\forall (i,j) \in A$) gültig und optimal ist. Mit einem Zielfunktionswert von 0 führt diese Wahl für λ allerdings zu keiner guten unteren Schranke. Um die beste mögliche untere Schranke der Relaxation zu finden, kann das Lagrange Dual Problems LDP_{cc} gelöst werden. Durch die Eigenschaft der Unimodularität ist der Zielfunktionswert der Lösung gleich dem einer optimalen LP-Lösung (s. Abschnitt 3.3).

$$LDP_{cc} : \quad \max_{\lambda \in \mathbb{R}} LR_{cc}(\lambda)$$

Zur Lösung des Lagrange Dual Problems kann eine Subgradientenmethode eingesetzt werden, die durch ein iteratives Vorgehen versucht geeignete Werte für λ zu finden. In jeder Iteration der Subgradientenmethode muss ein Lagrange Problem mit fixiertem Vektor λ gelöst werden. Zur Erreichung einer optimalen Lösung sind in der Regel viele Iterationen notwendig (vgl. [Kokott und Löbel, 1996]), so dass diese Relaxation nicht zur schnellen Bestimmung einer unteren Schranke geeignet. Allerdings kann durch eine Reformulierung des mathematischen Modells eine alternative Lagrange Relaxation gebildet werden, die im Folgenden vorgestellt wird.

Relaxation der Depotflusserhaltung

Die Modellierung des MD-VSPs als Mehrgüter Flussmodell verwendet (neben möglichen Kapazitätsrestriktionen) zwei Typen von Restriktionen, um auf der Basis eines Netzwerkmodells die Gültigkeit eines Umlaufplans sicherzustellen. Durch Knotenrestriktionen wird die Flusserhaltung innerhalb des Netzwerks sichergestellt (es gehen keine Flusseinheiten verloren) und durch die Überdeckungsrestriktionen wird die Bedienung jeder Servicefahrt gewährleistet. Beispielhaft für eine Servicefahrt $t \in T$ und zwei Depots ($D = \{d1, d2\}$) sind im Netzwerkmodell für jedes Depot zwei Knoten gegeben, die die Abfahrt bzw. Ankunft der Servicefahrt repräsentieren. Ein Abfahrtsknoten (bzw. Ankunfts-knoten) einer Servicefahrt t in der Netzwerkschicht des Depots d wird mit vd_t^d (bzw. va_t^d) bezeichnet.

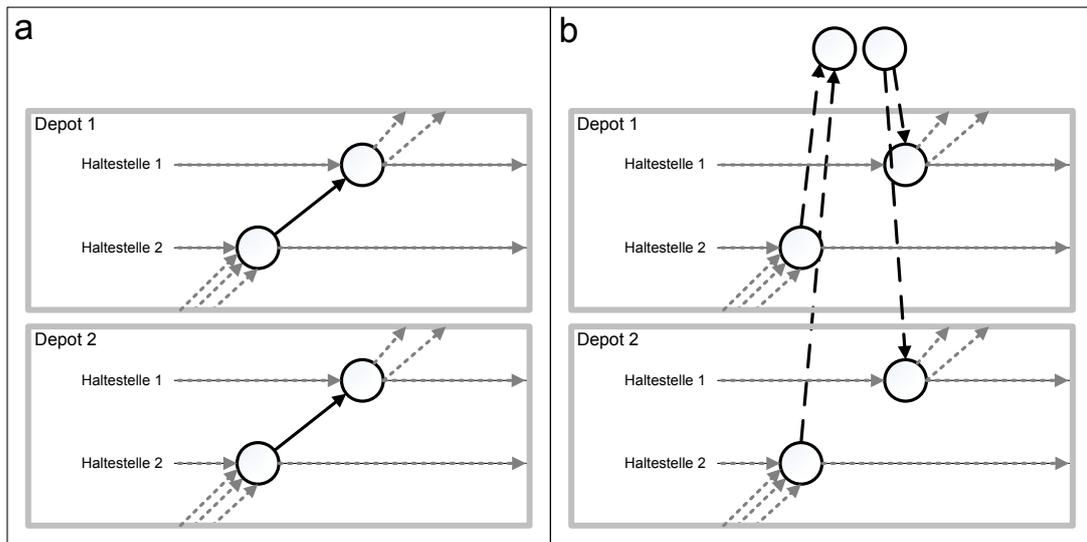


Abbildung 7.3: Alternative Netzwerkmodellierung von Servicefahrten

Zusätzlich existiert je eine Kante a_t^d pro Schicht, die die Bedienung der Servicefahrt t durch ein Fahrzeug des Depots d modelliert (s. Abbildung 8.1a). Unabhängig von den möglichen Verbindungs- und Depotfahrten, die vor oder nach dem hier beschriebenen Netzwerkausschnitt liegen können, wird diese Konstellation im Mehrgüter Flussmodell mit vier Restriktionen zur Erhaltung des Netzwerkflusses in den vier Knoten sowie einer Überdeckungsrestriktion für Fahrt t modelliert. An dieser Stelle soll angemerkt werden, dass die Netzwerkstruktur im Falle von Fahrzeugtypgruppen zwar nicht strukturgleich ist, da zum einen nicht jede Fahrt von jedem Depot aus bedient werden kann und zum anderen die Verbindungskanten in der aggregierten Time-Space-Netzwerk-basierten Modellierung auf unterschiedliche Weise aggregiert sein können. Die zuvor beschriebene Knotenkonstellation gilt allerdings auch für diesen Fall.

Eine alternative Modellierung kann durch zwei zusätzliche Knoten vd_t und va_t erreicht werden, die den Beginn und das Ende der Servicefahrt t darstellen. Statt Knoten in den Netzwerkschichten direkt durch eine Fahrtkante zu verbinden, modellieren die Kanten ad_t^d zwischen vd_t^d und vd_t ($\forall d \in D$) den Beginn der Bedienung von Fahrt t von Depot d . Entsprechend repräsentieren die Kanten aa_t^d zwischen va_t und va_t^d ($\forall d \in D$) das Ende der Bedienung der Fahrt t aus Depot d . Die Flussgröße auf einer Kante ad_t^d (bzw. aa_t^d) entspricht der Entscheidungsvariable x_{vt}^d (bzw. $x_{vt'}^d$). Die Kosten auf den neuen Fahrtkanten entsprechen jeweils den halben Kosten der ursprünglichen Fahrtkante. Die globalen Fahrtknoten vd_t bzw. va_t werden nicht wie die anderen Netzwerknoden als Umladeknoten modelliert, sondern haben einen Bedarf bzw. ein Angebot von einer Flusseinheit.

Dadurch kann sichergestellt werden, dass jede Servicefahrt von einem Umlauf begonnen wird und in einem Umlauf endet. Der Aufbau des Netzwerks ist für den beispielhaften Ausschnitt in Abbildung 8.1b illustriert.

Um für diese Modellierung die Lösbarkeit des MD-VSPs sicherzustellen, müssen neben den Knotenrestriktionen zur Flusserhaltung weitere Restriktionen zum mathematischen Modell hinzugefügt werden. Damit ein gültiger Umlauf aus der Bedienung einer Fahrt $t \in T$ entsteht, müssen der Beginn und die Weiterführung nach der Fahrt dem gleichen Depot zugewiesen sein. Da durch die Knotenbedarfe sichergestellt ist, dass genau ein Depot pro Fahrt ausgewählt wird, ist es je Fahrt ausreichend für alle Depots $d \in D$ die Gleichheit der Flüsse beider Fahrtrichten zu gewährleisten: $ad_t^d = aa_t^d$.

Seien N' und A' die Mengen der Knoten und Kanten des wie in oben beschriebener Weise veränderten Mehrgüter Flussnetzwerks. Sei $NTD' \subset N'$ die Menge der zusätzlichen Fahrtnoten vd_t und $NTA' \subset N'$ die Menge aller Knoten va_t ($\forall t \in T$). Enthalte zusätzlich die Menge $NT' \subset N'$ alle diese Fahrtnoten ($NT' = NTD' \cup NTA'$) und sei N^d wie zuvor die Menge der Knoten in Netzwerkschicht d (ohne die globalen Fahrtnoten). Das MD-VSP kann durch das mathematische Optimierungsmodell $MDVSP'$ definiert werden. Die optimale Lösung des neuen Modells entspricht einer optimalen Lösung der ursprünglichen Formulierung.

$$MDVSP' : \quad \min \sum_{(i,j)^d \in A'} c_{ij}^d x_{ij}^d \quad (7.13)$$

$$s.t. \quad \sum_{i:(i,j)^d \in A'} x_{ij}^d - \sum_{i:(j,i)^d \in A'} x_{ji}^d = 0 \quad \forall j \in N^d, \forall d \in D \quad (7.14)$$

$$\sum_{i:(i,j)^d \in A'} x_{ij}^d = 1 \quad \forall j \in NTD' \quad (7.15)$$

$$\sum_{i:(j,i)^d \in A'} x_{ji}^d = 1 \quad \forall j \in NTA' \quad (7.16)$$

$$x_{t'}^d = x_{t''}^d \quad \forall t \in T, \forall d \in D \quad (7.17)$$

$$x_d \leq w_d \quad \forall d \in D \quad (7.18)$$

$$x_{ij}^d \in \mathbb{N} \quad \forall (i,j)^d \in A' \quad (7.19)$$

In der Zielfunktion (7.13) werden, wie in der ausgehenden Formulierung, die Gesamtkosten über alle Netzwerkflüsse minimiert. Die Restriktionen zur Flusserhaltung (7.14) gelten für alle Knoten bis auf die neuen globalen Fahrtnoten. Diese müssen für den Fall, dass der Knoten den Beginn einer Servicefahrt repräsentiert, eine Flusseinheit als Zugang haben (Restriktionen 7.15) oder ein

Angebot von einer Flusseinheit bieten (Restriktionen 7.16). Durch (7.17) wird sichergestellt, dass dem Beginn und Ende einer Fahrt das gleiche Depot zugewiesen wird. Die Restriktionen für die Depotkapazitäten und die Ganzzahligkeit bleiben wie in der ausgehenden Formulierung erhalten.

Da die vorgestellte Reformulierung des Mehrgüter Flussmodells bei gleicher oder schlechterer theoretischen Problemhärte zu einem Optimierungsproblem mit deutlich mehr Variablen und Nebenbedingungen führt, verspricht diese Formulierung keine Verbesserung bezüglich einer exakten mathematischen Optimierung. Allerdings ist festzustellen, dass das Modell ohne die Restriktionen (7.17) ein großes Netzwerkflussmodell des SD-VSPs darstellt (s. Abschnitt 4.2.1), weil die einzelnen Schichten über die globalen Fahrtnoten miteinander verknüpft sind. Es bietet sich daher an diese Restriktionen zu relaxieren, woraus das Lagrange Problem LR_{df} resultiert. Der Vektor π enthält für alle $t \in T, d \in D$ die Lagrange-Multiplikatoren $\pi_{td} \in \mathbb{R}$ der relaxierten Restriktionen. Aufgrund der unimodularen Koeffizientenmatrix wird die Ganzzahligkeit der LP-Relaxation sichergestellt und Restriktionen (7.19) können entfallen.

$$LR_{df}(\pi) : \min \sum_{(i,j)^d \in A'} c_{ij}^d x_{ij}^d - \sum_{d \in D} \sum_{t \in T} \pi_{td} (x_{t'}^d - x_{t''}^d) \quad (7.20)$$

$$s.t. \quad \sum_{i:(i,j)^d \in A'} x_{ij}^d - \sum_{i:(j,i)^d \in A'} x_{ji}^d = 0 \quad \forall j \in N^d, \forall d \in D \quad (7.21)$$

$$\sum_{i:(i,j)^d \in A'} x_{ij}^d = 1 \quad \forall j \in NTD' \quad (7.22)$$

$$\sum_{i:(j,i)^d \in A'} x_{ji}^d = 1 \quad \forall j \in NTA' \quad (7.23)$$

$$x_d \leq w_d \quad \forall d \in D \quad (7.24)$$

Im Falle der Umlaufplanung mit mehreren Depotstandorten und einem Fahrzeugtypen entspricht eine gültige Lösung für $LR_{df}(\pi)$ einem Umlaufplan, bei dem die Fahrzeuge während einer Servicefahrt ihre Zuordnung zu einem Depotstandort wechseln können. Dadurch ist nicht mehr gewährleistet, dass die Fahrzeuge in das gleiche Depot zurückkehren von dem aus sie ihren Umlauf gestartet haben. Diese Formulierung ist daher eine Möglichkeit die Umlaufplanung mit mehreren Depots in polynomieller Zeit (als SD-VSP) zu optimieren, sofern das Verkehrsunternehmen auf die Restriktion verzichten kann, dass alle Fahrzeuge in ihr Startdepot zurückkehren müssen. Im Fall mit mehreren Fahrzeugtypen entspricht eine Lösung dieser Relaxation einem Umlaufplan, bei dem die Fahrzeuge neben der Zuordnung des Depotstandortes auch ihren Fahrzeugtypen ändern können.

Die Lagrange Relaxation liefert untere Schranken für die ursprüngliche Formulierung *MDVSP*. Da auch LR_{df} eine unimodulare Entscheidungsmatrix hat, ist die beste untere Schranke gleich der optimalen LP-Schranke. Das zugehörige Lagrange Dual Problem lautet:

$$LDP_{df} : \quad \max_{\pi \in \mathbb{R}} LR_{df}(\pi)$$

Auch in diesem Fall ist die Lösung des Lagrange Dual Problems zum Beispiel mit einem Subgradientenverfahren möglich. Aufgrund der veränderten Formulierung stellt allerdings bereits die initiale Lösung mit den Lagrange-Multiplikatoren $\pi_{td} = 0$ eine gute untere Schranke dar, da fixe und operative Kosten für jedes benötigte Fahrzeug in den Zielfunktionswert mit einfließen. In [Kokott und Löbel, 1996] wird eine vergleichbare Relaxation für eine verbindungs-basierte Modellierung gezeigt, bei der eine initiale Lösung in kurzer Laufzeit eine gute untere Schranke liefert.

7.2.3 Warmstart der Simplexmethoden

Untere Schranken können neben der Abschätzung der optimalen Lösung auch zur Unterstützung von Optimierungsmethoden eingesetzt werden. Zum Beispiel wird in Branch&Bound Algorithmen die Differenz von bekannten unteren und oberen Schranken dafür verwendet, Teilbäume der Lösungssuche abzuschneiden. In diesem Abschnitt soll eine spezielle Technik diskutiert werden, mit der die Lösung von LP-Relaxationen für *MDVSP* beschleunigt werden kann.

Zur Lösung von LP verfügen Optimierungsbibliotheken in der Regel über Simplexverfahren und Innere-Punkte-Verfahren. Die Simplex-Verfahren optimieren ein LP durch die iterative Suche nach besseren Basislösungen des Lösungsraums bis bewiesen wird, dass keine bessere Basislösung existiert. Unterschieden werden der primale Simplex, der primal gültige Basislösungen betrachtet und iterativ versucht die Optimalitätskriterien zu erfüllen, und der duale Simplex, der dual gültige Basislösungen betrachtet bis die Lösung gültig ist (s. Abschnitt 3.1). Da die Algorithmen generisch für jegliche Art von linearen Optimierungsproblemen geeignet sind, werden keine problemspezifischen Informationen in die Lösungssuche mit einbezogen. Einige Implementierungen versuchen über so genannte Crash-Verfahren gute Basislösungen zu identifizieren, so dass der Algorithmus schneller konvergiert (vgl. [Bixby, 1992]). Fortschrittliche Optimierungssysteme wie zum Beispiel ILOG CPLEX bieten auch die Möglichkeit Basislösungen oder Startwerte für die Suche zu übergeben (vgl. [ILOG, 2007]). Die im vorigen Abschnitt vorgestellte Lagrange-Relaxation bietet eine Lösung, die bei relaxierten

Restriktionen dual gültig ist und zusätzlich eine gute untere Schranke liefert. Daher bietet sich die Übergabe dieser Lösung als Startwert für den dualen Simplex Algorithmus an.

Da eine optimale Lösung des Lagrange Problems $LR_{df}(\pi)$ keine dual gültige Lösung für das ursprüngliche Modell darstellt, müssen die Startwerte für die Variablen der Originalformulierung aus der optimalen Lösung übertragen werden. Für die in beiden Formulierungen verwendeten Entscheidungsvariablen der Kanten, die keine Servicefahrten repräsentieren, können die Lösungswerte des Lagrange Problems als Startwerte für den dualen Simplex übergeben werden, da die Modelle für diese Kanten eine identische Struktur aufweisen. Zur Übergabe der Startwerte für die Entscheidungsvariablen der Fahrtkanten wird der Mittelwert der Lösungswerte von den beiden reformulierten Fahrtkanten verwendet. Da die Lösung von $LR_{df}(\pi)$ ganzzahlig ist, existiert je Fahrt eine Kante für den Anfang und das Ende der Fahrt. Daher können zwei Fälle auftreten:

1. Beide Variablen gehören zu derselben Netzwerkschicht: die Variable für die originale Kante wird für diese Schicht auf 1 gesetzt, alle anderen auf 0.
2. Durch Verletzung der relaxierten Restriktionen gehören die zwei Variablen unterschiedlichen Schichten an: die Variablen für die originalen Kanten werden für beide Schichten auf 0,5 gesetzt, alle anderen auf 0.

Das Vorgehen stellt ein problemspezifisches Crash-Verfahren dar und die ermittelten Werte können für einen Warmstart der dualen Simplexmethode verwendet werden.

Experimente unter Verwendung des dualen Simplex Algorithmus der Standardoptimierungssoftware CPLEX in Version 11 zeigen, dass mit Hilfe des beschriebenen Warmstarts in kurzer Zeit eine duale Basislösung mit dem Zielfunktionswert der unteren Schranke gefunden werden kann. Allerdings konvergiert der duale Simplex Algorithmus für alle Instanzen nur sehr langsam gegen die optimale Lösung, so dass die Lösungszeit mit Hilfe des Warmstarts im Vergleich zur ursprünglichen Optimierung größer oder nur wenig geringer ist. Zum Beispiel konnte für die Instanz *real_5*, deren LP-Lösungszeit mit dem dualen Simplex 10812 Sekunden beträgt, innerhalb von nur 18 Sekunden eine untere Schranke gefunden werden, die fahrzeugoptimal ist und bezüglich der operativen Kosten lediglich um 0,27% von der optimalen Lösung abweicht. Es konnte bei Übergabe der Startwerte direkt eine dual gültige Basis identifiziert werden. Allerdings benötigt der Algorithmus durch die langsame Konvergenz insgesamt 10552 Sekunden auf dem gleichen System bis die optimale LP-Lösung bewiesen wurde. Ein Vergleich des Konvergenzverhaltens abhängig von der Iterationen des dualen Simplex ist in Abbildung 7.4 gegeben.

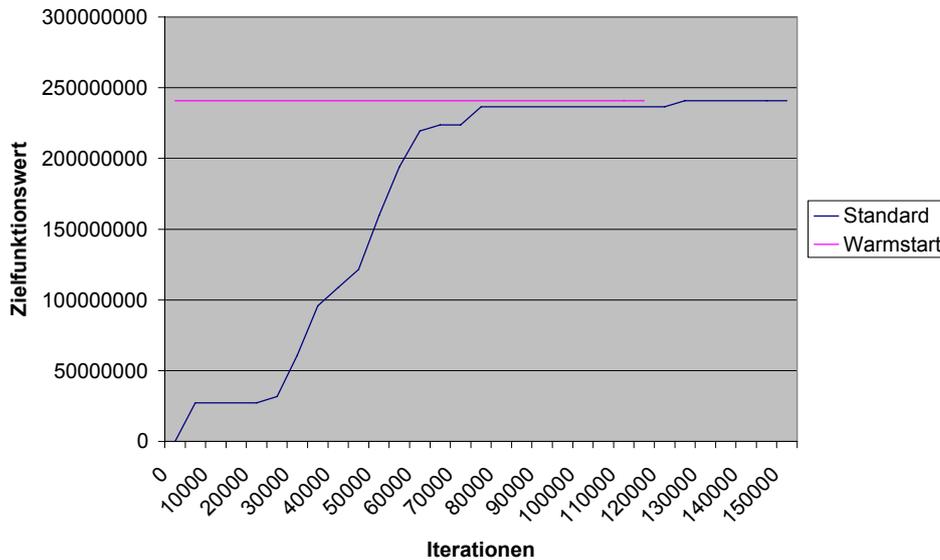


Abbildung 7.4: Konvergenzverhalten bei Warmstart von dualer Basis

Die beschriebene Technik kann für die Lösung von realen Umlaufplanungsproblemen mit Hilfe des dualen Simplex-Algorithmus eine geringe Verbesserung der Lösungszeit ermöglichen. Allerdings kann sie ausschließlich in Verbindung mit dem dualen Simplex eingesetzt werden, der (wie in Abschnitt 7.5 gezeigt wird) gerade für die Lösung von großen Problemstellungen nicht geeignet ist. Da der Warmstart des dualen Simplex zudem bedeutet, dass der weitere LP-Optimierungsprozess nicht zusätzlich gesteuert werden kann (Black-Box-Verhalten), wird dieser Ansatz nicht weiter betrachtet, wobei nicht ausgeschlossen ist, dass eine andere Crash-Heuristik besser für einen Warmstart geeignet ist. Da durch eine Reihe von Experimenten allerdings keine Verbesserung des Verhaltens erreicht wurde, wird im folgenden Abschnitt stattdessen ein alternatives Verfahren zur Lösung des MD-VSPs vorgestellt.

7.3 Lösungen durch Kantengenerierung

Die in Abschnitt 7.1 und 7.2 vorgestellten Verfahren zielen auf die schnelle Berechnung von guten heuristischen Lösungen und unteren Schranken ab. Sie sind auf eine schnelle Abschätzung der optimalen Lösung ausgerichtet und bilden die in der Initialisierungsphase benötigten Algorithmen (s. Abbildung 7.1). In diesem Abschnitt wird gezeigt, wie unter Verwendung der in der Initialisierung ermittelten Lösungen die Verbesserungsphase so gestaltet werden kann, dass für die

operative Planung ohne Zeitfenster bessere und ggf. optimale Lösungen berechnet werden können.

Wie in Abschnitt 6.2.3 gezeigt wurde, weisen die in dieser Arbeit betrachteten realen Problemstellungen für die Busumlaufplanung besondere Eigenschaften auf. Die Modellgröße der zu lösenden Problemstellungen wächst stark mit dem Umfang des Fahrplans (s. Abschnitt 6.2.2). Zusätzlich sind die Formulierungen stark degeneriert, so dass die lineare Relaxation schwierig zu lösen ist (s. Abschnitt 6.2.2). Durch die besondere Eigenschaft, dass die LP-Lösungen nur wenig fraktionale Variablenwerte aufweisen, ist die Berechnung einer optimalen ganzzahligen Lösung vielfach mit geringem Laufzeitaufwand möglich. Aufgrund des konträren Verhaltens der beiden Phasen bezüglich ihrer theoretischen Komplexität, ist das im Folgenden beschriebene Verfahren auf die Lösung des LPs fokussiert (s. Abschnitt 6.2.3).

Das Verfahren verwendet das Prinzip des Column Generation, dessen Funktionsweise in Abschnitt 3.2 beschrieben ist. Das MD-VSP ist als aggregiertes TSN-basiertes Mehrgüter Flussmodell aufgebaut, in dem für jede Kante eine Variable im mathematischen Modell definiert wird. Das Column Generation Prinzip betrachtet in jedem RMP eine Untermenge aller Kanten und löst die LP-Relaxation des MD-VSPs für diese Untermenge. Daher werden in jedem RMP eine Vielzahl von Verbindungsmöglichkeiten, Depotfahrten und die Fahrtenbedienung von einzelnen Depots nicht berücksichtigt. Durch das Generieren von neuen Spalten auf Basis der dualen Werte des RMP werden die Kanten in das Netzwerk aufgenommen, die zur Verbesserung der Lösung beitragen können. Durch das Verfahren werden genau die Verbindungen und möglichen Fahrten identifiziert, die für eine kostenoptimale Bedienung hinsichtlich der LP-Lösung notwendig sind. Da die Kanten nicht zur Laufzeit des Algorithmus generiert werden müssen, sondern das Netzwerk vollständig zu Beginn der Optimierung aufgebaut und gespeichert werden kann, handelt es sich um die Form des expliziten Column Generation. Dieser spezielle Einsatz des Column Generation wird im Folgenden *Kantengenerierung* genannt.

Neben der effizienteren Lösung der LP-Relaxation bietet sich die Kantengenerierung zur Einbeziehung in eine Methode für die operative Planung ohne Zeitfenster an, da sie auf die Berücksichtigung weiterer Planungsanforderungen (z.B. der Generierung von Lösungen zur Laufzeit) ausgerichtet werden kann. Die besondere Eigenschaft der betrachteten Modelle, dass die LP-Lösung nahezu ganzzahlig ist, kann in der Kantengenerierung genutzt werden. Die LP-Lösungen der RMP stellen in jeder Iteration Umlaufplanungsprobleme dar, bei denen nicht alle Freiheitsgrade betrachtet werden. Allerdings können zu den optimalen LP-Lösungen aufgrund der Fast-Ganzzahligkeitseigenschaft gültige MIP-Lösungen

gefunden werden, die eine heuristische Lösung für das Gesamtproblem darstellen. Diese Lösungen können sobald sie gefunden werden an den Planer ausgegeben werden, so dass zur Laufzeit immer bessere Lösungen angeboten werden können. Durch die Verwendung des TSN-basierten Modells können durch unterschiedliche Dekompositionsstrategien eine Vielzahl an Umlaufplänen mit identischen Kosten ausgegeben werden. Des Weiteren bietet das Verfahren im Gegensatz zu Standardoptimierungsmethoden auch bei einem Abbruch der Methode eine gültige Lösung. Die spezielle Optimierungsmethode in Verbindung mit der Ausnutzung der modellspezifischen Eigenschaften (z.B. der Fast-Ganzzahligkeitseigenschaft) zur Vermeidung eines Black-Box-Verhaltens kann den Planer effektiv bei seiner Arbeit unterstützen.

Das Kantengenerierungsverfahren wird im Folgenden beschrieben. Dazu ist eine Übersicht des Ablaufs in Algorithmus 7.1 dargestellt. Die Formulierungen entsprechen den in Abschnitt 3.2 gegebenen Definitionen. Weitere Details zu einzelnen Schritten sowie Beschleunigungstechniken werden in separaten Abschnitten diskutiert.

In *Schritt 1* des Algorithmus wird das Verfahren initialisiert. Dazu werden die zuvor beschriebenen Verfahren zur Bestimmung einer gültigen und fahrzeugoptimalen Lösung (s. Abschnitt 7.1) sowie einer unteren Schranke (s. Abschnitt 7.2) ausgeführt. Anschließend wird eine initiale Kantenmenge \tilde{A}^0 bestimmt. Mit diesen Kanten wird das erste RMP aufgebaut. Um in Schritt 2 duale Werte für die Modellrestriktionen zu erhalten, muss in der Initialisierung sichergestellt werden, dass RMP_0 gültig ist. Dafür müssen die in \tilde{A}^0 enthaltenen Kanten eine Flusslösung ermöglichen, die einen gültigen Umlaufplan darstellt.

In *Schritt 2* wird die LP-Relaxation des in der aktuellen Iteration betrachteten Master Problems ($RMP(A^t)$) optimal gelöst. Aus der optimalen LP-Lösung lassen sich die dualen Werte π^t der Restriktionen ablesen. Diese werden für die späteren Schritte gespeichert. Der Zielfunktionswert der Lösung stellt keine gültige untere Schranke für das ganzzahlige Modell dar, da nicht alle Freiheitsgrade (Kanten) berücksichtigt werden. Aufgrund der relaxierten Ganzzahligkeitsbedingungen stellt die Lösung auch keine gültige obere Schranke dar.

Um aus der LP-Lösung von $RMP(\tilde{A}^t)$ eine gültige Lösung zu generieren, wird in *Schritt 3* eine Heuristik verwendet, die – motiviert durch die Fast-Ganzzahligkeitseigenschaft der Modelle (s. Abschnitt 6.2.3) – versucht, eine gültige Lösung zu generieren. Die gefundene Flusslösung kann zur Laufzeit dekomponiert werden und die resultierenden Umlaufpläne ausgegeben werden.

In *Schritt 4* werden auf Basis der in Schritt 2 identifizierten dualen Werte für die Restriktionen die reduzierten Kosten \bar{c}_x für alle Kanten $x \in A$ berechnet. Aufgrund des expliziten Column Generation werden die reduzierten Kosten für

Algorithmus 7.1 : Kantengenerierungsalgorithmus

- (Schritt 1) **Initialisierung**
Berechne gültige Lösung UB_{best}
Berechne untere Schranke LB_{best}
Identifiziere initiale Kantenmenge $\tilde{A}^0 \subset A$
Setze $t = 0$
- (Schritt 2) **Master Problem**
Löse LP-Relaxation des $RMP(\tilde{A}^t)$
Speichere duale Werte π^t
- (Schritt 3) **IP Heuristik**
Versuche gültige IP-Lösung UB^t zu bestimmen
Setze ggf. $UB_{best} = UB^t$
- (Schritt 4) **Pricing Problem**
Löse $PP(\pi^t)$ und wähle Kanten zur Aufnahme in \tilde{A}^t
- (Schritt 5) **Abbruchkriterien**
Überprüfe Abbruchkriterien
Wenn Kriterium erfüllt, dann gehe zu Schritt 7
- (Schritt 6) **Spaltenmanagement**
Setze $t = t + 1$
Setze $\tilde{A}^t = \tilde{A}^{t-1}$
Eliminiere Kanten aus \tilde{A}^t
Füge in Schritt 4 identifizierte Kanten zu \tilde{A}^t hinzu
- (Schritt 7) **IP Phase**
Optimiere $RMP(\tilde{A}^t)$ als MIP
-

alle Kanten berechnet. Eine Kante $x \in A$, die negative reduzierten Kosten hat ($\bar{c}_x < 0$) und nicht in der aktuellen Kantenmenge des RMP ist ($x \notin \tilde{A}^t$), kann den Zielfunktionswert der LP-Relaxation verbessern und ist somit ein Kandidat zur Aufnahme in die aktuelle Kantenmenge \tilde{A}^t . Auf Basis dieser Informationen werden eine oder mehrere Kanten zur Aufnahme in das nächste RMP identifiziert.

Wenn im Pricing Problem keine Kante negative reduzierte Kosten hat, ist die optimale LP-Lösung des aktuellen Master Problems $RMP(\tilde{A}^t)$ gleichzeitig eine optimale LP-Lösung für das Modell mit allen Kanten $RMP(A)$. Neben diesem Abbruchkriterium können weitere definiert werden, die in *Schritt 5* des Kantengenerierungsalgorithmus geprüft werden. Zum Beispiel kann das Verfahren nach einer definierten Anzahl von durchgeführten Iterationen oder einer vorgegebenen maximalen Laufzeit abgebrochen werden. Wenn ein Abbruchkriterium erfüllt ist, wird der Algorithmus bei Schritt 7 weitergeführt und das letzte Master Problem wird als MIP gelöst. Andernfalls beginnt die nächste Iteration ($t = t + 1$).

Zur Vorbereitung der nächsten Iteration wird in *Schritt 6* ein Spaltenmanagement durchgeführt. Die Kantenmenge des letzten Master Problems wird für die Iteration übernommen ($\tilde{A}^t = \tilde{A}$) und um die in Schritt 4 identifizierten Kanten erweitert. Um die Dimension des zu lösenden Master Problems zu begrenzen, werden zusätzlich Kanten identifiziert, die aus der Kantenmenge entfernt werden. Dadurch wird sichergestellt, dass die zur Lösung des Master Problems erforderliche Lösungszeit nicht stark ansteigt.

Bei Abbruch des Kantengenerierungsverfahrens in Schritt 5 wird die optimale LP-Lösung von $RMP(\tilde{A}^t)$ verwendet, um in *Schritt 7* mit einem Branch&Cut Algorithmus eine ganzzahlige Lösung für die Kantenmenge \tilde{A}^t zu bestimmen. Im Folgenden werden weitere Details zu einzelnen Schritten des Kantengenerierungsalgorithmus gegeben.

7.3.1 Initialisierung

Für die Bestimmung der initialen Kantenmenge \tilde{A}^0 können verschiedene Verfahren in Betracht gezogen werden, die sich in Bezug auf Laufzeit, Lösungsqualität und Lösungseigenschaften unterscheiden. In Hinsicht auf die korrekte Funktionsweise des Kantengenerierungsalgorithmus ist eine Anforderung an die Identifikation, dass über die Kantenmenge eine gültige Flusslösung für das Modell existiert. Als sekundäre Anforderungen für das Verfahren lassen sich eine möglichst kurze Laufzeit und eine gute Lösungsqualität ableiten, da der Kantengenerierungsprozess durch diese Eigenschaften am besten unterstützt wird. Im Folgenden sollen mögliche Strategien zur initialen Kantenidentifizierung kurz vorgestellt und hinsichtlich ihrer Eignung diskutiert werden.

1. **Einfügen von künstlichen Kanten (Ini_k):** Eine Lösung der Mehrgüter Flussformulierung mit allen Variablenwerten gleich 0 verletzt die Überdeckungsrestriktionen des Modells. Um das Kantengenerierungsverfahren zu beginnen, kann das RMP daher für alle $t \in T$ um eine künstliche Variable κ_t erweitert werden, die als eine fehlende Bedienung der Servicefahrt interpretiert werden kann. Die Variable enthält in der Koeffizientenmatrix eine 1 in der Überdeckungsrestriktion von Fahrt t . Alle anderen Einträge sind gleich 0. Die Variablen ermöglichen daher ohne die Berücksichtigung weiterer Kanten eine gültige Lösung. Die Kosten für diese künstlichen Variablen werden so hoch gesetzt, dass sie alle anderen Kosten klar dominieren. Dadurch entstehen sehr große duale Werte und das Verfahren kann im Laufe der folgenden Iterationen Kanten generieren, die eine Verwendung der künstlichen in einer Lösung des RMP überflüssig machen. Die Strategie verursacht in der Initialisierungsphase keinen Laufzeitaufwand und stellt die Lösbarkeit des initialen RMP sicher.
2. **Verwendung von Kanten aus heuristischer Flusslösung (Ini_h):** Bei dieser Strategie wird zur Berechnung einer guten Lösung die in Abschnitt 7.1 vorgestellte Heuristik ausgeführt. Anschließend wird die initiale Kantenmenge aus den Kanten gebildet, die in der heuristischen Lösung verwendet werden. Das initiale RMP enthält daher bereits eine sehr gute (fahrzeugoptimale) Lösung und das Verfahren kann ausgehend von der Lösung nach weiteren Verbesserungen suchen. Diese Strategie erfordert einen geringen Laufzeitaufwand zur Durchführung der Heuristik, stellt die Lösbarkeit des Modells sicher und gibt dem Verfahren bereits eine gute Startlösung vor.
3. **Verwendung von Kanten aus heuristischer Flusslösung und Lagrange Lösungen (Ini_{hlg}):** Bei der zuvor vorgestellten Strategie fließt das Wissen aus der heuristischen Lösung in die Auswahl der Kanten ein. Diese Strategie kann dahingehend erweitert werden, dass neben Kanten einer heuristischen Lösung auch Kanten hinzugefügt werden, die in einer dual gültigen Lösung verwendet werden. Diese Kombination aus primal und dual motivierten Kanten kann zu einer besseren Konvergenz der Kantengenerierung führen. Der in Abschnitt 7.2.2 beschriebene Algorithmus zur Bestimmung einer unteren Schranke wird ausgeführt und die in der Lösung aktivierten Kanten werden mit in die initiale Kantenmenge für das erste RMP aufgenommen. Diese Strategie erfordert durch die zusätzliche Bestimmung der unteren Schranke mehr Laufzeitaufwand, aber verspricht aufgrund der zusätzlichen Kanten eine schnellere Konvergenz des Kantengenerierungsalgorithmus.

Numerische Ergebnisse für die oben vorgestellten Strategien werden in Abschnitt 7.5 gegeben. Neben den aufgeführten Strategien kann flexibel jedes heuristische Verfahren, das eine gültige Lösung für die Problemstellung bestimmt, verwendet werden. Dies stellt einen wichtigen Aspekt für die Integration der in diesem Kapitel vorgestellten Optimierungsmethodik in einen betrieblichen Planungsprozess dar, da bereits verwendete Methoden oder manuelle Lösungen als Startlösung eingebracht werden können. Das Verfahren kann dadurch mit der bereits bekannten Lösung beginnen und im Sinne der Zielfunktion bessere Lösungen vorschlagen. Damit kann die Akzeptanz von Planern zur Migration zu einem Entscheidungsunterstützungssystem mit der vorgestellten Methodik gefördert werden.

7.3.2 Master Problem

In jeder Iteration des Kantengenerierungsalgorithmus wird ein Mehrgüter Flussmodell mit einer Teilmenge der im aggregierten Time-Space-Netzwerk vorhandenen Kanten gelöst. Die Komplexität sowie seine Eigenschaften sind in Kapitel 6 detailliert beschrieben.

Um das RMP in einer Iteration zu lösen, bieten sich eine Vielzahl von möglichen Lösungsmethoden an. So werden zur Generierung der dualen Werte neben LP-Methoden auch Subgradienten- oder Bundle-Methoden eingesetzt (vgl. zum Beispiel [Borndörfer et al., 2004] und [Steinzen, 2008]). In der im Rahmen dieser Arbeit entwickelten Methode wird das Master Problem allerdings nicht nur zur Gewinnung von dualen Werten ausgeführt, sondern auch zur Bestimmung von ganzzahligen Lösungen verwendet. Dies ist durch die Fast-Ganzzahligkeitseigenschaft der LP-Lösungen möglich (s. Abschnitt 6.2.3) und erfordert primale Informationen, die LP-Methoden bereitstellen. Andere Verfahren zur Bestimmung von dualen Werten sind aus diesem Grund nicht für den Einsatz in der im Rahmen dieser Arbeit entwickelten Methodik geeignet.

Die Lösungsmethoden zur Lösung von LPs, Simplex- und Innere-Punkte-Verfahren, bringen für die Lösung des RMP unterschiedliche Vor- und Nachteile. Da in jeder Iteration nur ein kleiner Teil aller Modellvariablen angepasst wird, bleibt die Variablenmenge des Master Problems zu einem Großteil erhalten. Aus diesem Grund werden in vielen LP-basierten Column Generation Ansätzen in Anwendungsfeldern des Operations Research Simplex-Methoden zur Optimierung des RMP eingesetzt (vgl. [Desaulniers et al., 2005]). Diese haben den Vorteil, dass bei Veränderung der Spaltenmenge des mathematischen Modells nicht von neuem optimiert werden muss, sondern von der optimalen Basis der letzten Iteration aus gestartet werden kann. Diese Reoptimierung führt zu einer Beschleunigung

der Optimierung. Allerdings kann dieser Vorteil nur ausgenutzt werden, wenn das gesamte RMP gelöst wird und nicht durch ein Preprocessing verändert wird, in dem Variablen entfernt oder angepasst werden.

Der alternative Einsatz eines Innere-Punkte-Verfahrens zur Lösung des RMP wird in den meisten Anwendungen nicht betrachtet, da die optimale Lösung der vorigen Iteration nicht zur Reoptimierung verwendet werden kann und stattdessen jedes RMP neu optimiert werden muss. Für die Lösung des hier betrachteten RMP hat das Verfahren allerdings einige Vorteile. Im Gegensatz zu Simplex-Verfahren ist ein Innere-Punkte-Verfahren nicht so stark von dem Grad der Degeneration abhängig. Darüber hinaus können in Verbindung mit einer erneuten Ausführung in jeder Iteration die hochentwickelten LP-Preprocessing-Techniken aktueller Optimierungsbibliotheken genutzt werden, um die Optimierung des aktuellen RMP zu beschleunigen. Zudem ist die Lösung eines Simplex-Algorithmus eine Basislösung, weshalb die dualen Werte extreme Ausprägungen annehmen. Dieses Verhalten führt durch die sehr hohe Bewertung einzelner Restriktionen dazu, dass im Pricing Problem viele ähnliche Spalten bzw. Kanten generiert werden (vgl. [Desrosiers und Lübbecke, 2005]). Ein Innere-Punkte-Verfahren führt nach der Konvergenz gegen eine optimale LP-Lösung einen abschließenden Vorgang durch (Crossover), um eine optimale Basislösung zu bekommen. Die Verwendung von den dualen Werten ohne Ausführung des Crossover kann daher zu einer besseren Auswahl von Kanten und damit einer schnelleren Konvergenz des gesamten Algorithmus führen.

7.3.3 Pricing Problem

Im Pricing Problem werden basierend auf den dualen Werten des RMP die reduzierten Kosten für alle Kanten berechnet, die nicht in der aktuellen Kantenmenge vorhanden sind. Diese Berechnung kann sehr effizient durchgeführt werden, da jede Kante maximal drei Spalteneinträge ungleich 0 aufweist. Jede Kante enthält jeweils einen Eintrag mit dem Koeffizienten -1 (bzw. 1) für die Restriktion ihres Startknotens (bzw. Endknotens). Zusätzlich hat eine Servicefahrt-Kante noch einen Eintrag (1) in der zugehörigen Überdeckungsrestriktion und jede Rückfluss-Kante einen Eintrag (1) in einer Kapazitäts-Restriktion. Daher wird bei der Berechnung auf das Prüfen aller Zeileneinträge verzichtet und die Berechnung durch eine implizite Speicherung der dünn besetzten Entscheidungsmatrix beschleunigt.

Für das in Abschnitt 7.2.2 gegebene Modell (*MDVSP*) werden im Folgenden die dualen Werte definiert. Seien π_i^t in Iteration t die dualen Werte der Flussersparungsrestriktionen (7.6) für alle Knoten $i \in N$. Entsprechend seien ρ_s^t ($\forall s \in T$) und ω_d^t ($\forall d \in D$) die dualen Werte der Restriktionen (7.7) und (7.8). Für alle

in Iteration t nicht generierten Kanten werden die reduzierten Kosten berechnet:

$$\bar{c}_{ij}^t = c_{ij} + \pi_i^t - \pi_j^t \quad \forall (i, j) \in (A \setminus A^t) \quad (7.25)$$

Für alle Kanten (i, j) , die die Bedienung einer Servicefahrt $s \in T$ repräsentieren, werden zusätzlich die dualen Werte der Überdeckungsrestriktion ρ_s^t abgezogen. Entsprechend werden für Kanten, die den Rückfluss von Depot $d \in D$ repräsentieren, die dualen Werte der Kapazitätsrestriktion ω_d^t abgezogen.

Mit der expliziten Berechnung der reduzierten Kosten für alle (nicht generierten) Kanten, können eine oder mehrere Kanten ausgewählt werden, die in das RMP der nächsten Iteration aufgenommen werden. Theoretisch konvergiert der Algorithmus durch die Aufnahme der Kante mit den kleinsten (negativen) reduzierten Kosten gegen die optimale LP-Lösung. Dieses Vorgehen wird auch *Dantzig Pricing* (DP) genannt. Aufgrund der sehr großen Anzahl von Kanten und dem hohen Grad der Modelldeneration werden beim DP allerdings sehr viele Iterationen durchgeführt. Alternativ kann die Konvergenz durch die Verwendung von Pricing-Strategien, die mehrere Kanten identifizieren, beschleunigt werden. Die Auswahl von zu vielen Kanten führt zu einem so großen RMP, dass es nicht mehr effizient zu lösen ist. Eine Pricing-Strategie sollte daher einen geeigneten Trade-Off zwischen der von der Kantenanzahl abhängigen Laufzeit des RMPs und der Anzahl benötigter Iterationen finden. Im Folgenden werden mögliche Pricing-Strategien vorgestellt.

1. **Multiple Dantzig Pricing (MDP):** Bei der Strategie des *Multiple Dantzig Pricing* werden, statt der Auswahl der Kante mit den kleinsten reduzierten Kosten, mehrere Kanten ausgewählt. Dazu werden die im Pricing Problem betrachteten Kanten absteigend nach ihren reduzierten Kosten sortiert und die ersten x Kanten ausgewählt. Die Anzahl x wird dynamisch in Abhängigkeit von der Gesamtanzahl der Netzwerkkanten sowie der Größe des aktuellen RMP festgelegt. Dieser Strategie liegt die Annahme zugrunde, dass Variablen mit den geringsten reduzierten Kosten das größte Potential zur Verbesserung des RMP haben.
2. **Multiple Time Slot Pricing (MTSP):** Da die reduzierten Kosten abhängig von den dualen Werten aus dem RMP sind, können extreme Ausprägungen der dualen Werte dazu führen, dass viele ähnliche Kanten im MDP ausgewählt werden. Existiert zum Beispiel eine Knotenrestriktion mit einem sehr großen dualen Wert, dann werden die Kanten aus allen Netzwerkschichten ausgewählt, die ihn als Endknoten haben. Diese Auswahl kann dazu führen, dass vor allem die Möglichkeiten innerhalb eines kleinen

Zeitraums in das nächste RMP einbezogen werden. Da eine Verbesserung von Umläufen meist nicht lokal, sondern durch neue Kombinationen über den gesamten Tagesverlauf entsteht, wird durch das *Multiple Time Slot Pricing* eine ausgeglichene Verteilung der ausgewählten Kanten über den Planungszeitraum angestrebt. Der Planungstag wird in gleich große Zeitbereiche aufgeteilt (z.B. Stunden) und jede Kante dem Zeitbereich zugeteilt, innerhalb dessen die modellierte Aktion (z.B. Fahrt) stattfindet. Endet die Aktion in einem anderen Intervall als sie anfängt, entscheidet der größere Anteil über die Zuteilung. Für jeden Zeitbereich wird im Pricing Problem das MDP durchgeführt, wobei für jeden Bereich entsprechend weniger Kanten ausgewählt werden, so dass die Gesamtzahl der generierten Kanten in jeder Iteration der des MDP entspricht.

3. **Lagrangean Pricing** (LRP_{cc}/LRP_{df}): Innerhalb des Mehrgüter Flussmodells werden die möglichen Fahrten jeweils von einer Kante modelliert, die einer Variable im mathematischen Modell entspricht. Bei der Betrachtung der reduzierten Kosten von einzelnen Kanten werden daher nur lokale Verbesserungsmöglichkeiten innerhalb des Netzwerks untersucht. Da das Ziel des Kantengenerierungsalgorithmus die Verbesserung der Umlaufkosten ist und die Fahrten innerhalb eines Umlaufs voneinander abhängig sind, kann die gleichzeitige Betrachtung von mehreren Kanten vielversprechend sein. Im *Lagrangean Pricing* (vgl. [Löbel, 1998]) werden diese Kanten durch eine Lösung des Gesamtproblems identifiziert, bei dem den Kanten statt der ursprünglichen Kostenwerte die reduzierten Kosten der aktuellen Iteration zugewiesen werden. Durch die Minimierung der Zielfunktion werden Umläufe generiert, deren verwendete Kanten insgesamt die kleinsten negativen Kosten haben. Die Verwendung von Kanten in dieser Lösung stellt ein Indiz für die Eignung zur Verbesserung des RMP dar. Zur effizienten Durchführung dieser Optimierung wird das Problem durch die Relaxation von Restriktionen vereinfacht. Alle Kanten, die in einer Lösung des Hilfsproblems vorhanden sind und noch nicht im RMP vorhanden sind, werden zum aktuellen RMP hinzugefügt. Dabei können Kanten mit positiven Kosten ausgewählt werden, die in Kombination mit anderen Kanten insgesamt negative reduzierte Kosten haben.

Für die Lösung des beschriebenen Hilfsproblems können die in Abschnitt 7.19 vorgestellten Lagrange Relaxationen verwendet werden. Das Lagrangean Pricing mit Relaxation der Überdeckungsrestriktionen $LR_{cc}(\lambda)$ wird daher mit $c_{ij} = \bar{c}_{ij}^t$ und $\lambda = 0$ gelöst. Die gezeigte Umformulierung der Problemstellung in Verbindung mit der Relaxation der Restriktionen zur

Erhaltung der Depoteigenschaft $LR_{dc}(\pi)$ wird entsprechend mit $c_{ij} = \bar{c}_{ij}^t$ und $\pi = 0$ gelöst. Da die Kanten für die Servicefahrten in der Formulierung durch je zwei Kanten ersetzt wurden, werden die reduzierten Kosten der Servicefahrten zu gleichen Teilen auf diese Kanten aufgeteilt. Die Strategien werden im Folgenden LRP_{cc} und LRP_{df} bezeichnet.

Der in jeder Iteration anfallende Laufzeitaufwand für das Pricing ist abhängig von der Anzahl der Kanten, die nicht im RMP sind. Zum einen müssen für jede dieser Kanten die reduzierten Kosten berechnet werden und je nach Pricing-Strategie ein Optimierungsmodell mit diesen Kanten gelöst werden. Zur Beschleunigung des Pricing-Schritts sollte daher die Anzahl der zu betrachtenden Variablen reduziert werden.

Die Variablen der nicht im RMP vorhandenen Kanten entsprechen Nichtbasis-Variablen, die auf ihre untere Schranke (0) gesetzt sind. Um diese Variablen auf diesen Wert zu fixieren, so dass die entsprechende Kante nicht weiter betrachtet werden muss, kann das *Reduced Cost Fixing* (vgl. [Wolsey, 1998]) angewendet werden. Sei LB (bzw. UB) eine gültige untere (bzw. obere) Schranke für das gesamte Modell, dann gilt für das vorliegende Minimierungsproblem:

$$x_{ij} \leq \left\lfloor \frac{UB - LB}{\bar{c}_{ij}} \right\rfloor \quad \forall a \in (A \setminus A^t) \quad (7.26)$$

Um eine Variable auf 0 fixieren zu können, muss die rechte Seite der obigen Ungleichung gleich 0 sein. Da die rechte Seite abgerundet wird, reicht es aus, wenn sie echt kleiner 1 ist:

$$\frac{UB - LB}{\bar{c}_{ij}} < 1 \quad \Rightarrow \quad x_{ij} = 0 \quad (7.27)$$

Durch die Verfahren zur Bestimmung von unteren und oberen Schranken in der Initialisierungsphase sowie während der Kantengenerierung kann für jede Iteration der Gap ($UB - LB$) berechnet werden. Da im Pricing für jede Variable die reduzierten Kosten berechnet werden, kann die Prüfung ohne spürbaren Laufzeitaufwand für jede Variable vorgenommen werden. Eine Variable x_{ij} kann dauerhaft aus der Betrachtung genommen werden bzw. auf 0 fixiert werden, wenn die reduzierten Kosten den aktuellen Gap überschreiten: $\bar{c}_{ij} > (UB - LB)$.

7.3.4 Ganzzahlige Lösungen

In Schritt 2 des Kantengenerierungsalgorithmus von Seite 132 wird die LP-Relaxation mit einer Teilmenge der Netzwerk-Kanten des Mehrgüter Flussmodells

gelöst. Durch die besondere Eigenschaft in dieser Arbeit betrachteten Umlaufplanungsprobleme, dass in der Lösung des LP keine oder nur wenige Variablen fraktionale Werte aufweisen (s. Abschnitt 6.2.3), kann sowohl während der Kantengenerierung (Schritt 3) als auch zur abschließenden Lösung des MIP (Schritt 7) genutzt werden.

Während der Kantengenerierung können verschiedene Strategien zur Identifikation von gültigen Lösungen herangezogen werden. Sie unterscheiden sich nach der Intensität des Aufwands, der zur Suche nach einer Lösung aufgewendet wird. Je nachdem wie wichtig für einen Anwendungsfall die Ausgabe von heuristischen Lösungen während der Laufzeit ist, kann durch die Auswahl eines Verfahrens der Trade-Off zwischen Gesamtlaufzeit und gefundenen gültigen Lösungen gesteuert werden.

1. Im einfachsten Fall werden die Variablen der LP-Lösung auf ihre Ganzzahligkeit überprüft und für den Fall, dass keine fraktionellen Werte vorhanden sind, eine gültige Lösung gebildet. Durch diese Strategie entsteht ein minimaler Zusatzaufwand, da ggf. nicht nach gültigen Lösungen gesucht wird.
2. Als Erweiterung der ersten Strategie kann bei nicht gültigen Lösungen ein Rundungsverfahren ausgeführt werden, dass zur Bildung einer gültigen Lösung führen kann. Dazu werden die in der Lösung vorhandenen Variablenwerte absteigend nach ihrem Grad der Fraktionalität ξ_a der Kante a sortiert mit:

$$\xi_a = |[x_a] - x_a| \quad \forall a \in \tilde{A}^t \quad (7.28)$$

Die Variable mit der geringsten Fraktionalität ($\min_{a \in \tilde{A}^t} \xi_a$) wird auf den nächsten ganzzahligen Wert gerundet. Da durch die Veränderung die Restriktionen verletzt werden, an denen die Variable beteiligt ist, müssen alle anderen Variablen entsprechend angepasst werden und mit den in Folge der Anpassung ungültigen Restriktionen entsprechend verfahren werden. Anschließend wird der Grad der Fraktionalität für alle Variablen neu bewertet und diese Schritte solange wiederholt bis eine gültige Lösung entstanden ist oder aufgrund von Zeit- oder Ungültigkeit abgebrochen wird.

3. Eine alternative Strategie ist es, für die Suche nach einer gültigen Lösung auf Standardoptimierungsbibliotheken zurückzugreifen, die für die Suche nach gültigen Lösungen im ersten Knoten eines Branch&Bound oder eines Branch&Cut Algorithmus eine Reihe von Techniken implementiert haben. Diesem so genannten „Supernode Processing“ kann eine optimale LP-Lösung übergeben werden, so dass die Optimierungssoftware für eine vorgegebene Zeit nach gültigen Lösungen suchen kann. Da die Lösung des

RMP mit Algorithmen aus Standardoptimierungssoftware gelöst wird (s. Abschnitt 7.3.2), ist die optimale LP-Lösung direkt verfügbar. Ein weiterer Vorteil ist, dass diese Strategie durch die Angabe der verfügbaren Zeit flexibel skalierbar ist.

In Schritt 7 des Kantengenerierungsalgorithmus wird das mathematische Modell des letzten RMPs als MIP gelöst. Dabei wird ausgehend von der optimalen LP-Lösung auf B&B bzw. B&C Algorithmen aus Standardoptimierungssoftware zurückgegriffen. Numerische Tests sowie die Parametereinstellungen für die verwendeten Algorithmen sind in Abschnitt 7.5 beschrieben.

7.4 Systemintegration und Preprocessing

Für den Einsatz eines Optimierungsverfahrens wie des im vorigen Abschnitt vorgestellten Kantengenerierungsalgorithmus ist eine Integration der Methode in ein Planungssystem notwendig. Dieses System stellt unter anderem die Kommunikation zum Planer und die Verwaltung der Daten sicher. Der Optimierungsmethode werden die notwendigen Informationen übergeben, wobei das Datenmaterial in der Praxis häufig unvollständig, redundant oder für ein Optimierungsverfahren in ungeeigneter Weise zusammengestellt ist, so dass vorab eine Bereinigung um redundante oder nicht relevante Daten erfolgt.

Neben der geeigneten Aufbereitung der Daten können Techniken integriert werden, die zusätzliche Anforderungen des Planers berücksichtigen oder durch ein heuristisches Vorgehen die benötigte Optimierungslaufzeit senken. Diese Techniken können in Form eines „Preprocessing“ unabhängig von der Lösungsmethode eingesetzt werden. Das im Rahmen dieser Arbeit entwickelte Vorgehen basiert auf der Ausführung von unterschiedlichen Modulen, die unabhängig voneinander verwendet werden können. Durch eine vorgegebene Reihenfolge der Ausführung werden die Module so ineinander verschachtelt, dass sie ohne Kenntnis der in anderen Modulen durchgeführten Veränderungen die Korrektheit einer Lösung erhalten. Die Module sind dazu in verschiedenen Stufen angeordnet. Auf jeder können die Daten entsprechend der Module angepasst werden und an die nächste übergeben werden bis in der letzten Stufe die Umlafoptimierung auf Basis der resultierenden Informationen durchgeführt wird. Verschiedene Module können bewirken, dass die Problemstellung in mehrere voneinander unabhängige Problemstellungen aufgeteilt wird. Die nachfolgenden Stufen werden dann für jedes Teilproblem getrennt durchgeführt, so dass in der letzten Stufe eine Reihe von Optimierungsproblemen zu der ursprünglichen Problemstellung zu lösen ist. Nach

der Durchführung der Optimierung werden die Stufen in umgekehrter Reihenfolge durchlaufen und ggf. Anpassungen an den Ergebnissen getätigt, um die Konsistenz wiederherzustellen. Die Einbettung der Kantengenerierungsalgorithmus in dieses Verfahren ist in Abbildung 7.5 dargestellt. Durch die grau hinterlegten Module werden Techniken markiert, die zu einer Veränderung der Lösungsqualität führen (Heuristiken). Die weiß hinterlegten Module können die Daten verändern, aber erhalten dabei optimale Lösungen.

In der ersten Stufe werden die zur Durchführung einer Umlaufoptimierung notwendigen Daten aus einer Schnittstelle zum Entscheidungsunterstützungssystem ausgelesen. Diese beinhalten neben den in Kapitel 2 beschriebenen Informationen zu Fahrplan und Topologie auch die vom Benutzer einstellbaren Parameter zur Steuerung der Optimierung. Nach dem Einlesen werden die Daten hinsichtlich möglicher Redundanzen, überflüssiger Daten oder Unvollständigkeiten überprüft und entsprechend angepasst. Unter anderem werden nicht verwendete Haltestellen, Depots und Fahrzeugtypen gefiltert und die Verbindungsmatrix vervollständigt. Dazu wird für fehlende Verbindungsangaben ein kürzester-Wege-Algorithmus (vgl. [Ahuja et al., 1993]) über das Netzwerk der Haltestellen mit den vorhandenen Verbindungsangaben durchgeführt. Im letzten Modul dieser Stufe werden die Fahrzeugtypen entsprechend des in Abschnitt 7.1 vorgestellten Verfahrens zur Substitution von Fahrzeugtypen zusammengefasst. Das Verfahren kann unabhängig von der Heuristik zur Anpassung der gesamten Problemstellung eingesetzt werden, wobei je nach Anzahl der durchgeführten Phasen der Substitution eine optimale oder heuristische Lösung ermöglicht wird.

In der zweiten Stufe des „Preprocessing“ können Techniken zur Reduktion der numerischen Komplexität bei Planungsproblemen mit Zeitfenstern durchgeführt werden (vgl. [Bunte et al., 2005]). Zusätzlich kann in dieser Stufe eine Heuristik zur schnellen Fixierung von Linienanschlüssen ausgeführt werden. Das Verfahren basiert auf der Beobachtung, dass die Servicefahrten in realen Fahrplänen häufig in einer von der Linien abhängigen Taktung angelegt sind. In den Fällen beginnt eine neue Servicefahrt der gleichen Linie kurz nach der Ankunft der vorherigen. Die Heuristik identifiziert Anschlüsse, bei denen zwei Servicefahrten, die die gleiche Linie bedienen an der gleichen Haltestelle verknüpft werden können. Eine Fixierung dieser Anschlüsse führt zu einer geringeren numerischen Komplexität des zu lösenden VSPs. Außerdem entstehenden tendenziell Umläufe mit weniger Linienwechseln. Allerdings führt die Verknüpfung von Servicefahrten zu einer Verkleinerung des betrachteten Lösungsraums und damit zu einer heuristischen Lösung. Je kürzer die Zeit zwischen der Ankunft der ersten Fahrt und der Abfahrt der zweiten Fahrt ist, umso weniger Potential geht bei einer Fixierung dieses Anschlusses verloren. Die Entscheidung über eine Fixierung

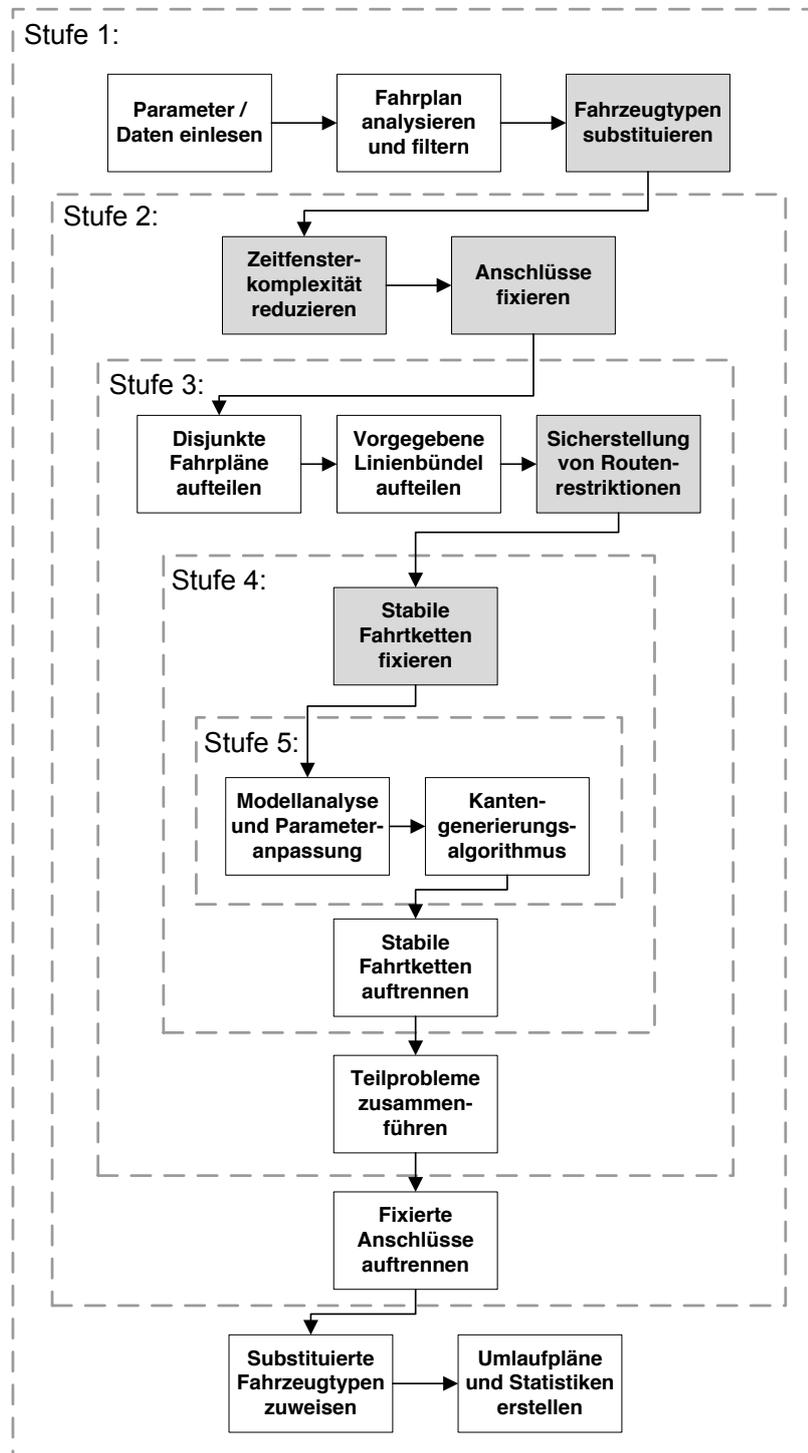


Abbildung 7.5: Einbettung des Kantengenerierungsalgorithmus

erfolgt daher über einen Parameter γ , der die akzeptierte Wartedauer zwischen den Anschlüssen definiert. Zwei Servicefahrten $a, b \in T$ werden fixiert, wenn sie

- die gleiche Linie bedienen,
- ohne Verbindungsfahrt verknüpft werden können ($h_a = g_b$) und
- innerhalb der akzeptierten Wartezeit stattfinden ($e_b - f_a \leq \gamma$).

Alle durch fixierte Anschlüsse verbundenen Servicefahrten werden für die folgenden Stufen als eine lange Fahrt modelliert. Durch Anpassung des Parameters γ kann ein Planer zwischen linienreinen Umläufen und Kosteneffizienz abwägen.

In Stufe 3 wird der bereinigte Fahrplan auf die Existenz von unabhängigen Teilproblemen hin überprüft. Wenn die Menge der Servicefahrten in zwei oder mehr Teilmengen aufteilbar ist und diese bezüglich Endhaltstellen, Verbindungsmöglichkeiten und Depots unabhängig voneinander sind, werden die Fahrpläne aufgeteilt und in den späteren Stufen getrennt voneinander betrachtet. Neben dieser Aufteilung können auch vom Planer vorgegebene Mengen von Linien (Linienbündel) getrennt voneinander optimiert werden, indem für jedes Linienbündel ein Fahrplan mit den Servicefahrten der angegebenen Linien erstellt wird. Zusätzlich kann in dieser Stufe die maximale Anzahl unterschiedlicher Linien pro Umlauf berücksichtigt werden, indem die Fahrpläne so aufgeteilt werden, dass keine Verletzung der Restriktion möglich ist. Die Begrenzung kann dadurch sichergestellt werden, dass in jedem Unterproblem maximal die vorgegebene Anzahl unterschiedlicher Linien vorhanden ist. An dieser Stelle sei angemerkt, dass durch dieses Vorgehen die Lösungsqualität drastisch verschlechtert werden kann, so dass dieses Modul nur eingesetzt werden sollte, wenn die Einhaltung der Routenrestriktion zwingend erforderlich ist.

In der vierten Stufe kann ein weiteres Modul zur Verringerung der zu betrachtenden Servicefahrten angewendet werden, in dem vereinfachte Problemstellungen zur Bildung von stabilen Fahrketten gelöst werden. Dabei werden die Anschlüsse zwischen zwei Servicefahrten identifiziert, die bei Lösung von unterschiedlichen vereinfachten Umlaufplanungen in allen Lösungen enthalten sind. Unter Annahme, dass diese Anschlüsse auch in einer guten oder optimalen Lösung vorhanden sind, werden sie fixiert und die resultierenden Fahrketten als jeweils einzelne lange Fahrt im Fahrplan betrachtet (vgl. [Gintner et al., 2005]).

In der fünften und letzten Stufe werden alle Problemstellungen, die in den vorhergehenden Stufen gebildet wurden, analysiert und mit individuell angepassten Parametern mit dem vorgestellten Verfahren gelöst. Durch die Aufteilung einer Instanz können sich Problemstellungen ergeben haben, die sehr unterschiedliche Charakteristika haben. So können Teilprobleme zum Beispiel als SD-VSP, für

die die Lösung der initialen Heuristik bereits eine optimale Lösung liefert. Durch die Anpassung der Parameter können die spezifischen Besonderheiten daher zur Steuerung der Lösungsmethode verwendet werden.

Alle vorgestellten Module sind unabhängig voneinander einsetzbar. Bei dem Übergang in eine andere Stufe des Preprocessings werden allerdings Änderungen an der originalen Problemstellung vorgenommen. Nach der Optimierung aller Unterprobleme müssen diese Änderungen in einem Postprocessing berücksichtigt werden, so dass eine gültige Lösung für die Originalformulierung gegeben werden kann. Die in Stufe 4 fixierten stabilen Fahrketten müssen dazu wieder in die ursprünglichen Servicefahrten und zusätzliche Verbindungsfahrten zwischen den Fahrten aufgespalten werden. Anschließend werden die Lösungen der in Stufe 3 gebildeten Teilprobleme in einer Lösung zusammengefasst. Entsprechend werden für alle Stufen die Anpassungen berücksichtigt, so dass ein gültiger Umlaufplan an das Entscheidungsunterstützungssystem zurückgegeben wird.

Die beschriebenen Schritte werden aufgrund der dynamischen Lösungsgenerierung im vorgestellten Lösungsverfahren nicht linear ausgeführt. Da während der Optimierung Informationen und gültige Lösungen an den Planer ausgegeben werden sollen, werden die nachgelagerten Schritte zur Herstellung der Konsistenz zur Laufzeit für jede gültige Lösung durchgeführt. Im Falle von mehreren zu optimierenden Unterproblemen, kann das Verfahren durch eine parallele Ausführung der Lösungsmethoden beschleunigt werden. Die Zusammenfassung und Ausgabe einer gültigen Lösung muss in diesem Fall von einer zentralen Instanz gesteuert werden. Die parallele Durchführung von Optimierungen wurde im Rahmen dieser Arbeit allerdings nicht implementiert, da die vorliegenden Problemstellungen ohne Verwendung einer heuristische Aufteilung keine Möglichkeit der Parallelisierung gezeigt haben.

7.5 Numerische Ergebnisse

Im folgenden Abschnitt werden die in den vorhergehenden Abschnitten vorgestellten Methoden validiert. Dazu wurden die beschriebenen Verfahren implementiert und anhand umfangreicher Tests auf den in Abschnitt 6.1.1 beschriebenen realen Probleminstanzen evaluiert. Der Aufbau des Abschnitts orientiert sich an der Gliederung der vorgestellten Methodik. Zur Bewertung der Lösungsqualität und des Laufzeitverhaltens, wurden die Ergebnisse mit der alternativen Optimierung unter Verwendung aktueller Standardoptimierungssoftware verglichen.

Die Implementierung der Methoden erfolgte in der Programmiersprache *C#* und verwendet die Klassen des *Microsoft .NET Frameworks* in der Version 2.0.

Um eine hochgradig effiziente Implementierung gewährleisten zu können, wurde die Verwendung dieses Frameworks auf die grundlegenden Datenstrukturen sowie generische Listen beschränkt. Alle im Folgenden vorgestellten numerischen Tests wurden auf einem Personal Computer (PC) durchgeführt, der mit einem *Intel Core 2 Duo*-Prozessor mit zwei auf 2,4 GHz getakteten Prozessorkernen und 8 Gigabyte Arbeitsspeicher ausgestattet war. Die Tests wurden unter dem 64-bit Betriebssystem *Microsoft Windows XP Professional x64 Edition* durchgeführt.

Alle Tests wurden auf der Basis des Mehrgüter Flussmodells durchgeführt (s. Abschnitt 4.2.2). Tabelle 7.2 zeigt die Ausprägungen der Problemdimensionen für die verbindungs-basierte (VBN) sowie die aggregierte Time-Space-Netzwerk-basierte (TSN) Modellierung. Die Anzahl der verwendeten Kanten und damit der Anzahl der Entscheidungsvariablen ist mit $|A|_{TSN}$ (bzw. $|A|_{VBN}$) überschrieben. Die Anzahl der Restriktionen ist durch $|M|_{TSN}$ (bzw. $|M|_{VBN}$) gegeben. Das bereits in [Kliwer et al., 2006b] beschriebene Verhalten einer erheblichen Modellreduktion durch die Verwendung des TSN kann auch auf den dieser Arbeit zugrunde liegenden Instanzen bestätigt werden (vgl. Spalte $\frac{|A|_{TSN}}{|A|_{VBN}}$). Daher wurde für die im Folgenden vorgestellten numerischen Tests das mathematische Optimierungsmodell auf Basis des aggregierten Time-Space-Netzwerks aufgebaut. Die möglichen Aktionen werden durch unterschiedliche Typen von Kanten modelliert. Tabelle 7.3 zeigt für die Instanzenmenge die Anteile der Menge aller Kantentypen an. Es werden die Kanten folgender Tätigkeiten unterschieden: Durchführung von Servicefahrten (*Service*), Umsetzfahrten (*Deadhead*), Ein- oder Ausrückfahrten ins Depot (*Depot*), Repräsentation einer Wartezeit (*Warte*) sowie eine Rückflusskante (*Rückfluss*) pro Schicht.

7.5.1 Optimierung mit Standardoptimierungssoftware

Zum Vergleich der im Rahmen dieser Arbeit entwickelten Verfahren wurden die mathematischen Modelle mit der führenden Standardoptimierungssoftware ILOG CPLEX in der Version 11.0.1 gelöst. Zur Lösung der initialen LPs wurden für alle Instanzen die Methoden der Standardsoftware *primaler Simplex*, *dualer Simplex* und *Barrier*¹ verwendet. Darüber hinaus wurden die Parameter basierend auf den Erfahrungen aus [Kliwer, 2005] angepasst, so dass die Lösungszeit gegenüber den Standardeinstellungen der Software verringert wurde.

Die Ergebnisse für die Optimierung mittels der Standardoptimierungssoftware sind in Tabelle 7.4 gegeben. Im Bereich *Optimale Kosten* sind die nach fixen und operativen Kosten aufgeschlüsselten Gesamtkosten einer optimalen Lösung für alle Instanzen dargestellt. Zusätzlich ist die optimale Anzahl der Umläufe dar-

¹Die *Barrier Methode* ist eine Implementierung eines Innere-Punkte-Verfahrens.

Instanz	$ A _{TSN}$	$ M _{TSN}$	$ A _{VBN}$	$ M _{VBN}$	$\frac{ A _{TSN}}{ A _{VBN}}$
real_1	4.482	1.409	65.596	1.274	6,8 %
real_2	2.915	1.445	80.898	1.280	3,6 %
real_3	40.291	2.663	559.908	4.347	7,2 %
real_4	25.754	1.458	767.330	4.670	3,4 %
real_5	544.425	6.068	18.941.169	48.842	2,9 %
real_6	56.334	4.461	2.699.067	15.195	2,1 %
real_7	781.180	7.810	30.426.456	59.874	2,6 %
real_8	99.305	7.847	5.150.600	12.264	1,9 %
real_9	134.194	8.610	8.676.615	18.437	1,5 %
real_10	127.462	8.791	7.031.922	14.129	1,8 %
real_11	46.897	4.102	1.498.547	9.211	3,1 %
real_12	4.525.530	231.846	183.753.676	213.174	2,5 %
real_13	2.191.866	161.590	194.977.869	162.610	1,1 %
real_14	12.764.066	653.700	939.389.952	665.556	1,4 %
real_15	1.657.423	17.010	63.437.955	154.300	2,6 %

Tabelle 7.2: Dimensionen der Netzwerke

Instanz	Service	Deadhead	Depot	Warte	Rückfluss
real_1	424	2.280	826	951	1
real_2	426	675	838	975	1
real_3	1.734	31.629	3.398	3.524	6
real_4	1.684	16.746	3.354	3.967	3
real_5	23.504	419.726	46.722	54.460	13
real_6	6.568	26.412	10.486	12.862	6
real_7	28.728	631.841	56.844	63.755	12
real_8	4.904	74.237	9.478	10.684	2
real_9	7.899	92.692	15.663	17.937	3
real_10	5.650	98.834	11.086	11.890	2
real_11	3.067	30.796	6.071	6.958	5
real_12	101992	4001798	203466	218241	33
real_13	75935	1820068	146228	149620	15
real_14	327367	11166200	633002	637441	56
real_15	71563	1282325	141655	161824	56

Tabelle 7.3: Anzahl modellierter Kanten im TSN

gestellt. Im Bereich *Laufzeiten LP-Phase* sind die Lösungsdauern (in Sekunden) der initialen LPs unter Verwendung der unterschiedlichen Verfahren aufgezeigt. Die Anzahl der Variablen in der LP-Lösung mit fraktionellen Werten sowie die resultierende Laufzeit zur Lösung des Branch&Cut Verfahrens in der IP-Phase sind in den Spalten des Bereichs *IP-Phase* dargestellt. Die schnellste Lösungszeit für eine Instanz ist fett dargestellt.

Alle durchgeführten Tests wurden nach einer maximalen Laufzeit von 10 Stunden abgebrochen. Dies galt für die Instanz *real_13*, von der das LP gelöst werden konnte, aber in der anschließenden IP-Phase keine gültige MIP-Lösung gefunden wurde. Für *real_12* wurde das LP nicht innerhalb der Zeit gelöst und für *real_14* konnte das mathematische Modell aufgrund limitierten Arbeitsspeichers nicht aufgebaut werden. Für die verbleibenden Instanzen wurde eine optimale Lösung bestimmt.

Bei der Lösung der LPs zeigt sich, dass der duale Simplex für die Mehrzahl der Instanzen die schnellste Lösungsmethode ist. Mit dem primalen Simplex können nur wenige Instanzen, und diese nur mit langer Laufzeit, gelöst werden. Für Instanzen mit großen Modelldimensionen (s. Tabelle 7.2) erweist sich allerdings tendenziell die Barrier Methode als schneller, so dass große Problemstellungen wie *real_13* nur mit dieser Methode lösbar sind. Eine Ausnahme bilden innerhalb der betrachteten Instanzenmenge die Probleme *real_5* und *real_7*, bei denen das Barrier Verfahren eine sehr lange Laufzeit aufweist bzw. keine optimale Lösung findet. Bei genauerer Betrachtung dieser Optimierungen zeigt sich, dass das Innere-Punkte-Verfahren innerhalb der Barrier Methode vergleichbar schnell konvergiert (nach 385 bzw. 843 Sekunden), aber der Einsatz des primalen Simplex im abschließenden Crossover zur großen Laufzeit führt. Dieses Verhalten könnte auf die Struktur dieser Instanzen zurückzuführen sein. Denn bei einer großen Anzahl an betrachteten Depots in Verbindung mit nur einem Fahrzeugtypen existiert eine große Anzahl von Kanten, die in allen Schichten (Depots) gleiche Kostenwerte aufweisen. Dies kann zu einer sehr starken Degeneration der Basislösungen im primalen Simplex und damit zu einer schlechten Konvergenz führen.

Wie bereits in Abschnitt 6.2.3 diskutiert wurde, weisen optimale LP-Lösungen der realen Probleminstanzen nur einen geringen Anteil an fraktionellen Variablen auf. Auffällig ist, dass mit steigender numerischer Komplexität einer Instanz neben der benötigten Laufzeit zur Lösung des LP auch der Grad der Fraktionalität steigt. So sind die LP-Lösungen von den Instanzen mit geringer Modellgröße bereits gültig und es ist kein Laufzeitaufwand innerhalb der IP-Phase notwendig. Bei großen Instanzen führt eine stärkere Fraktionalität der LP-Lösung zu Aufwand innerhalb der IP-Phase, der Anteil fraktioneller Variablen bleibt aber für

Instanz	Optimale Kosten				Laufzeiten LP-Phase			IP-Phase	
	Gesamt	Fixe	Operative	#Uml.	primal	dual	barrier	#Frak.	Laufzeit
real_1	2.942.925	2.900.000	42.925	29	0,2	0,0	0,2	0	0
real_2	1.934.176	1.920.000	14.176	32	0,1	0,0	0,1	0	0
real_3	71.003.097	71.000.000	3.097	71	5	2	5	0	0
real_4	54.271.201	51.600.000	2.671.201	47	3	0	2	0	0
real_5	236.494.606	234.300.000	2.194.606	55	- ¹	6.187	26.268	3.012	820
real_6	21.432.487	21.300.000	132.487	115	140	62	17	0	0
real_7	347.974.843	345.060.000	2.914.843	81	- ¹	13.185	- ¹	6.529	1.857
real_8	344.878.979	340.370.000	4.508.979	101	238	3	26	0	0
real_9	12.649.516	12.500.000	149.516	125	1.508	178	108	371	103
real_10	235.556.882	230.920.000	4.636.882	92	203	13	30	0	0
real_11	273.675.596	265.800.000	7.875.596	173	2	0	3	0	0
real_12	-	-	-	362 ²	- ¹	- ¹	- ¹	-	-
real_13	-	-	-	356 ²	- ¹	- ¹	6.808	36.954	- ¹
real_14	-	-	-	356 ²	- ³	- ³	- ³	-	-
real_15	1.494.268.460	1.477.760.000	16.508.460	393	- ¹	21.226	2.000	14.668	7.398

¹ Optimierung wurde nach 36.000 Sekunden Laufzeit abgebrochen.

² Optimale Fahrzeuganzahl wurde durch neue Heuristik berechnet.

³ Modell konnte aufgrund unzureichendem Arbeitsspeicher (8 GB) nicht aufgebaut werden.

Tabelle 7.4: Ergebnisse mit Standardoptimierungssoftware

alle Instanzen unter 1%.

Nach der Vorstellung der Ergebnisse, die mit aktueller Standardoptimierungssoftware erreicht werden konnten, werden in den Folgenden Abschnitten die Auswertungen für die in diesem Kapitel präsentierten Methoden aufgezeigt.

7.5.2 Initialisierungsphase

Im Rahmen Initialisierungsphase der in diesem Kapitel vorgestellten neuen Lösungsmethodik werden die in Abschnitt 7.1 beschriebene Heuristik sowie die in Abschnitt 7.2 präsentierten Methoden zur Bestimmung von unteren Schranken ausgeführt. Die Ergebnisse sind in Tabelle 7.5 dargestellt und werden im Folgenden diskutiert.

Heuristik

Unter Verwendung der neuen Heuristik konnten alle betrachteten Instanzen gelöst werden. Da die Methode die Verwendung der minimalen Anzahl eingesetzter Fahrzeuge sicherstellt und die Fixkosten der Fahrzeuge für reale Probleminstanzen die Zielfunktion dominieren, weichen die Kosten aller mit Hilfe der Heuristik gefundenen Lösungen weniger als ein Promille vom Zielfunktionswert optimaler Lösungen ab (s. Spalte *Gap (ges.)*). Aus diesem Grund sind in den Spalten *Gap (oper.)* die Abweichungen der gültigen Lösungen nach Phase 2 und 3 der Heuristik von den optimalen operativen Kosten gezeigt. Diese konnten für kleine Instanzen optimal bestimmt werden und lagen für größere Instanzen maximal 3,8% über den optimalen operativen Kosten.

Die Lösungszeiten für die Heuristik liegen für die meisten Instanzen im Bereich weniger Sekunden. Lediglich für die großen Instanzen, für die die Standardoptimierungssoftware keine Lösung gefunden hat, werden Zeiten von mehreren Minuten benötigt (bis zu ca. 45 Minuten für die größte Instanz). Die Lösungsdauern sind in den Spalten *Laufzeiten (Sekunden)* aufgeschlüsselt für die drei Phasen der Heuristik gegeben.

Untere Schranken

Im Bereich *Untere Schranken* der Tabelle 7.5 sind die Ergebnisse für die Verfahren zur Bestimmung von unteren Schranken mittels Modellaggregation (*AggrLB*) und durch Relaxation der Depotflusserhaltung ($LR_{df}(0)$) gegeben (s. Abschnitte 7.2.1 und 7.2.2). Da die Verfahren im besten Fall den Zielfunktionswert der LP-Relaxation erreichen können, sind in den Spalten *Gap (zum LP)* die relativen Abweichungen der Lösungen zum optimalen LP-Zielfunktionswert gegeben. In den

Instanz	Heuristik						Untere Schranken						
	Gap (ges.)			Gap (oper.)			Laufzeit (Sekunden)		Gap (zum LP)		Laufzeit (Sekunden)		
	Phase 3	Phase 2	Phase 3	Phase 3	Phase 2	Phase 3	Phase 1	Phase 2	Phase 3	AggrLB	LR _{diff} (0)	AggrLB	LR _{diff} (0)
real_1	0,000%	0,000%	0,000%	0,000%	0	0	0	0	0	0,000%	0,000%	0	0
real_2	0,000%	0,000%	0,000%	0,000%	0	0	0	0	0	0,000%	0,000%	0	0
real_3	0,000%	0,775%	0,000%	0,000%	2	0	0	0	0	0,000%	0,000%	1	1
real_4	0,000%	0,000%	0,000%	0,000%	2	0	0	0	0	0,000%	0,000%	1	0
real_5	0,027%	3,264%	2,914%	2,914%	1	0	0	9	9	0,006%	0,002%	1	15
real_6	0,013%	2,257%	2,135%	2,135%	5	0	0	1	1	5,609%	15,874%	1	1
real_7	0,032%	4,292%	3,763%	3,763%	1	0	0	23	23	0,001%	0,001%	1	24
real_8	0,022%	2,336%	1,715%	1,715%	1	0	0	2	2	0,001%	0,000%	1	1
real_9	0,031%	3,170%	2,630%	2,630%	1	0	0	2	2	0,004%	0,002%	0	7
real_10	0,022%	1,171%	1,128%	1,128%	1	0	0	2	2	0,000%	0,000%	1	2
real_11	0,000%	2,280%	0,000%	0,000%	3	0	0	1	1	0,000%	0,000%	1	1
real_12	-1	-1	-1	-1	2.711	1	1	38	38	-2	-2	1.423	73
real_13	-1	-1	-1	-1	1.451	1	1	14	14	0,175%	4,274%	5.069	173
real_14	-1	-1	-1	-1	772	3	3	13	13	-2	-2	45	239
real_15	0,020%	2,125%	1,778%	1,778%	9	2	2	58	58	0,030%	0,004%	12	63

¹ Lösung wurde gefunden, aber es ist keine optimale Lösung bekannt.

² Untere Schranke wurde berechnet, aber es ist keine optimale LP-Lösung bekannt.

Tabelle 7.5: Ergebnisse der Initialisierungsphase

meisten Fällen konnte der LP-Wert oder eine geringe Abweichung zu ihm in wenigen Sekunden erreicht werden (s. Spalten *Laufzeit (Sekunden)*). Eine Ausnahme bildet die Instanz *real_6*, bei der die unteren Schranken um mehrere Prozent vom LP-Optimum abweichen. Ein Grund hierfür ist die stark unterschiedliche Kostenstruktur der Fixkosten für die unterschiedlichen Fahrzeugtypen. Da die Methoden zur Bestimmung der unteren Schranken nicht die optimalen Fixkosten bestimmen, kann diese stark heterogene Kostenstruktur zu schlechten unteren Schranken führen.

Das Verfahren $LR_{df}(0)$ weist in der Regel eine größere Laufzeit auf als *AggrLB*, aber führt bei mehreren Instanzen zu besseren Lösungen. Für schwierige Instanzen (z.B. *real_6* mit einer heterogenen Struktur der Fixkosten) scheint $LR_{df}(0)$ allerdings ungeeignet zu sein. Bei diesen Instanzen liefert *AggrLB* zum Teil deutlich bessere untere Schranken.

7.5.3 Verbesserungsphase

Im Folgenden werden die Ergebnisse der in Abschnitt 7.3 vorgestellten Verbesserungsphase präsentiert. Da die Heuristik, wie im vorhergehenden Abschnitt dargestellt wurde, bereits für die Instanzen *real_1*, *real_2*, *real_3*, *real_4*, und *real_11* optimale Lösungen gefunden hat, werden diese in den folgenden Untersuchungen nicht betrachtet.

Pricing Problem

Zum Vergleich der in Abschnitt 7.3.3 vorgestellten Strategien zur Auswahl neuer Kanten für das Master Problem wurden die Instanzen mit bis auf die Pricing-Strategie identischen Parametern gelöst. Die Initialisierung des ersten RMP wurde mit den Kanten der Heuristik durchgeführt. Zur Lösung des RMP wurde der duale Simplex verwendet.

Für die geeignete Einstellung der jeweiligen Parameter wurden folgende Einstellungen auf Basis empirischer Studien identifiziert. Für das *Multiple Dantzig Pricing (MDP)* wurden maximal $2 \cdot |T|$ Kanten mit den geringsten negativen reduzierten Kosten ausgewählt. Innerhalb des *Multiple Time Slot Pricing (MTSP)* wurde ein Fahrplan in $\zeta = 25$ gleich große Zeitbereiche aufgeteilt und je Zeitbereich maximal $\frac{3 \cdot |T|}{\zeta}$ Kanten mit den geringsten negativen reduzierten Kosten ausgewählt. Für die Strategien des Lagrangean Pricing (LRP_{cc} und LRP_{df}) wurde nach Konvergenz des Verfahrens auf das Multiple Dantzig Pricing gewechselt (s. Abschnitt 7.3.3), damit nur noch Kanten mit negativen reduzierten Kosten in das RMP aufgenommen werden (vgl. [Löbel, 1997]). Der Wechsel wurde nach

drei Iterationen durchgeführt, in denen sich der optimale Zielfunktionswert des RMP nicht verbessert hat.

Da mit der Generierung von neuen Kanten in das RMP die Modellgröße und damit die notwendige Lösungszeit in jeder Iteration ansteigt, werden in jeder Iteration Spalten des aktuellen RMP gelöscht (s. Algorithmus 7.1, Schritt 6). Um möglichst die Kanten aus dem RMP zu löschen, die nicht zur Bildung einer guten bzw. optimalen Lösung notwendig sind, wurde die Entscheidung auf Basis der reduzierten Kosten sowie dem aktuell bekannten absoluten Gap zwischen bester bekannter gültiger Lösung und unterer Schranke getroffen. Eine Nicht-Basisvariable x_{ij} , deren reduzierte Kosten einen Anteil η des aktuellen Gaps überschritten, wurde gelöscht ($\bar{c}_{ij} > \eta \cdot (UB - LB)$). In den durchgeführten Tests wurden Kanten mit reduzierten Kosten über 60% des aktuellen Gaps gelöscht ($\eta = 0,6$), aber maximal die schlechtesten 10% der Variablen im aktuellen RMP. Zudem wurden die Löschvorgänge gespeichert, damit Variablen, die in früheren Iterationen bereits drei mal gelöscht (und wieder generiert) wurden nicht mehr gelöscht werden. Dadurch kann ein Stagnieren des Algorithmus verhindert werden.

Als Abbruchkriterien für den Kantengenerierungsalgorithmus wurde zum einen bei Überschreitung der Begrenzungen von 10.800 Sekunden (3 Stunden) oder 100 Iterationen abgebrochen. Darüber hinaus wurde das Verfahren nach fünf Iterationen ohne Verbesserung des aktuellen Zielfunktionswertes abgebrochen (Konvergenzabbruch).

Die beschriebenen Parametereinstellungen gelten, sofern nicht explizit anders angegeben, auch für die im Weiteren vorgestellten Untersuchungen.

Abbildung 7.6 zeigt das Verhalten der Lösungsmethode für die vorgestellten Pricing-Strategien. Die Funktionen verdeutlichen das Verhalten der im Master Problem gefundenen LP-Lösungen über die Zeit (Sekunden). Die Lösungsqualität ist als relative Abweichung der **operativen** Kosten zur optimalen LP-Lösung angezeigt. Aufgrund der nicht bekannten optimalen LP-Lösung der Instanz *real_12* sind in der zugehörigen Graphik absolute Werte angegeben. Für *real_14* konnten die Datenstrukturen zur Lösung der Pricing Probleme aufgrund unzureichenden Arbeitsspeichers nicht aufgebaut werden, so dass für die Instanz keine Verbesserungsphase ausgeführt werden konnte.

Wie in Abschnitt 7.3.3 diskutiert wurde, führt das *MDP* nur zu Beginn des Algorithmus zu deutlichen Verbesserungen und konvergiert sehr langsam. Bei Verwendung des *MTSP* konvergiert das Verfahren zu Beginn langsamer, allerdings können auch zu einem späten Zeitpunkt des Algorithmus noch deutliche Verbesserungen erreicht werden. Beide Verfahren sind für die Lösung von großen Instanzen nicht geeignet, da der Algorithmus abbricht, wenn über mehrere Iterationen keine Verbesserungen erreicht werden können. Unter Verwendung des La-

7 Lösungsmethodik für Busumlaufplanungsprobleme

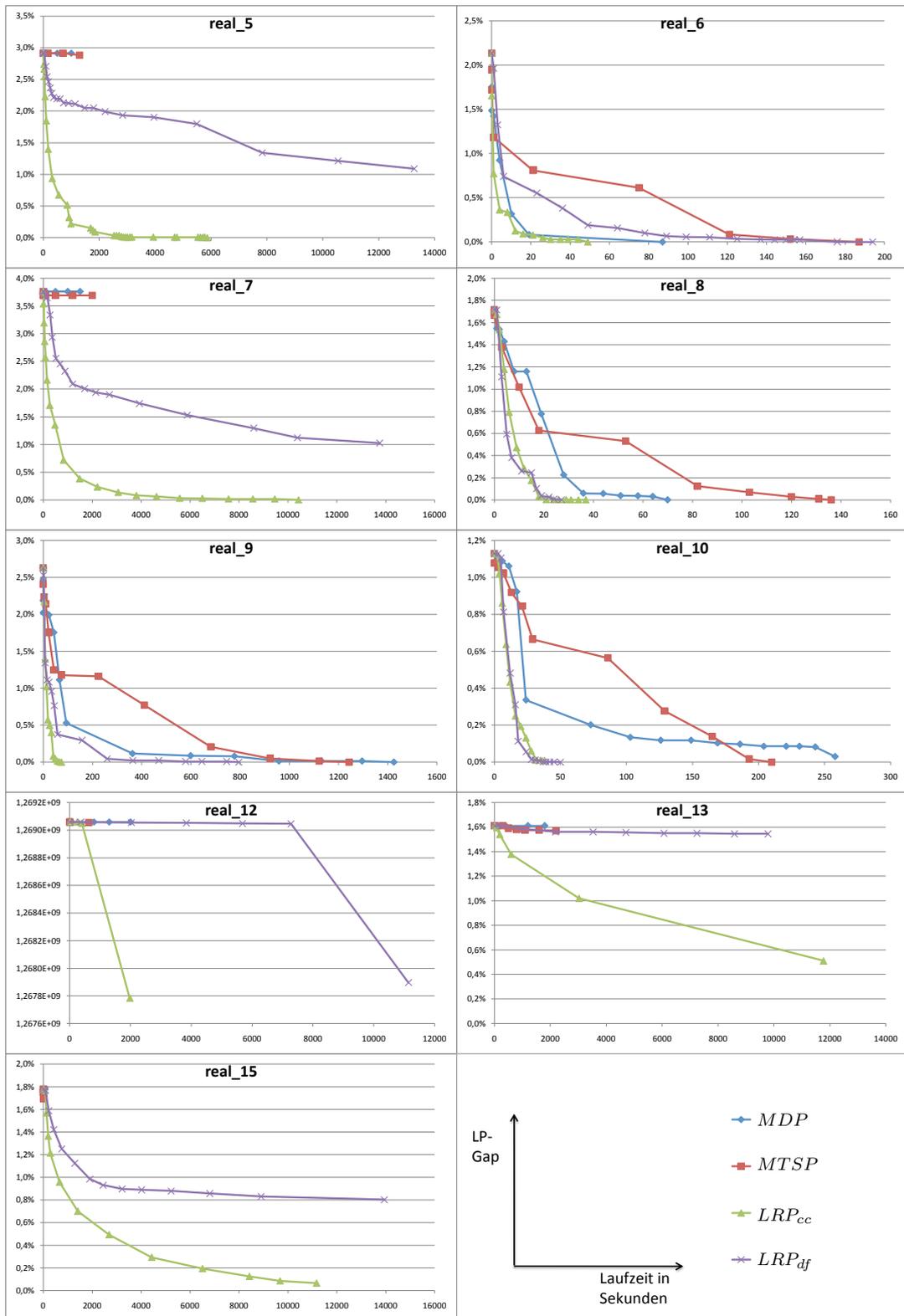


Abbildung 7.6: Dominanz der Pricingstrategie LRP_{cc} gegenüber alternativen Strategien in der Kantengenerierung

grangean Pricing (LRP_{cc} und LRP_{df}) können die Lösungen für alle dargestellten Instanzen deutlich verbessert werden. Die Verwendung des LRP_{cc} dominiert auf allen Instanzen die anderen Strategien, so dass für die im Weiteren präsentierten Untersuchungen nur die Ergebnisse für diese Strategie aufgeführt werden.

Durch das *Reduced Cost Fixing* (s. Abschnitt 7.3.3) können je nach Instanz zwischen 1% und 82% aller Netzwerk-Kanten auf 0 fixiert werden. Da diese Kanten nicht im RMP vorhanden sind, hat die Information keinen Einfluss auf das Master Problem. Im Pricing Problem werden die Kanten nicht betrachtet oder auf 0 fixiert. Die resultierende Vereinfachung des Pricing Problems führt allerdings nur zu einer geringen Reduzierung der Laufzeit.

Master Problem

Zur Lösung des Master Problems kommen unterschiedliche Methoden in Frage (s. Abschnitt 7.3.2). Abbildung 7.7 zeigt die Ergebnisse für unterschiedliche LP-Methoden, die in der Standardoptimierungssoftware ILOG CPLEX (Version 11.0.1) bereitgestellt werden. Neben der *primalen Simplex Methode (PS)* ist die *duale Simplex Methode mit (DS)* und *ohne Preprocessing (DS*)* sowie die *Barrier Methode ohne Durchführung des Crossover (BM)* dargestellt. Eine Diskussion der Vor- und Nachteile der Methoden ist in Abschnitt 7.3.2 aufgeführt. Die Darstellung der Konvergenz bei unterschiedlichen Lösungsmethode für das Master Problem ist analog zu Abbildung 7.6 aufgebaut.

Die Ergebnisse zeigen, dass bei kleinen Instanzen unabhängig von der Wahl der Lösungsmethode eine Konvergenz zur optimalen Lösung erreicht wird. Der duale Simplex erweist sich, mit und ohne Preprocessing, als beste Wahl zur schnellen Lösung dieser Instanzen, da sowohl der primale Simplex als auch die Barrier Methode eine längere Laufzeit pro Iteration benötigen (s. zum Beispiel *real_9*).

Bei Instanzen mit einer größeren numerischen Komplexität (s. Abschnitt 6.2.4) erreichen der duale Simplex mit Preprocessing und die Barrier Methode eine schnellere Konvergenz (s. zum Beispiel *real_5*). Vor allem auf großen Instanzen dominiert die Barrier Methode die anderen Alternativen. Dieses Verhalten ist durch das Verhalten des Lagrangean Pricing in Bezug auf die Anzahl der generierten Kanten pro Iteration zu erklären: Durch die Betrachtung des gesamten Netzwerks im Pricing Problem ist die Anzahl der ausgewählten Kanten proportional zur Gesamtanzahl der Kanten. Je komplexer eine Instanz in Bezug auf die Anzahl an Kanten im Netzwerk ist, desto größer sind also die im RMP zu lösenden LPs. Wie in Abschnitt 7.5.1 gezeigt, ist die Barrier Methode ohne Ausführung des Crossover am besten zur Lösung großer Instanzen geeignet. Aus diesem Grund zeigt der Einsatz der Barrier Methode im Kantengenerierungsalgorithmus auf den

7 Lösungsmethodik für Busumlaufplanungsprobleme

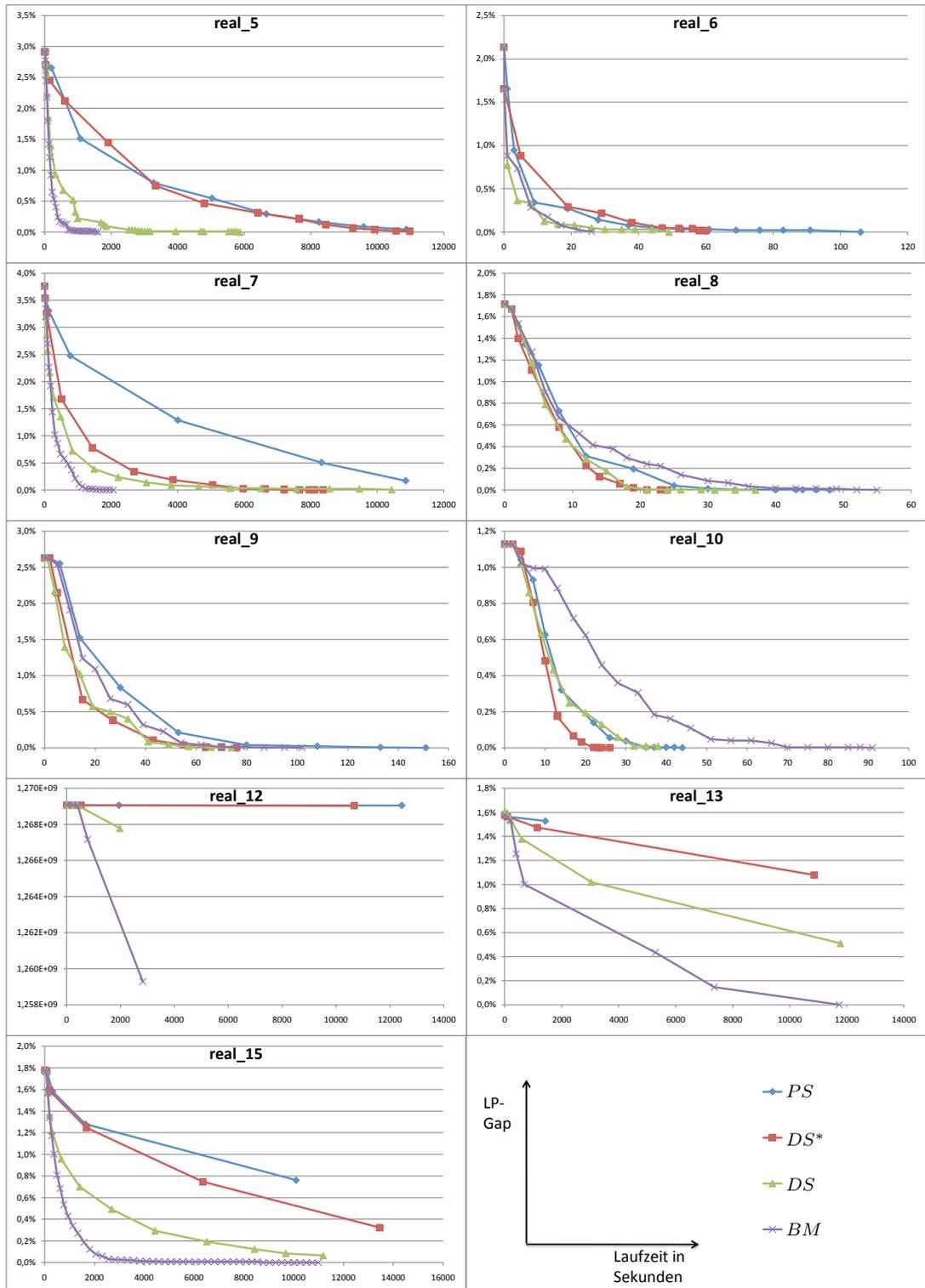


Abbildung 7.7: Vergleich der Strategien zur Lösung des Master Problems in der Kantengenerierung

großen Instanzen ein performantes Verhalten.

Für die folgenden Ergebnisse werden nur noch die Ergebnisse des dualen Simplex mit Preprocessing (*DS*) und der Barrier Methode (*BM*) dargestellt.

Initialisierung

Zur Initialisierung des Kantengenerierungsalgorithmus können unterschiedliche Methoden eingesetzt werden, die sicherstellen, dass das initiale RMP mindestens eine gültige Lösung enthält (s. Abschnitt 7.3.1).

Bei der Verwendung von zusätzlichen künstlichen Variablen (Ini_k), die jeweils eine Servicefahrt abdecken und mit großen Kosten versehen sind, führte die Anwendung des Kantengenerierungsalgorithmus zu schlechten Ergebnissen. Das Verfahren benötigte für alle Instanzen viele Iterationen, um eine Lösung ohne künstliche Variablen zu finden. Für große Instanzen konnte keine Lösung gefunden werden. In allen Fällen blieb der beste Gap zur optimalen Lösung weit über dem der in Abschnitt 7.1 vorgestellten Heuristik. Aus diesem Grund werden im Folgenden nur die Ergebnisse für die übrigen in Abschnitt 7.3.1 vorgestellten Strategien dargestellt. Die schlechte Konvergenz des Verfahrens bei Initialisierung mit künstlichen Kanten unterstreicht die Wichtigkeit der Heuristik, damit in der Kantengenerierung gute Lösungen gefunden werden können.

Abbildung 7.8 zeigt die Ergebnisse für den Kantengenerierungsalgorithmus unter Verwendung des dualen Simplex mit Preprocessing (*DS*) und der Barrier Methode (*BM*). Zur Initialisierung wurden jeweils die in der Heuristik verwendeten Kanten (Ini_h) oder zusätzlich die in der Lagrange-Lösung vorhandenen Kanten (Ini_{hlq}) verwendet (s. Abschnitt 7.3.1). Die verikal eingezeichnete Linie markiert den Zeitpunkt, an dem die jeweils beste LP-Methode eine optimale LP-Lösung für das Modell gefunden hat (s. Tabelle 7.4).

Die Ergebnisse bestätigen das im vorhergehenden Abschnitt gezeigte Verhalten, dass die Barrier Methode bei großen Instanzen eine schnellere Konvergenz aufweist. Zwischen den beiden unterschiedlichen Initialisierungen lässt sich dagegen für die meisten Instanzen kein dominierendes Verhalten einer Strategie erkennen. Allerdings führt die Verwendung der zusätzlichen Kanten aus der Lagrange Relaxation bei großen Instanzen zu einer langsameren Konvergenz.

Gegenüber der LP-Lösung, die unter Verwendung der jeweils schnellsten Methode aus der Standardoptimierungssoftware berechnet wurde, kann der Kantengenerierungsalgorithmus für viele Instanzen in kürzerer Zeit die optimale Lösung berechnen. Darüber hinaus ergeben sich weitere Vorteile zur Bestimmung von gültigen (MIP-) Lösungen, die im folgenden Abschnitt aufgeführt werden.

7 Lösungsmethodik für Busumlaufplanungsprobleme

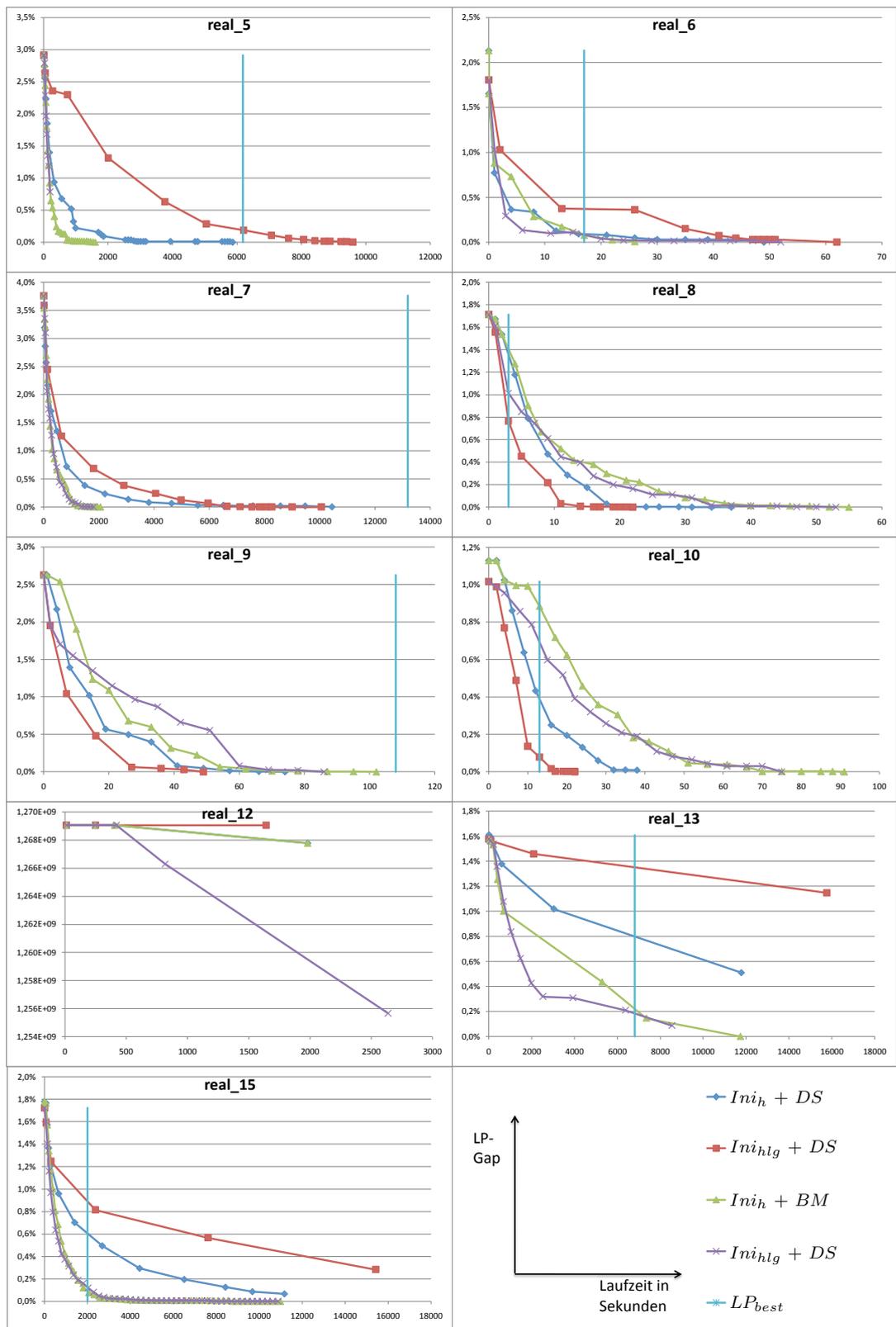


Abbildung 7.8: Vergleich von Initialisierungsstrategien hinsichtlich der Konvergenz zur optimalen LP-Lösung

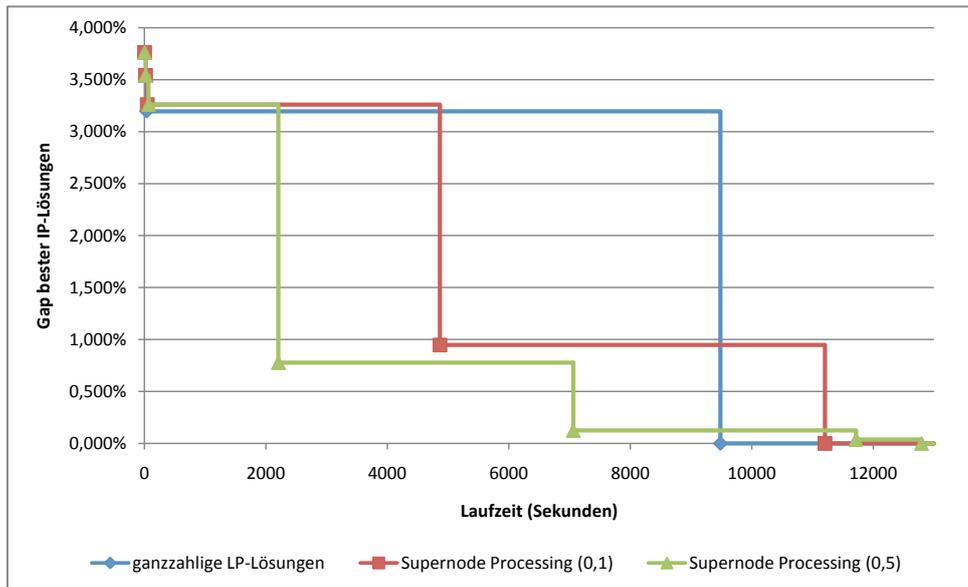


Abbildung 7.9: Aufhebung des Black-Box-Verhaltens durch den Einsatz von IP-Heuristiken

Ganzzahlige Lösungen

Durch die iterative Lösung des RMP und der Fast-Ganzzahligkeitseigenschaft der Problem instanzen (s. Abschnitt 6.2.3) führt der Einsatz einer IP-Heuristik zur Laufzeit des Algorithmus zu ganzzahligen Lösungen. Zur Identifikation gültiger Lösungen wurden die in Abschnitt 7.3.4 präsentierten Strategien untersucht.

Bereits bei der Anwendung der ersten Strategie, in der jede LP-Lösung lediglich auf Ganzzahligkeit geprüft wird, werden für alle Instanzen gültige Lösungen zur Laufzeit gefunden.

Bei Verwendung der Heuristik zur Konstruktion ganzzahliger Lösungen zeigte sich, dass nur in sehr seltenen Fällen und mit hohem Laufzeitaufwand eine weitere Lösung gefunden werden konnte. Diese Vorgehensweise ist für eine effiziente Suche nach gültigen Lösungen daher nicht geeignet und wird im Folgenden nicht weiter dargestellt.

Dagegen führte der gesteuerte Einsatz des „Supernode Processing“ aus dem Branch&Bound Algorithmus von Standardoptimierungssoftware zu einer geeigneten Möglichkeit, gültige Lösungen zu finden. Durch die Begrenzung der Laufzeit für das Verfahren ist eine Steuerung des Trade-Offs zwischen Lösungsdauer und der Anzahl gefundener Lösungen möglich. Abbildung 7.9 zeigt das für alle betrachteten Instanzen typische Verhalten dieses Trade-Offs anhand der Instanz *real_7*: In den ersten Iterationen des Algorithmus werden in der Regel viele ganz-

Instanz	Anzahl Variablen			IP-Laufzeit (Sekunden)	
	Standard	Generierung	Anteil	Standard	Generierung
real_5	544.425	85.629	15,7%	820	78
real_6	56.334	21.690	38,5%	0	0
real_7	781.180	116.069	14,9%	1.857	1.727
real_8	99.305	14.694	14,8%	0	0
real_9	134.194	25.764	19,2%	103	96
real_10	127.462	15.418	12,1%	0	0
real_12	4.525.530	290.008	6,4%	- ¹	- ¹
real_13	2.191.866	235.562	10,8%	- ¹	93
real_15	1.657.423	291.070	17,6%	7.398	2.590

¹ IP-Phase wurde nach 36.000 Sekunden ohne gültige Lösung abgebrochen

Tabelle 7.6: Kürzere Laufzeiten in der IP-Phase des Kantengenerierungsalgorithmus

zahlige LP-Lösungen gefunden, da die Modellgröße der RMPs relativ klein ist (s. Abschnitt 7.5.1). Sofern keine IP-Heuristik verwendet wird, können im weiteren Verlauf des Algorithmus keine oder nur wenige zusätzliche Lösungen gefunden werden bis durch die IP-Phase eine optimale Lösung berechnet wird. Bei Anwendung des „Supernode Processing“ werden weitere Lösungen gefunden, allerdings konvergiert das Verfahren erst zu einem späteren Zeitpunkt zur optimalen Lösung. Zur Steuerung gibt ein Parameter die maximale Laufzeit als relativen Anteil der Laufzeit des zuletzt gelösten RMPs an.

Neben der Findung von gültigen Lösungen zur Laufzeit, ermöglicht der Kantengenerierungsalgorithmus eine Verkürzung der Lösungszeit für die IP-Phase, da nur die Variablen berücksichtigt werden, die im letzten RMP vorhanden sind. Unter Verwendung dieser Variablenmenge kann eine optimale Lösung nicht bewiesen werden, aber für alle Instanzen, von denen eine optimale Lösung bekannt ist, konnte diese mit der Teilmenge der Variablen gefunden werden.

Die Modellgrößen sowie die Laufzeiten der Optimierungen sind für das jeweils beste Verfahren aus Standardoptimierungssoftware in Tabelle 7.6 dargestellt. Der Anteil der pro Instanz generierten Variablen zur Bildung einer optimalen Lösung liegt zwischen 6,4% und 38,5%. Aus diesem Grund reduzieren sich auch die Laufzeiten für die abschließende IP-Phase des Kantengenerierungsalgorithmus, so dass neben einer schnelleren LP-Phase und ganzzahliger Lösungen zur Laufzeit des Algorithmus auch die IP-Phase eine kürzere Laufzeit hat.

Einbettung und Preprocessing

Durch die Einbettung des Kantengenerierungsalgorithmus in den in Abschnitt 7.4 präsentierten Prozess können unabhängig von der Methode zusätzliche Techniken verwendet werden. Auf der einen Seite kann dies die Berücksichtigung zusätzlicher Restriktionen sein (z.B. Definition von Linienbündeln oder Festlegung von Anschluss-Bedingungen). Auf der anderen Seite kann die Planungskomplexität durch heuristische Preprocessing-Techniken reduziert werden (z.B. globale Substitution von Fahrzeugtypen oder Bildung stabiler Fahrketten).

Durch diese Kombination des Kantengenerierungsalgorithmus mit weiteren problemspezifischen Techniken können auch sehr komplexe Problemstellungen gelöst werden. Zum Beispiel kann für die Instanz *real_14* bei Kombination mit der Heuristik zur Bildung stabiler Fahrketten die initiale Lösung verbessert werden, was aufgrund zu hoher Speicheranforderungen vorher nicht möglich war.

7.5.4 Zusammenfassung und Bewertung

Abschließend wird eine Zusammenfassung der in den vorangegangenen Abschnitten dargestellten Ergebnisse gegeben sowie die Eignung des neuen Lösungsansatzes hinsichtlich der Anwendungsfälle der Umlaufplanung bewertet.

Der im Rahmen dieser Arbeit entwickelte Lösungsansatz umfasst eine Initialisierungs- und eine Verbesserungsphase. Zur schnellen Berechnung einer gültigen Lösung in der Initialisierungsphase wurde eine Heuristik präsentiert, die durch Lösung von drei vergleichsweise wenig komplexen Optimierungsproblemen eine Lösung findet, die bereits eine optimale Fahrzeuganzahl enthält (s. Abschnitt 7.1). Für die betrachteten realen Instanzen konnte in wenigen Sekunden (bzw. Minuten für große Instanzen) eine Lösung gefunden werden, von der auch die operativen Kosten nur wenige Prozent von den optimalen operativen Kosten abweichen. Durch die Heuristik konnten erstmals für große Instanzen gültige Lösungen gefunden werden (s. Abschnitt 7.5.2).

Zur Bestimmung unterer Schranken des optimalen Zielfunktionswertes wurden zwei Ansätze vorgestellt, die die Problemstellung in unterschiedlicher Form vereinfachen und lösen (s. Abschnitt 7.2). Die Verfahren können für den Großteil der Instanzen in wenigen Sekunden gute untere Schranken finden, so dass in Verbindung mit der Heuristik eine enge Abschätzung des optimalen Zielfunktionswertes möglich ist (s. Abschnitt 7.5.2). Neben dieser Einschätzung dienen die ermittelten Informationen der folgenden Verbesserungsphase als Möglichkeit zur Initialisierung einer gültigen Kantenmenge und zum Ausschluss ungeeigneter Kanten (s. Abschnitte 7.3.1 und 7.3.3).

Zur Verbesserung der gefundenen Lösungen wurde ein Kantengenerierungsalgorithmus vorgestellt, der ausgehend von einer Lösung mit einer kleinen Untermenge aller Kanten iterativ zusätzliche Kanten in die Optimierung einbezieht (s. Abschnitt 7.3). Das Verfahren kann durch speziell auf die Problemstellung ausgerichtete Lösungsstrategien bessere gültige Lösungen finden (s. Abschnitte 7.5.3).

Durch die Kombination dieser beiden Phasen und die Einbeziehung weiterer unabhängiger Beschleunigungskomponenten (s. Abschnitt 7.4) erfüllt die Methode alle Anforderungen für die Anwendungsfälle der Umlaufplanung ohne Zeitfenster. Durch die flexible Anpassbarkeit kann das Verfahren von der Berechnung einer gültigen Lösung in kurzer Zeit (nur Heuristik) bis zur Berechnung einer bewiesenen optimalen Lösung (Betrachtung aller Kanten) beliebig skaliert werden. So kann das Verfahren zum Beispiel auch in einer vorgegebenen Zeit eine möglichst gute Lösung suchen. Durch die Einbeziehung der problemspezifischen Eigenschaft wenig fraktioneller LP-Lösungen können bereits zur Laufzeit gültige Lösungen ausgegeben werden. Durch die Verwendung eines aggregierten Mehrgüter Flussmodells werden je Lösung eine Vielzahl von Umlaufplänen mit gleichen Kosten dekomponiert, so dass unterschiedliche Alternativen angeboten werden. Die Anzahl der zur Laufzeit gefundenen gültigen Lösungen kann durch die parametrisierte Intensität der Suche gesteuert werden. Durch diese Möglichkeit vermeidet das Verfahren ein Black-Box-Verhalten und kann den Planer je nach Anforderung unterstützen.

Für die Zusammenfassung der numerischen Ergebnisse sind die Instanzen, sortiert nach der in Abschnitt 6.2.4 vorgestellten Komplexitätskennzahl Λ , in Tabelle 7.7 aufgeführt. Die Instanzen sind in drei Kategorien – klein, mittel und groß – aufgeteilt. Die Tabelle zeigt in den drei letzten Spalten die Laufzeiten (in Sekunden) für die Optimierung mit der besten Einstellung von Standardoptimierungssoftware sowie für die Initialisierungsphase und die gesamte neue Methode. Die zusätzlichen zur Laufzeit gefundenen Lösungen sind nicht dargestellt.

Abhängig von der numerischen Komplexität einer Instanz eignen sich unterschiedliche Lösungsstrategien. Für alle Instanzen ist die vollständige Durchführung der Initialisierungsphase sinnvoll, da eine schnelle Rückmeldung des Optimierungsverfahrens mit fahrzeugoptimalen Lösungen erfolgt und diese für alle kleinen Instanzen bereits optimal sind. Allerdings ist eine Begrenzung der Laufzeit für die Algorithmen zur Berechnung unterer Schranken sinnvoll, da sie zum Teil stark schwankende Laufzeiten aufweisen (s. Tabelle 7.5).

Für das in der Verbesserungsphase zu lösende Pricing Problem hat sich das Lagrangen Pricing (CC) als effizient gezeigt (s. Abbildung 7.6). Die Auswahl der effizientesten Methode zur Lösung des Master Problems ist dagegen abhängig

Kategorie	Instanz	Λ	$t_{Standard}$	$t_{Initialisierung}$	$t_{Generierung}$
klein ($\Lambda \leq 5$)	real_1	0	0	0	0
	real_2	0	0	0	0
	real_11	1	0	4	4
	real_3	1	2	2	2
	real_4	2	0	2	2
mittel ($0 \leq \Lambda \leq 500$)	real_8	12	3	3	24
	real_10	21	13	3	23
	real_6	21	17	6	49
	real_9	32	211	3	170
groß ($500 \leq \Lambda$)	real_13	1.212	- ¹	1.466	5.376
	real_15	2.374	9.398	69	8.788
	real_7	3.614	15.042	24	3.788
	real_5	4.292	7.007	10	1.682
	real_12	8.615	- ¹	2.750	36.000 ²
	real_14	276.920	- ¹	788	36.000 ²

¹ Optimierung wurde nach 36.000 Sekunden Laufzeit ohne gültige Lösung abgebrochen.

² Optimierung wurde nach 36.000 Sekunden Laufzeit mit gültigen Lösungen abgebrochen.

Tabelle 7.7: Kategorisierung der Instanzen und Zusammenfassung der numerischen Ergebnisse

von der Problemkomplexität. Für kleine und mittlere Instanzen ist die Verwendung des dualen Simplex mit LP-Preprocessing, für große Instanzen die Barrier Methode mit LP-Preprocessing und ohne Durchführung des Crossover am besten geeignet (s. Abbildung 7.7).

Die Einstellung für die Intensität der Suche nach ganzzahligen Lösungen ist abhängig von der gewünschten Anzahl. Ein sinnvoller Kompromiss kann die Verwendung des *Supernode Processing* mit einer maximalen Laufzeit von 10% der zuletzt benötigten RMP-Laufzeit. Durch diese Einstellung investiert das Verfahren nur vergleichsweise wenig Zeit in die IP-Heuristik, aber findet auch zu gering fraktionellen LP-Lösungen eine gültige IP-Lösung (s. Abbildung 7.9).

Für die Lösung sehr großer Instanzen ($\Lambda \geq 10.000$) ist die Verwendung einer zusätzlichen Heuristik (z.B. Fahrketten-Heuristik) sinnvoll, da die exakten Netzwerk-Modelle andernfalls nicht auf üblichen PCs aufgebaut werden können (s. Abschnitt 7.5.3). Der Grenzwert ist stark abhängig von dem verwendeten System und muss individuell angepasst werden.

Abschließend kann festgestellt werden, dass die identifizierten Anwendungsfälle der Umlaufplanung ohne Zeitfenster durch den flexibel skalierbaren Einsatz der im Rahmen dieser Arbeit entwickelten Methodik abgedeckt werden können. Durch den Einsatz der Lösungsmethode können

- zu allen mit Standardoptimierungssoftware exakt gelösten Instanzen *optimale Lösungen* gefunden werden,
- für mittlere und große Instanzen *schneller fahrzeugoptimale Lösungen* berechnet werden,
- für große und zum Teil mittlere Instanzen *schneller optimale Lösungen* gefunden werden,
- *erstmalig gültige Lösungen* zu großen Instanzen gefunden werden und
- eine Vielzahl gültiger *Umlaufpläne zur Laufzeit* ausgegeben werden.

8 Optimierungsmethoden für die Umlaufplanung mit Zeitfenstern

Für die Umlaufplanung im öffentlichen Personennahverkehr sind die Abfahrts- und Ankunftszeiten für Servicefahrten bereits aus der vorhergehenden Phase der Fahrplanerstellung gegeben (s. Abschnitt 2.1). Bei der Planung des Fahrzeugeinsatzes werden getätigte Entscheidungen zu Abfahrts- und Ankunftszeiten durch die Berücksichtigung von Zeitfenstern für Servicefahrten zu einem Teil revidiert. Die Fahrzeugeinsatzplanung mit Zeitfenstern stellt somit eine Teilintegration der Fahrplanerstellung mit der Umlaufplanung dar (s. Abschnitt 2.3.5). Für die Optimierung von Problemstellungen mit Zeitfenstern sind in Abschnitt 2.2 zwei Anwendungsfälle aus operativen und taktischen Planungsphasen dargestellt. Für diese werden im Folgenden Lösungsansätze präsentiert, die die identifizierten Anforderungen der jeweiligen Anwendungsfälle berücksichtigen können.

8.1 Operative Umlaufplanung mit Zeitfenstern

In der *operativen Planung mit Zeitfenstern* werden Zeitfenster zur Senkung der geplanten operativen Fahrzeugeinsatzkosten betrachtet. Durch eine zeitliche Verschiebung von Servicefahrten können neue Verbindungsmöglichkeiten entstehen, so dass weitere Fahrtensequenzen zulässig werden und die Menge zulässiger Umläufe vergrößert wird. Im Vergleich zur Planung ohne Zeitfenster können diese zu einer weiteren Reduzierung der geplanten Kosten genutzt werden.

8.1.1 Organisatorische Aspekte und Anforderungen

Der Umfang akzeptierter zeitlicher Veränderungen im Fahrplan ist abhängig von den Freiheitsgraden, die ein Verkehrsunternehmen bei der Planung zulässt. Der Umgang mit den Änderungen im Fahrplan und die Anforderungen unterscheiden sich je nach Verkehrsunternehmen. Häufig werden individuelle Zeitfenster abhängig von der Art der Servicefahrt vergeben. So werden getaktete Fahrplanfahrten in der Regel mit keinen oder kleinen Zeitfenstern (z.B. ± 2 Minuten) betrachtet. Für nicht reguläre Fahrten wie Schul-, Sonder- oder Verstärkungsfahrten,

die unabhängig von einer Linientaktung ausgeführt werden, werden häufig größere zeitliche Verschiebungen akzeptiert, da die Erwartungshaltung der Fahrgäste hinsichtlich einer pünktlichen Bedienung der Fahrt nicht so stark ausgeprägt ist wie bei den Linienfahrten.

Unabhängig von der Größe der vom Verkehrsunternehmen akzeptierten Zeitfenster kann mit den resultierenden Verschiebungen auf unterschiedliche Weise umgegangen werden. Eine Möglichkeit besteht darin die Änderungen der geplanten Abfahrts- und Ankunftszeiten nicht im veröffentlichten Fahrplan auszuzeichnen. Der zusätzliche Freiheitsgrad in der Planung und damit verbundene Kosteneinsparungen können dadurch allerdings zu einer Verringerung der Kundenzufriedenheit führen, da die Fahrzeuge (geplant) weniger pünktlich sind. Da eine zu frühe Bedienung einer Servicefahrt im Gegensatz zu einer Verspätung zu ungleich größerer Unzufriedenheit führt, werden bei diesem Vorgehen in der Regel nur zeitliche Verschiebungen nach hinten zugelassen. Alternativ können die Änderungen im veröffentlichten Fahrplan ausgezeichnet werden, so dass die Fahrgäste informiert sind und keine (geplanten) Verspätungen wahrnehmen.

Neben der Berücksichtigung individueller Zeitfenster können weitere Anforderungen an die Planung gestellt werden. Aufgrund der unterschiedlichen Verschiebung von Servicefahrten können geplante Taktungen (z.B. innerhalb einer Linie) oder Anschlüsse zu anderen Linien verloren gehen. Aus diesem Grund kann die Anforderung bestehen, dass definierte Mengen von Servicefahrten nur in gleichem Ausmaß verschoben werden dürfen.

Da der zusätzliche Freiheitsgrad erlaubter Fahrtenverschiebungen auf der einen Seite die geplanten Kosten senken kann, aber auf der anderen Seite die in der Fahrplanerstellung geplante Struktur stört, kann ein Trade-Off zwischen Kostensenkung und Gesamtverschiebung angestrebt werden. So kann zum Beispiel die Anforderung bestehen, dass eine Verschiebung von Servicefahrten nur dann zuzulassen wird, wenn ein Fahrzeug eingespart werden kann.

Entsprechend der operativen Planung ohne Zeitfenster zielt die Optimierung mit Zeitfenstern auf die Generierung eines möglichst kostengünstigen Umlaufplans ab. Deshalb gelten die in Abschnitt 2.2.1 vorgestellten Anforderungen in gleicher Weise für die operative Planung mit und ohne Zeitfenster (z.B. Akzeptanz einer langen Optimierungsdauer und Vermeidung eines Black-Box-Verhaltens).

Die folgenden Abschnitte beschreiben Ansätze zur Modellierung der vorgestellten Anforderungen der operativen Planung mit Zeitfenstern und zeigen die Anpassung der in Kapitel 7 vorgestellten Methodik zur Lösung der Probleme.

8.1.2 Modellierung

Zur Modellierung von Umlaufplanungsproblemen mit Zeitfenstern (VSP-TW) werden im Rahmen dieser Arbeit Mehrgüter Flussmodelle verwendet, da sie, wie in Kapitel 6 dargestellt, bei Verwendung realer Instanzen effizient lösbar sind. Die Berücksichtigung von Zeitfenstern innerhalb der aggregierten Time-Space Netzwerk (TSN) basierten Mehrgüter Flussformulierung ist in [Kliwer, 2005] gegeben. Durch eine diskrete Betrachtung der möglichen Verschiebungen können die Ausprägungen als Fahrtkanten im Netzwerk modelliert werden (s. Abschnitt 4.4).

Um die Verschiebung von Fahrten nur für große Kosteneinsparungen durchzuführen, können die Kosten der Zeitfensterkanten abhängig von der Höhe der zeitlichen Verschiebung mit einem Faktor gewichtet werden. Durch eine flexible Wahl dieses Faktors kann der Grad der Bestrafung einer Verschiebung vom Planer gesteuert werden.

Um die gleiche Verschiebung von Servicefahrten zu berücksichtigen, kann das mathematische Modell aus [Kliwer, 2005] um zusätzliche Entscheidungsvariablen $y_k^l \in \{0, 1\}$ erweitert werden, die für alle Fahrtenmengen l und den jeweils zulässigen relativen Verschiebungen k definiert werden. Das erweiterte mathematische Modell für die Planung mit Zeitfenstern lautet:

$$VSPTW : \quad \min \sum_{(i,j)^d \in A} c_{ij}^d x_{ij}^d \quad (8.1)$$

$$s.t. \quad \sum_{i:(i,j)^d \in A} x_{ij}^d - \sum_{i:(j,i)^d \in A} x_{ji}^d = 0 \quad \forall j \in N^d, \forall d \in D \quad (8.2)$$

$$\sum_{k=-q_t}^{p_t} \sum_{d \in D} x_t^{d,k} = 1 \quad \forall t \in T \quad (8.3)$$

$$\sum_{t \in T_l} \sum_{d \in D} x_t^{d,k} \leq y_k^l M \quad \forall k \in [-\tilde{q}_l, \tilde{p}_l], l \in [1, b] \quad (8.4)$$

$$\sum_{k=\tilde{q}_l}^{\tilde{p}_l} y_k^l = 1 \quad \forall l \in [1, b] \quad (8.5)$$

$$x_d \leq w_d \quad \forall d \in D \quad (8.6)$$

$$x_{ij}^d \in \mathbb{N} \quad \forall (i, j)^d \in A \quad (8.7)$$

$$y_k^l \in \{0, 1\} \quad \forall k \in [-\tilde{q}_l, \tilde{p}_l], l \in [1, b] \quad (8.8)$$

Die in [Kliwer, 2005] erweiterten Überdeckungsrestriktionen (8.3) stellen sicher, dass für jede Servicefahrt genau eine Kante im Netzwerk verwendet wird.

Bei der optionalen Berücksichtigung von gleichen Verschiebungen aller Fahrtenmengen $l \in [1, b]$ werden die zusätzlichen Variablen y_k^l in Restriktion (8.4) mit den Entscheidungsvariablen für die Servicefahrten verknüpft. Der Wert M stellt dabei eine die mögliche Ausprägung der linken Seite der Restriktion dominierende Konstante dar. In Restriktion (8.5) wird sichergestellt, dass alle Fahrten einer Menge $l \in [1, b]$ die gleiche Ausprägung innerhalb ihrer zulässigen Verschiebung $[-\tilde{q}_l, \tilde{p}_l]$ haben, indem genau eine Verschiebungsausprägung zulässig ist.

8.1.3 Lösungsmethoden

Die Einbeziehung der Zeitfenster in die Modelle erhöht die numerische Komplexität je Ausprägung zu einem ähnlichen Grad wie für zusätzliche Servicefahrten. Durch die Behandlung der Zeitfensterkanten als Fahrkanten steigt die Anzahl der zusätzlichen Verbindungs- und Depotkanten. Es werden, im Fall der Zeitfenster ohne gleiche Verschiebung, zwar keine neuen Restriktionen in die Formulierung eingefügt, allerdings steigt die Anzahl der Einträge in den Überdeckungsrestriktionen. Die Umlaufplanungsprobleme mit Zeitfenstern weisen daher verglichen mit den gleichen Problemstellungen ohne Zeitfenster eine größere numerische Komplexität auf (s. Abschnitt 8.1.4).

In [Kliwer, 2005] wird zur Lösung der TSN-basierten Modelle auf Standard-Optimierungssoftware zurückgegriffen. Da mittlere und große Instanzen mit Zeitfenstern nicht mehr exakt gelöst werden konnten, werden Verfahren verwendet, die vor der Optimierung die Zeitfenster so einschränken, dass die Freiheitsgrade ohne vielversprechenden Nutzen ausgeschlossen werden. Zum Beispiel wird in der Schnitt-Heuristik der Auslastungsgraph des Fahrplans analysiert und zur Minimierung der Fahrzeuganzahl nur Zeitfenster für Servicefahrten in Bereichen hoher Auslastung zugelassen. Durch den Einsatz dieser Heuristiken können auch Instanzen mittlerer Größe gelöst werden.

Um anhand des Mehrgüter Flussmodells auf der Basis eines aggregierten TSNs auch große Probleme mit Zeitfenstern lösen zu können und gleichzeitig die Anforderungen der operativen Planung mit Zeitfenstern zu berücksichtigen, kann die in Kapitel 7 vorgestellte Methodik der Kantengenerierung für die Lösung von VSP-TW angepasst werden. Die Vorgehensweise und die Vorteile der Methodik entsprechen den im vorangegangenen Kapitel gegebenen Ausführungen, die in Algorithmus 7.1 zusammengefasst sind. Daher soll im Folgenden nur auf die notwendigen Anpassungen der Methodik eingegangen werden.

Anpassung des Kantengenerierungsalgorithmus

Bei der Initialisierung des mathematischen Modells muss sichergestellt werden, dass eine gültige Lösung für das erste RMP existiert. Die vorgeschlagenen Initialisierungsstrategien für das MD-VSP können übernommen werden, da eine gültige Lösung des MD-VSPs auch für das VSP-TW gültig ist. Allerdings berücksichtigt keines dieser Verfahren die Zeitfenster, so dass der zusätzliche Freiheitsgrad in der Initialisierung nicht genutzt wird. Da bereits das VSP-TW mit einem Depot und keinen Fahrzeugtypen NP-schwer ist (s. Abschnitt 2.3.5), ist die in der in Abschnitt 7.1 vorgestellten Heuristik durchgeführte Lösung einer vereinfachten Problemstellung zur Erstellung günstiger Fahrtensequenzen nicht sinnvoll. Statt die möglichen Freiheitsgrade zu reduzieren, wird daher eine Lösung durch die sequentielle Konstruktion von Umläufen erstellt. Wie im vorangegangenen Kapitel werden die Kombinationen aus Depotstandort und Fahrzeugtyp zur Vereinfachung als Depots bezeichnet. Der Ablauf des Verfahrens ist in Algorithmus 8.1 dargestellt.

Algorithmus 8.1 : Konstruktionsheuristik für VSP-TW

```

Sortiere Servicefahrten nach Abfahrtszeit  $e_t \forall t \in T$ 
Initialisiere  $U = \emptyset$ 
for  $t \in T$  do
    identifiziere gültige Depots  $D_t \subseteq D$ 
    for  $d \in D_t$  do
        identifiziere gültige Umläufe  $U_d^t$ 
        berechne Zusatzkosten  $c_d^t$ 
    if  $D_t = \emptyset$  then
        finde Depots mit freier Kapazität  $D'_t \subseteq D_t$ 
        for  $d \in D'^t$  do
            berechne Kosten  $c_d^t$  für neuen Umlauf mit  $t$ 
            erstelle Umlauf in  $d$  mit  $\min_{d \in D'^t} c_d^t$ 
    else
        füge  $t$  in Umlauf mit  $\min_{u \in U_d^t, d \in D_t} c_d^t$ 

```

Zur Initialisierung des Verfahrens werden die Servicefahrten nach ihrer Abfahrtszeit sortiert. Anschließend wird für jede Fahrt geprüft, wie sie mit den geringsten Zusatzkosten in den Umlaufplan eingefügt werden kann. Dazu werden zuerst die Depots identifiziert, deren Fahrzeugtyp in der Fahrzeugtypgruppe der Fahrt enthalten ist, da andernfalls keine gültige Zuweisung möglich ist. Für die

im bereits aufgebauten Plan vorhandenen Umläufe wird geprüft, ob die Fahrt in einer Ausprägung ihres gültigen Zeitfensters an den Umlauf angehängt werden kann (direkt oder durch eine Umsetzfahrt). Für diese Umläufe werden die zusätzlichen Kosten berechnet, die das Einfügen der Fahrt verursachen würde. Falls mindestens ein möglicher Umlauf existiert, wird die Fahrt in den Umlauf mit den minimalen Zusatzkosten eingefügt. Um primär die Anzahl der benötigten Fahrzeuge zu minimieren, wird die Fahrt so in den Umlauf eingefügt, dass sie im Rahmen ihres gültigen Zeitfensters möglichst früh beginnt. Dadurch wird die bestmögliche Ausgangssituation für die Einsortierung weiterer Fahrten geschaffen. Sollte kein Umlauf existieren, der die Fahrt aufnehmen kann, werden für alle zulässigen Depots die Kosten berechnet, die ein neuer Umlauf mit der Fahrt verursachen würde und der günstigste von ihnen wird angelegt. Die Fahrt wird in diesem Fall zu ihrem frühestmöglichen Zeitpunkt beginnen. Nach Abschluss der Einsortierung aller Fahrten, ist der resultierende Umlaufplan eine gültige Lösung für das VSP-TW. Allerdings kann durch die Vorgehensweise der Greedy-Heuristik keine gesicherte Aussage über die Qualität der Lösung gemacht werden.

Für die Lösung des Master Problems und der Findung ganzzahliger Lösungen können die gleichen Methoden eingesetzt werden wie für die Lösung des MD-VSPs. Dies gilt nicht für das Lösen des Pricing Problems. Die reduzierten Kosten können für jede Kante wie zuvor berechnet werden, so dass die Strategien *Multiple Dantzig Pricing (MDP)* sowie *Multiple Time Slot Pricing (MTSP)* geeignet sind. Die Anwendung des Lagrangean Pricing ist allerdings nicht vollständig übertragbar.

Bei der in Abschnitt 7.2.2 vorgestellten Umformulierung des Modells, anhand der die Bedingungen zur Erhaltung der Depotzugehörigkeit relaxiert werden konnten, sind für jede Fahrt zwei zusätzliche globale Fahrtknoten angelegt worden. Durch eine Erweiterung dieser Umformulierung mit Zeitfensterkanten würde eine Relaxation der Bedingungen dazu führen, dass eine Fahrt neben der Zugehörigkeit zu einem Depot auch unterschiedliche Verschiebungen der Abfahrts- und Ankunftszeit erhalten könnte. Abbildung 8.1a zeigt einen Ausschnitt aus einem TSN mit zwei Kanten für eine Fahrt, die eine geplante Abfahrtszeit und eine um eine Minute nach hinten verschobene Abfahrt repräsentieren. Der grün markierte Fluss kann durch die Verwendung der Zeitfensterkante realisiert werden, so dass eine neue Verbindungsmöglichkeit entsteht. In Abbildung 8.1b wird das umformulierte Netzwerk dargestellt. Anhand des roten Flusses ist beispielhaft dargestellt, dass sowohl die zugeordnete Netzwerkschicht als auch die zeitliche Zuordnung verändert werden kann. In einer optimalen Lösung der Relaxation würden daher viele Fahrten eine verkürzte Fahrtdauer enthalten, so dass die in der Lösung vorhandenen Kanten nicht als vielversprechend für die Verbesserung des RMP

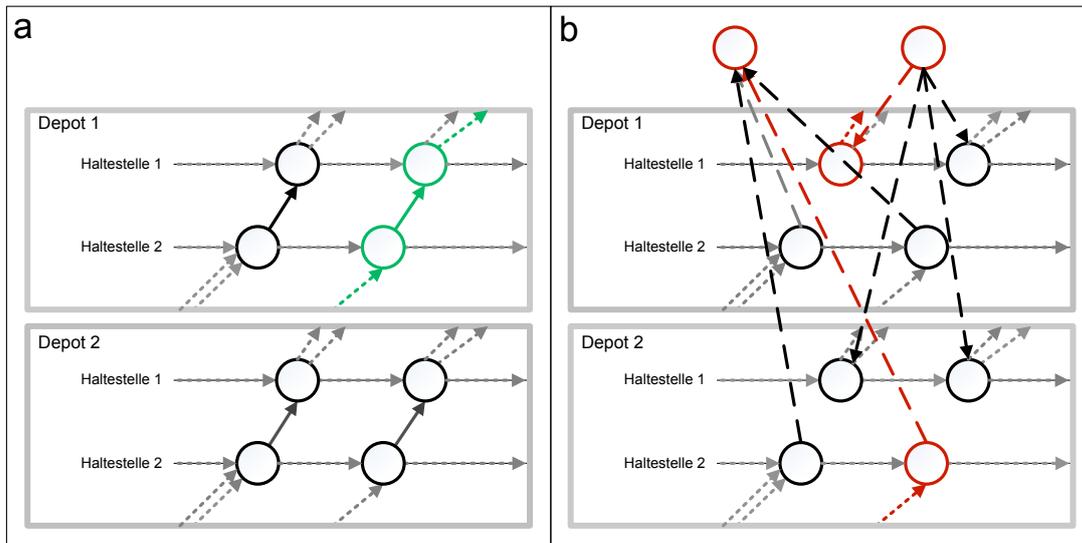


Abbildung 8.1: Alternative Netzwerkmodellierung für Servicefahrten mit Zeitfenstern

angesehen werden.

Im Gegensatz zur Lagrange Relaxation der Depotflusserhaltung kann das Lagrangean Pricing mit Relaxation der Cover Constraints für die Kantengenerierung verwendet werden. Durch die Relaxation können Fahrten mehrfach oder gar nicht abgedeckt werden, die Dauer einer Fahrt bleibt aber in jedem Fall erhalten.

Die in Abbildung 7.5 dargestellte Einbettung des Kantengenerierungsalgorithmus kann bis auf wenige Anpassungen für die Optimierung von VSP-TW übernommen werden. Die Fixierung von Anschlüssen gleicher Linien (Ebene 2) ist zwar auch für VSP-TW möglich, allerdings wird der Freiheitsgrad der Zeitfenster durch die fixierten Anschlüsse vollständig vernachlässigt. Das Verfahren kann daher nur zielführend eingesetzt werden, wenn die Zeitfenster der betroffenen Servicefahrten die gleichen Größen haben. In diesem Fall kann die neue Servicefahrt, die die fixierten Servicefahrten ersetzt, das gleiche Zeitfenster bekommen. Auch die Bildung von stabilen Fahrtketten (Ebene 4) erscheint für den Einsatz zur Lösung von VSP-TW nicht sinnvoll, da die innerhalb des Verfahrens zu lösenden Unterprobleme durch die Zeitfenster der Servicefahrten bereits schwierig zu lösen sind und somit eine lange Laufzeit benötigen oder nicht lösbar sind. Die Relaxation der Zeitfenster in den Unterproblemen würde wie bei der Lösung des relaxierten MD-VSPs (s. Abschnitt 7.2.2) zu einer vergleichsweise kurzen Laufzeit führen. Da die Fahrtketten allerdings ohne die Ausnutzung der durch die Zeitfenster entstehenden Freiheitsgrade gebildet werden, resultiert eine schlechte Lösungsqualität. Als zusätzliche Komponente können allerdings die in [Klie-

wer, 2005] vorgestellten Methoden zur Reduktion der Zeitfenster (z.B. Schnitt-Heuristik) eingebunden werden. Dadurch können diese Methoden gleichzeitig mit dem Kantengenerierungsalgorithmus verwendet werden.

8.1.4 Numerische Ergebnisse

Der folgende Abschnitt präsentiert die numerischen Ergebnisse für das Optimierungsproblem der *operativen Planung mit Zeitfenstern* (s. Abschnitt 2.2). Zur Lösung der in Abschnitt 8.1.2 vorgestellten mathematischen Formulierung *VSPTW* wird der angepasste Kantengenerierungsalgorithmus verwendet (s. Abschnitt 8.1.3). Um die Eignung dieser Lösungsmethode zu validieren, wurden die mathematischen Modelle der im Rahmen dieser Arbeit betrachteten realen Probleminstanzen (s. Abschnitt 6.1.1) zusätzlich mit Standardoptimierungssoftware sowie der Schnitt-Heuristik aus [Kliwer, 2005] gelöst.

Die numerischen Tests wurden auf einem PC durchgeführt, der mit einem *Intel Core 2 Duo*-Prozessor mit zwei auf 3,16 GHz getakteten Prozessorkernen und 8 Gigabyte Arbeitsspeicher ausgestattet war. Die Tests wurden unter dem 64-bit Betriebssystem *Microsoft Windows XP Professional x64 Edition* durchgeführt. Eine Optimierung wurde nach drei Stunden (10.800 Sekunden) abgebrochen.

Alle Instanzen wurden mit unterschiedlich großen Zeitfenstern optimiert, wobei für alle Servicefahrten gleich große Verschiebeintervalle gültig waren. Tabelle 8.1 zeigt die Ergebnisse für diskrete Zeitfenster von ± 1 , ± 2 und ± 5 Minuten für jede Fahrt. Die Einträge zeigen den relativen Abstand zur besten bekannten Lösung (*Gap*) und die Optimierungsdauer in Sekunden (*Sek.*) für die folgenden Lösungsverfahren:

- *Exakt*: Lösung mit Standardoptimierungssoftware ILOG CPLEX in Version 11.0.1 und der Barrier Methode zur Lösung des initialen LPs.
- *Schnitt*: Verwendung der Schnitt-Heuristik aus [Kliwer, 2005] zur Reduzierung der zulässigen Zeitfenster und Lösung des reduzierten Problems mit exaktem Verfahren.
- *Generierung*: Optimierung mit angepasstem Kantengenerierungsalgorithmus aus Abschnitt 8.1.3.
- *Gener.+Schnitt*: Verwendung der Schnitt-Heuristik aus [Kliwer, 2005] und Lösung mit angepasstem Kantengenerierungsalgorithmus.

Für die Instanzen *real_12* und *real_14* sind keine Ergebnisse angegeben, da die Modelle bereits für Zeitfenster von ± 1 aufgrund unzureichenden Arbeitsspeichers nicht aufgebaut werden konnten.

8.1 Operative Umlaufplanung mit Zeitfenstern

Instanz	TW	Exakt		Schnitt		Generierung		Gener.+Schnitt	
		Gap	Sek.	Gap	Sek.	Gap	Sek.	Gap	Sek.
real_1	± 1	0,0%	2	0,0%	1	0,0%	1	0,0%	1
	± 2	0,0%	3	3,6%	1	0,0%	1	3,6%	1
	± 5	0,0%	14	7,8%	5	0,0%	3	7,8%	2
real_2	± 1	0,0%	1	0,0%	1	0,0%	1	0,0%	1
	± 2	0,0%	2	0,0%	1	0,0%	1	0,0%	1
	± 5	0,0%	6	0,0%	3	0,0%	3	0,0%	2
real_3	± 1	0,0%	20	0,0%	11	0,0%	4	0,0%	3
	± 2	0,0%	37	0,0%	18	0,0%	8	0,0%	4
	± 5	0,0%	186	0,0%	80	0,0%	29	0,0%	10
real_4	± 1	0,0%	10	0,0%	7	0,0%	4	0,0%	4
	± 2	0,0%	27	0,0%	18	0,0%	7	0,0%	5
	± 5	0,0%	358	0,0%	144	0,0%	64	0,0%	148
real_5	± 1	-	-	0,0%	852	0,0%	11.831	0,0%	13.259
	± 2	0,0%	6.626	0,0%	2.937	4,0%	353	3,9%	5.824
	± 5	-	-	-	-	0,2%	842	0,0%	394
real_6	± 1	0,0%	103	0,0%	42	1,7%	383	0,0%	692
	± 2	-	-	0,0%	416	0,0%	3.122	0,0%	3.063
	± 5	-	-	-	-	0,0%	7.198	0,4%	3.072
real_7	± 1	-	-	-	-	0,0%	306	0,0%	12.774
	± 2	-	-	-	-	0,0%	573	0,0%	228
	± 5	-	-	-	-	0,0%	1.432	2,0%	432
real_8	± 1	0,0%	211	0,0%	100	1,2%	169	0,3%	104
	± 2	0,0%	525	0,0%	306	1,8%	482	0,0%	922
	± 5	-	-	-	-	0,0%	5.948	7,9%	46
real_9	± 1	0,0%	1.100	0,8%	359	1,9%	936	1,9%	476
	± 2	0,0%	3.266	0,9%	847	2,4%	2.184	3,0%	1.386
	± 5	-	-	0,0%	5.858	2,8%	10.337	7,9%	111
real_10	± 1	0,0%	224	0,0%	129	0,2%	169	0,0%	15
	± 2	0,0%	1.124	0,0%	412	0,3%	502	0,0%	25
	± 5	-	-	-	-	0,0%	811	0,8%	66
real_11	± 1	0,0%	20	0,4%	11	0,0%	6	0,4%	5
	± 2	0,0%	59	0,4%	27	0,0%	15	0,4%	8
	± 5	0,0%	647	0,4%	199	0,0%	126	0,4%	35
real_13	± 1	-	-	-	-	0,0%	5.839	0,1%	406
	± 2	-	-	-	-	0,0%	3.980	0,0%	2.751
	± 5	-	-	-	-	-	-	-	-
real_15	± 1	-	-	-	-	0,5%	649	0,0%	14.752
	± 2	-	-	-	-	0,0%	2.077	0,1%	578
	± 5	-	-	-	-	-	-	-	-

Tabelle 8.1: Ergebnisse für operative Zeitfenster

Die Ergebnisse zeigen, dass mit steigenden Zeitfenstergrößen die Laufzeit stark ansteigt. Mit dem exakten Verfahren können die Problemstellungen für kleine Instanzen (z.B. *real_1* - *real_4*) optimal gelöst werden. Bereits für mittelgroße Instanzen ist eine exakte Optimierung mit Zeitfenster von 5 Minuten nicht mehr möglich (z.B. *real_8* - *real_10*). Große Instanzen können bereits für Zeitfenster von einer Minute nicht gelöst werden.

Mit der zusätzlichen Verwendung der Schnitt-Heuristik können die Instanzen aufgrund der reduzierten Zeitfenster schneller gelöst werden, so dass auch die Optimierung mit größeren Zeitfenstern möglich ist (*real_6* und *real_9*). Allerdings können für große Instanzen oder mittlere Instanzen mit Zeitfenstern von fünf Minuten auch mit dieser Methode keine Lösungen innerhalb der zulässigen Zeit gefunden werden (z.B. *real_7*). Zudem weicht die Lösungsqualität zum Teil deutlich von der optimaler Lösungen ab (z.B. *real_1*).

Bei Verwendung des Kantengenerierungsalgorithmus werden optimale Lösungen für kleine Instanzen in kürzerer Zeit gefunden (z.B. *real_1* - *real_4*). Zudem können Lösungen für weitaus komplexere Problemstellungen gefunden werden, so dass erstmals auch große Instanzen mit Zeitfenstern gelöst werden können (z.B. *real_15*). Allerdings zeigen die Ergebnisse für mittlere Instanzen zum Teil deutliche Abweichungen von der optimalen Lösung (z.B. *real_9*). Das Verhalten resultiert aus einem, im Gegensatz zum MD-VSP, vergleichsweise großen Anteil fraktioneller Variablen in den LP-Lösungen, der mit fortschreitender Optimierung tendenziell zunimmt. Daher kann die IP-Heuristik des Kantengenerierungsalgorithmus während der LP-Phase keine gültigen Lösungen finden. Zudem ist die Anzahl der durchgeführten Iterationen im Vergleich zur Lösung des MD-VSPs größer, da das initiale RMP keine fahrzeugoptimale Lösung enthält und die Konvergenz daher länger dauert. Bei allen Instanzen, zu denen das Verfahren keine optimale Lösung finden kann, ist dieses Verhalten aufgetreten, wodurch die Optimierung aufgrund begrenzter Laufzeit während der IP-Phase abgebrochen wurde. Die in Tabelle 8.1 angegebenen Werte sind daher gültige Lösungen, die in frühen Iterationen gefunden wurde. Die Zielfunktionswerte der gefundenen LP-Lösungen entsprechen dagegen den optimalen IP-Lösungen, so dass das Verfahren bei zusätzlicher Laufzeit vermutlich eine optimale Lösung findet.

Bei der Verwendung der Schnitt-Heuristik vor dem Kantengenerierungsalgorithmus werden die zu betrachtenden Zeitfenster reduziert, so dass der Lösungsraum kleiner ist. Durch diese Beschränkung werden allerdings, wie oben beschrieben, in mehreren Fällen optimale Lösungen ausgeschlossen. Die Ergebnisse zeigen, dass bei der Kombination in den meisten Fällen vergleichsweise schlechte Lösungen gefunden werden und auch keine zusätzlichen Instanzen gelöst werden können.

8.2 Taktische Fahrplananalyse mit Zeitfenstern

Der Anwendungsfall der *taktischen Umlaufplanung mit Zeitfenstern* fordert eine Berücksichtigung von vergleichsweise großen Zeitfenstern zur Analyse von Potentialen zur Fahrzeugeinsparung in einem Fahrplan (s. Abschnitt 2.2.2). Aufgrund der Fokussierung auf die minimale Anzahl von Fahrzeugen, wird der Fahrplan unabhängig von Erweiterungen zur Senkung operativer Kosten (z.B. Depots) betrachtet. Wie im vorangegangenen Abschnitt für reale Probleminstanzen gezeigt wurde, können sowohl exakte Lösungsverfahren als auch heuristische LP-basierte Lösungsansätze, wie der im Rahmen dieser Arbeit entwickelte Kantengenerierungsalgorithmus, diese Problemstellungen nicht lösen (s. Abschnitt 8.1.4). Daher werden im folgenden Abschnitt drei neue Ansätze zur Lösung des VSP-TW mit großen Zeitfenstern vorgestellt. In Abschnitt 8.2.1 wird das Optimierungsproblem zur Findung der minimalen Fahrzeuganzahl auf ein Permutationsproblem zurückgeführt und dieses heuristisch gelöst. Abschnitt 8.2.2 zeigt ein Konzept für einen hybriden Ansatz zur Kombination von Meta-Heuristiken und effizienten Optimierungsmethoden. Die Ähnlichkeit vom Umlaufplanungsproblem mit Zeitfenstern und dem *vehicle routing problem* mit Zeitfenstern sowie die Portierung effizienter Methoden wird in Abschnitt 8.2.3 gegeben.

Die Methoden beschreiben das Vorgehen für das Umlaufplanungsproblem mit einem Depot/Fahrzeugtypen und Zeitfenstern (SD-VSP-TW), da dieses in der Regel zur Abschätzung von Einsparungen betrachtet wird. Auf die notwendigen konzeptionellen Erweiterungen der jeweiligen Methode wird kurz eingegangen.

8.2.1 Sortierheuristiken

Der im Folgenden vorgestellte Ansatz zur Lösung von Umlaufplanungsproblemen mit großen Zeitfenstern basiert auf der Beobachtung, dass unter Verwendung von Algorithmus 8.1 für das VSP-TW ein fahrzeugoptimaler Umlauf entstehen kann. Aufgrund der lokalen Betrachtung der Freiheitsgrade einer Fahrt wird in der Regel allerdings keine optimale Lösung gefunden. Im Folgenden wird ein Algorithmus präsentiert, der angelehnt an Algorithmus 8.1 den Aufbau eines Umlaufplans für das SD-VSP-TW vornimmt. Es wird gezeigt, dass eine Reihenfolge von Fahrten existiert, für die der Algorithmus die optimale Lösung ermittelt.

Für eine beliebig vorgegebene Sortierung der Servicefahrten, kann ein Umlaufplan erstellt werden, indem für jede Fahrt sukzessive alle bestehenden Umläufe geprüft werden, ob die Fahrt unter Berücksichtigung der möglichen Verschiebeintervalle an einen Umlauf angehängt werden kann. Ist dies der Fall, wird die Fahrt zum frühestmöglichen Zeitpunkt in einen gefundenen Umlauf aufgenom-

men. Andernfalls wird ein neuer Umlauf erstellt, der die Servicefahrt zu ihrem frühestmöglichen Zeitpunkt enthält. Dieses Vorgehen ist in Algorithmus 8.2 dargestellt.

Algorithmus 8.2 : Evaluierung für Sortierheuristiken

Eingabe : Reihenfolge der Servicefahrten

Initialisiere $U = \emptyset$

for $t \in T$ **do**

identifiziere zur Aufnahme von t gültige Umläufe U_t

if $U_t = \emptyset$ **then**

erstelle neuen Umlauf u mit t möglichst früh

$U = U \cup \{u\}$

else

füge t möglichst früh in $u \in U_t$ ein

Ausgabe : Anzahl benötigter Umläufe $|U|$

Es existiert eine Reihenfolge der Servicefahrten, für die der Algorithmus eine fahrzeugoptimale Lösung liefert. Um dies zu zeigen, sei ein optimaler Umlaufplan U^* angenommen, der $|U^*|$ Umläufe enthält. Jeder Umlauf $u^* \in U^*$ beinhaltet eine Sequenz von Servicefahrten $u_i^* = (t_{i1}, t_{i2}, \dots), \forall i = 1, \dots, |U^*|$. Dieser optimale Umlaufplan kann durch den Algorithmus mit Eingabe der Reihenfolge $(t_{11}, t_{12}, \dots, t_{21}, t_{22}, \dots, t_{31}, t_{32}, \dots)$ gebildet werden, da für die erste Fahrt des optimalen Umlaufs t_{11} ein neuer Umlauf im Algorithmus gebildet wird und jede folgende Fahrt t_{1*} aufgrund der möglichst frühen Abfahrt jeder weiteren Servicefahrt in den Umlauf aufgenommen werden kann. Dies gilt anschließend auch für alle weiteren optimalen Umläufe, die mit Fahrt t_{*1} beginnen.

Da in der für den Algorithmus vorgegebenen Reihenfolge der Servicefahrten die Teilsequenzen der optimalen Umläufe (t_{*1}, t_{*2}, \dots) beliebig kombinierbar sind, existieren mindestens $|U^*|!$ optimale Reihenfolgen für den Algorithmus.

Das SD-VSP-TW mit der Zielsetzung einer minimalen Fahrzeuganzahl kann auf der Basis der vorangegangenen Überlegungen als Permutationsproblem definiert werden: gesucht ist eine Reihenfolge für die Servicefahrten T , zu der Algorithmus 8.2 die optimale Fahrzeuganzahl liefert. Es existieren $|T|!$ mögliche Kombinationen aller Fahrten, von denen mindestens $|U^*|!$ optimal sind. Der Algorithmus kann für eine optimale Reihenfolge allerdings nicht feststellen, dass sie optimal ist. Mit Hilfe dieses Konzepts können beliebige Verfahren zur intelligenten Suche nach optimalen Reihenfolgen eingesetzt werden (z.B. Meta-Heuristiken), um das SD-VSP-TW zu lösen.

Zur Sicherstellung einer korrekten Zuordnung im Falle von Fahrzeugtypgrup-

pen (MVTG), kann der Algorithmus angepasst werden, so dass eine Fahrt nur in einem gültigen Umlauf zugeordnet werden kann. Zudem können Routenrestriktionen (RC) durch den sukzessiven Aufbau jedes Umlaufs exakt innerhalb des Verfahrens berücksichtigt werden. Dazu müssen bei der Identifizierung möglicher Umläufe die Routenrestriktionen beachtet werden und der Umlauf bei Verletzung einer Restriktion für die Fahrt ausgeschlossen werden.

Exemplarisch wurde im Rahmen dieser Arbeit ein evolutionärer Algorithmus entwickelt, der im Folgenden vorgestellt wird.

Evolutionärer Algorithmus für Sortierheuristik

Entsprechend der Modellierungskonzepte für Meta-Heuristiken (s. Abschnitt 3.4) definiert die *Darstellung* die Größe des betrachteten Lösungsraums. Für den evolutionären Algorithmus wird diese innerhalb eines Chromosoms kodiert. Für das beschriebene Permutationsproblem sei der Genotyp durch eine Sequenz der Zahlen von 1 bis $|T|$ definiert. Diese Kodierung repräsentiert den Phenotyp, der eine Sortierung der Servicefahrten darstellt. Jedes Gen gibt den Index einer Servicefahrt im Fahrplan an (s. Abschnitt 3.4.2). Das *Ziel* des Algorithmus ist das Finden einer Sequenz mit der minimalen Ausprägung der Bewertungsfunktion. Als *Bewertungsfunktion* wird das im vorangegangenen Abschnitt vorgestellte Verfahren zur Konstruktion eines Umlaufplans auf der Basis der Fahrtensequenz verwendet (s. Algorithmus 8.2).

Neben der Bewertung von Lösungen, bei denen die Fitness der Individuen entsprechend der Lösung der Bewertungsfunktion gesetzt wird, werden drei weitere Schritte im evolutionären Algorithmus durchgeführt (s. Abschnitt 3.4.2). Für die Konzeption des Algorithmus zur Lösung des Permutationsproblems für das VSP-TW haben sich nach der Analyse unterschiedlicher Ansätze die folgenden Techniken als effizient erwiesen:

Zur *Initialisierung* der ersten Population P_0 wird ein Individuum erzeugt, bei dem die Servicefahrten nach ihrer Abfahrtszeit sortiert werden, um eine vergleichsweise gute Lösung in der Population zu haben. Die weiteren Individuen werden durch eine zufällige Reihenfolge der Servicefahrten gebildet, damit möglichst unterschiedliche Individuen und somit heterogene Chromosome enthalten sind. Die Anzahl der Individuen in einer Generation ist konstant und sei durch $|P|$ gegeben.

Nach der Bewertung der Individuen wird in jeder Iteration eine Menge $Q_t \subset P_t$ identifiziert, die nicht in die nächste Population P_{t+1} übernommen wird. Diese *Selektion* wird durch eine *Wettbewerbsselektion* (engl: *tournament selection*) vorgenommen, bei der je zwei zufällig bestimmte Individuen miteinander vergli-

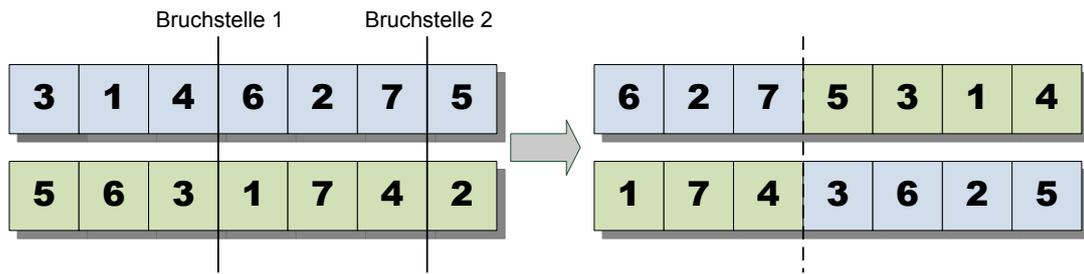


Abbildung 8.2: Variation zweier Individuen für Sortierheuristik

chen werden. Das Individuum mit der besseren Fitness ist ein Kandidat für die nächste Generation und das Individuum mit der schlechteren Fitness wird aus der Population entfernt. Sofern $Q_t > \frac{P_t}{2}$ wird das Verfahren für die identifizierten Kandidaten entsprechend wiederholt. Durch die Wettbewerbsselektion kann die Übernahme der besten Lösung sichergestellt werden und gleichzeitig eine Diversifikation durch eine mögliche Übernahme schlechter Lösungen erreicht werden.

Zur Bildung der nächsten Generation werden die durch die Selektion identifizierten Individuen aufgenommen und $|Q|$ neue Individuen (Kinder) durch *Variation* der bestehenden gebildet. Die einfache Kreuzung von zwei Chromosomen über eine Bruchstelle führt für das Permutationsproblem in der Regel zu einer ungültigen Lösung, da Elemente mehrfach und andere damit nicht enthalten sind. Aus diesem Grund werden je zwei zufällig bestimmte Individuen variiert, indem zwei Bruchstellen identifiziert werden und die Sequenz des ersten Chromosoms zwischen den Bruchstellen für das neue Chromosom übernommen wird. Alle nicht in dieser Sequenz enthaltenen Elemente werden in der Reihenfolge des Chromosoms des zweiten Individuums ergänzt, so dass ein zulässiges Individuum entsteht. Ein zweites Individuum wird analog unter Verwendung der Sequenz des zweiten Individuums und Ergänzung aus dem Chromosom des ersten gebildet. Das Vorgehen der Variation ist in Abbildung 8.2 anhand eines Beispiels mit sieben Fahrten illustriert. Zusätzlich zur Bildung neuer Individuen wird die Suche nach besseren Lösungen durch Mutation der Population diversifiziert. In jeder Iteration besteht für jedes Individuum eine geringe Wahrscheinlichkeit der Mutation, bei der zwei zufällig bestimmte Positionen des Chromosoms miteinander vertauscht werden.

8.2.2 Fix&Optimize Verfahren

Als alternativer Ansatz wird im Folgenden eine Heuristik vorgestellt, die durch die Arbeit von [Bokinge und Hasselström, 1980] inspiriert wurde. Die Autoren konstruieren eine Lösung für das VSP-TW, indem sie ausgehend von einer ungültigen

Lösung mit verkürzten Fahrtzeiten sukzessive Zeitfenster fixieren und die resultierende Problemstellung als SD-VSP lösen (s. Abschnitt 4.4).

Das *Fix&Optimize Verfahren* erweitert dieses Vorgehen, indem der gültige Lösungsraum als Kombination von unterschiedlichen zeitlichen Verschiebungen von Servicefahrten interpretiert wird und dieser mit Meta-Heuristiken durchsucht werden kann. Für eine fixierte Verschiebung einer Fahrt können die Abfahrts- und Ankunftszeit des originalen Fahrplans angepasst werden und der resultierende Fahrplan effizient als SD-VSP optimiert werden.

Sei r_i die Anzahl der möglichen Ausprägungen für die Verschiebung der Servicefahrt $i \in T$ einschließlich der originalen Abfahrtszeit. Durch die mögliche Kombination aller diskreten Ausprägungen ist die Anzahl der möglichen Lösungen für ein Fix&Optimize Verfahren: $\prod_{i \in T} r_i$. Für den Fall, dass alle Fahrten bis zu t Minuten früher oder später gefahren werden dürfen, existieren daher $(2t + 1)^{|T|}$ gültige Lösungen.

Zur Anwendung des Fix&Optimize Verfahrens kann, wie bei der Sortierheuristik, eine beliebige Meta-Heuristik verwendet werden. Exemplarisch wird im Folgenden die Konzeption eines evolutionären Algorithmus gezeigt.

Evolutionärer Algorithmus für Fix&Optimize Verfahren

Im evolutionären Algorithmus ist jede Lösung durch die Ausprägung des je Servicefahrt möglichen Zeitfenstern definiert. Wie bereits in Abschnitt 8.1 erläutert, können die möglichen zeitlichen Verschiebungen von Servicefahrten als diskrete Möglichkeiten unterschieden werden, die in Zeiteinheiten von je einer Minute modelliert sind. Zur *Darstellung* eines Individuums ist daher ein Chromosom der Länge $|T|$ gegeben, in dem jedes Gen die Verschiebung einer Servicefahrt relativ zur Abfahrtszeit im originalen Fahrplan angibt. Jedes Gen enthält eine ganze Zahl (Genotyp), die den Umfang der späteren Abfahrt in Minuten angibt (Phenotyp). Für negative ganze Zahlen entspricht die Angabe einer früheren Bedienung der Servicefahrt. Die Gene können die für eine Servicefahrt erlaubten relativen Verschiebungen als Ausprägungen annehmen.

Das *Ziel* der Heuristik ist, wie in der Sortierheuristik, das Finden einer Lösung mit der geringsten Anzahl an notwendigen Fahrzeugen zur Bedienung des Fahrplans.

Um die Fitness der Individuen zu berechnen, wird in der *Bewertungsfunktion* ein um die relativen Angaben der Verschiebungen angepasster Fahrplan erstellt und mit einem Algorithmus zur Optimierung von SD-VSP gelöst. Aus dieser Lösung kann die optimale Fahrzeuganzahl für die fixierten Zeiten gewonnen werden und der Fitness des Individuums zugewiesen werden.

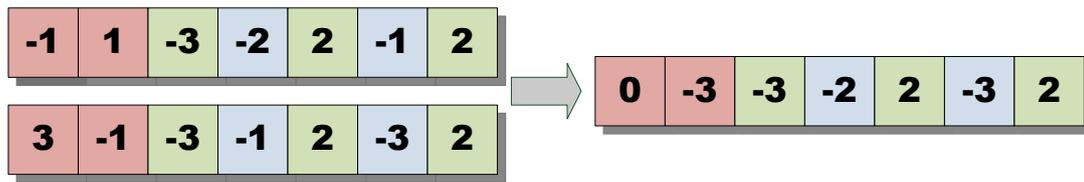


Abbildung 8.3: Variation zweier Individuen für das Fix&Optimize Verfahren

In der *Selektion* werden aus der aktuellen Generation P_t die $|Q_t|$ Individuen mit der schlechtesten Fitness ausgewählt und aus der Population entfernt. Die Größe der Population bleibt über den Verlauf des Algorithmus konstant.

Um in der *Variation* die neuen Individuen Q_t zu erzeugen, werden für die Bildung eines neuen zwei Eltern zufällig aus der bestehenden Population gewählt. Die Chromosomen der Eltern werden über einen *Fusion-Operator* kombiniert (vgl. [Beasley und Chu, 1996]). Die Vorgehensweise wird anhand eines in Abbildung 8.3 illustrierten Beispiels mit sieben Servicefahrten verdeutlicht: Die Gene der beiden Eltern werden paarweise miteinander verglichen und der Wert des neuen Gens anhand folgender Entscheidungsregeln bestimmt:

1. Wenn die Ausprägungen der beiden Gene gleich ist, wird der Wert für das neue Gen übernommen (grün dargestellt).
2. Wenn sich die Werte unterscheiden, aber die gleiche Tendenz hinsichtlich einer früheren (≤ 0) oder späteren Bedienung (≥ 0) der Servicefahrt aufweisen, wird ein zufällig bestimmter Wert zwischen denen der Eltern bestimmt (blau dargestellt).
3. Wenn die Ausprägungen der Elterngene eine unterschiedliche Tendenz aufweisen (d.h. ein Gen > 0 und das andere < 0), wird der Wert des neuen Gens zufällig im Bereich des zulässigen Zeitfensters bestimmt.

Durch dieses Vorgehen wird aus je zwei Individuen ein neues Individuum mit generiert, so dass ggf. Individuen mehrfach als Eltern verwendet werden. Da durch die zufällige Bestimmung des neuen Wertes in der dritten vorgestellten Entscheidungsregel bereits eine Diversifizierung der Suche vorgenommen wird, findet keine zusätzliche Mutation der Lösungen statt.

8.2.3 Adaption von VRP-Methoden

Neben dem VSP existieren eine Vielzahl von weiteren Problemstellungen im Bereich der Fahrzeugeinsatzplanung. Ein Beispiel ist das *vehicle routing problem*

(VRP) bei dem die Routen für Fahrzeuge geplant werden, die von einem zentralen Depot unterschiedliche Kunden beliefern und zusätzliche Restriktionen (z.B. maximale Weglänge oder Kapazität) berücksichtigen müssen. Ansätze für Optimierungsmethoden zur Lösung spezieller VRPs werden bereits seit Beginn der OR-Forschung entwickelt (vgl. [Dantzig und Ramser, 1959]), so dass sich eine Vielzahl unterschiedlicher Unterproblemklassen entwickelt haben (vgl. [Lenstra und Kan, 1981]). Aufgrund der langjährigen Forschung und der hohen Praxisrelevanz der VRP wurden bereits eine Vielzahl unterschiedlicher exakter und heuristischer Lösungsmethoden entwickelt (vgl. [Laporte, 1992] und [Cordeau et al., 2002]).

Die Problemstellungen VSP und VRP beinhalten eine Vielzahl von Gemeinsamkeiten, wenn die Servicefahrten des VSP als Kunden des VRP interpretiert werden:

- Jede/r Servicefahrt/Kunde muss genau ein mal bedient/beliefert werden.
- Jede Fahrzeugroute muss im gleichen Depot starten und enden.
- Nach der Bedienung/Belieferung von Servicefahrten/Kunden können alle anderen Haltestellen/Kundenstandorte angefahren werden (Deadheading).
- Durch unterschiedliche Fahrzeugtypen bzw. Fahrzeugtypgruppen können die Kostenstrukturen für Fahrzeuge sowie die zulässige Bedienung/Belieferung von Servicefahrten/Kunden erforderlich sein.

Neben den Gemeinsamkeiten der Problemstellungen unterscheiden sie sich allerdings in den folgenden Punkten:

- Im Gegensatz zu der Belieferung eines Kunden (VRP) ändert sich bei der Bedienung einer Servicefahrt sowohl die Zeit (Fahrtdauer) als auch ggf. der Ort (Abfahrtshaltestelle \neq Ankunftshaltestelle).
- Die Berücksichtigung von Kapazitäten für Fahrzeuge im VRP entspricht der Erweiterung des VSP-RCs.
- Die Servicefahrten müssen zu einem festgelegten Zeitpunkt bedient werden, die Belieferung der Kunden ist unabhängig von der Zeit.

Die beiden erstgenannten Unterschiede sind hinsichtlich der Konzeption von Optimierungsmethoden für beide Problemstellungen einheitlich modellierbar, da jeweils ein Spezialfall bei einer Problemstellung auftritt: im ersten Fall entspricht die Belieferung eines Kunden (VRP) einer Servicefahrt, die die gleiche Abfahrts- und Ankunftshaltestelle hat und eine Fahrtdauer von 0 Minuten hat. Im zweiten

Fall enthalten die im VSP verwendeten Fahrzeuge eine unendlich (oder ausreichend große) Kapazität für alle im VRP betrachteten Ressourcen. Dies gilt nicht für den dritten Unterschied, daher werden im Folgenden jeweils spezielle Problemklassen des VSPs und des VRPs betrachtet, die eine gleiche Betrachtung der formalen Problemklassen ermöglicht.

Bei der Berücksichtigung von Zeitfenstern für die Bedienung von Servicefahrten (VSP-TW) bzw. für die Belieferung von Kunden (VRP-TW) entfällt der dritte zuvor dargestellte Unterschied der beiden Problemstellungen. Für das VSP bedeutet die Einführung von Zeitfenstern eine Aufweichung der festgelegten Fahrtzeiten, so dass zusätzliche Freiheitsgrade zur Optimierung entstehen. Bei dem VRP führen Zeitfenster dagegen zu einer Einschränkung der möglichen Routen, da Kunden nicht zu einer beliebigen Zeit, sondern innerhalb ihres Zeitfensters bedient werden müssen. Daher entspricht das VSP-TW bis auf der Unterscheidung in Servicefahrt und Kunde dem VRP-TW. Da die innerhalb von VRP-TW betrachteten Zeitfenster in der Regel vergleichsweise groß sind (z.B. mehrere Stunden), ist im Rahmen dieser Arbeit die Idee entstanden, Konzepte erfolgreicher Lösungsmethoden für das VRP-TW auf den Anwendungsfall der taktischen Planung mit Zeitfenstern zu übertragen.

Aufgrund langjähriger Forschung existieren eine Vielzahl von Ansätzen zur Lösung des VRP-TWs. Für die Anwendung auf das VSP-TW wurden zwei Methoden ausgewählt, die auf Instanzen des VRP-TWs im Vergleich zu anderen Ansätzen die größte Effizienz zeigen (vgl. [Bräysy und Gendreau, 2005a] und [Bräysy und Gendreau, 2005b]). Dies ist zum einen die Tabu Suche (s. Abschnitt 3.4.1) von Cordeau, Laporte und Mercier, bei der in jeder Iteration genau ein Kunde in eine andere Route eingefügt wird und die durch zusätzliche Techniken (z.B. Aspirationskriterien und dynamische Bestrafungsfunktionen für ungültige Lösungen) ergänzt ist (vgl. [Cordeau et al., 2001]). Als zweite Methode ist eine Variable Nachbarschaftssuche (s. Abschnitt 3.4.1) von Polacek, Hartl, Doerner und Reimann ausgewählt worden, in der die Nachbarschaften Kundensequenzen aus zwei Routen auswählen und diese tauschen. Durch die Möglichkeit einer parametrisierten Bildung werden im Rahmen der VNS unterschiedlich große Nachbarschaften verwendet (vgl. [Polacek et al., 2004]).

Die zwei Methoden wurden im Rahmen dieser Arbeit auf der Basis der in den Veröffentlichungen dargestellten Informationen implementiert. Zusätzlich wurden problemspezifische Anpassungen durchgeführt, die Effizienz der Methoden hinsichtlich der Lösung von VSP-TW steigern konnten. Hierzu gehörten unter anderem die Anpassung der Aspirationskriterien, so dass ein tabu gesetzter Zug bei der Einsparung eines Umlaufs getätigt werden darf. Eine entscheidende Erweiterung stellt der Umgang mit so genannten Zielfunktionsplateaus dar, die für

viele ähnliche Lösungen einen gleichen Zielfunktionswert (Anzahl der Umläufe) haben, so dass die Methoden die Qualität dieser Lösungen nicht unterscheiden können. Durch eine Erweiterung der Zielfunktion um eine Steuerungsfunktion, die die Varianz der Umlauflängen mit in die Bewertung einbezieht, wird die Bildung von Umläufen mit wenigen Fahrten bevorzugt, so dass bei einer Verteilung der wenigen restlichen Fahrten ein Umlauf eingespart werden kann.

8.2.4 Numerische Ergebnisse

Für die numerischen Tests wurden die in Abschnitt 6.1.1 vorgestellten realen Problem instanzen zu SD-VSP konvertiert, da für den vorliegenden Anwendungsfall die Analyse der Fahrplanpotentiale unabhängig von zusätzlichen Nebenbedingungen betrachtet wird. Die Instanzen enthalten nur noch ein Depot und einen Fahrzeugtypen und werden im Folgenden mit „konv’ x “¹ bezeichnet. Die Tests wurden auf dem in Abschnitt 8.1.4 beschriebenen System ausgeführt.

Für die Überprüfung der in den vorangegangenen Abschnitten vorgestellten Heuristiken wurden die Instanzen mit großen Zeitfenstern von ± 10 und ± 20 Minuten für alle Servicefahrten durchgeführt. Nach einer maximalen Laufzeit von einer Stunde (3.600 Sekunden) je Instanz wurden die Verfahren abgebrochen. Die durch die großen Zeitfenster bedingten komplexen Problemstellungen sind nicht mehr mit LP-basierten Methoden, wie Algorithmen aus Standardoptimierungssoftware oder dem in Abschnitt 8.1.3 vorgestellten Kantengenerierungsansatz, lösbar. Aus diesem Grund sind die in den Abbildungen 8.4 und 8.5 dargestellten Ergebnisse relativ zu einem zufälligen Suchalgorithmus gegeben. Dieser Algorithmus ist eine Sortierheuristik (s. Abschnitt 8.2.1), bei der in jeder Iteration die Reihenfolge der Servicefahrten zufällig bestimmt wird.

Die Abbildungen zeigen für alle Instanzen den Grad der erreichten Verbesserung. Der evolutionäre Algorithmus zur Steuerung der Sortierheuristik erreicht eine Verbesserung von durchschnittlich ca. 20 %. Dies zeigt, dass Ansatz der Meta-Heuristik für das Konzept geeignet ist. Allerdings werden mit den anderen Verfahren deutlich größere Verbesserungen erreicht. Vor allem der evolutionäre Algorithmus für die Fix&Optimize Heuristik findet für die Mehrzahl der Instanzen die vergleichsweise beste Lösung. Durch die in der Fix&Optimize Heuristik notwendige Lösung von großen SD-VSPs mit Hilfe von LP-basierten Methoden, können große mathematische Modelle (*real_12* - *real_15*) aufgrund unzureichenden Speichers nicht aufgebaut werden. Die zwei angepassten Verfahren zur Lösung des VRP-TW erreichen für Zeitfenster von ± 10 in der Regel schlechtere Lösungen als das Fix&Optimize Verfahren, aber können auch die größten Instanzen lösen.

¹mit $x \in [1, 15]$ entsprechend der Nummerierung aus Abschnitt 6.1.1

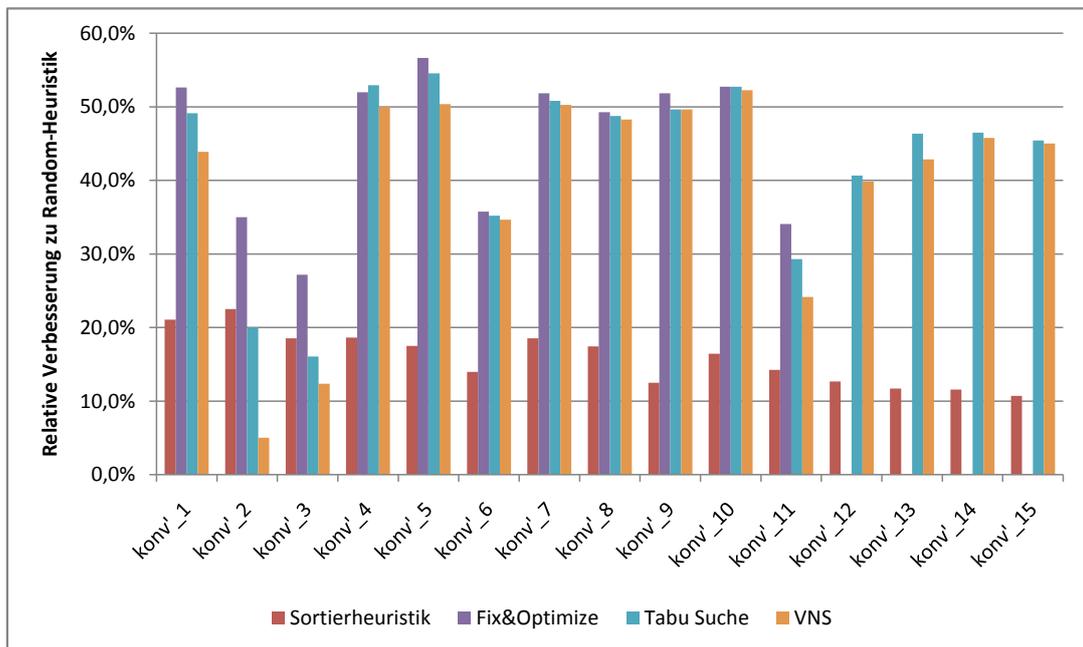


Abbildung 8.4: Ergebnisse für taktische Planung mit Zeitfenstern von ± 10

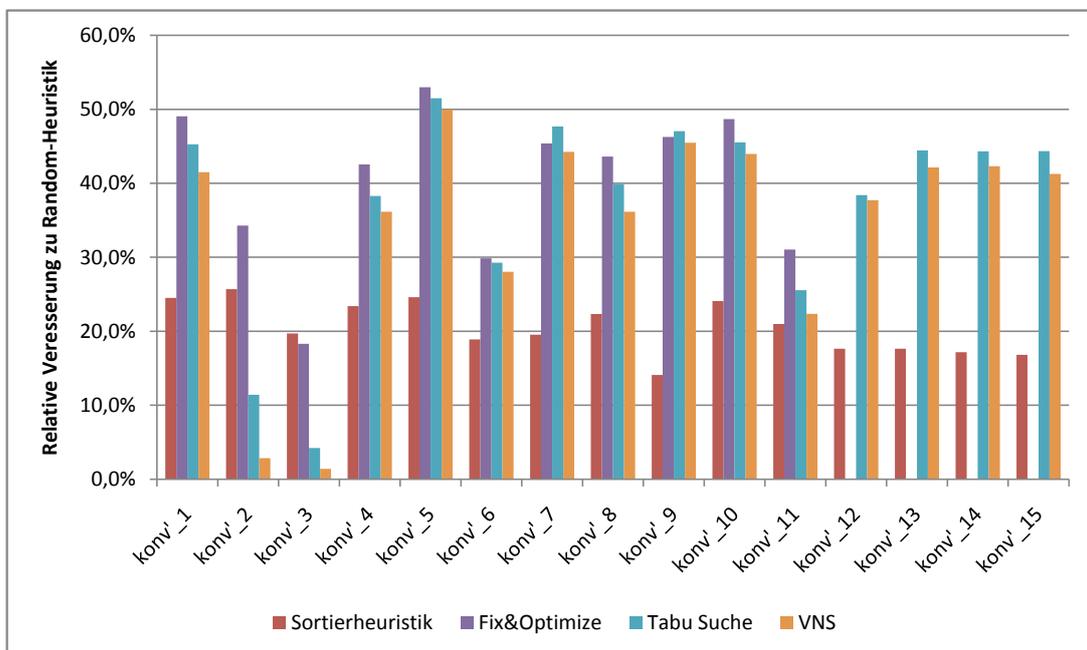


Abbildung 8.5: Ergebnisse für taktische Planung mit Zeitfenstern von ± 20

Für größere Zeitfenster (± 20) zeigen die VRP-Methoden ein vergleichsweise besseres Verhalten, so dass sie zu mehreren Instanzen die besten Lösungen finden. Die Tabu Suche erreicht für alle Instanzen bessere Lösungen im Vergleich zum VNS.

Zur Einschätzung der allgemeinen Lösungsqualität der vorgestellten Methoden wurden kleine Instanzen mit bis zu 350 Fahrten gebildet, für die ein exaktes Optimum bestimmt werden konnte. Für diese Instanzen konnte die Fix&Optimize Heuristik immer die optimale Lösung hinsichtlich der Anzahl der Fahrzeuge finden, so dass dies ein Indiz für ein hohes Qualitätsniveau der dargestellten Verfahren ist.

Die beschriebenen Beobachtungen können anhand der unterschiedlichen Ausrichtungen der Lösungsverfahren erklärt werden. Das Konzept der Sortierheuristik bietet den Vorteil, dass die Anzahl der möglichen Lösungsdarstellungen (die möglichen Sortierungen der Servicefahrten) unabhängig von der Größe der Zeitfenster ist. Aufgrund der vergleichsweise großen Anzahl ist eine Sortierheuristik daher nicht für die betrachteten Problemstellungen geeignet. Das Verfahren ist aufgrund der abhängig von der Fahrtenanzahl stark zunehmenden Lösungsraumgröße nur für Problemstellungen mit wenig Servicefahrten und großen Zeitfenstern geeignet.

Die vom VRP-TW adaptierten Methoden sind in ihrer Konzeption auf die Betrachtung von sehr großen Zeitfenstern ausgelegt. Dadurch ist das tendenziell bessere Verhalten auf den Problemstellungen mit größeren Zeitfenstern zu erklären. Im Gegensatz zum Fix&Optimize Verfahren, das auf den meisten Instanzen die besten Lösungen findet, haben die VRP-Verfahren keine hohen Anforderungen bezüglich des benötigten Speichers.

Für die Lösung von Problemen der *taktischen Planung mit Zeitfenstern* ist daher das Fix&Optimize Verfahren geeignet, sofern eine Lösung der Unterprobleme innerhalb des verfügbaren Speichers möglich ist. Sollte das Fix&Optimize Verfahren nicht anwendbar sein oder noch weitaus größere Zeitfenster betrachtet werden, ist die vom VRP-TW adaptierte Tabu Suche eine geeignete Wahl.

Durch das Konzept der iterativen Suche nach besseren Lösungen, das alle vorgestellten Verfahren verfolgen, existiert bei Abbruch der Optimierung zu einem beliebigen Zeitpunkt eine gültige Lösung. Da bessere Lösungen gefunden werden können, je mehr Laufzeit für ein Verfahren verfügbar ist, kann der Trade-Off zwischen Zeit und Qualität skaliert werden und somit die erforderliche Flexibilität für den Anwendungsfall der *taktischen Planung mit Zeitfenstern* gewährleistet werden.

9 Zusammenfassung und Ausblick

Im Rahmen der vorliegenden Arbeit wurden Optimierungsmethoden mit einer Ausrichtung auf reale Planungssituationen thematisiert. Eine Zielsetzung bestand in der Identifikation der Anwendungsfälle für den Einsatz von Optimierungsmethoden innerhalb des Planungsprozesses im ÖPNV sowie der Analyse realer Probleminstanzen hinsichtlich besonderer Eigenschaften. Darauf aufbauend sollte ein Methodenframework konzipiert und implementiert werden, das effiziente Lösungsmethoden für die Optimierung Umlaufplanungsproblemen enthält. Zur Anwendung der Methoden auf die identifizierten Anwendungsfälle sollten Strategien zur Erfüllung individueller Planungsanforderungen entwickelt werden, die eine effiziente Lösung der Problemstellungen unter Einbeziehung aller praxisrelevanten Nebenbedingungen ermöglichen. Darüber hinaus sollten erstmalig individuelle Anforderungen der Anwendungsfälle sowie die Vermeidung eines Black-Box-Verhaltens der Lösungsmethoden explizit berücksichtigt werden.

Das klassische Vorgehen innerhalb der Planung im ÖPNV umfasst eine Vielzahl von strategischen bis operativen Planungen, die in *Kapitel 3* beschrieben wurden. Erstmals wurden im Rahmen dieser Arbeit neue Anwendungsfälle für die Lösung von Umlaufplanungsproblemen gezeigt, die die Planung auch bei strategischen und taktischen Entscheidungen unterstützen und zu einer effizienteren Planung beitragen können. Für die sechs identifizierten Anwendungsfälle wurden Einsatzmöglichkeiten aufgezeigt und die Anforderungen an Optimierungsmethoden beschrieben. Darüber hinaus wurden unterschiedliche Problemstellungen und praxisrelevante Erweiterungen aufgezeigt und formal beschrieben.

Unter Verwendung der formalen Problembeschreibung wurde in *Kapitel 4* ein umfassender Überblick über den Stand der Technik gegeben. Erstmals wurden die Veröffentlichungen zur Umlaufplanung hinsichtlich der Problemstellung klassifiziert und in einer einheitlichen Notation dargestellt, so dass ein vollständiger Überblick über die Forschung aus mehr als 40 Jahren aufgezeigt wurde.

Auf der Basis der aufgestellten Anforderungen an Optimierungsmethoden für die Umlaufplanung und dem Stand der Technik wurden in *Kapitel 5* die Handlungsbedarfe dargestellt, um die Lücke zwischen realen Anforderungen und der bisherigen Forschung zu schließen. Aus den Bedarfen wurden die konkreten Ziele dieser Arbeit abgeleitet und das Vorgehen zur Umsetzung aufgezeigt.

Die bestehenden Optimierungsansätze aus der Literatur wurden entweder anhand künstlich generierter oder realer Instanzen validiert. In *Kapitel 6* wurden diese Instanzen miteinander verglichen und ihre Struktur analysiert, wodurch unterschiedliche Eigenschaften nachgewiesen werden konnten. Die Abhängigkeit der Effizienz gängiger Modellierungsarten von diesen Eigenschaften wurde nachgewiesen und durch geeignete Kennzahlen abgebildet. Zum ersten Mal wurde im Rahmen dieser Arbeit die Auswahl einer geeigneten Modellierung für die Abbildung realer Problemstellungen gezeigt. Einen wichtigen Beitrag stellt eine neu entwickelte Kennzahl zur Abschätzung der Komplexität von Umlaufplanungsproblemen dar, die unter Einbeziehung der Struktureigenschaften von Probleminstanzen eine Abschätzung der benötigten Optimierungslaufzeit ermöglicht.

Die unterschiedlichen Anwendungsfälle der Umlafoptimierung erfordern je nach Planungssituation eine stark unterschiedliche Ausrichtung der Lösungsmethoden, zum Beispiel hinsichtlich der akzeptierten Gesamtlaufzeit, einer schnellen Abschätzung der Lösung oder der Ausgabe gültiger Umlaufpläne zur Laufzeit. Aus diesem Grund wurde im Rahmen dieser Arbeit ein neuartiges Methodenframework entwickelt worden, mit dem die Vielzahl dieser Anforderungen berücksichtigt werden kann. Auf Basis der in [Kliwer, 2005] vorgestellten Modellierung mit Hilfe eines aggregierten Time-Space-Netzwerks, konnten Verfahren aus der Literatur angepasst und erweitert werden, so dass eine schnelle Berechnung von unteren Schranken und einer gültigen, sowie bezüglich der Fahrzeuganzahl optimalen, Lösung möglich war. Basierend auf diesen Methoden wurde ein Verbesserungsverfahren entwickelt, das durch die Ausnutzung der in Kapitel 6 identifizierten Eigenschaften realer Instanzen eine effiziente Optimierung ermöglicht. Durch die Konzeption eines iterativen Prozesses, in dem die Lösungsqualität schrittweise verbessert wird, kann das Verfahren flexibel auf die Anforderungen eines Anwendungsfalls angepasst werden und gleichzeitig die Konvergenz zu einer optimalen Lösung sicherstellen. Durch dieses in *Kapitel 7* vorgestellte Methodenframework konnten alle Anwendungsfälle der Umlaufplanung ohne Zeitfenster mit geeigneten Strategien unterstützt werden. Die geeignete Wahl der Parameter wurde anhand der Ergebnisse numerischer Tests gezeigt, so dass sich die Methode für eine konkrete Planungssituation automatisch auf die Eigenschaften der Instanz einstellen kann. Anhand umfangreicher numerischer Tests wurde die Effizienz der Methode belegt. Es wurde nachgewiesen, dass die Methode im Vergleich zu bestmöglich eingestellter Standardoptimierungssoftware eine Vielzahl von besseren Lösungen findet und die Instanzen optimal lösen kann. Für mittlere und große Instanzen konnte die optimale Lösung bei Vermeidung eines Black-Box-Verhaltens sogar in kürzerer Zeit gefunden werden. Durch die Kombination unterschiedlicher Lösungstechniken konnten erstmals gültige Lösungen

für sehr große Instanzen gefunden werden, so dass die vorliegende Arbeit auch die Optimierung komplexer Fragestellungen, die zum Beispiel in Großstädten mit einer heterogenen Fahrzeugflotte auftreten, ermöglicht.

Für die Unterstützung von Anwendungsfällen der Umlaufplanung, in denen Zeitfenster für Fahrplanfahrten zugelassen werden, sind spezielle Modellanpassungen und Lösungsmethoden erforderlich. In *Kapitel 8* wurde das Methodenframework auf das in [Kliwer et al., 2006a] um Zeitfenster erweiterte Modell angepasst und um spezielle Lösungstechniken erweitert. Zusätzlich wurde das Modell um die Bedingung einer gleichmäßigen zeitlichen Verschiebung von Fahrten einer Linie erweitert. Numerische Ergebnisse belegen, dass erstmalig auch für große Instanzen mit Zeitfenstern Lösungen gefunden werden konnten. Um im Rahmen von Potentialanalysen der taktischen Planung im ÖPNV auch Probleme mit großen Zeitfenstern lösen zu können, wurden in dieser Arbeit neuartige Ansätze vorgestellt, die einen Einsatz von Meta-Heuristiken für die Lösung ermöglichen. Exemplarisch konnte durch die Konzeption von zwei evolutionären Algorithmen und weiteren Meta-Heuristiken das Verhalten und die Lösungsqualität dieser Ansätze gezeigt werden, die im Vergleich zu einfachen Methoden eine deutliche Verbesserung der Lösung ermöglichen.

Zusammenfassend kann festgestellt werden, dass die Ziele der Arbeit wie folgt erfüllt sind: Durch die umfangreiche Analyse realer Problemstellungen konnten Eigenschaften identifiziert werden, mit denen die Entwicklung eines effizienten Methodenframeworks für die Lösung von Umlaufplanungsproblemen ermöglicht wurde. Durch die Ausrichtung auf die Anforderungen neu identifizierter realer Planungssituationen kann der Planungsprozess in vielen Bereichen verbessert werden, so dass Planer besser in ihrer Aufgabe unterstützt werden und der ÖPNV durch eine kosteneffizientere Planung nachhaltig gestärkt wird.

Die neuartige Ausrichtung der Forschung auf weitere Anwendungsfälle der Umlaufplanung im ÖPNV eröffnet die Möglichkeit weiterer Untersuchungen in diesem Bereich. So können durch eine konsequente Weiterentwicklung und Integration von Optimierungsmethoden weitere Planungsprobleme unterstützt werden. Durch die erweiterte Einbeziehung von Optimierungskomponenten in Planungssoftware für den ÖPNV kann der Planungsprozess über den bisherigen Grad hinaus unterstützt werden.

Eine Möglichkeit der Unterstützung strategischer Fragestellungen ist der Einsatz von Umlafoptimierungsmethoden in der Planung von Standortentscheidungen für Depots. Durch eine integrierte Betrachtung könnten, über die Abschätzung von Ressourcenkosten für Szenarien hinaus, optimale Standortentscheidungen zur Minimierung der Fahrzeugeinsatzkosten gefunden werden.

Darüber hinaus besteht Forschungsbedarf in der Sicherstellung von robusten

Einsatzplänen. Die im Rahmen dieser Arbeit entwickelten Erkenntnisse über die Instanzeigenschaften können Grundlage neuer Konzepte zur Erhöhung der Robustheit sein. Darüber hinaus bieten die neu entwickelten Methoden die Möglichkeit eine Vielzahl von Unterproblemen effizient zu lösen, so dass iterative Verfahren zur Steigerung der Robustheit entwickelt werden können.

Durch die neuen generischen Konzepte zur Lösung von Umlaufplanungsproblemen mit großen Zeitfenstern, ist die Verwendung beliebiger Meta-Heuristiken zur Lösung dieser Problemstellung möglich. Dadurch können im Rahmen weiterer Forschungen neue Lösungsmethoden entwickelt werden, die ohne Kenntnis der Problemeigenschaften eine Verbesserung der Lösungsqualität ermöglichen können.

Abkürzungsverzeichnis

Abkürzung	Bedeutung
B&B	Branch & Bound Verfahren
B&C	Branch & Cut Verfahren
B&P	Branch & Price Verfahren
IMCCF	Integer Multicommodity Flow Problem
IP	Integer Program
LDP	Lagrange Dual Problem
LP	Linear Program
MCCF	Multicommodity Flow Problem
MCF	Minimum Cost Flow Problem
MD-VSP	Multiple Depot Vehicle Scheduling Problem
MDP	Multiple Dantzig Pricing
MTSP	Multiple Time Slot Pricing
MVTG-VSP	Multiple Vehicletype Group Vehicle Scheduling Problem
O-D-Matrix	Origin-Destination-Matrix
ÖPNV	Öffentlicher Personennahverkehr
RMP	Restricted Master Problem
SA	Simulated Annealing
SCP	Set Covering Problem
SD-VSP	Single Depot Vehicle Scheduling Problem
SPP	Set Partitioning Problem
TP	Transportation Problem
TSN	Time Space Netzwerk (Modell)
TSP	Traveling Salesman Problem
VBN	Verbindungsbasiertes Netzwerk (Modell)
VNS	Variable Neighborhood Search
VRP	Vehicle Routing Problem
VRP-TW	Vehicle Routing Problem with Time Windows
VSP	Vehicle Scheduling Problem
VSP-RC	Vehicle Scheduling Problem with Route Constraints
VSP-TW	Vehicle Scheduling Problem with Time Windows

Abbildungsverzeichnis

2.1	Sequentieller Planungsprozess im ÖPNV	6
2.2	Fahrplan einer Linie der PaderSprinter (Paderborn)	7
2.3	Auslastungskurven für reale Fahrpläne	15
2.4	Reduzierung der benötigten Fahrzeuge durch Zeitfenster	20
3.1	Column Generation für MIPs	30
4.1	Zuweisungsmodell für das SD-VSP	41
4.2	Transportmodell für das SD-VSP	44
4.3	Netzwerkflussmodell für das SD-VSP	45
4.4	Einfaches Flussmodell für das MD-VSP	48
4.5	Mehrgüter Flussmodell für das MD-VSP	51
4.6	Time-Space-Netzwerk für das MVTG-VSP	65
4.7	Modellierung diskreter Zeitfenster nach [Kliwer, 2005]	70
4.8	Multilevel Modell für das VSP-RC	72
4.9	Rückkanten Modell für das VSP-RC	73
6.1	Qualität der Schranken durch unterschiedliche LP-Relaxationen	89
6.2	Vergleich der Auslastung beispielhafter Fahrpläne	91
6.3	Strukturelle Unterschiede von realen, konvertierten und künstlich generierten Instanzen	95
6.4	Auslastungsgraph eines Fahrplans mit Auslastungsspitze Ψ_{max}	99
6.5	Unterschiede in Fahrplanstrukturen bei gleicher maximalen Auslastung	100
6.6	Vergleich von Abschätzungen der optimalen Fahrzeuganzahl	101
6.7	Vergleich der Laufzeiten für LP- und IP-Phase	104
6.8	Struktur der Entscheidungsmatrix des Mehrgüter Flussmodells	106
7.1	Vorgehen zur Berücksichtigung der Anforderungen von Anwendungsfällen ohne Zeitfenster	113
7.2	Beispiel: Möglichkeiten der Zuweisung von Umläufen zu Depots und Fahrzeugtypen	118
7.3	Alternative Netzwerkmodellierung von Servicefahrten	124

7.4	Konvergenzverhalten bei Warmstart von dualer Basis	129
7.5	Einbettung des Kantengenerierungsalgorithmus	143
7.6	Dominanz der Pricingstrategie LRP_{cc} gegenüber alternativen Strategien in der Kantengenerierung	154
7.7	Vergleich der Strategien zur Lösung des Master Problems in der Kantengenerierung	156
7.8	Vergleich von Initialisierungsstrategien hinsichtlich der Konvergenz zur optimalen LP-Lösung	158
7.9	Aufhebung des Black-Box-Verhaltens durch den Einsatz von IP-Heuristiken	159
8.1	Alternative Netzwerkmodellierung für Servicefahrten mit Zeitfenstern	171
8.2	Variation zweier Individuen für Sortierheuristik	178
8.3	Variation zweier Individuen für das Fix&Optimize Verfahren . . .	180
8.4	Ergebnisse für taktische Planung mit Zeitfenstern von ± 10	184
8.5	Ergebnisse für taktische Planung mit Zeitfenstern von ± 20	184

Tabellenverzeichnis

4.1	Eigenschaften gelöster SD-VSP Probleminstanzen	47
4.2	Eigenschaften gelöster MD-VSP Probleminstanzen	61
4.3	Eigenschaften gelöster MVTG-VSP Probleminstanzen	66
6.1	Reale Probleminstanzen	85
6.2	Konvertierte Probleminstanzen	86
6.3	Beispiel 1 für einen Fahrplan	90
6.4	Beispiel 2 für einen Fahrplan	90
6.5	Analyse der Anzahl der Modellvariablen für ein Depot	92
6.6	Verhaltensunterschiede von künstlichen und realen Instanzen	97
6.7	Abschätzung der numerischen Komplexität realer Instanzen	108
7.1	Beispiel: Definition von Fahrzeugtypgruppen	116
7.2	Dimensionen der Netzwerke	147
7.3	Anzahl modellierter Kanten im TSN	147
7.4	Ergebnisse mit Standardoptimierungssoftware	149
7.5	Ergebnisse der Initialisierungsphase	151
7.6	Kürzere Laufzeiten in der IP-Phase des Kantengenerierungsalgorithmus	160
7.7	Kategorisierung der Instanzen und Zusammenfassung der numeri- schen Ergebnisse	163
8.1	Ergebnisse für operative Zeitfenster	173

Liste der Algorithmen

3.1	Column Generation Verfahren	29
3.2	Hill-climber Algorithmus	34
3.3	Variable Nachbarschaftssuche	35
3.4	Tabu Suche	36
3.5	Evolutionärer Algorithmus	37
4.1	Instanzengenerator nach [Carpaneto et al., 1989]	60
7.1	Kantengenerierungsalgorithmus	132
8.1	Konstruktionsheuristik für VSP-TW	169
8.2	Evaluierung für Sortierheuristiken	176

Literaturverzeichnis

- [Ahuja et al., 1993] Ahuja, R., Magnanti, T., und Orlin, J. (1993). *Network Flows - Theory, Algorithms, and Applications*. Prentice-Hall, Englewood Cliffs, NJ.
- [Banihashemi und Haghani, 2000] Banihashemi, M. und Haghani, A. (2000). A model for the multiple depot transit vehicle scheduling problem with route time constraints. Technical report, Civil and Environmental Engineering Department, University of Maryland.
- [Barnhart et al., 1998] Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W., und Vance, P. H. (1998). Branch-and-price: Column generation for solving huge integer programs. *Operations Research*, 46(3):316–329.
- [Bartlett, 1957] Bartlett, T. E. (1957). An algorithm for the minimum number of transport units to maintain a fixed schedule. *Naval Research Logistics Quarterly*, 4:139–149.
- [Beasley, 1993] Beasley, J. (1993). Lagrangean relaxation. In Reeves, C. R., editor, *Modern heuristic techniques for combinatorial problems*, chapter 6, pages 243–303. Blackwell, London.
- [Beasley und Chu, 1996] Beasley, J. und Chu, P. (1996). A genetic algorithm for the set covering problem. *European Journal of Operational Research*, 94(2):392–404.
- [Bennington und Rebibo, 1975] Bennington, G. und Rebibo, K. (1975). Overview of rucus vehicle scheduling program (blocks). In Bergmann, D. und Bodin, L., editors, *Preprints: Workshop on Automated Techniques for Scheduling of Vehicle Operators for Urban Public Transportation Services*.
- [Bertossi et al., 1987] Bertossi, A. A., Carraresi, P., und Gallo, G. (1987). On some matching problems arising in vehicle scheduling models. *Networks*, 17:271–281.
- [Bertram und Winckler, 1990] Bertram, H.-H. und Winckler, J. (1990). Scheduling on microcomputers using microbus. In Daduna, J. und Wren, A., editors,

- Computer-Aided Transit Scheduling: Procs. of the Fourth International Workshop on Computer Aided Scheduling of Public Transport*, volume 308 of *Lecture Notes in Economics and Mathematical Systems*, pages 188–199, Berlin. Springer.
- [Bianco et al., 1994] Bianco, L., Mingozzi, A., und Ricciardelli, S. (1994). A set partitioning approach to the multiple depot vehicle scheduling problem. *Optimization Methods and Software*, 3:163–194.
- [Bixby, 1992] Bixby, R. E. (1992). Das Implementieren des Simplex-Verfahrens: Die Startbasis. Technical report, Konrad-Zuse-Zentrum für Informationstechnik Berlin.
- [Bixby, 2002] Bixby, R. E. (2002). Solving real world linear programs: A decade and more of progress. *Operations Research*, 50(1):3–15.
- [Blais und Rousseau, 1990] Blais, J.-Y. und Rousseau, J.-M. (1990). Overview of hastus current and future versions. In Daduna, J. und Wren, A., editors, *Computer-Aided Transit Scheduling: Procs. of the Fourth International Workshop on Computer Aided Scheduling of Public Transport*, volume 308 of *Lecture Notes in Economics and Mathematical Systems*, pages 175–187, Berlin. Springer.
- [Blum und Roli, 2003] Blum, C. und Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, 35(3):268–308.
- [BMVBS, 2007] BMVBS (2007). Tiefensee: Busgewerbe sichert 750.000 Arbeitsplätze. *Pressemitteilungen Bundesministerium für Verkehr, Bau und Stadtentwicklung*, 2007(20).
- [Bodin et al., 1985] Bodin, L., Ball, M., Duguid, R., und Mitchell, M. (1985). The vehicle scheduling problem with interlining. In Rousseau, J.-M., editor, *Computer Scheduling of Public Transport 2*, pages 427–450. Elsevier Science Publishers B.V. (North-Holland).
- [Bodin und Golden, 1981] Bodin, L. und Golden, B. (1981). Classification in vehicle routing and scheduling. *Networks*, 11(2):97–108.
- [Bodin et al., 1983] Bodin, L., Golden, B., Assad, A., und Ball, M. (1983). Routing and scheduling of vehicles and crews: the state of the art. *Computers & Operations Research*, 10(2):63–211.

- [Bodin und Rosenfield, 1976] Bodin, L. und Rosenfield, D. (1976). Estimation of the operating cost of mass transit systems. Technical report, State University of New York.
- [Bodin et al., 1978] Bodin, L., Rosenfield, D., und Kydes, A. (1978). Ucost: A micro approach to a transit planning problem. *Urban Analysis*, 5:47–69.
- [Bokinge und Hasselström, 1980] Bokinge, U. und Hasselström, D. (1980). Improved vehicle scheduling in public transport through systematic changes in the time-table. *European Journal of Operational Research*, 5:388–395.
- [Borndörfer et al., 2004] Borndörfer, R., Grötschel, M., und Pfetsch, E. (2004). Models for line planning in public transport. Technical Report ZR 04-10, Konrad-Zuse Zentrum für Informationstechnik.
- [Borndörfer et al., 2004] Borndörfer, R., Löbel, A., und Weider, S. (2004). A bundle method for integrated multi-depot vehicle and duty scheduling in public transit. Technical Report ZR 04-14, Konrad-Zuse Zentrum für Informationstechnik, Berlin, Germany.
- [Bräysy und Gendreau, 2005a] Bräysy, O. und Gendreau, M. (2005a). Vehicle routing problem with time windows, part i: Route construction and local search algorithms. *Transportation Science*, 39(1):104–118.
- [Bräysy und Gendreau, 2005b] Bräysy, O. und Gendreau, M. (2005b). Vehicle routing problem with time windows, part ii: Metaheuristics. *Transportation Science*, 39(1):119–139.
- [Bunte und Kliwer, 2006] Bunte, S. und Kliwer, N. (2006). An overview on vehicle scheduling models. Technical Report 11/2006, University of Paderborn, DS&OR Lab.
- [Bunte et al., 2005] Bunte, S., Kliwer, N., und Suhl, L. (2005). Mehrdepot-Umlaufplanung: Berücksichtigung von Verschiebeintervallen für Fahrten in einem Time-Space-Netzwerk-basierten Modell. In Haasis H., Kopfer H., S. J., editor, *Operations Research Proceedings 2005 - Selected Papers of the Annual International Conference of the German Operations Research Society (GOR)*. Springer.
- [Carpaneto et al., 1989] Carpaneto, G., Dell’Amico, M., Fischetti, M., und Toth, P. (1989). A branch and bound algorithm for the multiple depot vehicle scheduling problem. *Networks*, 19:531–548.

- [Carraraesi und Gallo, 1984] Carraraesi, P. und Gallo, G. (1984). Network models for vehicle and crew scheduling. *European Journal of Operational Research*, 16(2):139–151.
- [Ceder, 1993] Ceder, A. (1993). Minimum cost vehicle scheduling with different types of transit vehicles. In Daduna, J. R., Branco, I., und Paixao, J. M. P., editors, *Computer-aided Transit Scheduling: Proceedings of the Sixth International Workshop on Computer-aided Scheduling of Public Transport*, pages 102–114. Springer.
- [Ceder, 2001] Ceder, A. (2001). Efficient timetabling and vehicle scheduling for public transport. In *Computer-Aided Scheduling of Public Transport*, volume 505 of *Lecture Notes in Economics and Mathematical Systems*, pages 37–52, Berlin. Springer.
- [Ceder, 2004] Ceder, A. (2004). Improved lower-bound fleet size for fixed and variable transit schedules. Technical report, Transportation Research Institute, Technion-Israel Institute of Technology.
- [Ceder et al., 1990] Ceder, A., Fjornes, B., und Stern, H. I. (1990). Optibus: A scheduling package. In Daduna, J. und Wren, A., editors, *Computer-Aided Transit Scheduling: Procs. of the Fourth International Workshop on Computer Aided Scheduling of Public Transport*, volume 308 of *Lecture Notes in Economics and Mathematical Systems*, pages 212–225, Berlin. Springer.
- [Ceder und Stern, 1981] Ceder, A. und Stern, H. (1981). Deficit function bus scheduling with deadheading trip insertions for fleet size reduction. *Transportation Science*, 15(4):338–364.
- [Ceder und Stern, 1985] Ceder, A. und Stern, H. (1985). The variable trip procedure used in the autobus vehicle scheduler. In Rousseau, J.-M., editor, *Computer Scheduling of Public Transport 2*, pages 371–390. Elsevier Science Publishers B.V. (North-Holland).
- [Chamberlain und Wren, 1992] Chamberlain, M. und Wren, A. (1992). Developments and recent experience with the busman and busman ii systems. In Desrochers, M. und Rousseau, J.-M., editors, *Computer-aided Transit Scheduling: Proceedings of the Fifth International Workshop on Computer-aided Scheduling of Public Transport*, volume 386 of *Lecture Notes in Economics and Mathematical Systems*, pages 1–16, Berlin. Springer.

- [Chvatal, 1983] Chvatal, V. (1983). *Linear Programming*. Freeman, New York, USA.
- [Cordeau et al., 2002] Cordeau, J.-F., Gendreau, M., Laporte, G., Potvin, J.-Y., und Semet, F. (2002). A guide to vehicle routing heuristics. *Journal of the Operational Research Society*, 53:512–522.
- [Cordeau et al., 2001] Cordeau, J.-F., Laporte, G., und Mercier, A. (2001). A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society*, 52:928–936.
- [Costa et al., 1995] Costa, A., Branco, I., und Paixao, J. M. P. (1995). Vehicle scheduling problem with multiple type of vehicles and a single depot. In *Proceedings of the Sixth International Workshop on Computer-Aided Scheduling of Public Transport*, pages 115–129.
- [Daduna und Mojsilovic, 1988] Daduna, J. R. und Mojsilovic, M. (1988). Computer-aided vehicle and duty scheduling using the hot programme system. In *Computer-Aided Transit Scheduling*, pages 133–146.
- [Daduna et al., 1993] Daduna, J. R., Mojsilovic, M., und Schütze, P. (1993). Practical experiences using an interactive optimization procedure for vehicle scheduling. In Du, D. und Pardalos, P., editors, *Network Optimization Problems: Algorithms, Applications and Complexity*, pages 37–52. World Scientific Publishing.
- [Daduna und Paixao, 1995] Daduna, J. R. und Paixao, J. M. P. (1995). Vehicle scheduling for public mass transit - an overview. In *Proceedings of the Sixth International Workshop on Computer-Aided Scheduling of Public Transport*, volume 430 of *Lecture Notes in Economics and Mathematical Systems*, pages 76–90. Springer, Heidelberg.
- [Daduna und Völker, 1997] Daduna, J. R. und Völker, M. (1997). Fahrzeugumlaufbildung im öpnv mit unscharfen abfahrtszeiten. *Der Nahverkehr*, 11:39–43.
- [Dantzig, 1951] Dantzig, G. (1951). Application of the simplex method to the transportation problem. In Koopmans, T. C., editor, *Activity analysis of production and allocation*, pages 359–373. New York: John Wiley and Sons.
- [Dantzig und Hoffman, 1956] Dantzig, G. und Hoffman, A. (1956). On a theorem of dilworth. *Annals of Math. Studies*, 38.

- [Dantzig et al., 1955] Dantzig, G., Orden, A., und Wolfe, P. (1955). The generalized simplex method for minimizing a linear form under linear inequality restraints. *Pacific Journal of Mathematics*, 5:183–195.
- [Dantzig und Fulkerson, 1954] Dantzig, G. B. und Fulkerson, D. R. (1954). Minimizing the number of tankers to meet a fixed schedule. *Naval Research Logistics Quarterly*, 1:217–222.
- [Dantzig und Ramser, 1959] Dantzig, G. B. und Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, 6(1):80–91.
- [Dantzig und Wolfe, 1960] Dantzig, G. B. und Wolfe, P. (1960). Decomposition principle for linear programs. *Operations Research*, 8(1):101–111.
- [Darwin, 1859] Darwin, C. (1859). *On the Origin of Species*. John Murray.
- [Dell’Amico et al., 1993] Dell’Amico, M., Fischetti, M., und Toth, P. (1993). Heuristic algorithms for the multiple depot vehicle scheduling problem. *Management Science*, 39(1):115–125.
- [Desaulniers et al., 2005] Desaulniers, G., Desrosiers, J., und Solomon, M. (2005). *Column Generation*. Springer, New York.
- [Desaulniers et al., 2002] Desaulniers, G., Desrosiers, J., und Solomon, M. M. (2002). Accelerating strategies in column generation methods for vehicle routing and crew scheduling problems. In Ribeiro, C. und Hansen, P., editors, *Essays and Surveys in Metaheuristics*, pages 309–324. Kluwer, Boston.
- [Desaulniers und Hickman, 2006] Desaulniers, G. und Hickman, M. D. (2006). Public transit. In Barnhart, C. und Laporte, G., editors, *Handbooks in Operations Research & Management Science: Transportation*, volume 14 of *Handbooks in Operations Research and Management Science*, pages 69–127. North Holland, The Netherlands.
- [Desaulniers et al., 1998] Desaulniers, G., Lavigne, J., und Soumis, F. (1998). Multi-depot vehicle scheduling problems with time windows and waiting costs. *European Journal of Operational Research*, 111:479–494.
- [Desrosiers et al., 1995] Desrosiers, J., Dumas, Y., Solomon, M. M., und Soumis, F. (1995). Time constrained routing and scheduling. In Ball, M., Magnati, T., Monma, C., und Nemhauser, G., editors, *Network Routing*, volume 8 of *Handbooks in Operations Research and Management Science*, chapter 2, pages 35–139. Elsevier Science, Amsterdam.

- [Desrosiers und Lübbecke, 2005] Desrosiers, J. und Lübbecke, M. (2005). A primer in column generation. In Desaulniers, G., Desrosiers, J., und Solomon, M., editors, *Column Generation*, chapter 1, pages 1–32. Springer, New York.
- [Desrosiers et al., 1982] Desrosiers, J., Soumis, F., und Desrochers, M. (1982). Routes sur un reseau espace-temps. In *Congrès de l'ASAC - Recherche opérationnelle*, volume 3, pages 28–32.
- [Desrosiers et al., 1984] Desrosiers, J., Soumis, F., und Desrochers, M. (1984). Routing with time windows by column generation. *Networks*, 14:545–565.
- [Desrosiers et al., 1985] Desrosiers, J., Soumis, F., Desrochers, M., und Sauve, M. (1985). Routing and scheduling with time windows solved by network relaxation and branch-and-bound on time variables. In Rousseau, J.-M., editor, *Computer Scheduling of Public Transport 2*, Computer Scheduling of Public Transport 2, pages 451–471. Elsevier Science Publishers B.V. (North-Holland).
- [Destatis, 2008a] Destatis (2008a). *Datenreport 2008 - Ein Sozialbericht für die Bundesrepublik Deutschland*. Statistisches Bundesamt.
- [Destatis, 2008b] Destatis (2008b). Öffentlicher Nahverkehr 2007: Einnahmen um 2,7% gestiegen. *Pressemitteilungen Statistisches Bundesamt Deutschland*, 438.
- [Dilworth, 1950] Dilworth, R. P. (1950). A decomposition theorem for partially ordered sets. *The Annals of Mathematics*, 51(1):161–166.
- [Domschke und Drexl, 2005] Domschke, W. und Drexl, A. (2005). *Einführung in Operations Research*. Springer, Berlin.
- [El-Azm, 1985] El-Azm, A. (1985). The minimum fleet size problem and its applications to bus scheduling. In Rousseau, J.-M., editor, *Computer Scheduling of Public Transport 2*, volume 2, pages 493–512. Elsevier Science Publishers B.V. (North-Holland).
- [Ferland und Fortin, 1989] Ferland, J. A. und Fortin, L. (1989). Vehicles scheduling with sliding time windows. *European Journal of Operational Research*, 38:213–226.
- [Ferland und Michelon, 1988] Ferland, J. A. und Michelon, P. (1988). The vehicle scheduling problem with multiple vehicle types. *Journal of the Operational Research Society*, 39(6):577–583.

- [Fischetti et al., 2001a] Fischetti, M., Lodi, A., Martello, S., und Toth, P. (2001a). DEIS operations research group: Multiple depot vehicle scheduling instances. http://www.or.deis.unibo.it/research_pages/ORinstances/MD-VSP.html.
- [Fischetti et al., 2001b] Fischetti, M., Lodi, A., Martello, S., und Toth, P. (2001b). A polyhedral approach to simplified crew scheduling and vehicle scheduling problems. *Management Science*, 47(6):833–850.
- [Fischetti et al., 1999] Fischetti, M., Lodi, A., und Toth, P. (1999). A branch-and-cut algorithm for the multi depot vehicle scheduling problem. Technical report, Dipartimento di Elettronica e Informatica, Universita di Bologna, Italy.
- [Forbes et al., 1994] Forbes, M., Holt, J. N., und Watts, A. M. (1994). An exact algorithm for multiple depot bus scheduling. *European Journal of Operational Research*, 72:115–124.
- [Freling und Paixao, 1995] Freling, R. und Paixao, J. M. P. (1995). Vehicle scheduling with time constraint. In *Computer-Aided Transit Scheduling*, volume 430 of *Lecture Notes in Economics and Mathematical Systems*, pages 130–144, Berlin. Springer.
- [Freling et al., 2001] Freling, R., Wagelmans, A. P., und Paixao, J. M. P. (2001). Models and algorithms for single-depot vehicle scheduling. *Transportation Science*, 35(2):165–180.
- [Friedl, 2003] Friedl, J. E. F. (2003). *Reguläre Ausdrücke*. O'Reilly Germany.
- [Friedrich und Mott, 1998] Friedrich, M. und Mott, P. (1998). Linienerfolgsrechnung mit dem planungsprogramm visum. *Der Nahverkehr*, 10.
- [Friedrich und Nökel, 2003] Friedrich, M. und Nökel, K. (2003). Erweiterung strategischer verkehrsmodelle für die operationale planung im öv. In *Tagungsband der 19. Verkehrswissenschaftlichen Tage*. Technische Universität Dresden.
- [Friedrich und Nökel, 2004] Friedrich, M. und Nökel, K. (2004). Extending transportation planning models: From strategic modelling to operational transit planning. In *Proceedings of the 9th International Conference on Computer-Aided Scheduling of Public Transport*.
- [Fulkerson, 1961] Fulkerson, D. R. (1961). An out-of-kilter method for minimal-cost flow problems. *Journal on Applied Mathematics*, 9(1):18–27.

- [Garey und Johnson, 1979] Garey, M. und Johnson, D. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. WH Freeman & Co., New York, USA.
- [Gavish et al., 1978] Gavish, B., Schweitzer, P., und Shlifer, E. (1978). Assigning buses to schedules in a metropolitan area. *Computers & Operations Research*, 5(2):129–138.
- [Gavish und Shlifer, 1979] Gavish, B. und Shlifer, E. (1979). An approach for solving a class of transportation scheduling problems. *European Journal of Operational Research*, 3:122–134.
- [Geoffrion, 1974] Geoffrion, A. (1974). Lagrangean relaxation for integer programming. *Mathematical Programming Study*, 2:82–114.
- [Gertsbach und Gurevich, 1977] Gertsbach, I. und Gurevich, Y. (1977). Constructing an optimal fleet for a transportation schedule. *Transportation Science*, 11(1):20–36.
- [Gintner, 2008] Gintner, V. (2008). *Modelle und Lösungsverfahren für die integrierte Ressourceneinsatzplanung im öffentlichen Personennahverkehr*. PhD thesis, Fakultät für Wirtschaftswissenschaften, Universität Paderborn.
- [Gintner et al., 2005] Gintner, V., Kliewer, N., und Suhl, L. (2005). Solving large multiple-depot multiple-vehicle-type bus scheduling problems in practice. *OR Spectrum*, 27(4):507–523.
- [Gintner et al., 2006] Gintner, V., Kramkowski, S., Steinzen, I., und Suhl, L. (2006). Adaptive dienst- und umlaufplanung im öpnv. In Haasis, H.-D., Kopper, H., und Schönberger, J., editors, *Operations Research Proceedings 2005*, pages 55–60, Berlin. Springer.
- [Glover, 1986] Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers*, 5:533–549.
- [Glover et al., 1974] Glover, F., Klingman, D., und Stutz, J. (1974). Augmented threaded index method for network optimization. *INFOR*, 12(3):293–297.
- [Glover und Laguna, 1993] Glover, F. und Laguna, M. (1993). Tabu search. In Reeves, C. R., editor, *Modern heuristic techniques for combinatorial problems*, chapter 3, pages 70–150. Blackwell, London.

- [Grötschel et al., 1997] Grötschel, M., Löbel, A., und Völker, M. (1997). Optimierung des fahrzeugumlaufs im öffentlichen nahverkehr. In Hoffmann, K.-H., Jäger, W., Lohmann, T., und Schunck, H., editors, *Mathematik-Schlüsseltechnologie für die Zukunft*, pages 609–624. Springer Verlag.
- [Hadjar et al., 2006] Hadjar, A., Marcotte, O., und Soumis, F. (2006). A branch-and-cut algorithm for the multiple depot vehicle scheduling problem. *Operations Research*, 54(1):130–149.
- [Hadjar und Soumis, 2005] Hadjar, A. und Soumis, F. (2005). Window reduction for the multiple depot vehicle scheduling problem with time windows. Technical Report G-2005-54, HEC Montreal, Montreal, Canada.
- [Haghani und Banihashemi, 2002] Haghani, A. und Banihashemi, M. (2002). Heuristic approaches for solving large-scale bus transit vehicle scheduling problem with route time constraints. *Transportation Research Part A: Policy and Practice*, 36:309–333.
- [Haghani et al., 2003] Haghani, A., Banihashemi, M., und Chiang, K.-H. (2003). A comparative analysis of bus transit vehicle scheduling models. *Transportation Research Part B: Methodological*, 37:301–322.
- [Hamer und Seguin, 1992] Hamer, N. und Seguin, L. (1992). The hastus system: New algorithms and modules for the 90s. In Desrochers, M. und Rousseau, J.-M., editors, *Computer-aided Transit Scheduling: Proceedings of the Fifth International Workshop on Computer-aided Scheduling of Public Transport*, volume 386 of *Lecture Notes in Economics and Mathematical Systems*, pages 17–29, Berlin. Springer.
- [Hanisch, 1999] Hanisch, J. (1999). Die Regionalverkehr Köln GmbH und HAS-TUS. Technical report, Regionalverkehr Köln GmbH.
- [Hansen und Mladenović, 2001] Hansen, P. und Mladenović, N. (2001). Variable neighborhood search: Principles and applications. *European Journal of Operational Research*, 130(3):449–467.
- [Held et al., 1974] Held, M., Wolfe, P., und Crowder, H. (1974). Validation of subgradient optimization. *Mathematical Programming*, 6:62–88.
- [Hoffstadt, 1981] Hoffstadt, J. (1981). Computerized vehicle and driver scheduling for the hamburger hochbahn aktiengesellschaft. In Wren, A., editor, *Computer Scheduling of Public Transport*, pages 35–52. North-Holland, Amsterdam.

- [Holland, 1975] Holland, J. (1975). *Adaption in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor.
- [Huisman et al., 2001] Huisman, D., Freling, R., und Wagelmans, A. P. (2001). A dynamic approach to vehicle scheduling. Technical report, Econometric Institute, Erasmus University of Rotterdam.
- [Huisman et al., 2004] Huisman, D., Freling, R., und Wagelmans, A. P. (2004). A robust solution approach for dynamic vehicle scheduling. *Transportation Science*, 38(4):447–458.
- [Huisman et al., 2005] Huisman, D., Jans, R., Peeters, M., und Wagelmans, A. P. (2005). Combining column generation and lagrangian relaxation. In Desaulniers, G., Desrosiers, J., und Solomon, M., editors, *Column Generation*, chapter 9, pages 247–270. Springer, New York.
- [ILOG, 2007] ILOG (2007). *ILOG CPLEX 11.0 User's Manual*. ILOG, Gentilly, France.
- [Karmarkar, 1984] Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395.
- [Kliwer, 2005] Kliwer, N. (2005). *Optimierung des Fahrzeugeinsatzes im öffentlichen Personennahverkehr - Modelle, Methoden und praktische Anwendungen*. PhD thesis, University of Paderborn.
- [Kliwer et al., 2006a] Kliwer, N., Bunte, S., und Suhl, L. (2006a). Time windows for scheduled trips in multiple depot vehicle scheduling. In Binetti, M., Civitella, F., Liddo, E. D., Dell'Orco, M., und Ottomanelli, M., editors, *Proceedings of the EWGT2006 Joint Conferences*, pages 340–346, Bari, Italy.
- [Kliwer et al., 2004] Kliwer, N., Gintner, V., und Suhl, L. (2004). Line change considerations within a time-space network based multi-depot bus scheduling model. Technical report, University of Paderborn - DS&OR Lab.
- [Kliwer et al., 2002] Kliwer, N., Mellouli, T., und Suhl, L. (2002). A new solution model for multi-depot multi-vehicle-type vehicle scheduling in (sub)urban public transport. In *Proceedings of the 13th Mini-EURO Conference and the 9th meeting of the EURO working group on transportation*.
- [Kliwer et al., 2006b] Kliwer, N., Mellouli, T., und Suhl, L. (2006b). A time-space network based exact optimization model for multi-depot bus scheduling. *European Journal of Operational Research*, 175(3):1616–1627.

- [Kokott und Löbel, 1996] Kokott, A. und Löbel, A. (1996). Lagrangean relaxations and subgradient methods for multiple-depot vehicle scheduling problems. Technical Report 22, Konrad-Zuse-Zentrum für Informationstechnik Berlin.
- [Kramkowski und Bunte, 2008] Kramkowski, S. und Bunte, S. (2008). DS&OR Lab: Data instances for vehicle and crew scheduling. <http://www.dsor.de/bustestset>.
- [Kuhn, 1955] Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97.
- [Kwan und Rahin, 1999] Kwan, R. S. und Rahin, M. A. (1999). Object oriented bus vehicle scheduling - the boost system. In Wilson, editor, *Computer-aided transit scheduling*, pages 177–191. Springer.
- [Lamatsch, 1992] Lamatsch, A. (1992). An approach to vehicle scheduling with depot capacity constraints. In M. Desrochers, J.-M. R., editor, *Computer-Aided Transit Scheduling: Procs. of the Fifth International Workshop on Computer Aided Scheduling of Public Transport*, Lecture Notes in Economics and Mathematical Systems, pages 181–195, Montreal, Canada. Springer-Verlag.
- [Laporte, 1992] Laporte, G. (1992). The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59:345–358.
- [Lasch et al., 2005] Lasch, R., Lemke, A., Jugelt, R., Probst, G., Wagener, N., und Winckler, J. (2005). *Endbericht zur Delphi-Studie ÖPNV-Markt der Zukunft*. Lehrstuhl für Betriebswirtschaftslehre, insbes. Logistik Technische Universität Dresden and Wagener&Herbst Management Consultants GmbH.
- [Lenstra und Kan, 1981] Lenstra, J. und Kan, A. R. (1981). Complexity of vehicle routing and scheduling problems. *Networks*, 11(2):221–227.
- [Levin, 1971] Levin, A. (1971). Scheduling and fleet routing models for transportation systems. *Transportation Science*, 5:232–255.
- [Lin und Kernighan, 1973] Lin, S. und Kernighan, B. (1973). An effective heuristic algorithm for the traveling-salesman problem. *Operations Research*, 21:498–516.
- [Löbel, 1996] Löbel, A. (1996). Solving large-scale real-world minimum-cost flow problems by a network simplex method. Technical Report SC96-7, Konrad-Zuse Zentrum für Informationstechnik (ZIB), Berlin.

- [Löbel, 1997] Löbel, A. (1997). *Optimal vehicle scheduling in public transit*. PhD thesis, Technische Universität Berlin.
- [Löbel, 1998] Löbel, A. (1998). Vehicle scheduling in public transit and lagrangean pricing. *Management Science*, 44(12):1637–1650.
- [Löbel, 1999] Löbel, A. (1999). Solving large-scale multiple-depot vehicle scheduling problems. In Wilson, N. H., editor, *Computer-Aided Transit Scheduling*, volume 471 of *Lecture Notes in Economics and Mathematical Systems*, pages 193–220, Berlin. Springer.
- [Löbel und Strubbe, 1995] Löbel, A. und Strubbe, U. (1995). Wagenumlaufoptimierung - Methodischer Ansatz und praktische Anwendung. Technical Report 38, Konrad-Zuse-Zentrum für Informationstechnik Berlin.
- [Luedtke, 1985] Luedtke, L. K. (1985). Rucus ii: A review of system capabilities. In Rousseau, J.-M., editor, *Computer Scheduling of Public Transport 2*, pages 61–116. Elsevier Science Publishers B.V. (North-Holland).
- [Magnanti, 1981] Magnanti, T. L. (1981). Combinatorial optimization and vehicle fleet planning: Perspectives and prospects. Technical report, Massachusetts Institute of Technology.
- [Martin-Löf, 1970] Martin-Löf, A. (1970). A branch-and-bound algorithm for determining the minimal fleet size of a transportation system. *Transportation Science*, 4(2):159–163.
- [Mesquita und Paixao, 1999] Mesquita, M. und Paixao, J. M. P. (1999). Exact algorithms for the multi-depot vehicle scheduling problem based on multicommodity network flow type formulations. In Wilson, N. H., editor, *Computer-Aided Transit Scheduling*, volume 471 of *Lecture Notes in Economics and Mathematical Systems*, pages 221–243, Berlin. Springer.
- [Mesquita und Paixao, 1992] Mesquita, M. und Paixao, J. P. (1992). Multiple depot vehicle scheduling problem: A new heuristic based on quasi-assignment algorithms. In Desrochers, M. und Rousseau, J.-M., editors, *Computer-Aided Transit Scheduling: Procs. of the Fifth International Workshop on Computer Aided Scheduling of Public Transport*, volume 386 of *Lecture Notes in Economics and Mathematical Systems*, pages 167–180, Berlin. Springer.
- [Michalewicz und Fogel, 2000] Michalewicz, Z. und Fogel, D. (2000). *How to Solve It: Modern Heuristics*. Springer-Verlag.

- [Mingozzi et al., 1995] Mingozzi, A., Bianco, L., und Ricciardelli, S. (1995). An exact algorithm for combining vehicle trips. In *Proceedings of the Sixth International Workshop on Computer-Aided Scheduling of Public Transport*.
- [Mladenović und Hansen, 1997] Mladenović, N. und Hansen, P. (1997). Variable neighborhood search. *Computers and Operations Research*, 24(11):1097–1100.
- [Nemhauser und Wolsey, 1999] Nemhauser, G. L. und Wolsey, L. A. (1999). *Integer and Combinatorial Optimization*. John Wiley and Sons, Inc.
- [Orloff, 1976] Orloff, C. S. (1976). Route constrained fleet scheduling. *Transportation Science*, 10(2):149–168.
- [Oukil et al., 2007] Oukil, A., Amor, H. B., Desrosiers, J., und Gueddari, H. E. (2007). Stabilized column generation for highly degenerate multiple-depot vehicle scheduling problems. *Computers & Operations Research*, 34:817–834.
- [Oukil et al., 2004] Oukil, A., Gueddari, H. E., Desrosiers, J., und Amor, H. B. (2004). Stabilized column generation for highly degenerate multiple-depot vehicle scheduling problems. Technical report, GERAD, Montreal.
- [Paixao und Branco, 1987] Paixao, J. P. und Branco, I. M. (1987). A quasi-assignment algorithm for bus scheduling. *Networks*, 17:249–269.
- [Paixao und Branco, 1990] Paixao, J. P. und Branco, I. M. (1990). Bus scheduling with a fixed number of vehicles. In Daduna, J. und Wren, A., editors, *Computer-Aided Transit Scheduling: Procs. of the Fourth International Workshop on Computer Aided Scheduling of Public Transport*, volume 308 of *Lecture Notes in Economics and Mathematical Systems*, pages 28–40, Berlin. Springer.
- [Pepin et al., 2006] Pepin, A.-S., Desaulniers, G., Hertz, A., und Huisman, D. (2006). Comparison of heuristic approaches for the multiple depot vehicle scheduling problem. Technical Report EI2006-34, Econometric Institute, Erasmus University Rotterdam.
- [Polacek et al., 2004] Polacek, M., Hartl, R. F., Doerner, K., und Reimann, M. (2004). A variable neighborhood search for the multi depot vehicle routing problem with time windows. *Journal of Heuristics*, 10:613–627.
- [Rattadilok und Wren, 2006] Rattadilok, P. und Wren, A. (2006). Dynamically configured lambda-opt heuristics for bus scheduling. Technical report, School of Computing, University of Leeds.

- [Reeves, 1993] Reeves, C. R. (1993). Genetic algorithms. In Reeves, C. R., editor, *Modern heuristic techniques for combinatorial problems*, chapter 4, pages 151–196. Blackwell, London.
- [Ribeiro und Soumis, 1994] Ribeiro, C. und Soumis, F. (1994). A column generation approach to the multiple-depot vehicle scheduling problem. *Operations Research*, 42(1):41–52.
- [Rousseau und Blais, 1985] Rousseau, J.-M. und Blais, J.-Y. (1985). Hastus: An interactive system for buses and crew scheduling. In Rousseau, J.-M., editor, *Computer Scheduling of Public Transport 2*, pages 45–59. Elsevier Science Publishers B.V. (North-Holland).
- [Rousseau et al., 1990] Rousseau, J.-M., Lessard, R., und Desilets, M. (1990). Aliages: A system for the assignment of bus routes to garages. In Daduna, J. und Wren, A., editors, *Computer-Aided Transit Scheduling: Procs. of the Fourth International Workshop on Computer Aided Scheduling of Public Transport*, volume 308 of *Lecture Notes in Economics and Mathematical Systems*, pages 8–14, Berlin. Springer.
- [Saha, 1970] Saha, J. L. (1970). An algorithm for bus scheduling problems. *Operational Research Quarterly*, 21(4):463–474.
- [Salzborn, 1972] Salzborn, F. J. M. (1972). Optimum bus scheduling. *Transportation Science*, 6:137–148.
- [Sanzo et al., 1990] Sanso, B., Desrochers, M., Desrosiers, J., Dumas, Y., und Soumis, F. (1990). Modeling and solving routing and scheduling problems: Gencol’s users guide. Technical report, GERAD, Montreal, Canada.
- [Savelsbergh, 1985] Savelsbergh, M. W. (1985). Local search in routing problems with time windows. *Annals of Operations Research*, 4(1):285–305.
- [Scott, 1985] Scott, D. (1985). A large scale linear programming approach to the public transport scheduling and costing problem. In Rousseau, J.-M., editor, *Computer Scheduling of Public Transport 2*, pages 473–491. Elsevier Science Publishers, Amsterdam.
- [Silva et al., 1999] Silva, G. P., Wren, A., Kwan, R. S., und Gualda, N. D. F. (1999). Bus scheduling based on an arc generation - network flow approach. Technical report, University of Leeds - School of Computer Studies.

- [Smith und Wren, 1981] Smith, B. M. und Wren, A. (1981). Vampires and tasc: Two successfully applied bus scheduling programs. In Wren, A., editor, *Computer Scheduling of Public Transport*, pages 97–124. North-Holland.
- [Song und Zhou, 1990] Song, T. und Zhou, L. (1990). A new algorithm for the quasi-assignment problem. *Annals of Operations Research*, 24:205–223.
- [Steinzen, 2008] Steinzen, I. (2008). *Topics in Integrated Vehicle and Crew Scheduling in Public Transport*. PhD thesis, Fakultät für Wirtschaftswissenschaften, Universität Paderborn.
- [Stern und Ceder, 1981] Stern, H. und Ceder, A. (1981). A deficit function approach for bus scheduling. In Wren, A., editor, *Computer Scheduling of Public Transport*, pages 85–96. North-Holland.
- [Stern und Ceder, 1983] Stern, H. und Ceder, A. (1983). An improved lower bound to the minimum fleet size problem. *Transportation Science*, 17(4):471–477.
- [Suhl und Mellouli, 2006] Suhl, L. und Mellouli, T. (2006). *Optimierungssysteme*. Springer, Berlin.
- [Suhl, 2000] Suhl, U. (2000). Mops - mathematical optimization system. *OR News*, 8:11–16.
- [Vanitchakornpong et al., 2008] Vanitchakornpong, K., Indra-Payoong, N., Sumalee, A., und Raathanachonkun, P. (2008). *Applications of Evolutionary Computing*, volume 4974 of *Lecture Notes in Computer Science*, chapter Constrained Local Search Method for Bus Fleet Scheduling Problem with Multi-depot with Line Change, pages 679–688. Springer Berlin / Heidelberg.
- [Williamson, 1985] Williamson, R. H. (1985). Busman: The united kingdoms integrated approach to transit scheduling. In Rousseau, J.-M., editor, *Computer Scheduling of Public Transport 2*, pages 19–43. Elsevier Science Publishers B.V. (North-Holland).
- [Wolsey, 1998] Wolsey, L. A. (1998). *Integer Programming*. Wiley Interscience, New York.
- [Wren, 1972] Wren, A. (1972). Bus scheduling: an interactive computer method. *Transportation planning and technology*, 1:115–122.

- [Wren, 1981] Wren, A. (1981). General review of the use of computers in scheduling buses and their crews. In Wren, A., editor, *Computer Scheduling in Public Transport*, pages 3–16.
- [Wren, 2003] Wren, A. (2003). Scheduling vehicles and their drivers - forty years' experience. Technical report, University of Leeds - School of Computing.
- [Wren und Chamberlain, 1990] Wren, A. und Chamberlain, M. (1990). The development of micro-busman: Scheduling on micro-computers. In Daduna, J. und Wren, A., editors, *Computer-Aided Transit Scheduling: Procs. of the Fourth International Workshop on Computer Aided Scheduling of Public Transport*, volume 308 of *Lecture Notes in Economics and Mathematical Systems*, pages 160–174, Berlin. Springer.
- [Wren und Kwan, 1999] Wren, A. und Kwan, R. S. (1999). Installing an urban transport scheduling system. *Journal of Scheduling*, 2:3–17.