

Peter Schoner

Operative Produktionsplanung in der verfahrenstechnischen Industrie

Die vorliegende Arbeit wurde vom Fachbereich Wirtschaftswissenschaften der Universität Kassel als Dissertation zur Erlangung des akademischen Grades eines Doktors der Wirtschafts- und Sozialwissenschaften (Dr. rer. pol.) angenommen.

Erster Gutachter: Prof. Dr. Richard Vahrenkamp

Zweiter Gutachter: Prof. Dr.-Ing. Sigrid Wenzel

Tag der mündlichen Prüfung

31. Januar 2007

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar

Zugl.: Kassel, Univ., Diss. 2007

ISBN 978-3-89958-335-9

URN: urn:nbn:de:0002-3354

© 2008, kassel university press GmbH, Kassel
www.upress.uni-kassel.de

Druck und Verarbeitung: Unidruckerei der Universität Kassel
Printed in Germany

Danksagung

Diese Arbeit ist am Fachbereich Wirtschaftswissenschaften der Universität Kassel unter der Betreuung von Herrn Prof. Dr. Richard Vahrenkamp entstanden. Ich danke ihm für die freundliche Unterstützung bei der Bearbeitung dieses Themas, in dem ich Aufgabenstellungen der Produktionsplanung und der Mathematischen Optimierung miteinander verknüpfen konnte.

Mein Dank gilt auch Frau Prof. Dr.-Ing. Sigrid Wenzel, daß sie sich freundlicherweise bereit erklärt hat, das Zweitgutachten zu dieser Dissertation zu erstellen.

Ich danke ebenfalls der Friedrich-Naumann Stifung, die meine Promotion durch ein Stipendium aus den Mitteln des Bundesministeriums für Bildung und Forschung gefördert hat.

Ebenso bedanke ich mich bei allen Menschen, die mich während der Erstellung dieser Arbeit begleitet haben. Ihre konstruktiven Gespräche zu dem Thema und die kritische Durchsicht dieser Dissertation haben bei der Erstellung dieser Arbeit weitergeholfen.

Mein besonderer Dank gilt meinen Eltern, die mich in allen Phasen der Promotion unterstützt haben.

Inhaltsverzeichnis

Einleitung	1
Problemstellung und Zielsetzung der Arbeit	1
Aufbau der Arbeit	2
1 Aufgaben der Produktionsplanung	5
1.1 Produktionsplanung	5
1.1.1 Operative Produktionsplanung	6
1.2 Produktionsprozeßplanung	8
1.2.1 Losgrößenplanung	8
1.2.2 Reihenfolgeplanung	10
2 Prozesse und Anforderungen der Verfahrenstechnik	15
2.1 Verfahrenstechnische Produktion	15
2.1.1 Grundverfahren der Verfahrenstechnik	15
2.1.2 Die Verfahrensindustrie	17
2.2 Charakteristika der verfahrenstechnischen Produktion	17
2.2.1 Prozeßablauf	17
2.2.2 Anlagenstruktur	21
2.2.3 Ablaufplanung	29
2.3 Implementierung in Systemen zur Produktionsplanung	32
2.3.1 Abbildung durch Rezepturen	32
2.3.2 IT-Systeme zur Produktionsplanung und -steuerung	33
3 Modelle und Verfahren zur Produktionsplanung	39
3.1 Volumenplanung	39
3.2 Modellierung von Reihenfolgeproblemen	42
3.2.1 Gemischt-ganzzahlige Programmierung	43
3.2.2 Constraint Based Programming	47

3.2.3	Graphentheoretische Modellierung	49
3.3	Optimierungsverfahren	53
3.3.1	Verbesserungsheuristiken	56
3.3.2	Literaturübersicht	62
3.4	Operative Produktionsplanung	64
3.4.1	Literaturübersicht zu Modellen der Mengen- und Reihenfolgeplanung	64
3.4.2	Technische Aspekte der Entwicklung eines Planungsverfahrens	68
4	Reihenfolgeplanung	71
4.1	Ein Optimierungsproblem der Reihenfolgeplanung	71
4.2	Modellierung des Reihenfolgeproblems	76
4.2.1	Modellierung der Reihenfolge auf Reaktoren	76
4.2.2	Modellierung der Reihenfolge auf Silos	79
4.2.3	Modellierung von Ressourcenpools	83
4.2.4	Zusammenfassung und Initial-Instanz	84
4.2.5	Schritte der Reihenfolgeoptimierung	88
4.3	Erzeugen einer Startlösung	90
4.4	Algorithmus zur Makespan -Berechnung	98
4.4.1	Der DIJKSTRA-Algorithmus	98
4.4.2	Die Makespan -Berechnung	99
4.5	Reihenfolgeoptimierung	110
4.5.1	Optimierungsaufgaben	111
4.5.2	Transformationen im Graphenmodell	112
4.5.3	Nachbarschaften der inneren Optimierung	116
4.5.4	Nachbarschaften der äußeren Optimierung	118
4.5.5	Anwendung eines heuristischen Suchverfahrens	120
4.5.6	Zwischenlagerung	124
5	Kombinierte Chargen- und Reihenfolgeplanung	127
5.1	Planungsgrundlagen	127
5.1.1	Kundenaufträge	129
5.1.2	Zielsetzungen	129
5.1.3	Rollierende Planung	131
5.2	Volumenberechnung	135

5.2.1	Volumenberechnung und Reihenfolgeplanung	141
5.3	Chargenplanung	144
5.3.1	Auswahl des Produktionsrezeptes	149
5.3.2	Bestimmung des Produktionsvolumens	150
5.3.3	Erstellen von neuen Chargen	153
5.3.4	Bestimmung des Chargenvolumens und Nachoptimierung	155
6	Schlußbetrachtung	169
A	Erzeugen einer Startlösung	173
	Literaturverzeichnis	177

Abbildungsverzeichnis

1.1	Reihenfolgebeziehungen	12
2.1	Darstellung von Produktionsstrukturen durch Gozinto-Graphen (ohne Gewichtung der Kanten)	19
2.2	Produktionsanlage zur Trockenkaffeeherstellung	22
2.3	Möglichkeiten der Zwischenlagerung	24
2.4	Mehrstranganlage mit drei Einzelsträngen	26
2.5	Nebenläufige Produktion mit 3 Zwischenprodukten	27
2.6	Parallele Produktion in 3 Chargen	28
2.7	Rezept-Entwicklung nach NAMUR Empfehlung NE 33	33
2.8	Produktionsplanung und -steuerung	34
2.9	Planung in Advanced Planning Systemen	35
3.1	Modellierung des Zeitverlaufs	43
3.2	Formulierung eines Flow Shop-Problems als Assignment Problem	46
3.3	Formulierung eines Job Shop-Problems mit OPL	48
3.4	GANTT-Diagramm und gerichteter Graph	50
3.5	Auftragsorientierter Graph und zugehöriger disjunktiver Graph	51
3.6	Lösungsraum mit <i>Golfloch</i> -Optimum	63
4.1	Parallele Operationen nach einer Endprodukt-Produktion	77
4.2	Umwandlung paralleler Prozesse in eine Knotenkette: Fallunterscheidung bei der Abfüllung	78
4.3	Umwandlung des Produktionsnetzwerkes aus Abb. 4.1 in eine Knotenkette	79
4.4	Silo Bestandskette	81
4.5	Lagerung mit Batch-Mix (Beispiel 4.2)	82
4.6	Modellierung mit Ressourcenpool Abfüllung	83
4.7	Initial-Instanz der Reihenfolgeplanung	85
4.8	Schichtübergreifende Produktion	86
4.9	Neuplanung (ohne Reinigung)	87

4.10	Ablaufschema der Reihenfolgeoptimierung	89
4.11	Einzuplanende Chargen (Beispiel 4.4)	95
4.12	Erzeugte Startlösung (Beispiel 4.4)	97
4.13	Ausgangssituation von Beispiel 4.5 zur Optimierung der Abfüllung	112
4.14	Optimierte Reihenfolge der Abfüllung in Beispiel 4.5	113
4.15	Ausgangssituation in Beispiel 4.6 zur Optimierung der Reihenfolge	114
4.16	Optimierte Reihenfolge der Reinigung in Beispiel 4.6	115
4.17	Ausgangssituation in Beispiel 4.9	119
4.18	Unzulässige Zwischenlösung in Beispiel 4.9	119
4.19	Abgeschlossene Move-Operation (Beispiel 4.9)	120
4.20	Zwischenlagerung von Chargen (Beispiel 4.10)	126
5.1	Ergebnis der Chargenplanung bei unterschiedlichen Optimierungskriterien (Beispiel 5.1)	132
5.2	Auftragsmengen in der rollierenden Planung	133
5.3	Zusammenfassung der einzuplanenden Aufträge	134
5.4	Berechnung der Volumengrenzwerte (Beispiel 5.2)	138
5.5	Gesamte Produktionszeit (Definition 5.2)	142
5.6	Zusätzlich einzufügende Charge (Beispiel 5.4)	143
5.7	Simultane Planung	148
5.8	Aufträge umverteilen (Beispiel 5.6)	162
5.9	Ausgangslösung (Beispiel 5.6)	162
5.10	Lösung nach dem Umverteilen (Beispiel 5.6)	162
A.1	Einzuplanende Chargen (Beispiel 4.4)	173

Tabellenverzeichnis

2.1	Produktionsverfahren in der Verfahrens- und Fertigungstechnik	16
2.2	Klassifikation der Anlagenstruktur	26
2.3	Vergleich Kampagnenfahrweise vs. Mischfahrweise	31
4.1	Zugeordnete Reaktoren (Beispiel 4.4)	95
4.2	Erzeugte Zwischenlösungen (Beispiel 4.4)	96
5.1	Mögliche Nachteile bei unterschiedlicher Volumenauslastung	158
5.2	Volumengrenzwerte der Reaktoren (Beispiel 5.7)	165
A.1	Zugeordnete Reaktoren (Beispiel 4.4)	174
A.2	Einzelschritte des Algorithmus (4.2) zur Erzeugung der ersten Zwischenlösungen aus Beispiel 4.4 (vgl. Tabelle 4.2 aus S. 96).	175

Verzeichnis der Algorithmen

3.1	Simulated Annealing	59
3.2	Threshold Accepting - Veränderung des SA	61
3.2	Sintflut-Algorithmus - Veränderung des SA	62
4.1	List Scheduling	90
4.2	Erzeugen einer Startlösung	91
4.3	Regel (ch) für Algorithmus 4.2	93
4.4	Regel (rk) für Algorithmus 4.2	94
4.5	DIJKSTRA Algorithmus	99
4.6	Makespan-Algorithmus	101
4.7	Auswahl des Iterationsknotens	102
4.8	Berechnung der Vorläuferknoten	103
4.9	Berechnung paralleler Aktionen	106
4.10	Bestimmung eines Parallelknotens	107
4.11	Berechnung der Abfüllzeiten	108
4.12	Bestimmung einer lokalen unteren Schranke	123
4.13	Zwischenlagerung im Silo	124
5.1	Rollierende Planung	132
5.2	Simultane Chargen- und Reihenfolgeplanung	146
5.3	Auswahl des Produktionsrezeptes	150
5.4	Volumenbestimmung und Chargenerstellung	154
5.5	Chargen zusammenfassen	157
5.6	Aufträge umverteilen	159
5.7	Volumen reduzieren	165
5.8	Bestimmung des Reduktionsvolumens, Chargen	167
5.9	Reduktionsvolumen, Abfüllung	167

Verzeichnis der Abkürzungen und Symbole

Abkürzungen

APS	<i>Advanced Planning System bzw. Advanced Planning and Scheduling System; S. 34</i>
ATP	<i>Available-to-Promise; S. 34</i>
B&B	<i>Branch & Bound-Verfahren; S. 54</i>
CP	<i>Constraint Programming; S. 47</i>
CTP	<i>Capable-to-Promise; S. 34</i>
DIN	<i>Deutsche Industrie Norm</i>
EN	<i>Euronorm</i>
EOQ	<i>Economic Order Quantity Model; s. Losgrößenplanung S. 8</i>
ERP	<i>Enterprise Resource Planning; S. 33</i>
FTL	<i>Fertigtanklager; S. 29</i>
GA	<i>Genetische Algorithmen; S. 56</i>
GD	<i>Great Deluge Algorithmus; siehe Sintflutalgorithmus S. 62</i>
ISO	<i>International Organization for Standardization</i>
IT	<i>Informationstechnik</i>
JiT	<i>Just-in-Time; S. 74</i>
LP	<i>Linear Programming; S. 54</i>
ME	<i>Mengeneinheiten</i>
MES	<i>Manufacturing Execution System; S. 35</i>
MILP	<i>Mixed Integer Linear Programming; S. 43</i>

MRP	<i>Material Requirement Planning</i> ; S. 33
MRP II	<i>Manufacturing Resource Planning</i> ; S. 33
NAMUR	<i>Normenarbeitsgemeinschaft für Meß- und Regelungstechnik in der Chemischen Industrie</i>
NE	<i>NAMUR Empfehlung</i>
\mathcal{NP}	non-deterministic polynomial; s. \mathcal{NP} -schwer S. 53
RTN	<i>Resource-Task-Network</i> ; S. 67
SA	<i>Simulated Annealing</i> ; S. 58
STN	<i>State-Task-Network</i> ; S. 66
TA	<i>Threshold Accepting</i> ; S. 61
TKW	<i>Tankkraftwagen</i> ; S. 28
TS	<i>Tabu Suche</i> ; S. 56
ZE	<i>Zeiteinheiten</i>

Symbole

Mengen

\emptyset	Symbol für die leere Menge
\mathcal{C}	Menge der Chargen; s. Definition 3.6, S. 57
$\mathcal{K}, \mathcal{K}_{rz}$	Menge der Kundenaufträge bzw. Menge der Kundenaufträge für ein Produkt mit dem Herstellungsrezept rz
$\mathcal{N}(\mathcal{S})$	Nachbarschaft einer Lösung; s. Definition 3.9, S. 58
\mathcal{S}	Lösung eines Reihenfolgeproblems (Instanz, Scheduling); s. Definition 4.4, S. 84
\mathcal{S}_0	Initial-Instanz; s. Definition 4.4, S. 84
\mathcal{S}_{empty}	leere Instanz; s. Definition 4.4, S. 84
\mathcal{U}	Produktionsanlage; s. Definition 3.6, S. 57
$UR \subset \mathcal{U}$	Menge aller Reaktoren der Produktionsanlage \mathcal{U}
ZT_{rz}	Menge der Zutaten eines Rezeptes rz .

Modellierung

○	Start-Knoten in der Knotenkette einer Anlagekomponente
*	Ende-Knoten in der Knotenkette einer Anlagekomponente
[CLEAN]	sog. <i>Hyperknoten</i> , d.h. eine feststehende Knotenkette, die in der Modellierung für eine notwendige Reinigung eingefügt wird; s. S. 76 ff.
[LAGER]	<i>Hyperknoten</i> , der in Abhängigkeit der eingelagerten Chargen und durchgeführten Entnahmen den jeweiligen Bestand eines Silos angibt; s. S. 79 ff.
[PROD]	<i>Hyperknoten</i> , der in der Modellierung für eine vordefinierte Abfolge von Knoten im Falle einer Produktion eingefügt wird; s. S. 76 ff.
kn_{ABFUELL}	Knoten zur Darstellung einer Abfüllung
kn_{CLEAN}	Knoten zur Darstellung einer Reinigung
kn_{LAGER}	Knoten zur Darstellung des Materialbestands im Silo
kn_{PROD}	Knoten zur Darstellung der Produktion
kn_{TRANSFER}	Knoten zur Darstellung des Materialtransfers zwischen Anlagekomponenten
kn_{WARTE}	Knoten zur Darstellung der Zwischenlagerung (Wartezeit bis zum nächsten Prozeßschritt) in einem Reaktor

Variablen

chg	Charge
kn	Knoten
u	Anlagekomponente, Scheduling-Unit
zt	Zutat

Sonstiges

$MKS(u)$	lokaler Makespan einer Anlagekomponente; s. Definition 4.2, S. 75
■ A(3.1)	Ende der Beschreibung eines Algorithmus (<i>hier: Algorithmus 3.1</i>)

Einleitung

Diese Arbeit befaßt sich mit der Planung von Produktionsprozessen bei der chargenorientierten Herstellung. Dabei wird die Berechnung des Chargenvolumens bei der Herstellung auf Anlagen mit unterschiedlichen Kapazitäten ebenso betrachtet wie die Planung der Reihenfolge, in der die Chargen produziert und abgefüllt werden. Für das weitere Verständnis dieser Arbeit wird nachfolgend ein Überblick über die Zielsetzung und den Aufbau dieser Arbeit gegeben.

Problemstellung und Zielsetzung der Arbeit

Durch die Globalisierung der Märkte werden von Produktion und Logistik immer kürzere Lieferfristen gefordert. Zulieferunternehmen, die kein Alleinstellungsmerkmal aufweisen können oder Zusatzleistungen anbieten, werden austauschbar. Gleichzeitig können diese Unternehmen bei einer steigenden Produktvielfalt ihrer Produkte nicht mehr auf Lager vorproduzieren. Stattdessen müssen sie mit einer flexiblen Produktionsplanung kurzfristig auf eine ständig wechselnde Auftragssituation reagieren.

Diese Anforderungen und Bedingungen betreffen auch Unternehmen in der Prozeßindustrie. Nach einer Studie der **initions AG** produziert ein Großteil von Unternehmen dieses Industriesektors permanent oder zumindest zeitweise am Kapazitätslimit. Geeignete computergestützte Optimierungssysteme werden aber nur bei wenigen Betrieben eingesetzt [Fri03]. Die Produktionsplanung bleibt weitestgehend den Anlagenfahrern überlassen. Die Planung basiert daher im wesentlichen auf Erfahrungswerten und Ermessensspielräumen. Eine Variantenplanung und Untersuchung unterschiedlicher Produktionspläne ist oftmals nicht möglich.

Zu dem Industriesektor der Prozeßindustrie werden alle Unternehmen gezählt, bei denen verfahrenstechnische Herstellungsprozesse zur Anwendung kommen. Dazu zählen zu einem Großteil Unternehmen der Chemischen Industrie, die in dem Verband der Chemischen Industrie (VCI) zusammengeschlossen sind. Der VCI vertritt die Interessen von 1.700 Mitgliedsunternehmen, von denen 90% klein- und mittelständische Betriebe sind [VCI03]. Mit weiterhin 530.000 Beschäftigten in 5.800 meist mittelständischen Unternehmen sind etwa ebensoviele Arbeitnehmer im Bereich der Ernährungsindustrie, der Bundesvereinigung der Deutschen Ernährungsindustrie (BVE), angestellt [WN03].

Die beschriebenen Probleme der Produktionsplanung betreffen also im Wesentlichen mittelständische Unternehmen. Bei der verfahrenstechnischen Produktion sind dabei mit der kontinuierlichen Produktion und chargenorientierten Herstellungsprozessen zwei Formen der Prozessführung zu unterscheiden [Blö99]. In mittelständischen Betrieben werden i.d.R. Mehrzweckanlagen für die Chargenproduktion verwendet. Diese bieten eine größere Flexibilität in der Steuerung. Bei der Optimierung muß jedoch zugleich eine größere Anzahl von Planungsvarianten berücksichtigt werden.

Die betriebswirtschaftliche und mathematische Forschung betrachten schon seit mehreren Jahrzehnten verschiedenste Probleme der Produktions- und Reihenfolgeplanung. Hierbei wurden aber zum Großteil nur Problemstellungen der Fertigungsindustrie erforscht. Die entwickelten Modelle und Verfahren können i.d.R. jedoch nicht auf die verfahrenstechnische Produktion übertragen werden [ALS96]. Erst seit Ende der 1980er Jahre werden auch Probleme der Produktionsplanung in der Prozeßindustrie näher erforscht. Von der *Normenarbeitsgemeinschaft für Meß- und Regelungstechnik in der Chemischen Industrie* (NAMUR) wurden einige Schwachstellen von bestehenden Systemen zur Produktionsplanung aufgelistet [NAM99].

- *Fehlende Abstimmung zwischen Grob- und Feinplanung:* Viele Systeme arbeiten nach der MRP-Methode¹, die lediglich den Materialbedarf berechnet, aber nicht die Kapazität der einzelnen Anlagen berücksichtigt.
- *Fehlende Kopplung zwischen Produktion und Abfüllung:* Wenn zu einem Produktionsauftrag entsprechende Abfüllaufträge erzeugt werden, müssen diese in der Reihenfolgeplanung, wie auch in Bezug auf die produzierende Ressource mit dem Produktionsauftrag verknüpft werden.

Das Ziel dieser Arbeit ist es, ein Verfahren zur operativen Produktionsplanung bei chargenorientierten Herstellungsprozessen auf Mehrzweckanlagen zu entwickeln. Dazu werden die Anforderungen an und Schwachstellen von bestehenden Planungssystemen analysiert. In einer Volumenplanung werden aus der Menge der (Kunden-) Aufträge für (End-) Produkte unter Berücksichtigung der verfügbaren Ressourcen die Chargen zusammengestellt. Diese Chargen werden wiederum in einer Reihenfolgeplanung im Sinne einer zeitlich bestmöglichen Anordnung auf den Produktionsanlagen eingeplant. Bei dieser Reihenfolgeplanung wird auch die Abfüllung als ein letzter Produktionsschritt berücksichtigt.

Aufbau der Arbeit

Die vorliegende Arbeit ist nach dieser Einleitung in fünf Kapitel und eine Schlußbetrachtung untergliedert. Die ersten drei Kapitel beschreiben die Grundlagen, Anforderungen und Verfahren, die bei der Erstellung dieser Arbeit berücksichtigt und verwendet werden. Die weiteren Kapitel beschreiben das Modell und das Verfahren, das in dieser Arbeit zur operativen Produktionsplanung in der verfahrenstechnischen Industrie entwickelt worden ist.

¹MRP = Material Requirement Planning, s.S. 33

Kapitel 1 beginnt mit der begrifflichen Definition der Produktionsplanung und ihrer Aufgaben aus betriebswirtschaftlicher Sicht. Hierbei wird vor allem die kurzfristige, d.h. operative Produktionsplanung beschrieben. Diese umfaßt die Losgrößen- und die Reihenfolgeplanung. Für erstere werden die Anforderungen bei der Erstellung eines Optimierungsverfahrens beschrieben. Für die Ablauf- oder auch Reihenfolgeplanung wird ein allgemeines Klassifizierungsmodell vorgestellt.

In Kapitel 2 werden die Prozesse und Anforderungen der verfahrenstechnischen Produktion dargestellt. Hierzu werden zunächst die verfahrens- und fertigungstechnische Fertigung voneinander abgegrenzt und der in dieser Arbeit verwendete Begriff der Verfahrensindustrie definiert. Für die Erstellung eines Modells zur operativen Produktionsplanung werden anschließend die Merkmale einer chargenorientierten Produktion beschrieben.

Zur Fertigungssteuerung müssen die Produktionsabläufe informationstechnisch abgebildet werden. Im Abschnitt 2.3 wird daher für die Darstellung der benötigten Informationen die von der NAMUR vorgeschlagene Darstellung einer Rezeptur vorgestellt. Zur Produktionsplanung und -steuerung können dabei unterschiedliche IT-Systeme eingesetzt werden. Zum Abschluß von Kapitel 2 wird deswegen eine Übersicht über die Anforderungen, den Einsatz und die Schwachstellen der bestehenden IT-Systeme in der Prozeßindustrie gegeben. Diese dient als Kriterienliste für die Entwicklung des Optimierungsmodells, das im zweiten Teil dieser Arbeit beschrieben wird.

Das Kapitel 3 ist der Beschreibung der mathematischen Grundlagen für die Entwicklung eines Optimierungsmodells und -verfahrens zur operativen Produktionsplanung gewidmet. Hier werden die Eigenschaften beschrieben, die bei einer, der Ablaufplanung vorgeschalteten Volumenplanung berücksichtigt werden müssen. Für die Reihenfolgeplanung müssen die Anforderungen aus Kapitel 2 zunächst mit einem entsprechenden Modell beschrieben werden. Hierzu werden verschiedene Möglichkeiten der Modellierung vorgestellt und deren Vor- und Nachteile diskutiert. Anschließend werden einige Eigenschaften aufgezeigt, für die heuristische Optimierungsverfahren gegen ein globales Optimum konvergieren. Mit dem Simulated Annealing und verwandten Verfahren werden anschließend die Heuristiken vorgestellt und untersucht, die im zweiten Teil dieser Arbeit verwendet werden. Das dritte Kapitel schließt mit einer Beschreibung und Untersuchung der bisher in der Literatur vorgestellten Modelle zur operativen Produktionsplanung in der verfahrenstechnischen Industrie.

Im zweiten Teil dieser Arbeit wird in Kapitel 4 zunächst ein Modell zur Reihenfolgeplanung entwickelt. Dieses berücksichtigt die in Kapitel 2 beschriebenen Anforderungen. Darüber hinaus wird u.a. mit der Berücksichtigung der Abfüllung eine der in Abschnitt 2.3 genannten Schwachstellen behoben. Anhand des Klassifikationsschemas aus Kapitel 1 wird dieses Modell in den Kontext der Reihenfolgeplanung gestellt.

Für die Optimierung wird anschließend ein Verfahren zur Bestimmung einer Startlösung vorgestellt und mehrere Verbesserungsheuristiken entwickelt. Hierbei ist zwischen der sogenannten *inneren Optimierung*, bei der die Abfolge von den Prozessen Abfüllung und Reinigung verändert wird, und der *äußeren Optimierung*, bei der die eigentliche Abfolge der Produktionsprozesse verändert wird, zu unterscheiden. Dies ist notwendig, da Ressourcen, die bei der Abfüllung und Reinigung benötigt werden, zu Engpässen werden können und somit Einfluß auf das Optimierungsziel der Reihenfolgeplanung, die Minimierung des Makespan², nehmen können.

²Der Makespan oder auch die *Zykluszeit* gibt die Gesamtdauer an, bis alle Aufträge einer Auftragsmenge

Die Berechnung des **Makespan** erfordert wiederum einige Vorwärts- und Rückwärtsberechnungen der Startzeitpunkte. Dies ist insbesondere bei der Batchmix-Zwischenlagerung, d.h. der Lagerung von Chargen unterschiedlicher Produktionsprozesse in einem Silo, und der JiT-Produktion³ der Fall. Daher wird hier ein Algorithmus zur **Makespan**-Berechnung entwickelt, der die Startzeitpunkte und die unbekannte Dauer der Zwischenlagerung berechnet.

Kapitel 5 schließt die Arbeit mit der Entwicklung eines Verfahrens, das die Reihenfolgeplanung des vorherigen Kapitels verwendet, um eine simultane Volumen- und Reihenfolgeplanung durchzuführen. In Anbetracht nicht vorhandener Informationen für eine Kostenbewertung von Planungsergebnissen werden unterschiedliche Ersatzziele evaluiert. In dieser Arbeit wird die Minimierung der Anzahl zu produzierender Chargen als Zielsetzung der Volumenplanung verwendet. Das Volumen einer Charge ist dabei nur von den technischen Restriktionen abhängig. Ausgehend von einer Startlösung wird ein mögliches Maximalvolumen schrittweise reduziert, bis auch eine mit der Reihenfolgeplanung gültige Lösung erreichbar ist. Weitere Schritte der Nachoptimierung verändern das Volumen der Chargen und damit die Zusammenstellung von Kundenaufträgen zu Produktionsaufträgen.

Das in dieser Arbeit entwickelte Verfahren soll zur operativen Produktionsplanung in Unternehmen eingesetzt werden können, deren Produkte verfahrenstechnisch in chargenorientierten Produktionsabfolgen hergestellt werden. Für eine solche Anwendung müssen weitere Schritte durchgeführt werden, die nicht mehr die Optimierung, aber die Datenerhaltung betreffen. Daher sind an einzelnen Stellen in Kapitel 4 und 5 Abschnitte mit der Bezeichnung *Praxis: Implementierung in einem Planungssystem* eingefügt. Diese beschreiben die notwendigen Erweiterungen, die nicht mehr Bestandteil dieser Arbeit sind.

Das in dieser Arbeit entwickelte Modell und die darauf aufbauenden Verfahren wurden in der Programmiersprache C++ mit dem **Borland C++ Builder** programmiert.

fertiggestellt sind. Eine Definition wird auf S. 13 gegeben.

³Der in dieser Arbeit verwendete Begriff der JiT-Produktion beschreibt für *no-wait*-Produkte, d.h. Produkte, die ohne weitere Lager- bzw. Wartezeit direkt weiterverarbeitet werden müssen (s.S. 30), zusätzlich die zeitliche Beziehung, die bei der Weiterverarbeitung in einem nachfolgendem Produkt berücksichtigt werden muß (vgl. auch Fußnote 6 auf S. 74).

Kapitel 1

Aufgaben der Produktionsplanung

Die Produktionsplanung ist der Prozeß der zielgerichteten Entscheidungsfindung zur Herstellung von Produkten und der Festlegung des Produktionsprozesses unter Berücksichtigung der dafür verfügbaren Produktionsfaktoren ([DSV93], S. 8). In sachlicher, wie auch in zeitlicher Hinsicht sind dazu verschiedenste Zielsetzungen, Eigenschaften und Parameter zu berücksichtigen. In dem nachfolgenden Abschnitt (Kapitel 1.1) wird daher für den weiteren Verlauf dieser Arbeit die Produktionsplanung mit dem Schwerpunkt der kurzfristigen Planung vorgestellt. Für die unterschiedlichen Aufgaben der Planung sind einzelne Optimierungsprobleme zu entwickeln. Die Eigenschaften, die bei der Modellierung¹ dieser Optimierungsprobleme zu berücksichtigen sind, werden anschließend in dem Kapitel 1.2 näher beschrieben.

1.1 Produktionsplanung

Die Tätigkeit eines Unternehmens ist im wesentlichen durch die *Produktion*, d.h. die Herstellung von Sachgütern oder die *Erstellung von Dienstleistungen* geprägt ([Wö90], S. 2). Im Rahmen dieser Arbeit wird von diesen beiden die (industrielle) Produktion betrachtet, d.h. die Erzeugung von Produkten aus materiellen und nichtmateriellen Produktionsfaktoren durch technische Verfahrensweisen ([GT04], S. 6). Materielle Produktionsfaktoren sind die Einsatzstoffe, die in Endprodukte umgewandelt werden. Weitere (indirekte) Produktionsfaktoren sind die eingesetzten Maschinen und Lagerorte. Die wesentlichen nichtmateriellen Produktionsfaktoren sind die benötigte Energie und die menschliche Arbeitskraft, z.B. in der Entwicklung und zur Überwachung der Produktion. Der *Produktionsprozeß* selbst ist die Abfolge der Arbeitsschritte, um mit diesen Produktionsfaktoren das gewünschte Endprodukt herzustellen.

Die Führung eines Produktionsbetriebes ist einem permanenten Kreislauf von *Planung*, *Steuerung* und *Kontrolle* unterworfen. Zur Erreichung des marktwirtschaftlich obersten Ziels, eines langfristig größtmöglichen Gewinns, ist für die Produktion daher zunächst ein zu erreichendes (Zwischen-) Ziel zu bestimmen. Die (Produktions-) Planung ist dann die „gedankliche Vorwegnahme zukünftigen Handelns durch Abwägen der verschiedenen Handlungsalternativen und die Entscheidung für den günstigsten Weg“ ([Wö90], S. 138) für die zuvor gewählte Zielsetzung. Sie gibt als Ergebnis den durchzuführenden Produkti-

¹Computergestützte Planungssysteme nutzen diese Modelle in ihrer modularen Softwarearchitektur. In Kapitel 2.3.2 werden daher IT-Systeme zur Produktionsplanung vorgestellt.

onsplan zurück, dessen Umsetzung durch entsprechende Steuerungssysteme gewährleistet wird. Zur Überprüfung der Qualität des Produktionsplans, wie auch der Steuerung muß abschließend ein Soll-Ist-Vergleich durchgeführt werden. Nur durch eine solche Kontrolle können Abweichungen des Planungsmodells von der Realität erkannt und die Ursachen hierfür gefunden werden. Die daraus gewonnenen Erkenntnisse sind in das erstellte Optimierungsmodell einzufügen, um bei der nächsten Planung entsprechend berücksichtigt zu werden. Die Produktionsplanung kann anhand des zu betrachteten Zeithorizontes zusätzlich in verschiedene Planungsstufen aufgeteilt werden ([DSV93], S. 2 ff.).

Strategische Planung: Die strategische Planung betrachtet einen längerfristigen, mehrjährigen Zeitraum. Dabei werden z.B. Entscheidungen über das Produktspektrum und die Produktionsstandorte gefällt.

Taktische Planung: Die taktische Planung hat eine Reichweite von mehreren Monaten bis hin zu wenigen Jahren. Dabei werden Entscheidungen über die Ausstattung des Unternehmens in Bezug auf Personal und Maschinen, sowie die Zusammenarbeit mit Lieferanten und externen Dienstleistern gefällt.

Operative Planung: Die operative Produktionsplanung beschäftigt sich mit der zeitlichen, mengenmäßigen und räumlichen Planung des Produktionsvollzugs auf Wochen-, Tage- oder Stundenbasis ([DSV93], S. 15). Sie umfaßt sowohl die *Bereitstellungsplanung* für die benötigten Werkstoffe und die Personalplanung, als auch die *Produktionsprozeßplanung* der eigentlichen Produktion ([DSV93], S. 9). Die operative Planung ist eng mit der *Produktionssteuerung*, z.B. der (Planung der) Auftragsfreigabe in der Reihenfolge der Produktion, verbunden.

Diese Arbeit befaßt sich mit der Produktionsprozeßplanung, d.h. der kurzfristigen Planungsebene. Für eine Beschreibung der Produktionsplanung längerfristiger Planungszeiträume wird daher auf die entsprechende Literatur verwiesen ([Zäp82], [DSV93], [GT04]). Einzelne Aspekte der Planung werden von verschiedenen Autoren in dem von STADTLER UND KILGER (HRSG.) herausgegebenem Buch zum „Supply Chain Management and Advanced Planning“ näher betrachtet [SK00].

1.1.1 Operative Produktionsplanung

Die operative Produktionsplanung betrachtet die Umwandlung der Auftragsmengen in *Fertigungs-* bzw. *Produktionsaufträge*. Während bei der längerfristigen Produktionsplanung noch anhand von *Bedarfsprognosen* geplant wird, sind im Rahmen der operativen Produktionsplanung die Bedarfsmengen klar vorgegeben. Für die kurzfristige Planung werden daher je nach Auftragsart die folgenden Daten benötigt.

Definition 1.1 *Die zu produzierende Bedarfsmenge (in der operativen Produktionsplanung) umfaßt die Menge der Kundenaufträge, d.h. der direkt für einen Kunden zu produzierenden Produkte und der Lageraufträge.*

Kundenaufträge umfassen Informationen über Produkt, Menge, (Liefer-) Termin und Verpackung. Sie werden speziell für einen Kunden gefertigt.

Lageraufträge werden dagegen aufgrund einer Absatzprognose gefertigt. Sie umfassen nur die Information über das Produkt und die Menge, die zu fertigen ist.

Je nach Variantenvielfalt der Produkte² können dabei einzelne Zwischenprodukte aufgrund von Bedarfsprognosen vorproduziert oder erst nach Auftragseingang hergestellt werden. Insbesondere bei einer Auftragsannahme für kurzfristig zu liefernde Endprodukte ist generell zu prüfen, ob die gesamte Produktionsdauer durch die Herstellungsdauer eines Zwischenproduktes stark verzögert würde. Hier sind die jeweiligen Zwischenprodukte nach Möglichkeit bereits früher aufgrund von Bedarf- bzw. Absatzprognosen zu produzieren. Ebenso ist dann eine Vorproduktion durchzuführen, wenn die Bedarfsmenge für ein Zwischenprodukt geringer als die technisch realisierbare Produktionsmenge ist. Basierend auf Definition 1.1 wird daher zusätzlich die folgende Definition gegeben.

Definition 1.2 *Ein interner Auftrag ist der betriebsinterne Lagerauftrag für ein Zwischenprodukt, das vorproduziert wird.*

Ein solches Zwischenprodukt wird in der Fertigungsindustrie auch als Komponente, in der verfahrenstechnischen Industrie als Halbzeug bzw. Halbprodukt bezeichnet.

In der Produktionsprozeßplanung werden die Kundenaufträge, Lageraufträge oder internen Aufträge³ für die zu erzeugenden Vor-, Zwischen- oder Endprodukte zu *Fertigungsaufträgen* zusammengefasst. Ein Fertigungsauftrag umfaßt dabei eine i.d.R. feste (evtl. technisch bedingte) Stückzahl, die in einem Arbeitsgang gefertigt wird. Diese Anzahl wird oftmals auch als Mindestmenge und „Schrittweite“ in der Auftragsannahme eines Kundenauftrags verwendet. Aufgrund der Terminvorgaben der zusammengefassten Aufträge eines Fertigungsauftrags wird in der *Durchlaufterminierung* für jeden Fertigungsauftrag ein Zeitfenster für die Bearbeitung bestimmt.

Die wesentlichen Aufgaben der Produktionsprozeßplanung sind die *Losgrößenplanung* und die *Ablauf- bzw. Reihenfolgeplanung*⁴. Ein Los ist die Zusammenfassung von Fertigungsaufträgen, die auf einer Maschine hintereinander gefertigt werden. Die Losgrößenplanung bestimmt dann die nach Vorgabe der Zielfunktion optimale Losgröße. Dabei wird als Zielfunktion entweder die *Minimierung der Kosten* oder die *Minimierung der Durchlaufzeit* betrachtet ([Vah00], S. 165).

Die Bearbeitungsdauer eines Loses ist linear abhängig von der Größe des Loses. Die Kenntnis der Losgrößen ist daher erforderlich zur Reihenfolgeplanung ([Brü02], S. 13). Die Losgrößenplanung wird daher vor oder simultan mit der Reihenfolgeplanung durchgeführt. Daneben sind weitere sachliche und zeitliche Interdependenzen zwischen den verschiedenen Teilbereichen der Produktionsplanung zu berücksichtigen ([Zäp82], S. 290 ff.).

²Die Variantenvielfalt der Produkte ist durch die Auflagehäufigkeit eines Produktes bestimmt. In der *Einzelfertigung* herrscht eine große Variantenvielfalt vor. Alle Zwischenprodukte werden erst nach Auftragseingang eines Kundenauftrags hergestellt. Dagegen wird bei der *Massenfertigung* komplett auf Lager bzw. der *Sortenfertigung* die benötigten Zwischenprodukte anhand einer Absatzprognose gefertigt ([Gro04], S. 214).

³Aufgrund der vorherigen Überlegungen ist es oftmals sinnvoll, den internen Aufträgen anhand einer Bedarfsprognose zusätzlich einen (spätesten Produktions-) Termin zuzuweisen.

⁴Das Problem der Reihenfolgeplanung kann weiter unterteilt werden in die *eigentliche Reihenfolgeplanung*, bei der die chronologische Abfolge der Aufträge auf den einzelnen Maschinen festgelegt wird und das *Maschinenbelegungsproblem*, das für die einzelnen Arbeitsschritte die Start- und Endzeitpunkte berechnet ([DSV93], S. 16). Die Berechnung der Zeitpunkte wird auch als *Terminplanung* bezeichnet ([GT04], S. 212). Bei dieser Berechnung ist insbesondere darauf zu achten, ob die Startzeitpunkte frühestmöglich oder spätestmöglich berechnet werden.

1.2 Produktionsprozeßplanung

1.2.1 Losgrößenplanung

Die Losgrößenplanung wurde zuerst von HARRIS (1913) und ANDLER (1929) unter dem Namen des *Economic Order Quantity Model (EOQ-Modell)* vorgestellt ([DSV93], S. 70). Ziel einer Losgrößenplanung ist es, die im Sinne einer vorgegebenen Zielfunktion optimale Losgröße zu bestimmen. Hier wird i.d.R. eine der beiden folgenden Funktionen verwendet.

- *Kostenminimierung*: In der Herstellung werden verschiedene Kostenblöcke berücksichtigt. Durch die Zusammenfassung der Produkte in einem Los muß über einen längeren Zeitraum eine Vorfinanzierung erfolgen, welche *Kapitalbindungskosten* erzeugt. Bis zum Abverkauf der Produkte entstehen dann zusätzlich *Lagerkosten*. Andererseits können durch eine Zusammenfassung vor der Produktion notwendige Rüstzeiten und damit verbundene *Rüstkosten* bzw. abschließend auftretende *Reinigungskosten* reduziert werden.
- *Durchlaufzeitminimierung*: Die Durchlaufzeit ist die Dauer von dem Beginn des ersten Arbeitsschrittes bis zur Beendigung des letzten Arbeitsschrittes in dem gesamten Herstellungsprozeß eines Produktes. Bei einer mehrstufigen Produktion⁵ können Rüst- und Wartezeiten einen erheblichen Einfluß auf diese Gesamtdauer haben. Wesentlich ist bei der Losgrößenplanung daher der Modellierungsparameter, ob ein Los nur geschlossen oder auch offen (d.h. einzelne Fertigungsaufträge des Loses) weitergegeben werden kann. Bei einer offenen Losweitergabe können nachfolgende Produktionsschritte direkt anschließend ohne weitere Wartezeiten durchgeführt werden.

Das EOQ-Modell beruht auf sehr strengen Annahmen, wie z.B. der Planung für *nur ein einzelnes Produkt bei statischem Bedarf*. Zur Berücksichtigung eines schwankenden Bedarfs wurde von WAGNER UND WHITHIN (1958) eine entsprechende Erweiterung des Modells entwickelt ([Vah00], S. 173). Beide Modelle berücksichtigen jedoch nicht Kapazitätsbeschränkungen der Produktionsanlage. Weiterhin ist gerade bei einer *Mehrprodukt-Losgrößenplanung* aufgrund der Rüstzeiten auch die Reihenfolge der Produktionsprozesse zu berücksichtigen.

Diese erweiterten Modelle werden auch als *Lotsizing and Scheduling Modelle* bezeichnet. Dabei sind die nachfolgend beschriebenen Aspekte zu beachten ([Haa94], [Der95], [Me99]).

Produkte und Produktionsanlage: Je nach Anzahl der zu planenden (End-) Produkte wird von einem *Einprodukt-* oder *Mehrproduktproblem* gesprochen. Wenn in der Herstellung zusätzlich Zwischenprodukte hergestellt werden müssen, wird dies als *mehrstufige Produktionsstruktur*⁶ bezeichnet. Ansonsten spricht man von einer *einstufigen Produktionsstruktur*.

⁵Eine mehrstufige Produktion ist gegeben, wenn der Produktionsprozeß in einzelne Arbeitsschritte unterteilt werden kann und die jeweils hergestellten (Zwischen-) Produkte separat gelagert und unterschiedlich weiterverarbeitet werden könne. Im Verlauf dieser Arbeit werden hierzu entsprechende Definitionen (Definition 4.1 auf S. 73 und Definition 4.5 auf S. 93) gegeben.

⁶Die unterschiedlichen Typen einer mehrstufigen Produktionsstruktur werden im Abschnitt *Stofffluß* auf S. 18 näher beschrieben.

Weiterhin wird zwischen *Ein-* und *Mehrmaschinenproblemen* unterschieden. In letzterem Fall ist bei der Modellierung des Problems zu beachten, wie die Maschinen miteinander vernetzt sind. Wichtig ist auch, daß die *Kapazitätsbeschränkungen* der Ressourcen (Maschinen und Lagerplätze) als Nebenbedingung berücksichtigt werden. Hier ist für die Maschinen insbesondere die zeitliche Verfügbarkeit zu beachten, die neben der eigentlichen Produktion (*Stückbearbeitungszeit*) auch durch die Rüstzeit reduziert wird.

Planungsdaten: Bei der Planung kann ein unendlicher oder endlicher *Planungshorizont* betrachtet werden. In letzterem Fall wird zusätzlich unterschieden zwischen der Wahl eines festen Planungshorizonts und einem Horizont variabler Länge. Die Daten können im Verlauf dieses Horizonts *statisch/deterministisch* oder *dynamisch* sein. Bei statischem Bedarf kann eine Losgrößenplanung mit zyklischen Auflagenmustern durchgeführt werden [Ma96].

Die extern bedingten Zustandsänderungen des Systems oder interne Änderungen bedingen eine Unterteilung des Planungshorizonts in einzelne Planungsperioden (*time buckets*), wobei häufig noch einmal zwischen Makro- und Mikroperioden (bzw. *big buckets* und *small buckets*) unterschieden wird. In einem *big bucket Modell* können mehrere Lose in einer Periode gefertigt werden, während in einem *small bucket Modell* höchstens eine Umrüstung pro Periode erfolgt ([Me99], S. 52). Bei der Erstellung eines Optimierungsmodells ist zusätzlich zu entscheiden, ob die Produktion eines Loses nur innerhalb einer Periode erfolgen kann oder auch über diese zeitlichen Grenzen hinaus reichen kann. In letzterem Fall muß aber eine permanente Produktion z.B. im Sinne eines 3-Schicht-Betriebes gegeben sein.

Die Planung wird dann über den Verlauf der Perioden hinweg *rollierend* durchgeführt. Das Optimierungsmodell muß bei der Betrachtung der letzten Planungsperiode zusätzlich beachten, ob die einzuplanenden Bedarfsmengen komplett eingeplant werden müssen oder auch die Möglichkeit von *Anbruchmengen* besteht. Die verbleibende Menge wird dann in einer späteren Optimierung gemeinsam mit neuen Aufträgen erneut eingeplant.

Modellierungsparameter: Für die Optimierung ist zu berücksichtigen, ob die *Produktionskosten* im Zeitverlauf variieren. In diesem Fall sind sie entscheidungsrelevant, und die Losgrößenplanung muß mit einer kostenminimierenden Zielfunktion durchgeführt werden ([Der95], S. 23).

Bei einem Produktionswechsel entstehen Rüstzeiten, die sowohl bei einer Minimierung der Kosten, wie auch bei der Berücksichtigung der zeitlichen Kapazitäten der Maschinen beachtet werden müssen. Hierbei muß das Optimierungsmodell berücksichtigen, ob in einer neuen Planungsperiode der Rüstzustand beibehalten wird, oder ob erneut gerüstet werden muß. Letzteres ist nicht nur bei einem Produktwechsel notwendig, sondern bspw. auch in der Lebensmittelindustrie erforderlich, um den Hygienezustand einer Anlage nach einem längeren Stillstand zu erhalten.

Bei einer mehrstufigen Produktion ist weiterhin die *Losweitergabe* entscheidend für den Startzeitpunkt der nachfolgenden Produktionsstufen. Bei geschlossener Losweitergabe wird nur das vollständig produzierte Los für die nachfolgende Produktion freigegeben. Bei einer offenen Losweitergabe kann sie dagegen schon früher mit Teilmengen begonnen werden.

Eine genauere Übersicht über die verschiedenen Losgrößen und Reihenfolgeplanungsmodelle anhand der hier skizzierten Strukturmerkmale wird von MEYR gegeben [Me99].

1.2.2 Reihenfolgeplanung

Zur Beschreibung von Problemen der *Reihenfolge-* oder auch *Ablaufplanung* wurde erstmals von CONWAY ET AL. [CMM67] ein Klassifikationsschema entwickelt. Für die Klassifikation des Reihenfolgeproblems in Kapitel 4 wird dieses Schema auf der Basis der weiterentwickelten Notation von GRAHAM ET AL. [GL+79]⁷ vorgestellt.

In der *Reihenfolge-* oder auch *Ablaufplanung* ist eine vorgegebene Anzahl von k (Produktions-) Aufträgen (Jobs $\mathcal{J} := \{j_1, \dots, j_k\}$) mit jeweils $n(j)$ Operationen auf n Maschinen (m_1, \dots, m_n) zu verteilen⁸. Jeder Operation $o_{j,i}$ ist eine Maschine m_i und eine Dauer $t(o_{j,i}, m_i)$ für die Bearbeitung zugeordnet.

Ein Reihenfolgeproblem wird dann durch ein Tripel $\alpha/\beta/\gamma$ beschrieben. Dabei bezeichnet α die Maschinen-Charakteristik, β die Auftrags-Charakteristik und γ die Zielsetzung des Optimierungsproblems. Diese Charakteristika werden nachfolgend mit ihren weiteren Merkmalen erläutert.

Maschinen-Charakteristik: Bei der Maschinen-Charakteristik $\alpha = (\alpha_1, \alpha_2)$ werden die beiden Merkmale α_1 für die Anordnung der Maschinen und α_2 für deren Anzahl unterschieden. Hier finden sich in der Literatur verschiedenste Beschreibungen der Merkmale ([Bru95], S. 4 ff.; [DSV93], S. 254 f.). Die folgende Auflistung für α_1 folgt BŁAZEWCZ ET AL. ([BE+93], S. 47).

$\alpha_1 = \emptyset$: Für die Aufträge steht nur eine einzige Maschine zur Verfügung.

$\alpha_1 = P$: Die Aufträge bestehen nur aus einer Operation, für die mehrere identische parallele Maschinen mit gleichen Produktionsgeschwindigkeiten zur Verfügung stehen.

$\alpha_1 = Q$: Die Aufträge bestehen nur aus einer Operation, für die mehrere identische parallele Maschinen mit maschinenabhängigen Produktionsgeschwindigkeiten zur Verfügung stehen.

$\alpha_1 = R$: Die Aufträge bestehen nur aus einer Operation, für die mehrere identische parallele Maschinen mit maschinen- und auftragsabhängigen Produktionsgeschwindigkeiten zur Verfügung stehen.

Wenn ein Auftrag dagegen aus mehreren Operationen besteht, dann sind die folgenden Planungsmerkmale vorhanden:

$\alpha_1 = J$: Bei einem *Job Shop-Problem* hat jeder Auftrag eine eigene Reihenfolge, in der die Operationen auf den einzelnen Maschinen bearbeitet werden.

$\alpha_1 = F$: Bei einem *Flow Shop-Problem* haben alle Aufträge dieselbe Reihenfolge, in denen die Operationen den einzelnen Maschinen zugeordnet werden.

$\alpha_1 = O$: Bei einem *Open Shop-Problem* ist keine Reihenfolge für die Operationen auf den Maschinen vorgegeben.

⁷Alle Literaturangaben zitiert nach [DSV93], S. 250.

⁸Dieses Optimierungsproblem wird in der englischsprachigen Literatur auch als *Scheduling Problem* bezeichnet.

Weitere mögliche Ausprägungen der Maschinen-Charakteristik sind das *General Shop*-, das *Mixed Shop*-, das *Permutation Job Shop*- und *Hybrid Job Shop*-, das *Permutation Flow Shop*- und das *Hybrid Flow Shop*-Problem. Zur weiteren Erklärung wird auf die Beschreibung von BRUCKER ([Bru95], S. 5) und BRÜSSAU ([Brü02], S. 15) verwiesen.

Das zweite Merkmal α_2 hat den Wert $\alpha_2 = \emptyset$, wenn die Anzahl der Maschinen beliebig ist, oder $\alpha_2 = n$, wenn die Anzahl der Maschinen mit einem festen Wert⁹ n vorgegeben ist.

Auftrags-Charakteristik: Bei der Auftrags-Charakteristik finden sich in der Literatur unterschiedlich viele Charakteristika. BRUCKER ([Bru95], S. 3f.) beschreibt hier 6 Merkmale, BŁAZEWCZ ET AL. ([BE+93], S. 48) dagegen 8, und DOMSCHKE ET AL. ([DSV93], S. 257 ff.) definieren 10 Eigenschaften zur Beschreibung eines Auftrags. Nachfolgend werden die wesentlichen Eigenschaften in Anlehnung an DOMSCHKE ET AL. ([DSV93], S. 257 ff.) wiedergegeben.

β_1 , **Anzahl:** Die Anzahl der Aufträge, die betrachtet werden, kann beliebig oder konkret¹⁰ gegeben sein.

β_2 , **Unterbrechbarkeit:** Es gibt drei Konzepte der Unterbrechbarkeit in diesem Klassifikationsschema: Die einzelnen Operationen der Aufträge können beliebig unterbrechbar¹¹, ohne Unterbrechung, aber mit Wartezeiten zwischen den Operationen oder ohne Unterbrechung und ohne Wartezeit zwischen den Operationen sein¹²

β_3 , **Reihenfolgebeziehung:** Es können zwischen den Aufträgen \circ - keine Reihenfolgebeziehungen; *tree* - Reihenfolgebeziehung in Form eines gerichteten Baumes oder *prec* - Reihenfolgebeziehungen¹³ in beliebiger Form vorhanden sein.

Sofern eine Reihenfolgebeziehung besteht, ist zusätzlich die zeitliche Abfolge der Operationen untereinander zu berücksichtigen. Anhand von Start- und Endzeitpunkt können vier verschiedene Grundformen definiert werden (vgl. Abb. 1.1 und [Cor98], S. 464). Die Zeitpunkte können zusätzlich um einen zeitlichen Abstand, eine sog. *Offset-Zeit*, verschoben sein.

β_4 , **Vor- und Nachlaufzeiten:** Die Angabe einer Vorlaufzeit (Auftragsfreigabe) $\beta_4 = a_j$ gibt an, ob ein Auftrag j bereits zu Beginn hergestellt werden kann, oder ob die Herstellung erst nach einer zeitlichen Verzögerung gestartet werden darf. Wenn ein Auftrag eine Nachlaufzeit $\beta_4 = n_j$ hat, dann muß der Auftrag noch für n_j Zeiteinheiten (ZE) warten, bevor das Produkt des Auftrages weiter verarbeitet werden kann.

⁹Für den Spezialfall $\alpha_2 = 2$ entwickelte JOHNSON den nach ihm benannten *Johnson-Algorithmus* [Joh54]; ([Vah00], S. 207), der für das Ziel der Minimierung der Zykluszeit (s.S. 13) eine optimale Lösung bestimmt.

¹⁰Für den Fall $\beta_1 = 2$ stellen DOMSCHKE ET AL. ([DSV93], S. 270 ff.) spezielle Verfahren vor. Diese können als Relaxationen komplexerer Probleme dienen. Bei der Relaxation eines Problems werden einige Nebenbedingungen gelockert oder komplett weggelassen. Siehe hierzu u.a. auch Fußnote 59 auf S. 67.

¹¹Die Operationen werden dann mit *pmtn* gekennzeichnet; *pmtn* ist die Abkürzung für *preemption* (engl.: Zuvorkommen). Dies ist in dem Sinne zu verstehen, daß ein anderer Auftrag dem aktuellen Auftrag zuvorkommt, und der aktuelle Auftrag daher unterbrochen werden muß.

¹²Bei den letzten beiden Fällen der Unterbrechbarkeit werden die Operationen mit \circ bzw. *no wait* bezeichnet.

¹³*prec* für *precedence*; engl.: Vorausgehen

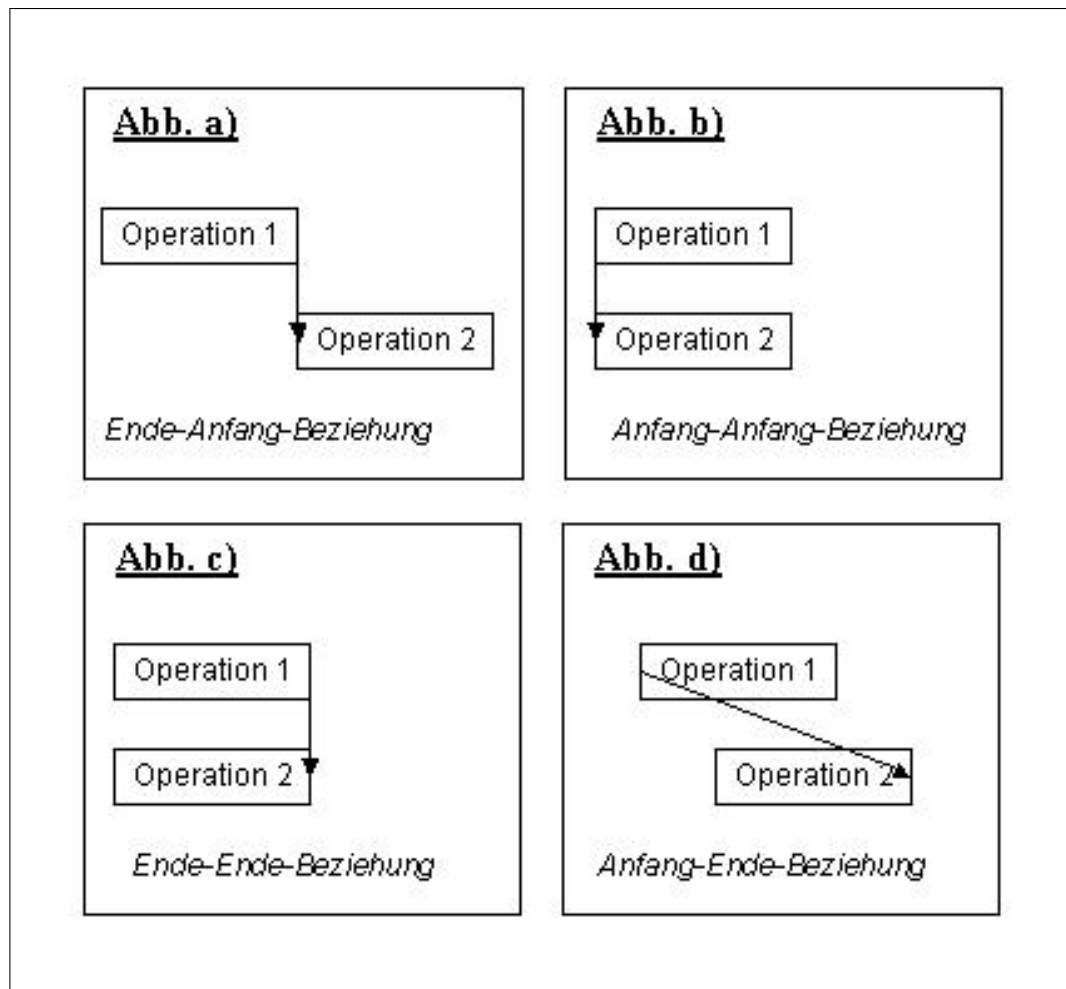


Abbildung 1.1: Reihenfolgebeziehungen

- β_5 , **Bearbeitungsdauer:** Die Bearbeitungsdauer $t(o_{j,i}, m_i)$ jedes Auftrags oder jeder Operation kann funktional abhängig von dem Auftragsvolumen oder konstant sein.
- β_6 , **Rüstzeiten:** Die Rüstzeiten zur Vorbereitung einer Anlage für die nächste Produktion werden nicht betrachtet, oder können maschinen- und/oder reihenfolgeabhängig sein.
- β_7 , **Ressourcen:** Dieser Parameter beschreibt, in welchem Umfang Ressourcen besonders berücksichtigt werden müssen. Es wird angegeben, welche zusätzlichen Ressourcen (Mitarbeiter, Hilfsmittel) in der Planung berücksichtigt, wie weit sie verfügbar sind und durch einzelne Operationen beansprucht werden.
- β_8 , **Fertigstellungstermine:** Es können Fertigstellungstermine (*Deadlines*) für die einzelnen Aufträge ($\beta_8 = f_j$) gegeben sein, die eingehalten werden müssen.
- β_9 , **Operationen:** Dieser Parameter gibt an, ob nur Aufträge mit einer bestimmten Anzahl von Operationen betrachtet werden, oder ob diese Anzahl beliebig ist.
- β_{10} , **Lagerkapazitäten:** Sofern die Aufträge aus mehreren Operationen bestehen, gibt dieser Parameter an, ob und welche Lagerkapazitäten für einzelne Maschinen zur Verfügung stehen.

BRÜSSAU ([Brü02], S. 16) beschreibt darüber hinaus zwei weitere Kriterien, die bei der Modellierung beachtet werden können:

Transportzeiten: Zwischen den einzelnen Operationen eines Auftrags bestehen Transportzeiten, die berücksichtigt werden müssen, oder nicht.

Input-Output-Beziehung: Mit diesem Parameter beschreibt BRÜSSAU die Art des Produktionsprozesses. Dies kann als Erweiterung von Parameter β_3 (Reihenfolgebeziehung) verstanden werden und wird in dem Abschnitt *Stofffluß* dieser Arbeit auf Seite 18 näher beschrieben.

Zielsetzung: Bei Problemen der Ablaufplanung werden nur zeitliche Vorgaben gemacht. Entsprechend können bei der Optimierung keine Kosten, sondern nur zeitorientierte Ziele berücksichtigt werden. In einem bestehenden Ablaufplan können u.a. die folgenden Zeitpunkte ermittelt werden.

Produktionsbeginn: Der Produktionsbeginn P_j eines Auftrags liegt zum oder nach dem Termin der Auftragsfreigabe a_j eines Auftrags (vgl. β_4).

Fertigstellungszeitpunkt: Für jeden Auftrag j resultiert aus dem Ablaufplan ein Fertigstellungszeitpunkt F_j .

Durchlaufzeit: Die Durchlaufzeit D_j berechnet sich als die Differenz zwischen Fertigstellungszeitpunkt und Produktionsbeginn oder Auftragsfreigabe; $D_j := F_j - P_j$ oder $D_j := F_j - a_j$.

Zykluszeit: Die Zykluszeit oder auch **Makespan** ist die Gesamtdauer bis der letzte Auftrag fertiggestellt worden ist; $\text{Makespan} := \max \{ F_j \mid j \in \mathcal{J} \}$

Leerlaufzeit: Jede Maschine m_i hat eine Leerlaufzeit $L_i := \text{Makespan} - \sum_{j \in \mathcal{J}} t(o_{j,i}, m_i)$. Entsprechend ist die gesamte Leerlaufzeit in einem Ablaufplan $L := \sum_{i=1}^n L_i$.

Wartezeit: Wenn die Produktion eines Auftrags oder einer seiner Operationen zu früh begonnen wird, können für die einzelnen Operation $o_{j,i}$ Wartezeiten $w(o_{j,i})$ dadurch entstehen, daß die Maschine für die nachfolgende Operation noch durch einen anderen Herstellungsprozeß belegt ist und somit blockiert ist.

Terminabweichung: Sofern für die Aufträge Fertigstellungstermine gegeben sind, kann die Terminabweichung $T_j := F_j - f_j$ bestimmt werden. Wenn $T_j > 0$ besteht eine Verspätung (engl.: *tardiness*). In diesem Fall entstehen möglicherweise Vertragsstrafen. Bei einer vorzeitigen Produktion (mit $T_j < 0$; engl.: *earliness*) erhöhen sich dagegen evtl. die Kapitalbindungskosten.

Diese Meßgrößen können nun in dem Optimierungsproblem der Ablaufplanung bezüglich ihrer Summe, eines (gewichteten) Durchschnittswertes oder des Maximalwertes minimiert werden. Allgemein wird eine Optimierung anhand einer der folgenden Zielsetzungen vorgenommen.

- Bei der *durchlaufzeitbezogenen* Optimierung werden die Kriterien der Durchlaufzeit oder der Wartezeit verwendet. Dies ist eine Betrachtung, die die Produktionsdauer der einzelnen Aufträge in den Mittelpunkt stellt.

- Wenn *kapazitätsorientierte* Ziele verfolgt werden, z.B. die Minimierung des **Makespan** oder der Leerlaufzeit, orientiert sich die Optimierung besonders an der Auslastung der Produktionsanlage.
- Bei der *terminorientierten* Optimierung wird versucht, eine möglichst große Anzahl von Kunden zufriedenzustellen, indem die maximale oder mittlere Verspätung minimiert wird.

Im Rahmen der betrieblichen Produktionsplanung sind diese Optimierungsziele in der Ablaufplanung jedoch nur Ersatzziele, anhand derer versucht wird, *eine* Kostengröße zu minimieren. So sinken mögliche Strafkosten, wenn weniger Verspätungen entstehen; bei einer Reduktion der Wartezeiten oder der Durchlaufzeit eines Auftrags sinken hingegen die Lagerkosten.

Eine solche, auf ein einseitiges Ziel orientierte Optimierung kann jedoch dazu führen, daß wiederum andere Kosten steigen. So kann die Optimierung der Durchlaufzeit zu hohen Rüstzeiten führen, eine möglichst hohe Auslastung der Maschinen mit geringen Leerlaufzeiten ist nur durch große Losgrößen zu erreichen, oder aus einer Optimierung der Terminabweichung resultiert eine höhere Zykluszeit.

Diese sich widersprechenden Zielsetzungen sind allgemein auch als das von GUTENBERG ([Gut83], S. 216) beschriebene *Dilemma der Ablaufplanung* bekannt. Im Rahmen der Produktionsplanung werden dann möglicherweise nur suboptimale Ergebnisse erreicht.

Um diese Zielkonflikte zu vermeiden, kann das Problem anhand einer *multi-kriteriellen Zielfunktion* bewertet werden. Dann werden die einzelnen Optimierungsziele in absteigender (lexikographischer) Ordnung oder gewichtet miteinander verknüpft (vgl. [DD95], S. 48 ff.).

Kapitel 2

Prozesse und Anforderungen der Verfahrenstechnik

Die Produktion als Herstellungsprozeß von Gebrauchs- und Verbrauchsgütern kann anhand verschiedener Kriterien charakterisiert werden. In dieser Arbeit wird die Unterscheidung anhand der Produktionstechnologie vorgenommen, die die eingesetzten Verfahren im Herstellungsprozeß in Fertigungstechnik und Verfahrenstechnik trennt¹ ([Vah00], S. 3).

In diesem Kapitel werden nachfolgend zunächst die Grundverfahren der Verfahrenstechnik vorgestellt, um damit zugleich die *Verfahrensindustrie* zu umschreiben (Kapitel 2.1). Zur Entwicklung eines Optimierungsmodells für die Produktionsplanung in der Verfahrensindustrie werden anschließend die *Charakteristika verfahrenstechnischer Produktionsprozesse*, insbesondere bei der Chargenproduktion dargestellt (Kapitel 2.2). Abschließend wird in Abschnitt 2.3 die Umsetzung der Anforderungen in verschiedenen IT-Systemen zur Produktionsplanung und -steuerung diskutiert.

2.1 Verfahrenstechnische Produktion

2.1.1 Grundverfahren der Verfahrenstechnik

Verfahrenstechnische Produktionsverfahren bewirken eine stoffliche Umwandlung der bearbeiteten Materialien ([Hem99], S.9). Im Gegensatz zur Fertigungstechnik werden die Eingangsstoffe dabei substanziell verändert. Dabei kommen chemische, physikalische und biologische Grundverfahren zum Einsatz ([DIN88], Teil 1)². Eine Übersicht über die Grundverfahren der Fertigungstechnik und Verfahrenstechnik ist in Tabelle 2.1 zusammengestellt ([DIN87] und [Hem99], S.11).

¹Oftmals wird als dritter Bereich die Energietechnik genannt, bei der durch eine Umwandlung von Energie z.B. Strom produziert wird.

²Die Teile 1-3 der DIN 28004, [DIN88], wurden inzwischen zusammengefaßt und ersetzt durch die DIN EN ISO 10628, [DIN01].

Grundverfahren der Verfahrenstechnik <i>Stoffänderung durch</i>	Grundverfahren der Fertigungstechnik <i>Formänderung durch</i>
<ul style="list-style-type: none"> • Physikalische Grundverfahren (<i>mechanisch, thermisch, elek- trisch/elektromagnetisch</i>): zum Trennen (<i>Filtrieren, Zen- trifugieren, Destillieren, Elek- trosmose, ...</i>) und zum Mi- schen (<i>Rühren, Kneten, Sintern, ...</i>) • Chemische Grundverfahren: thermisch (<i>Brennen, Rösten, ...</i>), katalytisch (<i>Polymerisie- ren, Oxydieren, ...</i>), photo- chemisch (<i>Chlorierung, ...</i>), elektrolytisch und Hochdruck- Verfahren • Biologische Grundverfahren (<i>Gären</i>) 	<ul style="list-style-type: none"> • Urformen (<i>Gießen, Sintern</i>) • Umformen (<i>Schmieden, Walzen, Falten, ...</i>) • Trennen (<i>Sägen, Feilen, Bohren, Hobeln, ...</i>) • Fügen (<i>Schweißen, Lötten, Ver- schrauben, ...</i>) • Beschichten (<i>Lackieren, Galva- nisieren, ...</i>) • Änderung von Stoffeigenschaf- ten (<i>Härten, Glühen, Stauchen, ...</i>)

Tabelle 2.1: Produktionsverfahren in der Verfahrens- und Fertigungstechnik

Einige Verfahren können dabei beiden Produktionstechniken zugeordnet werden³. Als zusätzliche Unterscheidung kann daher die Anwendung auf das verwendete Material dienen. Verfahren der Fertigungstechnik werden auf geometrisch bestimmte, feste Körper angewandt. In der Verfahrenstechnik werden dagegen zumeist flüssige oder pulverisierte Materialien⁴ eingesetzt. Die einzelnen verfahrenstechnischen Prozesse können zeitgleich ablaufen, so etwa die Polymerisierung zeitgleich mit dem Emulgieren ([Hem99], S. 11).

Die verfahrenstechnische Produktion setzt sich aus einzelnen Schritten zusammen, die wiederum eine oder mehrere Grundoperationen umfassen ([DIN88], Teil 1). Die Dauer der Durchführung der einzelnen Schritte in einem Produktionsprozeß wird dabei durch die jeweiligen Grundoperationen und den damit verbundenen naturgesetzlichen Vorgängen bestimmt. Die Dauer einer Produktion kann daher - im Gegensatz zur diskreten Produktion in der Fertigungstechnik - unabhängig von dem verarbeiteten Volumen immer die gleiche Zeit benötigen.

Die Arbeitsanweisungen in der Produktion enthalten zusätzliche regelungstechnische Vorgaben⁵. Diese beziehen sich auf die jeweiligen verfahrenstechnischen Produktionsanlagen. Die technischen Anforderungen und Besonderheiten der verfahrenstechnischen Produktion werden in dem Kapitel 2.2 näher beschrieben.

³Das *Sintern* wird je nach Literaturstelle der Fertigungstechnik oder der Verfahrenstechnik zugeordnet. Sintern ist ein Prozeß unter Wärmebehandlung, bei der ein feinkörniger Feststoff nahe an die Schmelztemperatur erhitzt wird. Die Moleküle bilden dann Festkörperbrücken; die Körner „sintern“ aneinander ([Hem99], S. 122).

⁴Weitere Grundstoffe sind Pasten, Granulate, Mahlgüter oder Mischungen.

⁵Dazu zählen u.a. Rührgeschwindigkeiten oder Fertigungsunterbrechungen.

2.1.2 Die Verfahrensindustrie

Die Verfahrensindustrie, die synonym auch als *Prozeßindustrie* bezeichnet wird, umfaßt alle Betriebe, die Gebrauchs- oder Verbrauchsgüter mit verfahrenstechnischen Produktionsmethoden herstellen. Zu diesen Produkten zählen u.a. Arzneimittel, Nahrungsmittel, Raffinerieprodukte, bauchemische Produkte, ...etc. Grob kann dieses Produktionsspektrum (für die Chemische Industrie) in die Bereiche der anorganischen und organischen Industriechemikalien, sowie der chemischen Spezialerzeugnisse (pharmazeutische Produkte, Waschmittel, Farben, etc.) unterteilt werden. Ein Großteil von ca. 70% dieser Produkte wird dabei durch andere Betriebe weiterverarbeitet [VCI03].

Der Begriff der Prozeßindustrie wird oft auch nur eingeschränkt auf die Chemische Industrie verwendet. Die wichtigsten Industriezweige der Verfahrensindustrie sind daneben jedoch auch die Pharmazeutische Industrie, die Nahrungs- und Genußmittelindustrie, die Zement-, Papier-, Glas-, Stahl-, Hütten- sowie die Mineralöl verarbeitende Industrie ([CM94], S. 874).

Die Produktion in der Prozeßindustrie stellt aufgrund der Produktionsprozesse und gesetzlicher Regelungen besondere Anforderungen. So muß zum Verbraucherschutz besonders bei der Herstellung von pharmazeutischen Produkten und Lebensmitteln eine *Chargen-Rückverfolgbarkeit* über alle Produktionsstufen gewährleistet werden ([Blö99], S. 8). Auch die *Sicherheits- und Umweltschutzbestimmungen* erfordern besondere Maßnahmen in der Lagerung und Entsorgung der eingesetzten Materialien. So werden z.B. für die Zusammenlagerung bestimmter Chemikalien Einrichtungen mit Explosionsschutz benötigt [VCI98]. Abfallprodukte der Produktion müssen einem besonderen Recyclingprozeß zugeführt werden.

2.2 Charakteristika der verfahrenstechnischen Produktion

Die verfahrenstechnischen Grundverfahren erfordern besondere Produktionsformen. Nachfolgend werden daher zunächst die möglichen *Prozeßabläufe* der Verfahrensindustrie beschrieben und anschließend die dazu notwendigen *Produktionsanlagen* vorgestellt. Abschließend werden aus diesen Charakteristika die Voraussetzungen für die *Ablaufplanung* in der Chargenproduktion abgeleitet.

2.2.1 Prozeßablauf

Die Produktion kann mit einem Input-Output-Modell beschrieben werden (vgl. [DoS03], S. 85): Produktionsfaktoren (*Input*) werden in einem *Produktionsprozeß* kombiniert und transformiert, um die gewünschten Produkte (*Output*) herzustellen.

In diesem Abschnitt werden die Besonderheiten der verfahrenstechnischen Produktion bei der Zusammensetzung aller Produkte in dem *Stofffluß* erläutert, die dazu notwendig sind. Anschließend werden die Charakteristika der *Prozeßführung* in dem eigentlichen Produktionsprozeß beschrieben.

Stofffluß

In der mehrstufigen Produktion ist der Stofffluß ein maßgeblicher Inputfaktor für die Gestaltung einer Produktionsanlage und die Materialbedarfsrechnung der Rohstoffe und Zwischenprodukte. Die Beziehung der einzelnen Produkte untereinander kann allgemein in der Form von Gozinto-Graphen⁶ dargestellt werden ([DoS03], S. 95 f. und Abb. 2.1 dieser Arbeit). In diesem gerichteten Graph stellen die Knoten die einzelnen Materialien (Rohstoffe oder Produkte) dar. Durch eine Gewichtung der Kanten kann angegeben werden, in welchem Anteil des Produktes ein (direktes) Zwischenprodukt⁷ oder ein Rohstoff in der Herstellung verwendet wird.

In der Klassifizierung von Gozinto-Graphen, und somit der Materialverarbeitung werden vier (vgl. [DoS03], S. 96; [Dyc94], S. 187) bis sechs (vgl. [BCT83], S. 1127, [HS86], S. 151) Elementartypen unterschieden:

- Bei *durchgängiger Produktion* (vgl. Abb. 2.1, a) wird genau ein Inputprodukt in genau ein Outputprodukt umgewandelt. Wenn nur ein Endprodukt hergestellt wird, wird von *serieller Produktion*, ansonsten von *paralleler Produktion* gesprochen.
- Bei *konvergierender Produktion* (Abb. 2.1, b) werden mehrere Inputprodukte in der Herstellung nur eines Endproduktes verwendet.
- In der *divergierenden Produktion* (Abb. 2.1, c) wird ein Inputprodukt dagegen in der Herstellung mehrerer Produkte weiterverarbeitet.
- In einer *allgemeinen Netzwerkstruktur* (Abb. 2.1, d) ist Vermischung der zuvor genannten Strukturen gegeben.

Die beiden folgenden Typen können als Sonderfälle der allgemeinen Netzwerkstruktur betrachtet werden:

- Wenn in dem Produktionsprozeß aus n Inputprodukten zeitgleich m verschiedene Produkte hergestellt werden, spricht man auch von einer *umgruppierenden Produktion* (Abb. 2.1, e).
- Wenn ein Nebenprodukt der Produktion wieder als Inputprodukt verwendet wird, wird dies als *zyklische Produktion* (Abb. 2.1, f) bezeichnet.

Von den hier genannten Produktionsstrukturen ist die umgruppierende und zyklische Produktion nur in der Verfahrensindustrie zu finden. Eine umgruppierende Produktion findet statt, wenn in der Herstellung eines Endproduktes mehrere Nebenprodukte entstehen wie z.B. in der Kokerei oder bei Raffinerien.

In der zyklischen Produktion fließt ein Nebenprodukt wieder in den Herstellungsprozeß eines Vorproduktes ein. Wenn das Nebenprodukt in derselben Produktionsstufe wiederverwendet wird, spricht man von einem einstufigen Zyklus, ansonsten von einem mehrstufigen Zyklus ([CM94], S. 877). Die Wiederverwendung eines Produkts in einer zyklischen Produktion kann folgendermaßen unterschieden werden:

⁶Der Begriff *Gozinto-Graph* entstammt der englischen Beschreibung *the part that goes into* zur Beschreibung der Erzeugnisstruktur ([GT04], S. 180).

⁷Nachfolgend werden Produkte, die direkt in der Produktion eines nachfolgenden Produktes verwendet werden, immer als *Zwischenprodukte* bezeichnet. Produkte, die dagegen über eine oder mehrere Zwischenproduktstufen in der Herstellung eines Produktes verwendet werden, werden als *Vorprodukte* bezeichnet.

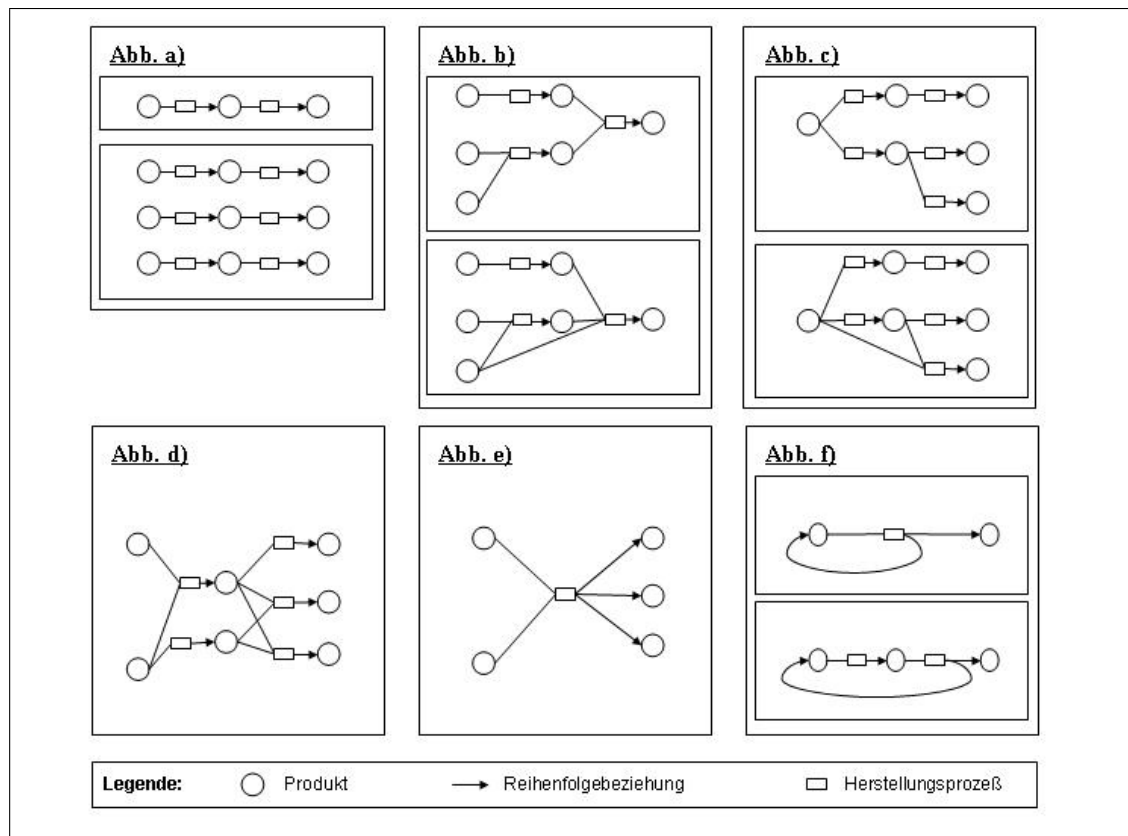


Abbildung 2.1: Darstellung von Produktionsstrukturen durch Gozinto-Graphen (ohne Gewichtung der Kanten)

- Es handelt sich um ein katalytisches Produkt. Dieses wird unverändert in Form und Menge wieder in dem Produktionsschritt verwendet.
- Das Nebenprodukt oder ein Anteil des Endprodukts ist von minderer Qualität, der aber nicht entsorgt werden muß, sondern als Input erneut verwendet und damit „recycled“ werden kann.
- Ein Anteil des Endprodukts wird von der produzierten Menge abgezweigt und als Ansatz in einem vorhergehenden Produktionsschritt wiederverwendet.

Sofern ein Gozinto-Graph zyklensfrei ist, kann ausgehend von dem Volumen eines Produktes rekursiv der Bedarf an Zwischenprodukten und Rohstoffen in der aktuellen (*Primärbedarf*) und den vorgelagerten Produktionsstufen (*Sekundärbedarf*) berechnet⁸ werden (vgl. [GTe04], S. 184 ff.). In der Fertigungsindustrie wird die Auflistung des Materialbedarfs anhand eines Gozinto-Graphen als Stückliste, in der Verfahrenindustrie als *Rezeptur*⁹ bezeichnet.

Neben der Berechnung des Materialbedarfs beschreiben Gozinto-Graphen zugleich auch die Reihenfolgebeziehung, die in der Herstellung der einzelnen Produkte berücksichtigt werden muß. In diesem Fall ist zu beachten, daß Zyklen innerhalb des Graphen nicht in

⁸In der englischsprachigen Literatur wird hierfür der Begriff *bill-of-materials* verwendet.

⁹Eine genauere Beschreibung von Rezepturen erfolgt in Kapitel 2.3.1 dieser Arbeit.

der gleichen Weise in der Produktionsreihenfolge abgebildet werden können. Ein Nebenprodukt kann erst in einer später stattfindenden Produktion wiederverwendet werden.

Prozeßführung

Bei der Prozeßführung betrachtet man den Materialfluß in der Produktionsanlage. In den einzelnen Produktionsschritten können die Zutaten entweder *kontinuierlich* oder *diskontinuierlich* hinzugefügt werden. Entsprechend werden das gefertigte Produkt und mögliche Nebenprodukte entweder kontinuierlich oder diskontinuierlich in die nächste Produktionsstufe oder ein Zwischenlager überführt. Die diskontinuierliche Produktion wird auch als *Batch-* oder *Chargenproduktion* bezeichnet.

In der **kontinuierlichen Produktion** durchlaufen die Materialien alle Produktionsstufen in einem stetigen Fluß. Um diesen Materialfluß aufrecht zu erhalten, muß ein reibungsloser Produktionsablauf und Materialtransport zwischen den einzelnen Produktionsstufen gewährleistet sein.

Anlagen zur kontinuierlichen Produktion weisen daher eine hohe automatisierte Steuerung auf. Der Ablauf ist bereits durch den Prozeß vorgegeben und wird mit der Konstruktion der Produktionsanlage fest installiert. Eine kontinuierliche Produktion kann somit nur auf *Mono-* oder *Einproduktanlagen* stattfinden, die nur für ein Produkt oder eine Produktgruppe konzipiert sind. Die Planung weist nur wenige Freiheitsgrade in der Ablaufplanung auf, die durch das Umrüsten zwischen den einzelnen Varianten entstehen.

Kontinuierliche Produktion findet u.a. in der Massenproduktion von chemischen Grundstoffen und in der Mineralölindustrie statt.

In der **Chargenproduktion** werden die einzelnen Produktionsschritte im wesentlichen zeitentkoppelt voneinander durchgeführt. Die Gesamtheit der in einem Produktionsschritt produzierten Menge wird als *Charge* oder *Batch* bezeichnet. Die Zutaten werden einmalig (oder zu definierten Zeitpunkten in vorgegebenen Mengen) der Produktion hinzugefügt und das Produkt als Ganzes wieder entnommen.

Die Chargenproduktion ist sehr flexibel einsetzbar und weniger anfällig gegen Störungen im Produktionsbetrieb. Sie findet u.a. auf flexibel umrüstbaren *Mehrzweckanlagen* statt. Ein Produktionsschritt benötigt dabei jeweils nur eine Teilanlage. Innerhalb der Chargenproduktion können dabei einzelne Produktionsschritte auch (semi-) kontinuierlich stattfinden¹⁰, etwa wenn direkt im Anschluß an eine Produktion ein Mahlvorgang stattfindet. In diesem Fall können zeitgleich auch mehrere Teilanlagen benötigt werden. Die Ablaufplanung muß daher insgesamt eine große Anzahl an Freiheitsgraden berücksichtigen.

Chargenproduktion findet oftmals auftragsorientiert statt, wenn eine hohe Anzahl von Produkten in (evtl.) geringerem Volumen hergestellt werden soll. Sie wird unter anderem in der Nahrungs-, Kunststoff- und Pharmaindustrie eingesetzt.

Da in dieser Arbeit die Produktionsplanung in der Chargenproduktion betrachtet wird,

¹⁰KIESSWETTER bezeichnet diese Produktionstypen als *Batch-Intern*, wenn eine Produktionsanlage kontinuierlich mit Einsatzstoffen beschickt wird und die Endprodukte diskontinuierlich in Chargen ausgebracht werden ([Kie99], S. 23 ff.), bzw. als *Batch-Extern* bei umgekehrter Handlungsweise. Er zeigt für Lösungsverfahren die Dominanzreihenfolge *Batch-Extern* → *Batch-Intern* → *Chargen* → *Konti* auf, d.h. ein Lösungsverfahren, das ein Optimierungsproblem mit dem Produktionstyp *Batch-Extern* lösen kann, kann auch ein Optimierungsproblem mit dem Produktionstyp *Batch-Intern* lösen, usw.

werden in der weiteren Beschreibung nur die Anforderungen der Chargenproduktion beschrieben.

2.2.2 Anlagenstruktur

Zur Planung verfahrenstechnischer Produktionsprozesse müssen die beschriebenen Grundverfahren und Produktionsabläufe auf eine geeignete Produktionsanlage abgebildet werden. Hierzu wird in Anlehnung an die NAMUR-Empfehlung¹¹ NA 33 der Aufbau einer Produktionsanlage beschrieben [NAM93].

Die kleinste technische Einheit (Bauteil) einer Produktionsanlage wird Anlageteil genannt. Man kann zwischen *aktiven* und *passiven* Anlageteilen unterscheiden. Aktive Anlageteile haben technische Funktionalitäten, die den Anforderungen zur Durchführung eines verfahrenstechnischen Produktionsprozesses entsprechen. Sie dienen also der Produktion oder dem Erhalt eines Produkts. Passive Anlageteile sind z.B. Kessel und Rohre. Sie werden zur Lagerung oder zum Transport verwendet. Mehrere Anlageteile werden zu einer technischen Einrichtung, nachfolgend *Anlagekomponente* genannt, zusammengeschlossen, wobei ein Anlageteil immer nur einer Anlagekomponente zugeordnet sein sollte.

Für die weitere Beschreibung werden vier verschiedene Typen von Anlagekomponenten unterschieden:

- Anlagekomponenten zur **Produktion**,
- Anlagekomponenten zur **Lagerung**,
- Anlagekomponenten für den **innerbetrieblichen Transport** und
- Anlagekomponenten zur **Abfüllung**.

Mehrere Anlagekomponenten sind in einer Teilanlage zusammengefasst. Eine Teilanlage kann dabei „zumindest zeitweise selbständig betrieben werden“ ([DIN88], Teil 1). Alle wegen ihrer örtlichen Nähe zueinander befindlichen Teilanlagen bilden daher die (Produktions-) Anlage eines Betriebs.

Anhand der Verrohrung können Produktionsanlagen klassifiziert werden¹². Abbildung 2.2 zeigt beispielhaft den Aufbau einer Produktionsanlage. Dieser umfasst die Gesamtheit aller Einrichtungen und Bauten zur Durchführung eines Verfahrens, d.h. zur Herstellung oder Beseitigung von Stoffen durch die beschriebenen physikalischen, chemischen oder biologischen Vorgänge. Mehrere Anlagen sind abschließend zu einem Anlagenkomplex und diese in einem Werk verbunden.

In dieser Arbeit wird ein Optimierungsmodell für die Produktionsplanung auf Ebene einer Produktionsanlage entwickelt. Die Anlagekomponenten sind dabei die kleinste betrachtete Basis in der Planung einzelner Produktionsschritte.

¹¹NAMUR ist die Normenarbeitsgemeinschaft für Meiß- und Regeltechnik in der Chemischen Industrie.

¹²Dies wird in dem Abschnitt *Innerbetrieblicher Transport* auf S. 25 näher beschrieben.

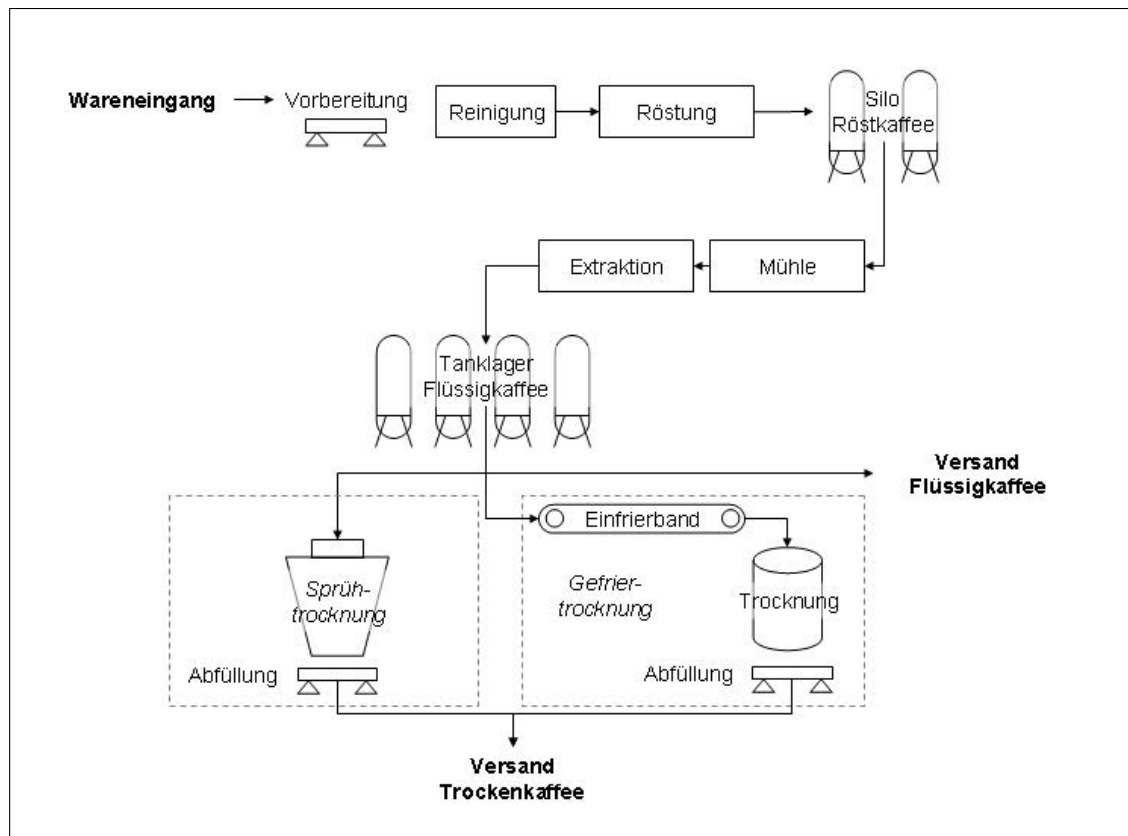


Abbildung 2.2: Produktionsanlage zur Trockenkaffeeherstellung

Produktion

Die Anlagekomponenten zur Produktion unterscheiden sich in Funktionalität, Leistung und Kapazität. In ihnen findet der eigentliche Herstellungsprozeß statt. Zur Liste der Produktionsanlagen werden u.a. Dissolver¹³, Extruder¹⁴, Trockner, Mühlen, Filter und Öfen gezählt. Diese werden nachfolgend verallgemeinernd als *Reaktoren* bezeichnet.

Für die Losgrößen- oder Volumenplanung ist besonders wichtig zu beachten, daß einzelne Typen dieser Reaktoren ein Minimal- (sog. *Sumpfmenge*) und Maximalvolumen haben. Insbesondere muß das Volumen einer Charge mindestens so groß wie die Sumpfmenge sein, da sonst die Produktion nicht durchgeführt werden kann (z.B. wenn im Reaktor ein Rührer ist, welcher zumindest bedeckt sein muß).

Lagerung

Im Unterschied zur Fertigungsindustrie sind bei verfahrenstechnischen Produktionsprozessen aufgrund der stofflichen Eigenschaften der Produkte die Möglichkeiten der Lagerung explizit zu berücksichtigen. Hierbei wird eine Trennung von Rohstoffen, Zwischenprodukten und Endprodukten vorgenommen.

¹³Ein *Dissolver* vermischt unter hoher Geschwindigkeit zwei oder mehrere Stoffe zu einem fluiden Gemisch, wobei die Stoffe keine chemische Reaktion miteinander eingehen.

¹⁴Ein *Extruder* ist eine Schneckenpresse, bei der dickflüssige oder feste Materialien durch eine formgebende Öffnung gepresst werden.

Rohstoffe werden in einem separaten Rohstofflager mit z.B. Schüttgutbehältern oder *Big Bags* für pulverisierte Rohstoffe und Silos (Tanks) für Flüssigkeiten oder Gase gelagert. Bei Anlieferung der Rohstoffe wird oftmals noch eine Qualitätskontrolle durchgeführt, so daß diese erst nach Freigabe für die weitere Produktion zur Verfügung stehen. Entsprechend müssen Rohstoffe, die einer Qualitätskontrolle unterzogen werden, separat gelagert und der Zugriff gesperrt werden.

Die Verwaltung und Beschaffung der Rohstoffe unterliegt dabei dem Disponenten. Eine Berücksichtigung der verfügbaren Lagerplätze in der Produktionsplanung ist daher i.d.R. nicht notwendig. Dies ist nur dann notwendig, wenn sich die Lagerplätze als Engpaß erweisen. Dann müsste bspw. die Produktion eines oder mehrerer Aufträge vorgezogen werden, wenn bekannt ist, daß eine Einlagerung die Kapazität des Rohstofflagers übersteigen würde.

Für die Zwischenlagerkapazitäten gibt BLÖMER eine Klassifikation an ([Blö99], S. 21 ff.). Bei *no intermediate storage* sind keine Kapazitäten zur Zwischenlagerung vorhanden. Entsprechend sind bei *finite intermediate storage* endliche und bei *unlimited intermediate storage* unbegrenzte Lagerkapazitäten gegeben. Weiter gibt er eine Übersicht über verschiedene Lagertypen an, die teilweise die Möglichkeit der gleichzeitigen Einlagerung unterschiedlicher Produkte zulassen. Es werden also nicht die einzelnen Lagerplätze betrachtet.

Die Zwischenlagerung muß bei der Produktionsplanung differenziert berücksichtigt werden, wenn spezielle Lagermöglichkeiten (d.h. Lagerung in fest installierten Silos oder Transportbehältern¹⁵) eingesetzt werden. So kann es notwendig sein, daß ein Lagertank über ein Rührwerk verfügen muß¹⁶, damit ein Zwischenprodukt nicht verklumpt oder sich Schwebestoffe absetzen.

In dieser Arbeit werden die Lagerplätze für Zwischenprodukte explizit betrachtet. Für Zwischenprodukte stehen zwei Möglichkeiten der Lagerung zur Verfügung. Entweder werden die Produkte in der herstellenden Anlagekomponente (Reaktor) zwischengelagert, bis der nachfolgende Produktionsschritt durchgeführt werden kann¹⁷, oder sie werden in speziellen Silos zwischengelagert. Die Einbindung der Silos in der gesamten Produktionsanlage kann unterschiedlich erfolgen. Für diese Arbeit wurden hier drei mögliche Anlagenkonfigurationen identifiziert:

- Jedem Reaktor sind ein oder mehrere Silos zur Zwischenlagerung zugeordnet. Die Silos können nur von einem Reaktor aus befüllt werden. Die Entnahme kann für jede nachfolgende Produktion in einen beliebigen Reaktor erfolgen (vgl. Abb. 2.3, Bild a).

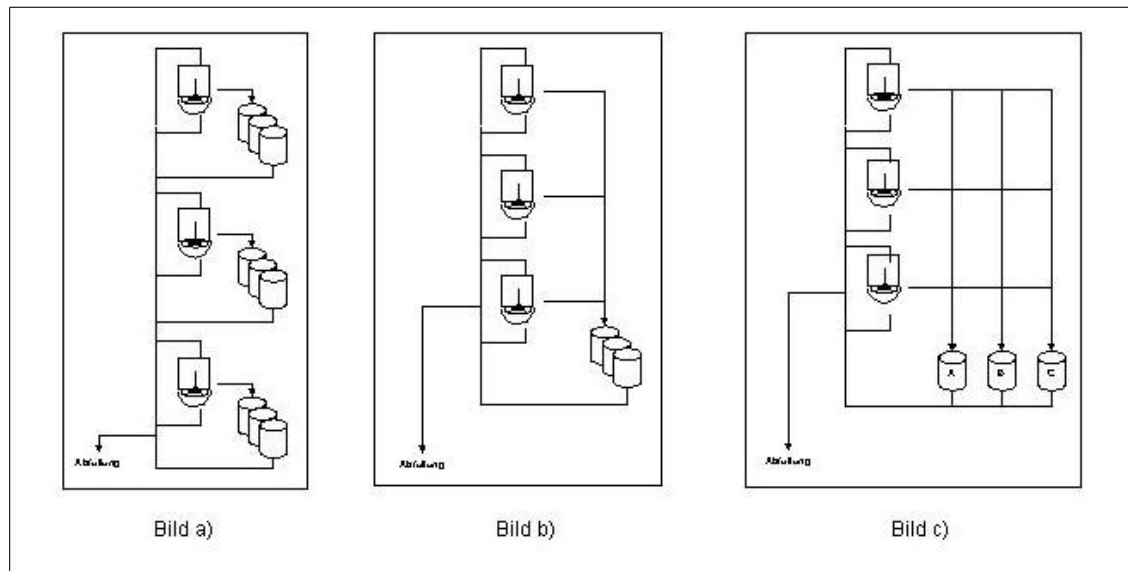
Diese Anlagenkonfiguration wird vor allem in Produktionsprozessen verwendet, die in Kampagnenfahrweise betrieben bzw. die als Flow Shop-Problem formuliert werden. Die Silos sind dabei i.d.R. produktspezifisch definiert, d.h. nur für die Zwischenlagerung bestimmter Produkte zugelassen.

¹⁵Vgl. hierzu die Beschreibung im nachfolgenden Abschnitt *Innerbetrieblicher Transport*.

¹⁶Weitere notwendige Funktionseigenschaften eines Silos können die Temperierung des Zwischenproduktes, Lagerung unter Druck/Vakuum oder explosionsgeschützte Lagerung sein.

¹⁷Dies bedingt, daß das gesamte Volumen des hergestellten Produktes möglichst nur in einem nachfolgendem Produktionsprozeß verarbeitet wird. Entsprechend ist der Reaktor für die Verwendung durch weitere Produktionsprozesse so lange gesperrt bis das Produkt in der nachfolgenden Produktionsstufe verarbeitet wurde.

- Jeder Reaktor ist an jedes Silo angebunden und jedes Produkt kann in jedem Silo zwischengelagert werden (vgl. Abb. 2.3, Bild b). Dies ist die allgemeinste Formulierung. Alle weiteren Anlagenkonfigurationen sind Spezialfälle durch Einschränkung in der Verrohrung oder Einlagerung der Produkte.
- Jeder Reaktor ist an jedes Silo angebunden. Ein Produkt kann aber nur in jeweils spezifischen Silos zwischengelagert werden (vgl. Abb. 2.3, Bild c). Durch Angabe der Verrohrung können hier Einschränkungen berücksichtigt werden.



Die Silos zur Zwischenlagerung können (a) einem einzelnen Reaktor zugeordnet sein, (b) global für alle Reaktoren zugänglich sein oder (c) produktspezifisch sein.

Abbildung 2.3: Möglichkeiten der Zwischenlagerung

Für die weitere Beschreibung des Optimierungsmodells sei mit der nachfolgenden Definition noch eine produktspezifische Lagereigenschaft gegeben.

Definition 2.1 Wenn bei der Lagerung eines Produkts Batch-Mix erlaubt ist, dann können Chargen desselben Produkts aus verschiedenen Produktionsvorgängen zeitgleich in einem Silo gelagert (und somit vermischt) werden.

Die Lagerung von *Endprodukten* ist abhängig von der jeweiligen Strategie der operativen Produktionsplanung. Bei auftragsorientierter Produktionsplanung können die Produkte direkt abgefüllt werden. Die weitere Vorgehensweise für diesen Fall wird in dem nachfolgenden Abschnitt *Abfüllung* näher beschrieben.

Bei Planung auf Lager müssen die Produkte dagegen vor der Abfüllung und dem endgültigen Versand noch einmal zwischengelagert werden. Hierzu sind im allgemeinen separate Lagerplätze vorgesehen, so daß die Verfügbarkeit und der Füllgrad der Silos zur Lagerung der Endprodukte in der Produktionsplanung nicht weiter berücksichtigt werden muß. Die Produktion der jeweiligen Produkte wird dann nur auf Bedarf und mit dem Auftragsvolumen durchgeführt, das durch das freie Volumen eines Endprodukt-Silos vorgegeben wird.

Innerbetrieblicher Transport

Der innerbetriebliche Transport zwischen den einzelnen Anlagekomponenten wird unterschieden in *Förderwege* und *Fördermittel* ([Hem99], S. 13).

Allgemein wird unter den Förderwegen die *Verrohrung* zum Transport von Flüssigkeiten oder Gasen zusammengefasst. Durch Schieber oder Ventile können einzelne Rohrleitungen zu einem flexiblen Rohrleitungssystem geschaltet werden, so daß i.d.R. zwischen allen Anlagekomponenten die Möglichkeit des Transports von Rohstoffen oder Produkten besteht.

Fördermittel bewegen das Material¹⁸. Feststoffe können durch diskontinuierliche Förderer oder Stetigförderer transportiert werden. Stetigförderer sind fest installierte Band- oder Kettenförderanlagen zwischen einzelnen Anlagekomponenten. Bei diskontinuierlicher Förderung durch Kräne oder Aufzüge muß das Material in einem entsprechenden Transportbehälter zwischengelagert werden.

Bei der Produktion kann sowohl die Verrohrung, als auch die Anzahl der Transportbehälter einen Engpaß darstellen. Da die Investitionskosten für solche Transportbehälter jedoch weitaus geringer sind, als für die produzierenden Anlagekomponenten, ist hier die Beschaffung weiterer Transportbehälter und die Installation eines Kanban-Systems¹⁹ zur Verwaltung der Behälter die einfachste Möglichkeit, Engpässe in der Produktion zu vermeiden. Daher werden diese oftmals nicht in der Produktionsplanung berücksichtigt.

Klassifikation einer Produktionsanlage anhand der Verrohrung In einer *Einproduktanlage* wird mit jeder Charge das gleiche Produkt hergestellt ([NAM99], S. 15 f.). Hier ist nur eine Variation der Steuerungsparameter in den einzelnen Verfahrensschritten möglich. Die Anlagekomponenten einer Einproduktanlage sind daher oft auch fest als *Einstranganlage* verschaltet. In dieser durchlaufen alle Chargen dieselbe Produktionsreihenfolge. Es werden aber unterschiedliche Produkte hergestellt. Die Verwendung von Einproduktanlagen ist nur bei einer Massenproduktion wirtschaftlich sinnvoll. Einstranganlagen werden dagegen bei einem Produktspektrum von Produkten mit gleichen Produktionsschritten eingesetzt.

Eine *Mehrstranganlage* (vgl. Abb. 2.4) besteht dann aus mehreren parallel geschalteten Einstranganlagen. Hier kann für jede Charge der einzelne Produktionsstrang ausgewählt werden. Innerhalb dieses Produktionsstrangs durchlaufen alle Chargen wieder die gleiche Reihenfolge.

Wenn dagegen in jedem Produktionsschritt der zu verwendende Reaktor für jede Charge separat ausgewählt werden kann, wird die Anlage als *Mehrstrang-Mehrweg-Anlage* bezeichnet. Hierbei ist die Reihenfolge des Produktionsdurchlaufs für alle Chargen und Produkte gleich. Man spricht hier auch von einer *Mehrproduktanlage*.

Die größte Anzahl an Freiheitsgraden im Produktionsdurchlauf besteht bei einer *Mehrzweckanlage*. Hier kann jedes Produkt und sogar jede Charge in einer eigenen, den Pro-

¹⁸In einem Rohrleitungssystem sind dies die Pumpen zum Transport von Flüssigkeiten und Kompressoren, Gebläse oder Ventilatoren für die Förderung von Gasen.

¹⁹Ein Kanban-System dient der Produktionssteuerung. Jeder Produktionsstelle ist ein Bestandspuffer mit genau festgelegter Menge der Materialien zugeordnet. Wird in diesem Puffer der Mindestbestand unterschritten, wird ein Nachschubauftrag ausgelöst. Die als Nachschub zu liefernde Menge ist auf einer Karte (japan. *Kanban*) verzeichnet. Diese wird dann von der vorgelagerten Produktionsstelle hergestellt bzw. aus dem Lager geliefert. Durch den rückwärts gerichteten Informationsfluß richten sich alle vorgelagerten Produktionsstellen auf den Bedarf der jeweils nachgelagerten Stelle ein.

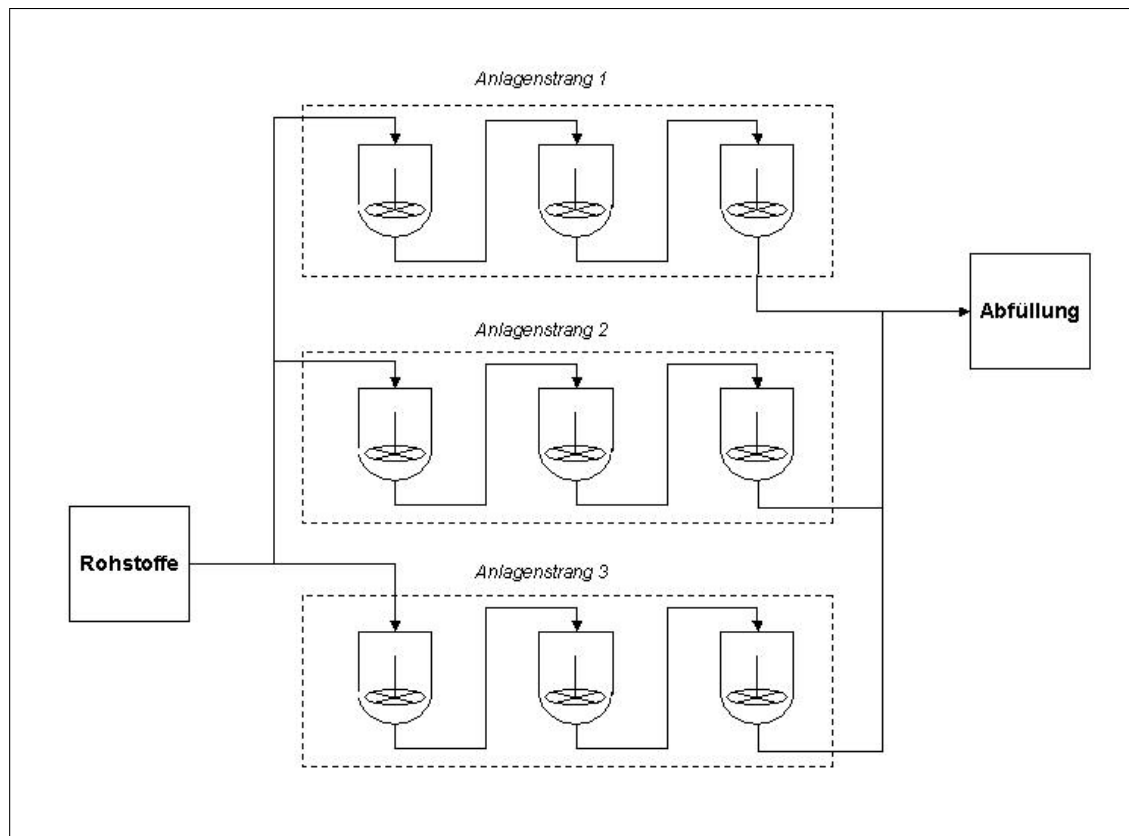


Abbildung 2.4: Mehrstranganlage mit drei Einzelsträngen

duktionsschritten entsprechenden Reihenfolge die Produktionsanlage durchlaufen. Mehrzweckanlagen bieten eine hohe Flexibilität zur Produktion unterschiedlichster Produkte. Allerdings bedeutet dies auch einen größeren Aufwand in der Steuerung der Anlage, für die ein hoher Automatisierungsgrad vorteilhaft ist.

Die Klassifikation der Produktionsanlagen ist in Tabelle 2.2 zusammengefasst. In Anlehnung an das Klassifikationsschema für Scheduling-Probleme in Kapitel 1.2.2 ist bei Verwendung einer Mehrzweckanlage ein Reihenfolgeproblem dann als Job Shop-Problem zu formulieren. In den anderen Fällen genügt dagegen die Formulierung eines Flow Shop-Problems.

<i>Anlage bei .. Verrohrung für</i>	spezielles Produkt	Produktfamilie	beliebiges Produkt
fest	Einprodukt-, Einstrang-	Mehrprodukt-, Einstrang-	Mehrprodukt-
parallel	-	Mehrstrang-	Mehrzweck-
beliebig	-	Mehrstrang- Mehrweg-	Mehrzweck-

Tabelle 2.2: Klassifikation der Anlagenstruktur

Koordination der Produktionsstufen In der Entwicklung eines Optimierungsmodell ist bei Mehrzweckanlagen zusätzlich die *nebenläufige*, *parallele* oder *serielle Produktion* zu berücksichtigen. Wenn eine Anlage nicht nur eine sequentielle Verrohrung aufweist, dann ist neben dem bereits genannten höheren Steuerungsaufwand auch eine höhere Komplexität in der Planung vorhanden. Hier muß dann für jede Produktionsstufe separat eine Entscheidung über die zu verwendenden Anlagekomponenten getroffen werden. Dies ist bei einer vorgegeben Durchlaufreihenfolge einer Einstrang- oder Mehrstranganlage nicht der Fall.

Die **nebenläufige Produktion** bezeichnet die Herstellung eines Produkts, bei der zwei oder mehr Zwischenprodukte (zeitgleich oder nacheinander auf einem oder mehreren Reaktoren) vorproduziert werden müssen (und nicht als Rohstoff einem Lager entnommen werden können; vgl. Abb. 2.5). Hier ist in der Erstellung eines Ablaufplans vor allem die Zulässigkeit einer Lösung zu prüfen. Bei der Berechnung der Produktionszeiten sind die mehrfachen Reihenfolgebeziehungen zwischen einem Produkt und seinen Zwischenprodukten zu berücksichtigen.

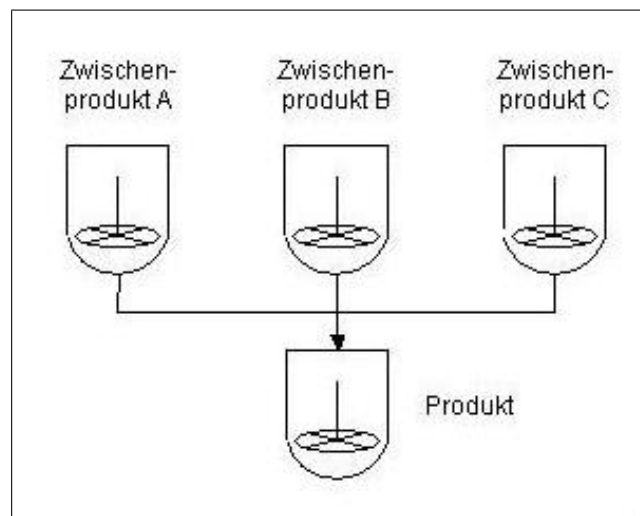


Abbildung 2.5: Nebenläufige Produktion mit 3 Zwischenprodukten

Bei der **parallelen** oder **seriellen Produktion** wird die Gesamtmenge einer Charge in der nachfolgenden Produktionsstufe aus Kapazitätsgründen in mehrere Produktionsprozesse auf mehreren Reaktoren (vgl. Abb. 2.6) oder (bei serieller Produktion) auf einem Reaktor aufgeteilt. Im Fall der seriellen Produktion muß zusätzlich noch die Möglichkeit der Lagerung der Zwischenprodukte gewährleistet sein. In beiden Fällen ist zusätzlich zu der Ablaufplanung in der Volumenplanung die Aufteilung des Chargenvolumens zu berechnen.

Abfüllung

Die Abfüllung ist kein eigentlicher Schritt der Produktion, sondern bereits ein Zwischenschritt zum Versand der Endprodukte. Die Einbindung der Abfüllung in die Produktionsplanung ist daher abhängig von der jeweiligen Strategie der operativen Produktionsplanung.

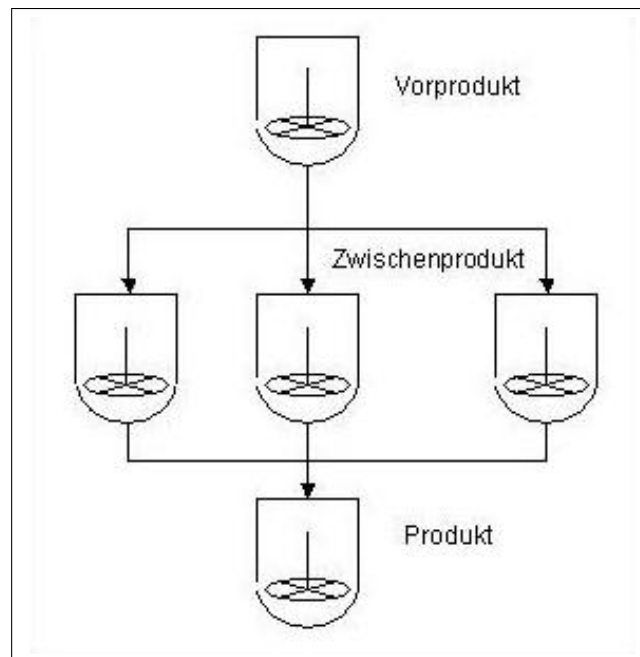


Abbildung 2.6: Parallele Produktion in 3 Chargen

Bei einer Produktion auf Lager werden die Endprodukte in eigenen Lagerplätzen (zwischen-)gelagert und erst nach Auftragseingang eines Kundenauftrags auftragsspezifisch abgepackt. Die Planung (und Durchführung) der Abfüllung ist damit entkoppelt von der eigentlichen Produktionsplanung. Bei auftragsorientierter Produktionsplanung können die Endprodukte dagegen direkt aus dem produzierenden Reaktor heraus abgefüllt werden.

Aufgrund oftmals vieler unterschiedlicher Verpackungsbehälter und entsprechender Vorschriften²⁰ steigt die Produktvielfalt künstlich auf ein Vielfaches der ursprünglich hergestellten Endprodukte. Generell können zwei Typen Verpackungsbehälter für den Versand von Endprodukten klassifiziert werden.

- **Abfüllung in Gebinde:** Das Endprodukt wird in lagerbare Behälter (z.B. Flaschen, Dosen, Eimer, ...) abgefüllt. Die Abfüllung in ein Gebinde ist zeitlich nicht terminiert und kann zu jedem Zeitpunkt nach dem Produktionsende durchgeführt werden.
- **Abfüllung in TKW:** Das Endprodukt wird in einen LKW (Tankkraftwagen, TKW) abgefüllt. Die Abfüllung in einen TKW ist zeitlich terminiert, da solche Transportfahrzeuge nur eingeschränkt und in Absprache mit einem Spediteur zur Verfügung stehen.

Sofern nicht sofort nach dem Produktionsende abgefüllt werden kann, kann in beiden Fällen das Endprodukt in dem Reaktor oder in einem Silo zwischengelagert werden. In beiden Fällen muß ein Optimierungsmodell daher die Zeiten der *Abfüll-Stutzen* planen, der in dieser Arbeit folgendermaßen zu definiert ist.

²⁰z.B. landesspezifische und sprachliche Kennzeichnungen oder Verkauf eines Produktes unter verschiedenen Markennamen.

Definition 2.2 Eine Abfüllanlage besteht aus einem oder mehreren Abfüll-Stutzen und den entsprechenden Abfüll-Linien. Ein Abfüll-Stutzen ist das Absperrorgan, an dem das Endprodukt in ein Gebinde oder einen TKW abgefüllt wird. Eine Abfüll-Linie bezeichnet die Verrohrung zwischen einem Reaktor und einem Abfüll-Stutzen.

Jede Abfüllanlage²¹ hat eine eigene (durchschnittliche) Abfüllgeschwindigkeit, aus der sich in Zusammenhang mit dem abzufüllenden Volumen die Dauer der Abfüllung ergibt. In der Produktionsplanung ist zu berücksichtigen, daß eine Abfüllanlage nur für einen der beiden Verpackungstypen zugelassen ist; die Abfüll-Stutzen befinden sich i.A. an unterschiedlichen Orten im Betrieb.

In der Volumenplanung werden die (Endprodukt-) Chargen aus den einzelnen Kundenaufträgen zusammengestellt. Dabei kann für eine Charge das (Teil-) Volumen eines einzelnen Auftrags genügen, oder es müssen mehrere Aufträge (mit evtl. unterschiedlichen Verpackungsbehältern) zu einer Charge zusammengestellt werden. Entsprechend muß bei der Planung der Abfüll-Stutzen auch das Volumen einer Charge in die einzelnen Kundenaufträge disaggregiert werden. Das produzierte Volumen einer Charge kann dann zeitgleich über die Abfüll-Stutzen der verschiedenen Verpackungstypen abgefüllt werden²².

Neben dieser Planung auf Basis vorliegender Kundenaufträge, kann auch eine Mischform zwischen auftragsorientierter Planung und Produktion auf Lager bestehen. Dabei kann die Entscheidung über Abfüllung oder Lagerung produktspezifisch oder sogar situativ getroffen werden. Die Lagerung der Endprodukte im Silo wird dann auch als *Abfüllung in FTL* (Fertigtanklager) bezeichnet. Da der Zeitpunkt der Auslagerung aus dem FTL bei der Durchführung des Planungsverfahrens nicht bekannt ist, muß ein Produktionsplaner bei einer späteren Auslagerung und Abfüllung die aktuelle Situation der Abfüllanlage berücksichtigen²³.

2.2.3 Ablaufplanung

In der Ablaufplanung der Chargenproduktion setzt sich die Gesamtdauer aus den zwei Zeitkomponenten der *produktbezogenen* und der *ablaufbezogenen Produktionsdauer* zusammen.

Produktionsdauer

Die Herstellung einer einzelnen Charge ist durch die verfahrenstechnischen Prozesse bestimmt. Die Abfolge der einzelnen Verfahrensschritte wird in Rezepturen festgelegt und bestimmt somit in der Summe der einzelnen Prozeßzeiten die Gesamtzeit der Herstellung. Die Dauer eines einzelnen Prozeßschritts ist wiederum von den Eigenschaften²⁴ der Roh-

²¹Eine Produktionsanlage kann mehrere Abfüllanlagen haben. Dabei werden die Abfüll-Stutzen des gleichen Behältertyps (Gebinde oder LKW) zu einer Abfüllanlage zusammengefasst.

²²Eine parallele Abfüllung über mehrere Abfüll-Stutzen *desselben Verpackungstyps* wird in dieser Arbeit nicht betrachtet.

²³Hierbei wird angenommen, daß ein FTL keine eigene Abfüllanlage hat. Ansonsten könnte wieder die zuvor beschriebene Trennung angewandt werden.

²⁴Wenn ein Rohstoff in einem Außen-Silo gelagert wird und die Rezeptur thermische Prozesse beinhaltet, so ist die Prozeßdauer abhängig von der Temperaturdifferenz zwischen dem Rohstoff und der Produktionstemperatur.

stoffe, dem Volumen und der Eintragung²⁵ im Produktionsprozeß abhängig. Auch muß bei der Berechnung der Prozeßzeiten die Leistung (z.B. beim Erhitzen) der einzelnen Anlagekomponenten berücksichtigt werden. Als weiterer Zeitfaktor müssen u.U. schließlich manuelle Eingriffe durch das Bedienpersonal beachtet werden.

In der Ablaufplanung werden jedoch i.A. nicht die einzelnen Prozeßzeiten berechnet. Die Produktionszeiten sind einem Produktionsplaner i.d.R. als Erfahrungswerte bekannt. Für die Produktionsdauer einer Charge wird dann entweder eine konstante Zeit festgesetzt oder ein linearer Zusammenhang in Abhängigkeit vom Volumen der Charge (und evtl. der Leistung des Reaktors) unterstellt²⁶.

Nach der Herstellung eines Produkts sind weitere Zeitpunkte für die nachfolgende Verarbeitung zu berücksichtigen. Durch die Eigenschaften des Produkts wird für die Verwendung ein Zeitfenster aus Liegezeit und Haltbarkeit definiert. Der nachfolgende Produktionsschritt bestimmt zudem den Zeitpunkt der Eintragung.

Wenn ein Produkt eine *Liegezeit* benötigt, ist die Produktion auf dem Reaktor selber beendet, aber die Herstellung des Produkts noch nicht abgeschlossen. Die Charge wird dann in dem herstellenden Reaktor oder häufiger an einem separaten Lagerplatz zwischengelagert. Eine Liegezeit ist z.B. durch Gärprozesse bedingt, oder ist notwendig, wenn ein Produkt nach einem Brennvorgang abkühlen muß.

Die *Haltbarkeit* eines Produkts gibt einen oberen Zeitpunkt für die weitere Verarbeitung an. Dieser Wert kann zwischen Null und unendlich liegen. Speziell bei einer **no-wait** (oder auch *zero-wait*) Haltbarkeit muß das Produkt also *just-in-time* produziert und sofort in dem nachfolgenden Produktionsschritt verarbeitet werden. Dies ist sowohl bei der Volumenplanung als auch bei der Ablaufplanung zu berücksichtigen. Das Produkt kann nicht in größeren Chargen auf Vorrat hergestellt werden, und der nachfolgende Produktionsschritt muß in der Ablaufplanung direkt nachfolgend eingeplant werden.

Sofern eine nur endliche Haltbarkeit gegeben ist, so ist diese i.d.R. immer noch länger als der Planungshorizont der kurzfristigen Ablaufplanung. Bei **finite-wait** oder **unlimited-wait** Haltbarkeit muß diese daher in der Ablaufplanung nicht weiter betrachtet werden.

Die *Eintragung* eines Rohstoffs oder Zwischenprodukts muß nicht sofort zu Beginn des Herstellungsprozesses einer Produktionsstufe geschehen. Dementsprechend kann die Herstellung eines Zwischenprodukts auch verspätet um diese *offset-Zeit* begonnen werden.

Produktionsablauf

Die ablaufbezogene Produktionsdauer umfasst alle Tätigkeiten, die nicht direkt mit der Herstellung eines Produkts verbunden sind, aber für die weitere Verarbeitung notwendig sind.

Aus dem innerbetrieblichen Transport resultiert hier die *Transferzeit* zwischen den einzelnen Produktionsstufen. Bei einer Verrohrung ist diese linear abhängig von dem Volumen der Charge und der Leistung der Pumpe. Bei Verwendung von Transportbehältern ist hier dagegen eine konstante Zeit anzusetzen.

²⁵Die *Eintragung* bezeichnet die Art und Weise und den Zeitpunkt, wie ein Rohstoff in der Produktion hinzugefügt wird. Dies kann z.B. in einer einzelnen oder mehreren Zugaben geschehen, um z.B. die Oberfläche des Rohstoff für die Reaktion zu variieren.

²⁶Vgl. ([Ans01], S. 33) und ([Brü02], S. 150)

Zwischen den einzelnen Produktionsprozessen eines Reaktors können einzelne ressourcenbezogene *Rüst-* oder *Reinigungsvorgänge* erforderlich werden. Ein Rüstvorgang wird *vor* dem Beginn einer Produktion notwendig, wenn eine Anlagekomponente für die nachfolgende Herstellung eingerichtet werden muß. Eine Reinigung wird durchgeführt, um eine Anlagekomponente *nach* Verwendung durch eine Charge wieder in den Urzustand zu versetzen.

Bei beiden Prozessen kann die Dauer abhängig *reihenfolgeabhängig* oder *reihenfolgeunabhängig* (von gefertigten Chargen) sein. Das Volumen der gefertigten bzw. zu fertigenden Chargen hat hier keinen Einfluß auf die Dauer der Reinigung, jedoch evtl. der Reaktor und die Kapazität des Reaktors.

Wenn die Produktionsanlage automatisiert durch einen Fertigungsleitstand gesteuert wird, dann ist die Anzahl der Mitarbeiter geringer als die Anzahl der Reaktoren. Ein Mitarbeiter überwacht mehrere Produktionsabläufe. Dementsprechend muß bei der Ablaufplanung der Einsatz der Mitarbeiter in den Rüst- und Reinigungsvorgängen als eigene Ressource berücksichtigt werden.

Die Berücksichtigung der Reinigungsvorgänge resultiert schließlich auch in der Anlagenfahrweise.

Anlagenfahrweise Bei der Anlagenfahrweise in der Chargenproduktion wird unterschieden zwischen der Kampagnenplanung und Mischfahrweise.

In der *Kampagnenplanung* werden über einen längeren Zeitraum auf einem Reaktor mehrere Chargen eines Produkts oder einer Produktgruppe hergestellt. Erst nach Ablauf der Kampagne werden die betroffenen Anlagekomponenten gereinigt und für die nächste Kampagne umgerüstet.

In der *Mischfahrweise* ist dagegen jederzeit ein Wechsel in der Herstellung eines Produkts möglich. Die Mischfahrweise kann daher auch als Kampagne der Länge 1 verstanden werden.

Die Anwendung, Vor- und Nachteile und der Einsatz von Kampagnenplanung und Mischfahrweise sind in Tabelle 2.3 zusammengestellt. Einen genaueren Überblick über die Kampagnenfahrweise geben PAPAGEORGIOU UND PANTELIDES [PP96]. Die Vor- und Nachteile bei der Mischfahrweise werden u.a. von ANSCHÜTZ ([Ans01], S. 28) aufgeführt.

	Kampagnenfahrweise	Mischfahrweise
Eigenschaften	zyklische Auflagenmuster	flexible Produktion
Anwendung	Mehrproduktanlage langfristige Planung	Mehrweckanlage kurzfristige Planung
Vorteile	Minimierung von Rüst- und Reinigungszeiten	niedrige Lagerbestände
Nachteile	hohe Lagerbestände	häufige Rüst- und Reinigungswechsel

Tabelle 2.3: Vergleich Kampagnenfahrweise vs. Mischfahrweise

2.3 Implementierung in Systemen zur Produktionsplanung

Die Planung und Durchführung der verfahrenstechnischen Produktion benötigt eine entsprechende informationstechnische Umsetzung. Dazu wird im folgenden die Rezeptur als Voraussetzung zur Mengen- und Reihenfolgeplanung beschrieben. Anschließend werden verschiedene IT-Systeme vorgestellt, die zur Produktionsplanung eingesetzt werden können. Die hierbei verwendbaren Möglichkeiten der Modellierung werden vertiefend in dem nachfolgenden Kapitel 3 beschrieben.

2.3.1 Abbildung der verfahrenstechnischen Prozesse durch Rezepte

In der Produktionsplanung werden die benötigten Materialien und Arbeitsschritte geplant. Der Bedarf an benötigten Materialien wird in der *Mengenplanung der fertigungstechnischen Produktion* anhand der Stücklisten ermittelt. Je nachdem, ob in der Mengenplanung der Bedarf strukturell nach *Primär-* und *Sekundärbedarf*²⁷ gegliedert wird, können hier unterschiedliche Arten von Stücklisten verwendet werden ([Vah00], S. 133). Die zeitliche Abfolge der Produktionsschritte erfolgt in der *Ablauf-* oder *Reihenfolgeplanung* aus den Arbeitsplänen ([Zäp82], S. 79).

Bei der verfahrenstechnischen Produktionsplanung werden dagegen *Rezepturen* verwendet, die den engen Zusammenhang zwischen Prozeßtechnologie und dem Produktionsprozeß in einer Kombination von Stückliste und Arbeitsplan darstellen ([Pre96], Sp. 1923). Ein Rezept beinhaltet dann die folgenden Informationen:

- Liste aller Einsatzstoffe (Rohstoffe und Zwischenprodukte)
- Beschreibung des erzeugten Produkts und weiterer Nebenprodukte
- Beschreibung der durchzuführenden Verfahrensschritte

Nach der NAMUR-Empfehlung NE 33 werden drei Generationen von Rezepten unterschieden ([NAM93], S. 9). Das *Urrezept* wird in der Entwicklungsphase eines Produkts erstellt. Es definiert anlagenneutral das Ziel und den Aufbau eines Herstellungsprozesses. Durch Projektierung auf eine Anlage wird aus dem Urrezept das *Grundrezept* entwickelt, das den Maßstäben der Anlage gerecht wird. Für den einzelnen Produktionsauftrag einer Charge wird das *Steuerrezept* erstellt, das zusätzlich die Zeitpunkte der verwendeten Anlagenteile angibt. Eine Übersicht über die Rezeptgeneration nach NE 33 wird in Abb. 2.7 gegeben. Die Umsetzung der Empfehlung wird in dem NAMUR-Arbeitsblatt NA 46 [NAM94] beschrieben.

Diese Definition der Rezeptgenerationen stellt ein Problem für die Ablaufplanung dar, weil das Grundrezept, obwohl nicht mehr anlagenneutral, noch keinen Bezug auf die Teilanlagen angibt ([NAM93], S. 11). Dagegen ist ein Steuerrezept bereits zu spezifisch, da es sich nur noch auf eine einzelne Charge bezieht ([Ans01], S. 35).

²⁷Der Primärbedarf ergibt sich aus den Erzeugnismengen des geplanten Produktionsprogramms. Der Sekundärbedarf umfaßt alle Rohstoffe und Bauteile, die für den Primärbedarf benötigt werden ([Cor98], S. 416).

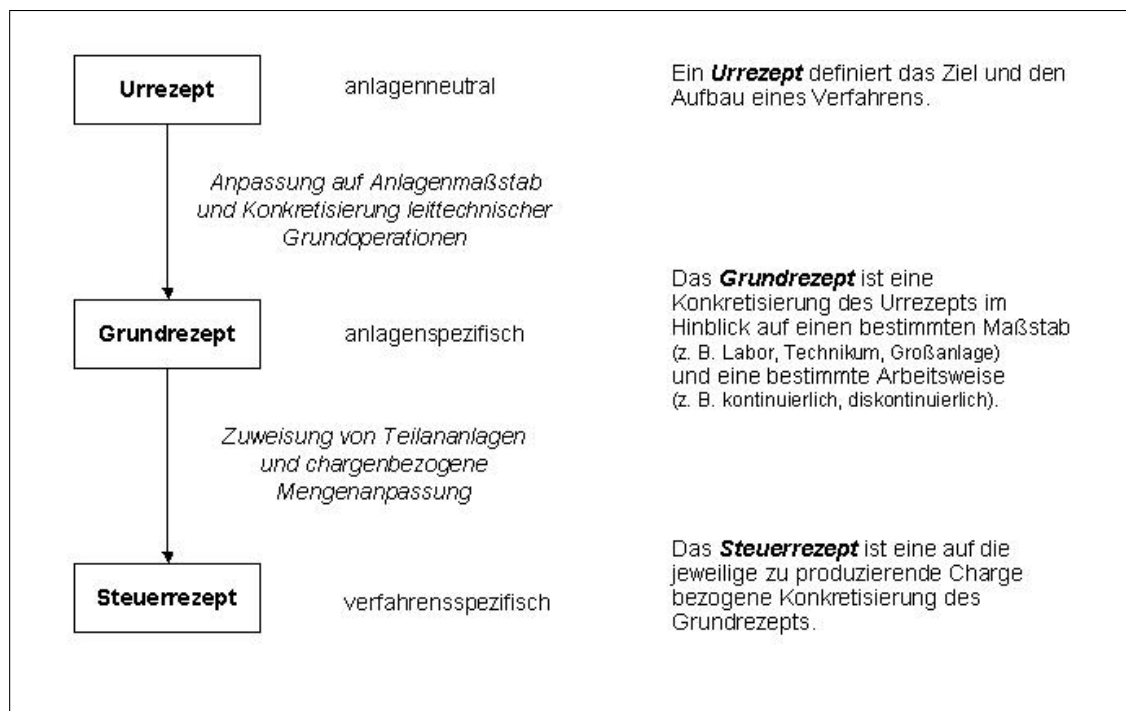


Abbildung 2.7: Rezept-Entwicklung nach NAMUR Empfehlung NE 33

2.3.2 IT-Systeme zur Produktionsplanung und -steuerung

Die Aufgaben der Produktionsplanung können aufgrund der historischen Entwicklung in drei unterschiedlichen IT-Systemen durchgeführt werden. Die betriebliche Datenhaltung für Finanzen und Controlling oder Personalmanagement wird in *Enterprise Resource Planning* (ERP) Systemen durchgeführt. Solche Systeme beinhalten auch eigene Module zur Produktionsplanung und -steuerung (PPS). Da diese jedoch nur eine sukzessive Planung für einen Produktionsbetrieb durchführen, wurden die *Advanced Planning Systeme* (APS)²⁸ für die betriebsübergreifende Planung in der Supply Chain entwickelt. Aus der Überwachung der Produktionsprozesse haben sich schließlich in den vergangenen Jahren die innerbetrieblich eingesetzten *Manufacturing Execution Systeme* (MES) entwickelt, bei denen Leitstände mit Funktionalitäten der Produktionsplanung erweitert wurden.

Produktionsplanung in ERP-Systemen: Die Produktionsplanung in ERP-Systemen²⁹ ist nach dem Konzept des *Manufacturing Resource Planning* (MRP II) aufgebaut ([Gro04], S.96). Dieses verfolgt einen sukzessiven Lösungsansatz (vgl. Abb. 2.8), bei dem die Ergebnisse einer jeweils höhergelegenen Planungsebene Planungsvorgaben für die darunter liegenden Ebenen sind.

In der obersten Ebene der *Programmplanung*³⁰ wird das zu produzierende Produktspektrum für einen festen Planungszeitraum aufgrund von Absatzprognosen oder vorliegenden Kundenaufträgen bestimmt ([Cor98], S. 494). Aus diesen Auftragsmengen wird der Pri-

²⁸Die Abkürzung APS wird in dem gleichen Zusammenhang auch für *Advanced Planning and Scheduling* verwendet.

²⁹ERP = Enterprise Resource Planning

³⁰In der englischsprachigen Literatur wird hier auch von dem *Master Production Schedule* (MPS) gesprochen.

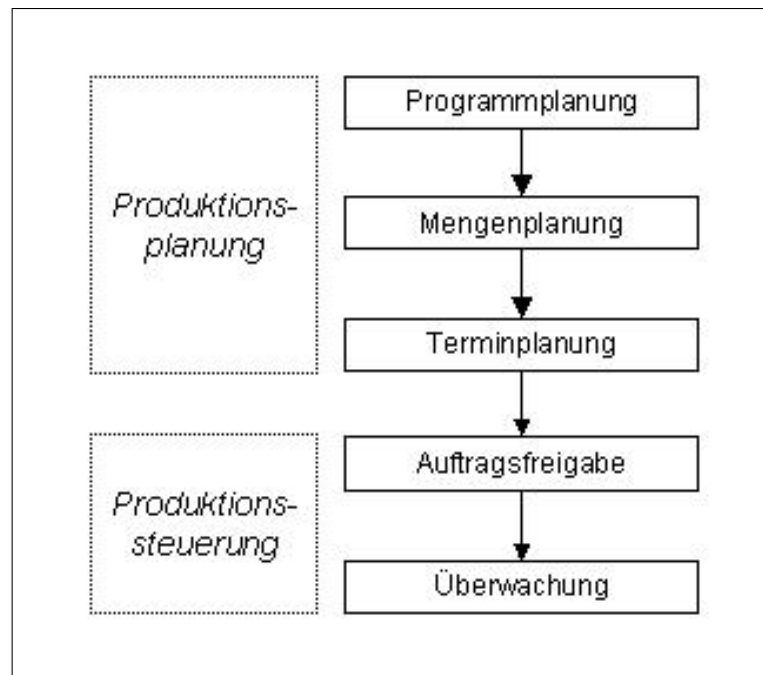


Abbildung 2.8: Produktionsplanung und -steuerung

märbedarf abgeleitet. Anschließend wird in der *Mengenplanung* (Material Requirement Planning; MRP) der Bedarf für die einzelnen Produktionsstufen aufgelöst. Erst wenn dadurch ein ausreichend großer Bestand für die weitere Produktion nachgewiesen wird, werden in der *Terminplanung* die Verfügbarkeiten der Produktionskapazitäten geprüft und die Termine festgelegt. Diese beiden Schritte der Terminplanung können sukzessiv ([Cor98], S. 494 ff.) oder simultan erfolgen ([GT04], S. 306).

Die eigentliche Produktionssteuerung wird dann mit Fertigungsleitständen und Systemen zur Betriebsdatenerfassung durchgeführt. Wichtige Funktionen nach der Auftragsfreigabe sind hier die automatische Fortschreibung der Materialentnahmedaten zum Soll-Ist-Vergleich und die Rückmeldung von Fertigungsdaten ([Gro04], S. 108).

Produktionsplanung in Advanced Planning Systemen: Die Schwäche vieler PPS-Systeme liegt genau in der zuvor beschriebenen Planungshierarchie. Mit der fortschreitenden Entwicklung der Rechnerkapazitäten zur Datenhaltung und Rechengeschwindigkeit konnten simultane Planungsalgorithmen entwickelt werden, die die Engpässe in der gesamten Lieferkette berücksichtigen ([Gro04], S. 96). Die Daten werden dabei dem unterlagerten ERP-System entnommen. Wie zuvor ist die Optimierung in einzelne Module unterteilt (vgl. Abb. 2.9; [Tem01]). Die Aufgaben der Produktionsplanung werden dabei in verschiedenen Software-Systemen durch ein oder zwei Module abgedeckt [MRW00]. Durch eine *Available-to-Promise* (ATP) Planung kann dem Kunden eine schnelle Rückmeldung über die Auftragsannahme gemacht werden. Während ATP dabei nur den Lagerbestand fortschreibt, werden stattdessen bei einer Auftragsannahme durch ein *Capable-to-Promise* (CTP) Modul auch die freien Produktionskapazitäten in der Planung berücksichtigt. Für die Beschreibung der weiteren APS-Module sei auf die entsprechende Literatur (z.B. [GT04], 330 ff.) verwiesen.

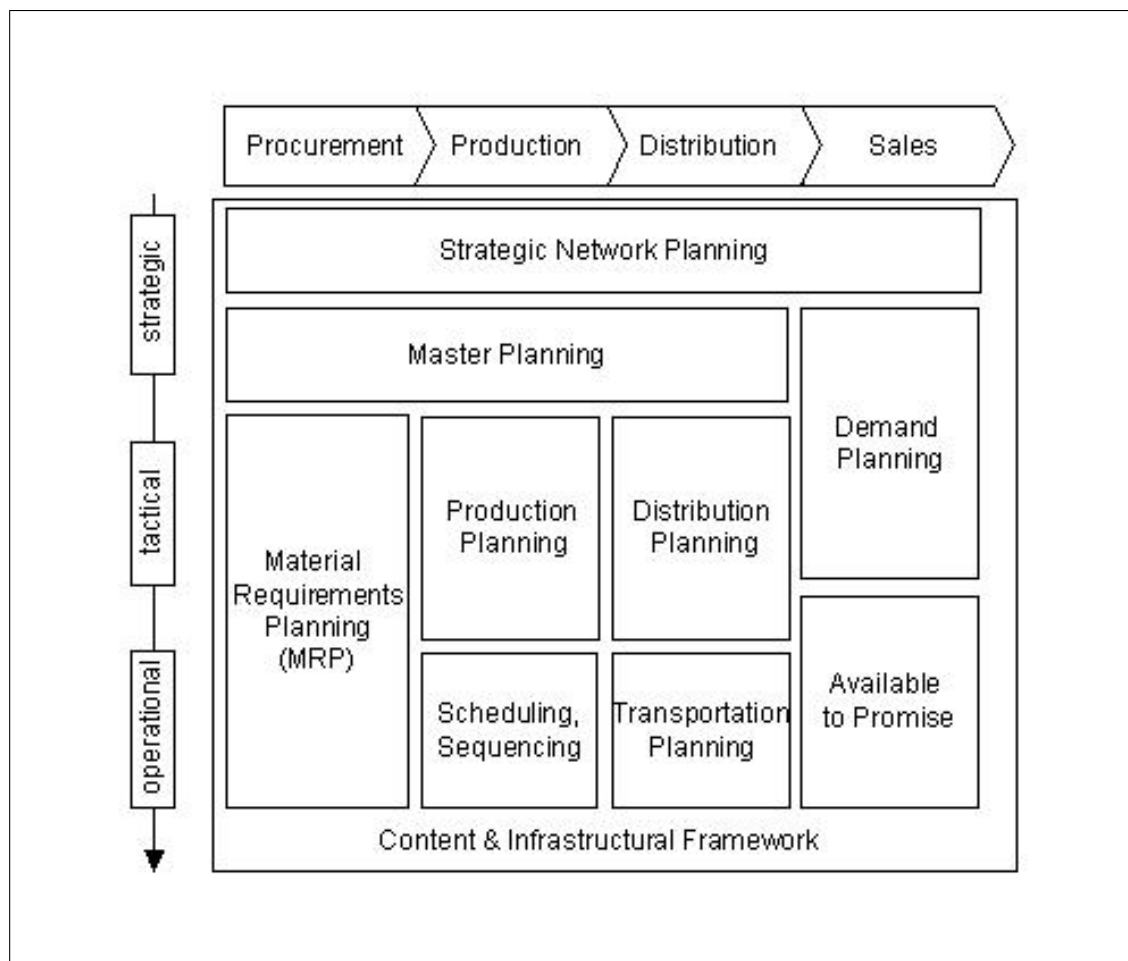


Abbildung 2.9: Planung in Advanced Planning Systemen

Die Schwierigkeiten von Advanced Planning Systemen beschreibt TEMPELMEIER [Tem01]. So bemängelt er, daß in den allgemein vorherrschenden IT-Strukturen die *Mengenplanung* der Beschaffungsseite zugeordnet wird, obwohl dies eine klassische Aufgabe der Produktionsplanung ist ³¹. Ebenso wird laut TEMPELMEIER die Losgrößenplanung nur auf der Ebene des *Master Planning* durchgeführt. Die unteren Planungsebenen werden dabei erneut nur durch zusätzliche Setup-Kosten berücksichtigt.

Produktionsplanung in Manufacturing Execution Systemen: Die Kernidee von Manufacturing Execution Systemen ist die Bereitstellung einer reaktionsschnellen Informationsverarbeitung [WMK05]. Einige Anbieter entwickeln hier zusätzliche Module zur Personal- und Ressourcenplanung, andere beschränken sich auf Leistungsanalysen und Datensammlung [Lin05]. ME Systeme umfassen damit sowohl Planungsaufgaben der klassischen PPS- und APS-Systeme, wie auch Aufgaben der Leitstände und von Systemen zur *Betriebsdaten-* (BDE), *Maschinendaten-* (MDE) und *Arbeitszeiterfassung* (AZE). H.-H. WIENDAHL ET AL. haben hier drei, historisch unterschiedlich entwickelte Ausprägungen

³¹In dem System Advanced Planner and Optimizer (APO) der SAP AG werden bspw. nur die relevanten Produkte, d.h. die Endprodukte und einige ihrer kritischen Komponenten geplant. Die komplette Ermittlung und Planung der Sekundärbedarfsmengen wird dann in dem ERP-System R/3 durchgeführt [RS00].

von ME Systemen analysiert [WMK05]:

- Ehemals als Leitstände eigenständig entwickelte Softwarelösungen zur Werkstatt- oder Fertigungssteuerung, die an übergeordnete ERP-Systeme angekoppelt werden oder eigenständig betrieben werden,
- Erweiterungen von ERP-Systemen mit Funktionalitäten der Werkstattsteuerung zur verbesserten Feinplanung *und*
- Systeme der Betriebsdatenerfassung mit speziellen Analysefähigkeiten, die um planerische Funktionalitäten erweitert werden.

ME Systeme werden also als Bindeglied zwischen einem betrieblichen ERP-System und der Fertigung eingesetzt [KW03]. Sie bieten dadurch insgesamt eine höhere Transparenz über die Prozesse, da sie die Daten in Echtzeit zur Verfügung stellen, die zur Produktionsplanung entlang der Supply Chain benötigt werden [Sth05]. Zudem können durch Analysen Stillstandszeiten in der Fertigung bestimmt und somit Anlagen und Arbeitsgruppen optimiert werden [MG05]. Einen Überblick über die Funktionalitäten und Optimierungsverfahren in Manufacturing Execution Systemen geben LINDEMANN UND SCHMID [LS05].

Einsatz, Anforderungen und Schwachstellen bestehender Systeme

Bei der Auswahl eines ERP-Systems für einen Einsatz in der Prozeß- oder Ernährungsindustrie müssen vor allem die *Chargenverfolgung*, die *Verwaltung von Mindesthaltbarkeitsdaten*, ein *Qualitätsmanagement* und eine *mengenmäßige Bestandsführung* gewährleistet werden. Nach einer Studie von WEIDENHAUN UND NIEHSEN decken diese Anforderungen jedoch weniger als 50% aller Systeme³² ab [WN03].

Völlig ungeeignet waren die MRP-Konzepte schon seit langem im Bereich der Prozeßindustrie, wo der Fokus der Planung in der optimalen Organisation und Steuerung einfacher Artikel über komplexe und verzweigte Fertigungs- und Lagerstrukturen hinweg liegt ([Gro04], S. 222). Die SAP AG hat daher für den Einsatz des ERP-Systems R/3 in der Prozeßindustrie das Modul PP/PI³³ entwickelt [BNS98]. In diesem wird ein zusätzlicher Rezepttyp, das *Planungsrezept* eingeführt und verwendet, um die Problematik der fehlenden Ressourcenzuordnung eines Grundrezeptes³⁴ zu beseitigen ([Loo99], S. 395).

Für die bestehenden PPS-Systeme (1999) hat die NAMUR in ihrem Arbeitsblatt NA 85 exemplarisch einige Schwachstellen in der Feinplanung aufgelistet [NAM99]. Diese lassen sich im wesentlichen in fünf Kategorien von Problemfällen³⁵ zusammenfassen:

- *Fehlende Abstimmung zwischen Grobplanung und Feinplanung*: PPS Systeme untergliedern in Grob- und Feinplanung. Viele Systeme arbeiten in der Grobplanung nur nach der MRP-Methode, die lediglich den Materialbedarf berechnet. Die Anlagenkapazitäten werden nicht berücksichtigt bzw. mit einer unendlichen Kapazität versehen ([NAM99], Beispiel 2 *Berücksichtigung der Anlagenkapazitäten in der Grobplanung*).

³²Untersucht wurden 26 ERP-Systeme.

³³PP/PI = Produktionsplanung Prozeßindustrie

³⁴Zur Beschreibung des Grundrezeptes siehe S. 32 und [NAM93].

³⁵Insgesamt werden zehn Beispiele aufgeführt, die aber teilweise mehrere Problemfälle gleichzeitig beinhalten.

- *Bereitstellung von Materialien:* Bei Aufträgen in der kontinuierlichen Produktion (oder bei einer anschließenden Verpackung, die länger andauert) muß die Gesamtmenge der Zutaten (oder der Verpackungsmaterialien) bereits zu Produktionsbeginn zur Verfügung stehen. Dies ist bei einer kontinuierlichen Produktion (oder Abfüllung) nicht notwendig und führt zu unnötig höheren Lagerbeständen ([NAM99], Beispiel 1 *kontinuierliche Materialversorgung*).
- *Fehlende Kopplung zwischen den Anlagen eines Anlagenverbunds:* In einem Produktionsverbund von zwei oder mehr Anlagen ohne große Zwischenlager wirken sich Störungen verstärkt aus. Die Feinplanungssysteme aller Anlagen müssen dies berücksichtigen ([NAM99], Beispiel 5 *Synchronisation von Betrieben*).
- *Fehlende Kopplung zwischen Produktion und Abfüllung oder anderen nachfolgenden (vorgelagerten) Prozessen:* Wenn zu einem Produktionsauftrag mehrere Abfüllaufträge (ohne Zwischenlager) erzeugt werden, müssen diese sowohl zeitlich, wie auch in Bezug auf die produzierende Ressource mit dem Produktionsauftrag verknüpft werden ([NAM99], Beispiel 4 *Verknüpfung von Produktion und Abfüllung*).
- *Simulation der Auslastung bereits im Design einer Produktionsanlage:* Bei dem Design einer Anlage, die die Produktion in einem Verbund mit anderen Anlagen ohne große Zwischenlager durchführt, muß bereits dem Anlagenplaner ein Simulationswerkzeug zur Verfügung stehen, um die Auslastung und Kapazität der Anlage richtig zu planen ([NAM99], Beispiel 3 *Design einer Anlage, eines Anlagenverbunds*).

Viele Probleme und technologisch unzulässige Lösungen resultieren also aus den Mängeln im Grobplanungsbereich, bei dem Vorgänge oder Maschinen lediglich kapazitativ berücksichtigt werden³⁶.

Entsprechend häufig wird die Feinplanung von einem Produktionsplaner interaktiv anhand des GANTT-Diagramms³⁷ eines Leitstands durchgeführt. Hierfür haben kleinere Softwarefirmen³⁸ zusätzliche Visualisierungen geschaffen, die zugleich den Materialbestand und die Ressourcenauslastung anzeigen [Jän03]. So hat OR Soft das *Add-on*³⁹ *Schedule++* zu SAP R/3 entwickelt, in dem durch einen Backtrackalgorithmus⁴⁰ zusätzlich die Möglichkeit geschaffen wird, nicht nur die aktuelle Planungssituation anzuzeigen, sondern eine technologisch zulässige Lösung zu erzeugen. Durch eine mehrfache Anwendung dieses Verfahrens kann dann eine simultane Material- und Ressourcenplanung durchgeführt werden [MH+01].

Wenn solche Add-ons nicht bestehen, werden von den Planern alternativ sogar Lösungen mit Tabellenkalkulationsprogrammen erzeugt⁴¹, da die Ergebnisse der Planung die Ist-Situation nicht widerspiegeln und daher nicht akzeptiert werden [B-H03]. Dies ist wahr-

³⁶Bei einer kapazitativen Zusammenfassung mehrerer Maschinen wird auch von einer Poolressource gesprochen [JR02].

³⁷Ein GANTT-Diagramm ist ein Balkendiagramm, das von dem Unternehmensberater Henry L. Gantt (* 1861, † 1919) als Werkzeug zum Projektmanagement entwickelt wurde [Bro05]. Ein solches Diagramm kann maschinen- oder auftragsorientiert sein. Diese Orientierung resultiert aus dem Kennzeichen, das auf der Ordinatenachse (y-Achse) aufgetragen wird. Die Abhängigkeiten zwischen den einzelnen Operationen über die Ordinaten hinweg wird zusätzlich durch Pfeile dargestellt.

³⁸Eine ausführliche Übersicht ist bei LINDEMANN UND SCHMID zu finden [LS05].

³⁹Ein Add-on ist ein Paket von Modulen, das die Funktionalität einer bestehenden Software erweitert.

⁴⁰Ein Backtrack-Algorithmus ist ein Lösungsverfahren, bei dem die Gesamtlösung schrittweise aus der Lösung von Teilproblemen ermittelt wird.

⁴¹Wenn bei einer solchen Planung nur Wert auf die Minimierung der Rüstzeiten gelegt wird, werden

scheinlich auch der Grund, warum in vielen Unternehmen weiterhin auf Lager produziert wird. 50% der Lagerprodukte haben allerdings dennoch Lieferzeiten von einigen Tagen oder länger [AF02b].

Nach einer Studie der **initions AG** (2002) werden in der Prozeßindustrie nur ERP-Systeme zur Planung eingesetzt; APS Systeme werden nicht verwendet. Eine vollständige und unternehmensübergreifende Planung im Sinne des Supply Chain Managments besteht größtenteils noch nicht ([Fri03], S. 159). Die Beratungsfirma **A.T. Kearney** hat in Projekten und Studien festgestellt, daß hier durch eine Vernetzung der Standorte die Fertigungskomplexität einzelner Standorte um 30-35% gesenkt werden kann. Dabei wurden teilweise nur Gesamtauslastungen von 35% in der Galenik⁴² von Pharmafirmen festgestellt [HBV02]. Durch simulationsbasierte Feinplanung kann die Durchlaufzeit um bis zu 30% gesenkt werden [KNS00] und mit optimalen Beständen Einsparungen von bis zu 20% erreicht werden [HBV02].

Nach **RICHTER UND STOCKRAHM** wurde das Modul **PP/DS**⁴³ von **SAP APO** in der Prozeßindustrie erstmals erfolgreich bei der Produktionsplanung von synthetischen Granulaten eingesetzt [RS00]. Hierbei konnte das Ergebnis der Planung im Sinne des **Makespan** vor allem durch Nutzung der zuvor verwendeten Pufferzeiten verbessert werden. Zusätzlich stieg vor allem die Flexibilität der Planungsprozesse. Die Planung wird innerhalb einer Stunde von nur noch einem Planer durchgeführt. Die Reihenfolge kann daher noch eine Stunde vor dem Produktionsstart verändert werden [Sto02].

oftmals gleiche Aufträge zusammengelegt. Dies führt dazu, daß Aufträge mit einer Produktionsdauer von wenigen Tagen bei mehreren Produktionsstufen eine Durchlaufzeit von mehreren Wochen benötigen [KNS00].

⁴²Pharmazeutische Technologie

⁴³Produktionsplanung Detailed Scheduling

Kapitel 3

Modelle und Verfahren zur Produktionsplanung

In dieser Arbeit werden innerhalb der Produktionsplanung die beiden Ebenen der Losgrößen- und Reihenfolgeplanung betrachtet. Für jede dieser Planungsaufgaben kann ein eigenes Optimierungsmodell aufgestellt werden. Zwischen diesen beiden Modellen besteht jedoch eine zeitliche Abhängigkeit. Bei der Planung der Produktionsaufträge wird nicht die Produktionsabfolge festgelegt. Entsprechend können reihenfolgeabhängige Rüst- und Reinigungszeiten nicht bei der Berechnung der genutzten Zeit einer Planungsperiode berücksichtigt werden. Die Kombination beider Planungsaufgaben wird später in dem Kapitel 5 behandelt.

Vorausbedingung für eine Reihenfolgeplanung ist die Erstellung der Produktionschargen. Zur Unterscheidung zwischen einer Losgrößenplanung und der Erstellung der Produktionschargen werden in Kapitel 3.1 daher zunächst die Anforderungen und Eigenschaften der *Volumenplanung* beschrieben. Anschließend werden verschiedene Möglichkeiten der *Modellierung von Reihenfolgeproblemen* vorgestellt (Kapitel 3.2). Zur Lösung eines solchen Problems werden anschließend einige *Optimierungsverfahren* beschrieben (Kapitel 3.3). In Abschnitt 3.4 werden abschließend einige aus der Literatur bekannte Modelle vorgestellt.

3.1 Volumenplanung

In Kapitel 1.1.1 wurde festgestellt, daß eine Losgrößenplanung Vorbedingung für die Reihenfolgeoptimierung ist. DOMSCHKE ET AL. definieren ein Los als die Zusammenfassung gleichartiger Objekte, die auf einem Arbeitsträger ohne weitere Rüstvorgänge hintereinander gefertigt werden ([DSV93], S. 63). Diese Definition ist jedoch nicht mit der im vorigen Kapitel gegebenen Beschreibung einer Charge¹ (s. S. 20) vergleichbar. Für die

¹STADTLER listet die Chargenfertigung in einer Übersicht von Modellen zur Losgrößenplanung auf. Er bezeichnet sie dabei als *gemeinsame, gleichzeitige Fertigung mehrerer Lose auf einer Ressource* [Sta01]. Hierbei ist anzumerken, daß eine solche Definition nur bei Betrachtung von Losen *desselben* Produktes zutreffend ist. Im Sinne einer Losgrößenplanung kann hier die *Kampagnenplanung* verstanden werden, bei der mehrere Chargen nacheinander auf derselben Teilanlage gefertigt werden. Die Anzahl der Chargen in einer Kampagne bestimmt dann die Losgröße. Eine Übersicht über Lösungsansätze zur Kampagnenplanung gibt BLÖMER ([Blö99], S. 61 ff.).

Ablaufplanung bei der Chargenproduktion müssen dennoch in einer vorhergehenden oder simultan stattfindenden Planung die Anzahl und das Volumen der Chargen bestimmt werden. KONDILI ET AL. bezeichnen diesen vorhergehenden Schritt als *Sequencing* [KPS93], bei TRAUTMANN wird er *Batching* genannt [Tra01]. In dieser Arbeit wird der vorgeschaltete Planungsschritt als *Volumenplanung der Chargen* oder *Chargenplanung*² bezeichnet. Die bei der Volumenplanung zu berücksichtigenden Aspekte für ein Modell zur Volumenplanung werden nachfolgend näher betrachtet.

Verknüpfung mit der Reihenfolgeplanung: Im Prozeß der Produktionsplanung können die beiden Aufgaben der Volumenplanung und der Reihenfolgeplanung auf zwei unterschiedliche Arten miteinander verknüpft werden.

Bei einer *hierarchischen Planung* wird das Gesamtproblem in verschiedene Teilprobleme (Ebenen) zerlegt, die wechselseitig voneinander abhängen³. Wesentlich ist hierbei, welche Kopplung zwischen den beiden Ebenen stattfindet. In einer streng hierarchischen Planung werden die Planungsaufgaben *sukzessive* durchgeführt. Durch sog. *Feedback*-Funktionen können die Ergebnisse einer unteren Ebene zur Revision der jeweils höherliegenden Planungsebene herangezogen werden [Sta00b].

Bei einer *simultanen Planung* werden alle Entscheidungen in einem sog. *monolithischen Modell* getroffen. Bei einer solchen Vorgehensweise sind dann keine wiederholten Abstimmungsvorgänge wie bei einer hierarchischen Planung notwendig ([Zäp82], S. 298 ff.). Allerdings sind solche Planungsmodelle aufgrund ihrer Größe ungeeignet, um in kurzer Zeit ein optimales oder zumindest akzeptables Ergebnis zu erreichen.

Zielfunktion: Modelle zur Losgrößenplanung werden im allgemeinen unter dem Kriterium der *Kostenminimierung* oder der *Durchlaufzeitminimierung* betrachtet ([Vah00], S. 165). Daher soll hier diskutiert werden, welches dieser Kriterien auf die Volumenplanung übertragen werden kann.

Bei einer *Minimierung der Kosten* werden i.A. die Produktions-, Lager- und Rüstkosten betrachtet. Zu den Rüstkosten können auch die Reinigungskosten gezählt werden. Sie unterscheiden sich lediglich durch den Zeitpunkt der bewerteten Aktion. Eine Produktionsanlage wird *vor* Beginn der Produktion gerüstet und *nach* der Produktion gereinigt. Jeder der genannten Kostenblöcke ist aus einem festem und einem (i.d.R.) linear proportionalen Anteil zusammengesetzt. Weitere Kostenblöcke, die in der Planung berücksichtigt werden können, sind *Leerkosten* für den Stillstand eines Reaktors und *Vertragsstrafen* im Falle der Nicht-Einhaltung von Lieferterminen.

Sofern die Produktionskosten im Verlauf des Planungshorizontes *nicht* variieren⁴, sind sie für die weitere Entscheidungsfindung irrelevant ([Der95], S. 22 ff.). Hier ist weiterhin zu prüfen, ob eine Abhängigkeit von dem Produktionsvolumen oder dem genutzten Reaktor besteht. Unter der Annahme, daß das gesamte Bedarfsvolumen auch hergestellt werden soll, sind volumenabhängige Produktionskosten daher nicht entscheidungsrelevant.

²Die Volumenplanung kann also mit der Bestimmung der einzelnen Fertigungsaufträge gleichgesetzt werden.

³Diese Interdependenzen in der Produktionsplanung sind entweder sachlicher oder zeitlicher Art. Sie werden u.a. von ZÄPFEL näher beschrieben ([Zäp82], S. 290 ff.).

⁴Die Produktionskosten können z.B. ansteigen, wenn zusätzlich an Wochenenden und Feiertagen produziert wird und damit Zuschläge für die Mitarbeiter berücksichtigt werden müssen.

Reaktorabhängige Produktionskosten können dagegen bei der Volumenplanung nicht berücksichtigt werden, da diese Zuordnung erst in der Reihenfolgeplanung erfolgt. Wichtig ist dagegen die Entscheidung, ob eine weitere Charge erstellt wird. Dadurch entstehen zusätzliche Fixkosten. Die Minimierung der insgesamt entstehenden Produktionskosten ist damit mit der Minimierung der Gesamtzahl der Chargen gleichzusetzen.

Auch die weiteren Kostenblöcke sind aufgrund der engen Verknüpfung zur Reihenfolgeplanung nur schwer quantifizierbar. So ist zu bestimmen, welche Lagerhaltungskosten entstehen, wenn ein Zwischenprodukt bis zur Verarbeitung im nächsten Produktionsschritt kurzzeitig in einem Silo zwischengelagert wird, oder welche Opportunitätskosten berücksichtigt werden müssen, wenn es stattdessen in einem Reaktor zwischengelagert wird. In letzterem Falle wäre der Reaktor für die weitere Nutzung durch eine andere Produktion blockiert.

Eine Betrachtung der Kostenminimierung als Optimierungsziel der Volumenplanung erscheint nicht möglich, da die einzelnen Kostenfaktoren in der verfahrenstechnischen Planung im allgemeinen schwer quantifizierbar sind. Lediglich BRUCKER UND HURINK minimieren die Gesamtkosten für die Produktion aller Chargen⁵ (ohne Rüst-, Reinigungs- und Lagerhaltungskosten) [BH00]. Sie berücksichtigen dabei die Einhaltung unterschiedlicher Liefertermine⁶, wodurch die Notwendigkeit entsteht, weitere Chargen zu erstellen. Sonst würde ihr Ergebnis der (minimalen) Anzahl an Chargen n_{chg} entsprechen. Dieser Wert wird aus dem Bedarfsvolumen V_{bed} und dem maximalen Chargenvolumen \bar{V}_{chg} als $n_{chg} := \lceil V_{bed} / \bar{V}_{chg} \rceil$ berechnet.

Die Werte der zeitlichen Nutzung sind dagegen bereits im Rahmen der Volumenplanung bekannt. Diese werden auch in der (bei einem hierarchischen Planungsansatz anschließenden) Reihenfolgeplanung verwendet. Das Optimierungskriterium der Volumenplanung kann daher so gewählt werden, daß in Verbindung mit der Reihenfolgeplanung eine möglichst gleichmäßige *Auslastung der Produktionsanlage* erreicht wird ([Tra05], S. 31 f).

Technische Restriktionen: Bei der Planung des eigentlichen Volumens einer Charge müssen die technischen Restriktionen der Produktionsanlage berücksichtigt werden. Die Problematik der Volumenbestimmung ist in der Prozeßindustrie besonders zu beachten, wenn eine kundenauftragsbezogene Planung durchgeführt wird, da die Volumina der Kundenaufträge meistens nicht dem (optimalen Maximal-) Volumen einer Charge entsprechen ([BNS98], S. 229). Sofern für eine Charge also nicht eine feste Größe⁷ vorgegeben ist, sind eine Unter- wie auch eine Obergrenze für das Volumen zu berücksichtigen. Hier ist zu prüfen, ob Produktionsanlagen mit unterschiedlichen Volumengrenzwerten zur Verfügung stehen.

TRAUTMANN modelliert das Batching Problem anhand eines nichtlinearen Optimierungsproblems [Tra01]. Die Zuordnung der Chargen zu den Produktionsanlagen erfolgt dabei erst in der Reihenfolgeplanung. Dies führt dazu, daß das Maximalvolumen der Chargen immer so gewählt wird, daß die Herstellung auf *allen* Produktionsanlagen durchgeführt werden kann. Erst bei weiteren Untersuchungen hat TRAUTMANN das Modell als lineares Optimierungsproblem beschrieben [Tra05]. Bei dieser Weiterentwicklung werden dann

⁵Hierzu verwenden sie Testdaten aus einer Projektpartnerschaft mit der Bayer AG.

⁶Anstelle dieser Nebenbedingung könnten auch zusätzliche *Kosten für eine vorgezogene Produktion* oder *Verspätungskosten* definiert werden [DKV99].

⁷Dies ist z.B. bei einem der ersten Modelle von EGLI UND RIPPIN [ER86] der Fall; vgl. [KPS93].

die unterschiedlichen Volumina der Produktionsanlagen bereits in dem Batching Problem berücksichtigt.

ANSCHÜTZ weist darauf hin, daß im Falle eines variablen Chargenvolumens bei der Erstellung des Optimierungsmodells auch beachtet werden muß, ob die Herstellungsdauer volumenabhängig ist ([Ans01], S. 33). Hier kann dann eine feste Zeit oder eine (linear) volumenabhängige Dauer für die Produktion benötigt werden. Neben der Volumenunterscheidung muß aber zudem berücksichtigt werden, daß auch die Leistung der einzelnen Komponenten (z.B. Rührwerk, Heizspirale) divergieren kann. Die Produktionsdauer muß dann nicht nur volumenabhängig, sondern auch leistungsabhängig definiert werden.

3.2 Modellierung von Reihenfolgeproblemen

Wie aus der Beschreibung der Anforderungen der verfahrenstechnischen Produktion im vorigen Kapitel ersichtlich wird, müssen in der Reihenfolgeoptimierung meistens mehrstufige Produktionsprozesse geplant werden. Bei der Modellierung müssen dabei zusätzlich zu den Charakteristika der Reihenfolgeplanung aus Kapitel 1.2.2 weitere Aspekte beachtet werden.

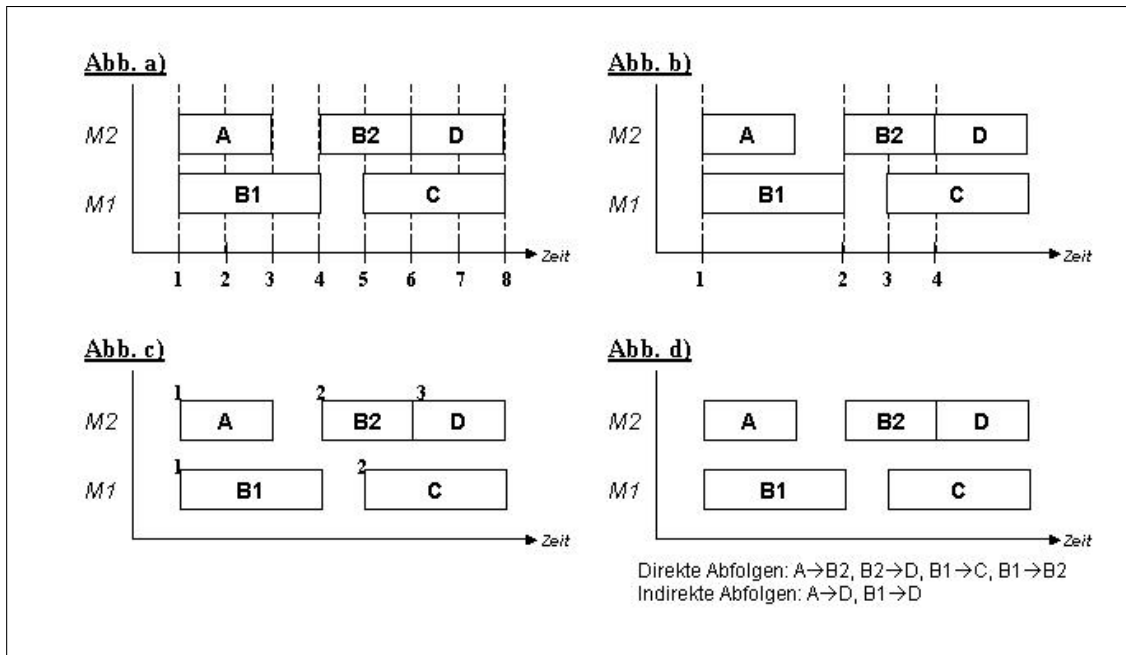
Modellierung des Zeitverlaufs: Ein wichtiger Aspekt bei der Modellierung von Reihenfolgeproblemen ist die Frage, wie die Zeitpunkte innerhalb der Planung repräsentiert werden [MC+05]. Die Entscheidung über diese Darstellung bestimmt dabei mit über den Umfang eines Optimierungsmodells in Bezug auf die Anzahl der Variablen und Nebenbedingungen.

Bei der Modellierung des Zeitverlaufs ist zu unterscheiden zwischen Modellen, bei denen nur *diskrete Zeitpunkte* im Horizont betrachtet werden⁸, und solchen mit einer *kontinuierlichen Betrachtung* des gesamten Zeitverlaufs. Bei einer Betrachtung diskreter Zeitpunkte wird dabei vorausgesetzt, daß die Prozesse jeweils nur zu diesen Zeitpunkten beginnen. Entsprechend müssen nur diese Zeitpunkte in der Planung berücksichtigt werden. Die Zeitpunkte können dann in globalen, äquidistanten Zeitintervallen festgelegt werden (vgl. Abb. 3.1 a) oder globale, fest definierte Punkte sein (vgl. Abb. 3.1 b). Alternativ dazu können diese Zeitpunkte auch anlagenspezifisch festgelegt werden. Zur Implementierung einer solchen Modellierung muß zuvor die Zeitdauer jedes einzelnen Prozesses bekannt sein, damit die Zeitpunkte bzw. das Zeitraster passend bestimmt werden können.

Bei einer kontinuierlichen Betrachtung des Zeitverlaufs ist die Zeitdauer der einzelnen Prozesse dagegen irrelevant. Für jeden Prozeß muß entsprechend eine Variable definiert werden, die den Startzeitpunkt des jeweiligen Prozesses darstellt. Hierbei kann die Reihenfolge dann entweder in Form einer numerischen Zuordnung (vgl. Abb. 3.1 c) oder in einer Bestimmung der Abfolgen der Prozesse (vgl. Abb. 3.1 d) dargestellt werden. Für eine solche Repräsentation sind *nach* Bestimmung der Reihenfolge daher zusätzlich mit einem entsprechenden Verfahren die Zeitpunkte der Prozesse zu bestimmen.

Methode der Modellierung: Die Möglichkeiten der Modellierung von Reihenfolgeproblemen sollen nachfolgend näher vorgestellt werden. Die "klassische" Methode, ist die

⁸Der Zeithorizont wird in endlich viele Intervalle unterteilt.



Modellierung des Zeitverlaufs anhand (extern vorgegebener) *äquidistanter Zeitpunkte* (Abb. a), *globaler Zeitpunkte* (Abb. b), *anlagenspezifischer Zeitpunkte* (Abb. c) oder *reihenfolgeabhängiger, kontinuierlicher Zeitpunkte* (Abb. d)

Abbildung 3.1: Modellierung des Zeitverlaufs

Beschreibung des Problems als *gemischt-ganzzahliges lineares Programm*. Mit dem *Constraint Programming* wurde ein relativ junges Verfahren aus der Forschung der Künstlichen Intelligenz in das Gebiet des Operations Research übernommen, die inzwischen auch vermehrt in kommerziellen Software Systemen (z. B. ILOG OPL Studio, ECLiPSe) implementiert wird. Bei der dritten Vorgehensweise wird schließlich auf die Theorie und Methoden der Graphentheorie zurückgegriffen, um das Problem als *gerichteten Graphen* darzustellen.

3.2.1 Gemischt-ganzzahlige Programmierung

Bei der Beschreibung eines Reihenfolgeproblems als gemischt-ganzzahlige Programmierung (*Mixed Integer Linear Programming*, MILP) wird ein Modell in der Form

$$\begin{aligned} \min \quad & c^T x \\ \text{unter den Nebenbedingungen} \quad & Ax \geq b \\ & c \in \mathbb{R}^n, \quad A \in \mathbb{R}^{m \times n}, \quad b \in \mathbb{R}^m, \quad x \in \mathbb{R}^n \end{aligned}$$

gelöst. Meistens gilt zusätzlich die Nichtnegativitätsbedingung $x \geq 0$. Dieses Modell heißt lineares Modell, da Zielfunktion und Nebenbedingungen von linearer Form sind. Gemischt-ganzzahlige Modelle haben zusätzlich für einige Entscheidungsvariablen x_i eine Ganzzahligkeitsbedingung $x_i \in \mathbb{N}$ oder sind sogar von binärer Form $x_i \in \{0, 1\}$.

Bei der Modellierung der Ablaufplanung als gemischt-ganzzahlige Programmierung sind nun zwei wesentliche Eigenschaften zu berücksichtigen.

Reihenfolge: Alle Aufträge müssen in der vorgegebenen Reihenfolge ihrer Operationen abgearbeitet werden.

Ausschließlichkeit: Eine Maschine kann keine zwei Operationen gleichzeitig durchführen⁹.

Diese Bedingungen können nun auf unterschiedliche Weise in einem MILP implementiert werden.

Das Assignment Problem: Bei dem Assignment Problem wird die Abfolge der Aufträge und ihrer Operationen durch die Zuordnung¹⁰ in einer Rangliste bestimmt. Die Rangliste hat ebenso viele Einträge wie Aufträge in der Ablaufplanung sortiert werden sollen. Entsprechend müssen in einem Optimierungsproblem mit k Aufträgen k^2 binäre Entscheidungsvariablen definiert werden. HEIPCKE ([Hei02], S. 117 ff.) formuliert ein Flow Shop-Problem¹¹ anhand von Ranglisten. Dieses Modell wird als Beispiel für eine *Assignment Modellierung* in Abb. 3.2 vorgestellt.

Es ist offensichtlich, daß eine solche Formulierung durch den quadratischen Faktor schnell in einer hohen Anzahl von Entscheidungsvariablen resultiert. Weiterhin werden durch die Reihenfolge in der Zuordnungsmatrix noch nicht die Startzeitpunkte der einzelnen Aufträge bestimmt. Dies bedarf zusätzlicher Variablen, um die Wartezeit von Aufträgen oder die Leerzeit der Maschinen zu bestimmen.

Das Zeitraster: Eine immer wieder verwendete Formulierung der Reihenfolgeplanung ist die Aufteilung des Planungszeitraums in diskrete Zeitintervalle¹² (oder *Perioden*). In einem Intervall kann dabei pro Maschine immer nur ein Auftrag bearbeitet werden. Entsprechend müssen bei t Intervallen für jeden Auftrag t Binärvariablen definiert werden, die bestimmen, ob die Produktion des Auftrags zu Beginn der Periode begonnen wurde. Da die Aufträge unterschiedliche Bearbeitungszeiten haben, muß hier zunächst der größte gemeinsame Teiler aller Produktionszeiten als Periodenlänge bestimmt werden [BG00].

Eine solche Modellierung bietet eine einfache Möglichkeit, die Anzahl der verfügbaren Mitarbeiter als Restriktion für die Anzahl zeitgleich durchführbarer Reinigungsprozesse zu modellieren (vgl. Abschnitt *Produktionsablauf* auf S. 30). Doch analog zu der Modellierung in einem Assignment Problem steigt auch hier die Anzahl der Entscheidungsvariablen schnell an.

⁹Hier ist zu beachten, daß die Zwischenlagerung in Silos zwar ebenfalls in der Modellierung berücksichtigt Dabei können bei erlaubtem Batch-Mix aber auch zwei oder mehr Chargen gleichzeitig gelagert werden. Der „Prozeß“ der Lagerung kann also nicht auf die gleiche Weise modelliert werden.

¹⁰Der Name *Assignment Problem* beschreibt die Aufgabenstellung der Zuordnung (engl.: *assignment*) in der Rangliste.

¹¹Zur Definition von Flow Shop-Problemen siehe Abschnitt *Maschinen-Charakteristik* auf Seite 10.

¹²Die häufige Verwendung einer Aufteilung in Zeitintervalle ist maßgeblich durch die Formulierung des State-Task-Netzwerkes von KONDILI ET AL. bedingt [KPS93], welches von verschiedensten Autoren verwendet und weiterentwickelt wurde. Das State-Task-Netzwerk wird im Abschnitt 3.4.1 auf S. 66 beschrieben.

Die disjunktive Formulierung: Bei einer disjunktiven Formulierung des Reihenfolgeproblems wird die zeitliche alternative Abfolge zwischen jeweils zwei Aufträgen formuliert. Für die Startzeitpunkte $st(j_1)$ und $st(j_2)$ von zwei Aufträgen j_1 und j_2 mit den Produktionsdauern t_{j_1} und t_{j_2} , die auf derselben Maschine gefertigt werden, gilt dann

$$st(j_1) + t_{j_1} \leq st(j_2) \quad \text{oder} \quad st(j_2) + t_{j_2} \leq st(j_1) \quad (3.1)$$

Eine solche Formulierung kann in ganzzahlig linearen Programmen nicht abgebildet werden, da hier immer alle Nebenbedingungen erfüllt sein müssen. Daher werden solche disjunktive Nebenbedingungen durch zusätzliche Binärvariablen und eine sogenannte *Big-M* Konstante ersetzt. SCHUSTER beschreibt die Umwandlung solcher Disjunktionen allgemein für lineare Optimierungsprobleme ([Schu03], S. 16) und HEIPCKE speziell anhand eines Job Shop-Problems ([Hei02], S. 124). Die obigen Bedingungen in (3.1) werden demnach ersetzt durch

$$st(j_1) + t_{j_1} \leq st(j_2) + M \cdot x_{j_1,j_2} \quad (3.2)$$

$$st(j_2) + t_{j_2} \leq st(j_1) + M \cdot (1 - x_{j_1,j_2}) \quad (3.3)$$

$$x_{j_1,j_2} \in \{0, 1\} \quad (3.4)$$

Bei einer Modellierung anhand von Disjunktionen sind daher für k Aufträge, deren Reihenfolge bestimmt werden sollen, $\frac{k \cdot (k-1)}{2}$ Binärvariablen und Nebenbedingungen zu implementieren. Insgesamt wächst der Umfang des Modellierungsproblems und die Anzahl der Entscheidungsvariablen mit jeder weiteren Operation, die in der Reihenfolgeplanung berücksichtigt werden muß, stark an.

Für das Flow Shop-Problem mit den Auftragsmenge \mathcal{J} ($|\mathcal{J}| = n_{\mathcal{J}}$), der Menge der Maschinen \mathcal{M} ($|\mathcal{M}| = n_{\mathcal{M}}$), und der Rangliste \mathcal{R} mit $|\mathcal{J}| = |\mathcal{R}|$, definiere die Entscheidungsvariable $x_{j,r}$ mit

$$\begin{aligned} \forall j \in \mathcal{J}, r \in \mathcal{R} & : x_{j,r} \in \{0, 1\} \\ \forall r \in \mathcal{R} & : \sum_{j \in \mathcal{J}} x_{j,r} = 1 \\ \forall j \in \mathcal{J} & : \sum_{r \in \mathcal{R}} x_{j,r} = 1 \end{aligned}$$

Hierbei bestimmt $x_{j,r}$, welcher Rang einem Auftrag j zugeordnet ist.

Zur Berechnung der Gesamtdauer definiere die reellwertige Variable $e_{m,r}$, die die ungenutzte (*empty*) Zeit der Maschine m zwischen den zwei Aufträgen vom Rang r und $r+1$ angibt, und die reellwertige Variable $w_{m,r}$, die die Wartezeit des Auftrags r zwischen den Operationen auf den Maschinen m und $m+1$ angibt.

$$\begin{aligned} \forall m \in \mathcal{M}, r = 1, \dots, (n_{\mathcal{J}} - 1) & : e_{m,r} \geq 0 \\ \forall r = 1, \dots, (n_{\mathcal{J}} - 1) & : e_{1,r} = 0 \\ \forall m = 1, \dots, (n_{\mathcal{M}} - 1), r \in \mathcal{R} & : w_{m,r} \geq 0 \\ \forall m = 1, \dots, (n_{\mathcal{M}} - 1) & : w_{m,1} = 0 \end{aligned}$$

Bei der Produktionsdauer $t_{m,j}$ eines Auftrags j auf einer Maschine m werden jetzt die Warte-, Leer- und Produktionszeiten miteinander verknüpft durch

$$\forall m = 1, \dots, (n_{\mathcal{M}} - 1), r = 1, \dots, (n_{\mathcal{J}} - 1) : \\ e_{m,r} + \sum_{j \in \mathcal{J}} t_{m,j} \cdot x_{j,r+1} + w_{m,r+1} = w_{m,r} + \sum_{j \in \mathcal{J}} t_{m+1,j} \cdot x_{j,r} + e_{m+1,r}$$

Für eine Minimierung des Makespan lautet das Optimierungsproblem dann

$$\min \sum_{m=1}^{n_{\mathcal{M}}-1} \sum_{j \in \mathcal{J}} t_{m,j} \cdot x_{j,1} + \sum_{r=1}^{n_{\mathcal{J}}-1} e_{n_{\mathcal{M}},r}$$

Abbildung 3.2: Formulierung eines Flow Shop-Problems als Assignment Problem

3.2.2 Constraint Based Programming

Das *Constraint Programming* (CP) oder auch *Constraint Based Programming* ist ein Lösungsverfahren, das basierend auf der logischen Programmierung im Forschungsgebiet der Künstlichen Intelligenz entwickelt wurde. Im Unterschied zur mathematischen Optimierung umfaßt die Forschung über CP nicht nur die Lösungssuche für restriktive Probleme, sondern auch die Darstellung der Problemstellung in einer entsprechenden Programmiersprache und die Speicherung der Daten in Datenbanken. CP bietet damit einen Rahmen, Problemstellung und Daten getrennt voneinander zu formulieren [Kle00]. Einen Überblick über CP und die Unterschiede in der Modellierung zur mathematischen Optimierung geben HEIPCKE [Hei99] und LUSTIG UND PUGET [LP01]. Zum weiteren Verständnis des Constraint Programming wird in Abb. 3.3 die Modellierung eines Job Shop-Problems von VAN HENTENRYCK ET AL. ([VM+99], Fig. 8) anhand der Modellierungssprache¹³ OPL vorgestellt und erläutert¹⁴.

Der wesentliche Unterschied zwischen CP und Verfahren der mathematischen Optimierung liegt darin, daß Problemstellungen des CP primär nach einer zulässigen, nicht jedoch nach einer optimalen Lösung suchen. Weiterhin ist es einfacher, disjunktive Nebenbedingungen zu implementieren. CP ist daher besonders geeignet, in schwierigen Problemstellungen eine zulässige Lösung zu finden, und ist somit für Anwendungen bei Problemstellungen des *Sequencing and Scheduling* empfehlenswert [LP01]. Die Zielfunktion kann durch zusätzliche Nebenbedingungen implementiert werden. Die Optimierung wird dann durch das Erstellen verschiedener Lösungen und sukzessives Hinzufügen weiterer Nebenbedingungen eingegrenzt.

Lösungsqualität und Rechendauer von CP sind noch nicht von gleicher Qualität wie in der mathematischen Optimierung [BPS99]. Durch Anwendung heuristischer Verfahren des Operations Research kann die Lösungssuche in Suchverfahren jedoch effizient verbessert werden.

¹³Eine Übersicht über *Software und Tools zur Optimierung* wird in dem entsprechenden Abschnitt auf S. 68 gegeben.

¹⁴Die Erläuterungen wurden anhand von [ILOG03] erstellt.

```

// Hinweis: Die Anweisung = ...; gibt an, daß Daten eingelesen werden.
// Gegeben ist eine Menge (range) von nbJobs Jobs, eine Menge
// von nbMachines Maschinen und eine Menge von nbTasks Tasks
int nbJobs = ...;
range Jobs 1..nbJobs
int nbMachines = ...;
range Machines 1..nbMachines
int nbTasks = ...;
range Tasks 1..nbTasks

// Über das zweidimensionale Array ressource wird für jeden Task
// der einzelnen Jobs eine Machine zugewiesen.
Machines ressource[Jobs,Tasks] = ...;
// Über das Array duration wird die Dauer je Job und Task eingelesen.
int+ duration[Jobs,Tasks] = ...;

// Für die Modellierung wird die Gesamtdauer totalDuration berechnet.
int totalDuration = sum(j in Jobs, t in Tasks) duration[j,t];
// Der Planungshorizont ScheduleHorizon ist dann genau
// totalDuration Zeiteinheiten lang.
ScheduleHorizon = totalDuration;

// Durch Activity werden die Variablen des Modells definiert.
// Die Variablen task übernehmen dabei die Dauer aus dem
// Array duration. Der makespan hat die Dauer 0.
Activity task[j in Jobs, t in Tasks](duration[j,t]);
Activity makespan(0);

// Das Schlüsselwort UnaryRessource weist das Modell an, daß den
// Machines jeweils nur eine Activity je Zeiteinheit zugewiesen
// werden darf.
UnaryRessource tool[Machines];

// OPL unterstützt mit den Zeitpunkt .end und der Zuweisung der
// Reihenfolge durch precedes die Modellierung von Reihenfolgeproblemen.
// Über requires werden die Zeitpunkte mit den
// Maschinenzuordnungen verknüpft.
minimize makespan.end;
subject to
{
  forall(j in Jobs)
    task[j,nbTasks] precedes makespan;
  forall(j in Jobs & t in 1..nbTasks-1)
    task[j,t] precedes task[j,t+1];
  forall(j in Jobs & t in Tasks)
    task[j,t] requires tool[ressource[j,t]];
};

```

Abbildung 3.3: Formulierung eines Job Shop-Problems mit OPL

3.2.3 Graphentheoretische Modellierung

Die letzte zu beschreibende Methode der Reihenfolgemodellierung nutzt die Graphentheorie¹⁵. Die Lösung eines Reihenfolgeproblems kann als Netzplan¹⁶ oder graphisch anschaulich als GANTT-Diagramm dargestellt werden. Diese Darstellung entspricht einem zusammenhängenden, gerichteten und zyklensfreien Graphen bzw. kann in einen solchen umgewandelt werden. Ein Netzplan kann wiederum auf Basis eines *disjunktiven Graphen* erstellt werden. Nach der folgenden Definition von BRUCKER ([Bru95], S. 143 ff.) ist ein disjunktiver Graph ein Graph mit gerichteten und ungerichteten Kanten.

Definition 3.1 *Ein disjunktiver Graph ist ein Graph $G = (V, C, D)$ mit der Menge V der Knoten, die die durchzuführenden Operationen darstellen. Die Menge C ist die Menge der gerichteten (konjunktiven) Pfeile, die die (vorgegebenen) Reihenfolgebeziehungen zwischen den Operationen darstellen. Die ungerichteten Pfeile der Menge D beschreiben die disjunktive Verknüpfung zwischen den Operationen, die auf derselben Maschine gefertigt werden und für die keine Reihenfolgebeziehung durch die Pfeile aus C vorgegeben ist.*

Bei der beschriebenen Modellierung ist also die Reihenfolge der Operationen der Aufträge bekannt, aber nicht die Abfolge der Operationen auf den einzelnen Maschinen. Für die Abfolge auf einer Maschine werden in der Modellierung die ungerichteten, *disjunktiven Kanten* eingefügt und der resultierende Graph daher als *disjunktiver Graph* bezeichnet.

Die hier beschriebene Modellierung wurde zuerst von ROY UND SUSSMANN (1964) und BALAS (1969) entwickelt. Sie haben diese Möglichkeit verwendet, um Probleme des Job Shop Scheduling zu beschreiben ([DSV93], S. 365 ff.). FORTEMPS UND HAPKE haben diese Modellierung auf Probleme der *ressourcenbeschränkten Projektplanung*¹⁷ übertragen [FH97]. Da eine Modellierung mit disjunktiven Graphen nicht auf Job Shop-Probleme mit *no wait*-Bedingungen¹⁸ angewandt werden kann, haben MASCIS UND PACCIARELLI schließlich das *alternative Graphenmodell* weiterentwickelt [MP02].

Zur Bestimmung einer Lösung anhand des disjunktiven Graphenmodells kann ein Branch- & Bound-Verfahren verwendet werden, bei dem die disjunktiven Kanten fixiert, d.h. in gerichtete Kanten umgewandelt werden. Dabei spielt die Reihenfolge der Fixierung allerdings eine erhebliche Rolle für die Leistungsfähigkeit dieses Lösungsverfahrens. Die Beschreibung eines solchen Verfahrens geben DOMSCHKE ET AL. ([DSV93], S. 384).

Zum weiteren Verständnis wird nachfolgend zu einer gegebenen Lösung das zugehörige disjunktive Graphenmodell beschrieben.

¹⁵Eine Einführung in die Graphentheorie ist bei AIGNER [Aig93] zu finden. Eine Zusammenfassung der hier benötigten Begriffe geben auch DOMSCHKE ET AL. ([DSV93], S. 52 ff.).

¹⁶Eine ausführliche Einführung in die Netzplantechnik wird z.B. von DOMSCHKE UND DREXL ([DD95], S. 86 ff.) oder HENNICKE [Hen91] gegeben.

¹⁷Probleme der ressourcenbeschränkten Projektplanung (engl.: *Resource Constrained Project Scheduling; RCPS*) sind eine Verallgemeinerung von Job Shop-Problemen [HD95]. Bei der Planung solcher Probleme werden den Operationen erneuerbare und nicht erneuerbare Ressourcen zugewiesen ([BE+93], S. 193 ff.). Es wird nicht mehr zwischen Maschinen und Einsatzstoffen unterschieden. Daher können für die Durchführung einer Operation auch mehrere Maschinen gleichzeitig verwendet werden, z.B. für die Planung eines Materialtransfers zwischen einzelnen Anlagekomponenten. Ein Anwendung wird z.B. von WITT für ein Problem der Stahlherstellung gegeben [Wi01].

¹⁸Siehe hierzu die Beschreibung der Auftrags-Charakteristik β_2 auf S. 11.

Beispiel 3.1 (Entwicklung eines disjunktiven Graphen) Gegeben ist ein Job Shop-Problem mit 3 Aufträgen J1, J2 und J3 (vgl. [DSV93] S. 367 f.). Die Aufträge sind mit jeweils 3 Operationen auf den Maschinen M1, M2 und M3 einzuplanen. Die Abfolge der Aufträge (mit der jeweiligen Dauer in Zeiteinheiten) ist:

$$\begin{aligned} J1 &: M1 (3) \rightarrow M2 (3) \rightarrow M3 (2), \\ J2 &: M2 (2) \rightarrow M3 (3) \rightarrow M1 (3) \text{ und} \\ J3 &: M2 (4) \rightarrow M1 (3) \rightarrow M3 (1). \end{aligned}$$

Abbildung 3.4 (a) zeigt eine Lösung des Reihenfolgeproblems in einem maschinenorientierten GANTT-Diagramm. In Abbildung 3.4 (b) wurde das Diagramm in den entsprechenden Netzplan umgewandelt. Dabei wurden zwei zusätzliche Knoten, \circ als Startknoten und $*$ als Endknoten hinzugefügt. Der Startknoten verweist auf die jeweils erste Operation der einzelnen Maschinen. Von den jeweils letzten Operationen der Maschinen wird auf den Endknoten verwiesen. Die gestrichelt dargestellten Pfeile stellen die Reihenfolgebeziehungen zwischen den Aufträgen dar.

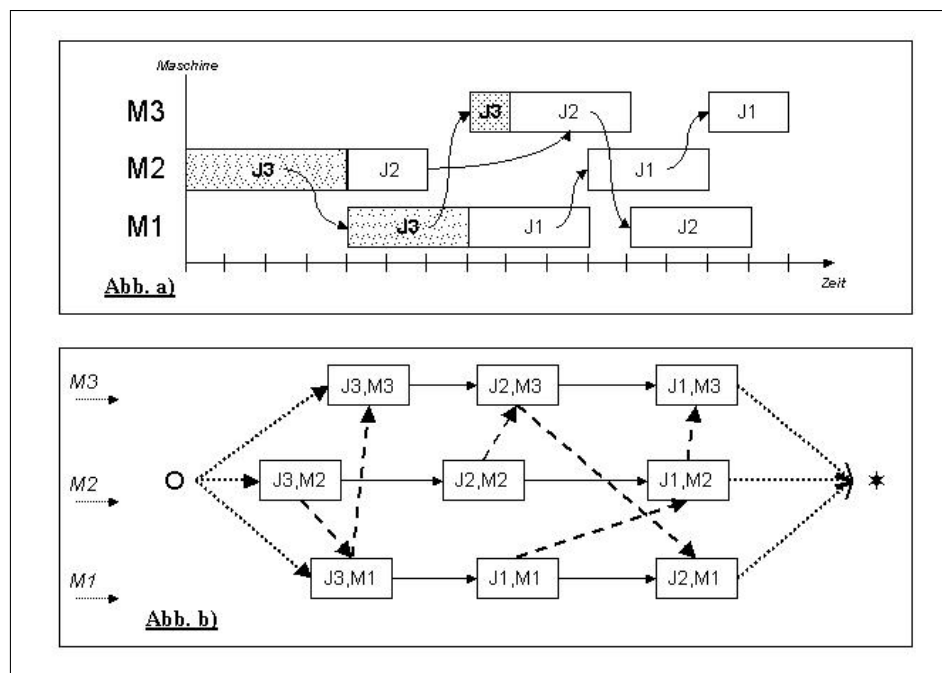


Abbildung 3.4: GANTT-Diagramm und gerichteter Graph

Mit der Fortsetzung des vorherigen Beispiels wird nun die Konstruktion eines disjunktiven Graphen beschrieben.

Die in Abb. 3.4 (a) gegebene Lösung eines Reihenfolgeproblems kann alternativ auch durch einen auftragsorientierten Netzplan beschrieben werden. Dieser ist in Abb. 3.5 (a) dargestellt. Hierbei verweist der Startknoten nicht auf die erste Operation einer Maschine, sondern auf die jeweils erste Operation eines Auftrags. Entsprechend weisen die jeweils letzten Operationen auf den Endknoten. Die gestrichelten Kanten geben hier die Reihenfolgebeziehung zwischen den Operationen einer Maschine wieder.

Wenn diese Reihenfolgebeziehungen (also die gerichteten Kanten) zwischen den Operationen einer Maschine aufgehoben werden (also durch ungerichtete Kanten ersetzt werden), erhält man mit Abb. 3.5 (b) den disjunktiven Graphen des Reihenfolgeproblems. Zusätzlich müssen hier noch ungerichtete Kanten zwischen den Operationen eingefügt werden, die in

der Lösung nicht in direkter Abfolge verbunden sind. Der disjunktive Graph enthält somit die gerichteten Kanten zwischen den einzelnen Operationen eines Auftrags und einen vollständigen¹⁹ Teilgraphen mit ungerichteten Kanten für alle Operationen, die auf derselben Maschine durchgeführt werden.

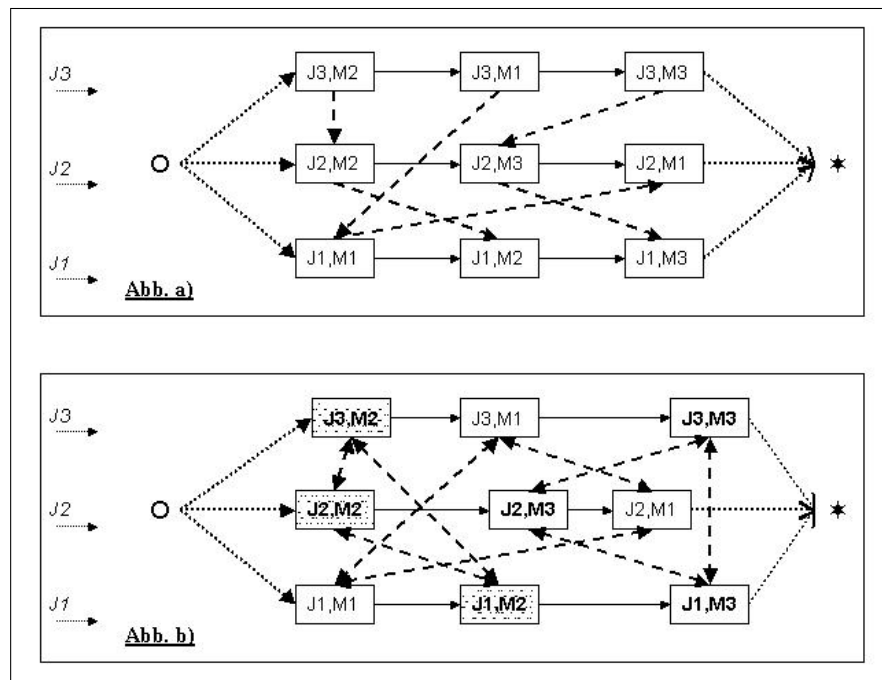


Abbildung 3.5: Auftragsorientierter Graph und zugehöriger disjunktiver Graph

Die Modellierung der Reihenfolge mit Mitteln der Graphentheorie bietet die Möglichkeit, daß deren Eigenschaften und Verfahren eingesetzt werden können. Darüber hinaus können solchermaßen formulierte Problemstellungen durch Interpretation oder Umwandlung in bereits bekannte Probleme überführt werden.

Das hier betrachtete Problem der Ablaufplanung kann als ein Problem der Tourenplanung interpretiert werden. Eine Lösung kann dann z.B. anhand von klassischen Tour-Konstruktionsverfahren gefunden und schrittweise verbessert werden ([Vog98], S. 59 ff.). Anhand eines Kürzeste-Wege-Verfahrens wie dem DIJKSTRA-Algorithmus²⁰ kann die *Entfernung* jedes Knotens von einem Start-Knoten (und somit der Startzeitpunkt der einzelnen Operationen) bestimmt werden. Dadurch kann auch der längste Weg, also der *Makespan* (d.h. die Entfernung zwischen Start- und Endknoten) berechnet werden ([DD95], S. 95).

Eigenschaften zur Optimierung

Für eine graphentheoretische Modellierung der Reihenfolgeplanung werden nun einige Definitionen und Eigenschaften vorgestellt, die bei der weiteren Optimierung eingesetzt

¹⁹Ein Graph heißt vollständig, wenn jeder Knoten mit jedem anderen Knoten durch eine Kante verbunden ist.

²⁰Dieses von DIJKSTRA 1959 vorgestellte Verfahren „besucht“ nacheinander alle Knoten des Graphen und berechnet deren Entfernung von einem Startpunkt ([Häs79], S. 23). Die berechneten Knoten werden entsprechend als „besucht“ markiert. Dieser Algorithmus wird in Kapitel 4.4.1 genauer beschrieben.

werden können. Es wird dabei jeweils eine bestehende Lösung, d.h. ein zyklensfreier ungerichteter Graph, vorausgesetzt.

Definition 3.2 *Ein bewerteter oder gewichteter Graph ist ein Graph, in dem entweder die Kanten oder die Knoten mit einer (positiven) Gewichtung versehen sind.*

Bei der Modellierung des Reihenfolgeproblems sind hierbei die Knoten, die die einzelnen Operationen darstellen, mit der Dauer der durchzuführenden Operation gewichtet.

Definition 3.3 *Ein Pfad oder Weg in einem Graph ist die Abfolge von Knoten, die jeweils paarweise durch Kanten miteinander verknüpft sind.*

Mit Definition 3.2 ergibt sich dann die Länge eines Weges als die Summe aller Gewichte des Pfades. Der längste Pfad in einem Graphen wird auch als kritischer Pfad bezeichnet ([Häs79], S. 36).

HENNICKE ([Hen91], S. 38) gibt eine äquivalente Definition des kritischen Pfades: Eine Abfolge von Vorgängen in einem Netzplan wird dann als kritisch bezeichnet, wenn der Gesamtpuffer Null ist. Dabei errechnet sich der Puffer aus der Differenz zwischen der frühesten und spätesten Startzeit einer Operation²¹. Die Verzögerung eines Ereignisses auf dem kritischen Pfad führt demnach zu einer Verlängerung der gesamten Produktionszeit ([Häs79], S. 36).

Durch die Verzweigungen an einem Knoten kann in einem gewichteten, zyklensfreien Graphen mehr als ein kritischer Pfad vorhanden sein [FH97]. Wenn das Gewicht eines Knotens als die Dauer des Prozesses interpretiert wird, der durch diesen Knoten repräsentiert wird, kann leicht nachgewiesen werden, daß die Summe der Gewichte entlang des kritischen Weges genau dem **Makespan** entsprechen. Dazu muß allerdings vorausgesetzt werden, daß die Knoten eine strikte Ende-Anfang-Beziehung²² haben, d.h. es gibt keine Wartezeiten und Überlappungen zwischen den Zeitpunkten der Knoten. Bei Reihenfolgeproblemen, die als Zielsetzung die Minimierung des **Makespan** verfolgen, kann aufgrund der Gleichheit mit der Länge des kritischen Pfades bei einer Modellierung mit disjunktiven Graphen das Ziel auch entsprechend undefiniert werden als Optimierungsproblem zur Minimierung der Länge des kritischen Weges [NS05].

Für die weitere Betrachtung wird angenommen, daß die Operationen so eingeplant und die Startzeitpunkte so berechnet werden, daß der resultierende Ablaufplan *semi-aktiv* ist ([Bru95], S. 7).

Definition 3.4 *Ein Ablaufplan wird als semi-aktiv bezeichnet, wenn jede Operation zu dem frühestmöglichen Zeitpunkt beginnt, der durch die vorhergehende Operation auf derselben Maschine und notwendig vorhergehende Operationen auf anderen Maschinen möglich ist, und jeder entsprechend frühere Zeitpunkt einer Operation nur durch Veränderung der Reihenfolge möglich ist.*

²¹Der späteste bzw. spätestmögliche Startzeitpunkt aller Operationen bzw. Ereignisse wird dabei ausgehend vom Zeitpunkt des Projektendes (i.S. der Produktionsplanung also das geplante bzw. erforderliche Produktionsende) durch eine retrograde Zeitrechnung über alle vorhergehenden Ereignisse bis zum Projektstart hin berechnet. Eine solche Berechnung ist nur möglich, wenn das Produktionsende bereits bekannt ist. Da eine Pufferzeit zugleich auch eine Wartezeit vor oder nach der Ausführung einer Operation bedingt, kann die Definition von HENNICKE ohne Veränderung des Inhaltes entsprechend verändert werden.

²²Zu den zeitlichen Beziehungen zwischen zwei Knoten bzw. Operationen vgl. die Abb. 1.1 auf S. 12.

Wenn die Prozesse nicht so geplant sind, daß ein semi-aktiver Plan gegeben ist, so muß ein entsprechender *left-shift* (d.h. eine lokale Linksverschiebung) durchgeführt werden [Cra96]. Mit diesem Verfahren werden entsprechend frühere Startzeitpunkte berechnet ohne die Reihenfolge auf den Maschinen selber zu verändern.

Das Problem der Reihenfolgeoptimierung ist bei einem semi-aktiven Ablaufplan somit äquivalent zu dem Problem, den längsten Weg zu minimieren. Hierzu muß die Lösung in dem bestehenden Ablaufgraph entsprechend verändert werden. Dazu werden im wesentlichen zwei Verfahren verwendet. Eine *Move*-Operation löscht einen Knoten aus der bestehenden Reihenfolge und fügt ihn an anderer Stelle wieder ein. Eine *Swap*-Operation vertauscht zwei Knoten²³ innerhalb eines Ablaufgraphen. Eine Swap-Operation ist somit die Abfolge von zwei Move-Operationen.

Für eine Optimierung würde es also genügen, nur Verschiebungen durch Move-Operationen zu verwenden. Weiterhin ist zu berücksichtigen, daß anstelle der Verschiebung eines Prozesses im Ablaufgraphen *hinter* einen Knoten auch eine oder mehrere Verschiebungen von einem oder mehreren anderen Prozessen *vor* einen Knoten durchgeführt werden kann.

Auf Basis der nachfolgenden Überlegungen²⁴ kann die Anzahl der Move-Operationen verringert werden. Dadurch wird der Lösungsraum, d.h. die Anzahl der Lösungen, reduziert. Es werden allerdings nur schlechtere Lösungen entfernt. Dazu sei zunächst eine weitere Definition gegeben.

Definition 3.5 *Ein Block ist eine maximale Teilfolge eines kritischen Pfades von (wenigstens zwei) Knoten, die Prozesse auf derselben Maschine darstellen.*

Wie aus der Definition des kritischen Pfades leicht ersichtlich wird, kann eine *bessere* Lösung nur erreicht werden, wenn ein Knoten verschoben wird, der auf dem kritischen Pfad des Ablaufgraphen liegt. Aufgrund der vorherigen Überlegungen verwenden wir dazu nur Move-Operationen, die einen Prozeß in dem Ablaufgraphen *vor* einen anderen Knoten verschieben. Der Startzeitpunkt des ersten Prozesses in einem Block oder eines Knoten, der auf dem kritischen Pfad, aber nicht in einem Block liegt, ist abhängig von einem Knoten auf einer anderen Maschine in dem Ablaufgraphen, oder er ist selber der erste Prozeß, der auf dieser Maschine ausgeführt wird. Dieser erste Prozeß bzw. dieser einzelne Knoten kann also in keiner Weise nach vorne verschoben werden, so daß eine bessere Lösung erzeugt wird. Damit folgt, daß es bei einer Optimierung eines semi-aktiven Ablaufplans genügt, nur Knoten zu verschieben, die in einem Block liegen.

3.3 Optimierungsverfahren

Mit der Anzahl der einzuplanenden Chargen in einer Reihenfolgeplanung wächst auch die Anzahl der möglichen Lösungen. Die Schwierigkeit, die optimale Lösung eines Planungsproblems in akzeptabler Rechenzeit zu finden, wird mit der Komplexitätstheorie²⁵ beschrieben. So heißt ein Optimierungsproblem²⁶ polynomiell lösbar, wenn es einen (deter-

²³Dabei müssen diese Knoten nicht direkt aufeinanderfolgend sein.

²⁴Diese wurden zuerst von GRABOWSKI ET AL. [GNZ86] vorgeschlagen; vgl [BHH03].

²⁵Für eine genauere Beschreibung der Komplexitätstheorie wird auf DOMSCHKE ET AL. ([DSV93], S. 47 ff.) und auf BRUCKER ([Bru95], S. 37 ff.) verwiesen.

²⁶Die Komplexitätstheorie wird anhand von Entscheidungsproblemen beschrieben, bei denen jeweils gefragt wird, *ob* es eine Lösung zu einem vorgegebenem Problem gibt. Bei Optimierungsproblemen wird

ministischen) Algorithmus gibt, der für alle möglichen Eingabewerte mit der Problemgröße²⁷ n eine Lösung in polynomialer Zeit findet, d.h. für ein Problem mit n Inputdaten kann eine Lösung dann also nach maximal n^k (für ein Polynom des Grades k) Lösungsschritten gefunden werden. Probleme, die dagegen nur durch nicht-deterministische Verfahren in polynomialer Zeit gelöst werden können, heißen \mathcal{NP} -schwer²⁸.

Mit wenigen Ausnahmen²⁹ sind Probleme der Reihenfolgeplanung schwer lösbar. So ist das Job Shop-Problem als \mathcal{NP} -schwer bekannt [Pin95]. Anstelle exakter Verfahren müssen daher Suchverfahren mit systematischer oder heuristischer Suchstrategie verwendet werden. Diese Optimierungsverfahren können anhand ihrer Vorgehensweise folgendermaßen klassifiziert werden.

Systematische Suche

Systematische Suchverfahren versuchen die optimale Lösung eines Planungsproblems zu finden, indem der Lösungsraum sukzessive eingeschränkt wird. Sie sind auch als Verfahren der mathematischen Optimierung bekannt. Für eine mathematische Programmierung ist darüber hinaus die Formulierung als (lineares) gemischt-ganzzahliges Optimierungsproblem erforderlich (vgl. Kapitel 3.2.1).

Das bekannteste systematische Suchverfahren ist das Branch & Bound-Verfahren³⁰ (B&B). Bei diesem Verfahren werden die einzelnen Entscheidungsvariablen nacheinander fixiert, und ausgehend von einer dann neu erhalten Zwischenlösung weitere Teillösungen (*branch*) separat untersucht. Wenn eine Lösung gefunden wurde, können alle Lösungen, die erwiesenermaßen einen schlechteren Zielfunktionswert haben, in der weiteren Lösungssuche unberücksichtigt bleiben (*bound*). Für eine ausführlichere Beschreibung der einzelnen Schritte wird auf VAHRENKAMP verwiesen ([Vah03], S. 29 ff.).

Weitere systematische Optimierungsmethoden sind das Schnittebenenverfahren³¹ oder Dekompositionsverfahren³². Verfahren der mathematischen Programmierung finden die optimale Lösung eines Optimierungsproblems. Dazu benötigen sie allerdings häufig eine hohe Rechenleistung und eine lange Rechenzeit.

dagegen nach der *besten* Lösung gesucht. Optimierungsprobleme können allerdings in Entscheidungsprobleme umgewandelt werden, indem gefragt wird, ob eine Lösung mit einem bestimmten (nämlich dem unbekannten optimalen) Wert existiert.

²⁷Die Problemgröße bezeichnet die Anzahl der zu optimierenden Daten.

²⁸ \mathcal{NP} für engl.: non-deterministic polynomial

²⁹Der Spezialfall des Flow Shop-Problems mit zwei Maschinen kann mit dem JOHNSON-Algorithmus exakt gelöst werden [Joh54]. Für den Spezialfall des Job Shop-Problem hat AKERS ein Verfahren zur Minimierung des Makespan entwickelt [Ak56]; vgl. [Vah00], S. 210.

³⁰Hierzu ist nicht notwendigerweise die Formulierung eines gemischt-ganzzahligen Optimierungsproblems erforderlich. Für Reihenfolgeprobleme stellen DOMSCHKE ET AL. ein B&B-Verfahren auf Basis der disjunktiven Graphen vor ([DSV93], S. 384 ff.).

³¹ Schnittebenenverfahren sind auch als Branch & Cut- oder GOMORY-Cut-Verfahren bekannt. Im Gegensatz zum B&B-Verfahren werden hierbei nicht die ganzzahligen Entscheidungsvariablen fixiert, sondern eine Lösung unter Mißachtung der Ganzzahligkeitsbedingung (sog. LP-Relaxation; LP = Linear Programming) gesucht. Basierend auf der Lösung dieses Problems werden dem Ursprungsproblem dann sukzessiv weitere Nebenbedingungen hinzugefügt ([NW88], S. 367 ff.).

³²Hierzu zählen die LAGRANGE- und die BENDERS-Dekomposition. Für eine weitere Beschreibung dieser Verfahren wird auf NEMHAUSER UND WOLSEY ([NW88], S. 409 ff.) verwiesen.

Heuristische Suche

Heuristische Suchverfahren benötigen keine Problembeschreibung in Form eines gemischt-ganzzahligen Optimierungsproblems. Sie werden oftmals dann eingesetzt, wenn eine systematische Suche aufgrund des Speicherbedarfs und Rechenaufwands nicht mehr möglich ist. Es kann nicht garantiert werden, daß diese Verfahren eine optimale Lösung finden. Sie werden daher auch als Approximationsverfahren bezeichnet.

Heuristische Verfahren benutzen oftmals eine prozedurale Herangehensweise zur Lösungsfindung. Auch wenn ein systematisches Suchverfahren nicht vollständig ausgeführt wird, sondern vor dem Nachweis der Optimalität abgebrochen wird, ist dieses ein approximatives Verfahren. Die Reihenfolgeplanung verwendet Heuristiken meistens in einer zweistufigen Vorgehensweise. In einem ersten Schritt wird durch eine *Konstruktionsheuristik* eine *Ausgangs-* oder *Startlösung* erzeugt. Diese Lösung wird in einem zweiten Schritt durch ein *Austausch-* oder *Verbesserungsverfahren* verändert. Ein Austauschverfahren, das nur bessere Lösungen akzeptiert, wird auch als *Iterative Improvement* bezeichnet ([Bru95], S. 51). Wenn stattdessen auch schlechtere Lösungen zugelassen werden, wird das Verfahren auch als *Meta-Heuristik* bezeichnet.

Am häufigsten werden in der Reihenfolgeplanung für die Erzeugung einer Startlösung Prioritätsregeln in Verbindung mit der *Greedy*-Suche³³ verwendet. Hier sind z.B. die kürzeste bzw. längste Bearbeitungsdauer, die kürzeste bzw. längste Restbearbeitungsdauer oder die Anzahl der noch auszuführenden Arbeitsvorgänge als Auswahlkriterien zu nennen³⁴. Nach JEHLE ET AL. erzielt hier eine Auswahl nach der kürzesten Bearbeitungsdauer gute Ergebnisse für das Optimierungsziel der minimalen Durchlaufzeit ([JMM90], S. 72). Planungssysteme können durch Anwendung unterschiedlicher Prioritätsregeln eine Simulation durchführen und dem Planer so unterschiedliche Pläne zur Auswahl stellen [SFR02].

Meta-Heuristiken

Meta-Heuristiken (Verbesserungsheuristiken) bestehen aus zwei wesentlichen Bestandteilen. Eine *Veränderungsvorschrift* transformiert eine bestehende Lösung in eine neue. Eine *Akzeptanzvorschrift* entscheidet dann, ob die neue Lösung als Ausgangspunkt der weiteren Suche verwendet wird, auch wenn diese einen schlechteren Zielfunktionswert hat.

Die Entscheidungsregeln über die Akzeptanz einer neuen Lösung sind oftmals Phänomenen der Natur nachgebildet. Hier sind vor allem das *Simulated Annealing* und seine Weiterentwicklungen *Threshold Accepting* und *Sintflutalgorithmus* zu nennen. Die einfachste Entscheidungsvorschrift ist sicherlich mit dem *Stabilisierungs-Destabilisierungs-Verfahren* von ABLAY (1987) gegeben ([Brü02], S. 42 ff.). Bei diesem wird zunächst mit einem Iterative Improvement Verfahren ein lokales Minimum gesucht. Wenn dieses erreicht ist, werden für eine vorgegebene Anzahl von Iterationen auch schlechtere Lösungen zugelassen.

Verbesserungsheuristiken werden weiterhin in mengenbasierte Verfahren und Verfahren,

³³ Die Bezeichnung Greedy-Search charakterisiert die Auswahl (*greedy* = engl.: gierig) des nächsten zu berücksichtigenden Elements in der Suche. Es wird jeweils nur der zusätzliche Gewinn betrachtet, der durch Hinzufügen eines Elementes (und nicht einer Kombination von Elementen) erreicht wird. Die Prioritätsregel bestimmt dann das Auswahlkriterium.

³⁴Weitere Prioritätsregeln sind bei JEHLE ET AL. ([JMM90], S. 70 ff.) und HOLZER ([Hol99], S. 92 ff.) zu finden.

die jeweils nur eine Lösung betrachten, unterschieden³⁵. Zu den mengenbasierten Verfahren zählen u.a. *Genetische Algorithmen* (GA) und *Evolutionsstrategien* (ES), sowie die *Tabu Suche* (TS).

Bei **Genetischen Algorithmen** und **Evolutionsstrategien** wird eine jeweils große Anzahl (sog. *Population*) von Lösungen³⁶ betrachtet. Diese Lösungen werden entsprechend verändert und die besten nach dem Darwinschen Prinzip des *survival of the fittest* für die weitere Suche ausgewählt. Bei Evolutionsstrategien wird eine neue Lösung jeweils durch Veränderung (sog. *Mutation*) von nur einer bestehenden Lösung erzeugt. Bei Genetischen Algorithmen wird eine neue Lösung dagegen durch *genetische Rekombination* und *Crossover*-Mechanismen aus zwei oder mehreren Individuen erzeugt ([SHF94], S. 219). Die neue Population besteht dann vollständig oder zu einem bestimmten Anteil aus den neu erzeugten und den bereits bestehenden Lösungen³⁷. Die Entscheidung über die Anwendung Genetischer Algorithmen oder einer Evolutionsstrategie ist auch abhängig von der Modellierung des Optimierungsproblems³⁸.

Im Gegensatz zu den vorigen mengenbasierten Meta-Heuristiken verändert die **Tabu Suche** jeweils nur eine Lösung, speichert aber Informationen über frühere Lösungen in eine sog. *Tabu Liste*. Dies sind Lösungen, die in der lokalen Suche nicht wieder berücksichtigt werden sollen. Das Verfahren verfügt also über ein Gedächtnis der bereits erzeugten Lösungen. Erst nach einer vorgegeben Anzahl von Iterationen werden diese wieder „vergessen“ ([Schu03], S. 94). Dies verhindert ein Zirkulieren zwischen wenigen Lösungen. Bei der Anwendung dieses Verfahrens ist dabei zu entscheiden, ob eine explizite Lösung, die Teilstruktur einer Lösung oder die Veränderungsvorschrift, die diese Lösung erzeugt hat, als *tabu* gelten soll.

Insbesondere bei einer großen Anzahl von Elementen in der Reihenfolgeplanung muß hier daher ein schnelles Verfahren gefunden werden, um alle Lösungen der Tabu Liste mit einer neu erzeugten Lösung zu vergleichen. Insgesamt erfordert die Tabu Suche einen höheren Implementierungsaufwand ([Me99], S. 105). Auch ist zu berücksichtigen, daß durch das Tabu, andere Lösungen, die eine Tabu-Lösung als Zwischenschritt in der lokalen Suche benötigen, nicht erreicht werden können ([BHH03], [BH00]).

3.3.1 Verbesserungsheuristiken

Verbesserungsheuristiken, die nicht mengenbasiert arbeiten, sind das *Simulated Annealing* und seine Abwandlungen das *Threshold Accepting* bzw. der *Sintflutalgorithmus*. Da diese Verfahren einfacher als mengenbasierte Heuristiken zu implementieren sind, sollen sie für die Reihenfolgeplanung in der Chargenproduktion verwendet werden. Nachfolgend werden daher die beiden Bestandteile einer Verbesserungsheuristik, die Veränderungsvorschrift und die Akzeptanzregeln der zuvor genannten Methoden, näher betrachtet.

³⁵Vgl. [Schn99], S. 18 f.

³⁶Entsprechend zu dem Begriff Population wird eine Lösung dann als Individuum bezeichnet.

³⁷Die Entscheidung, ob eine neue Lösung Teil einer neuen Generation der Population wird, wird anhand von sog. *Mutations- und Selektions-Verfahren* (MUSE) durchgeführt. Als solche werden die weiteren Meta-Heuristiken wie Simulated Annealing oder Threshold Accepting bezeichnet (vgl. ANSCHÜTZ ([Ans01], S. 72 ff.).

³⁸Für eine weitere Beschreibung dieser beiden Verfahren wird auf die ausführliche Einführung von SCHÖNEBURG ET AL. hingewiesen [SHF94].

Erzeugung einer Nachbarschaft

Die Optimierung mittels Verbesserungsheuristiken beruht auf der Suche in der Nachbarschaft einer bekannten Lösung. Für die Erzeugung einer solchen Nachbarschaft werden in dieser Arbeit die folgenden Bezeichnungen verwendet.

Definition 3.6 *Gegeben ist eine Menge von Chargen³⁹ \mathcal{C} und eine Produktionsanlage \mathcal{U} . Die Produktionsanlage umfaßt dabei alle Anlagekomponenten⁴⁰ zur Produktion, Lagerung, den innerbetrieblichen Transport und die Abfüllung.*

Eine Lösung der Reihenfolgeplanung⁴¹ \mathcal{S} ist die Zuordnung der Chargen $chg \in \mathcal{C}$ zu den Reaktoren $u \in UR \subset \mathcal{U}$

$$\mathcal{S} : \mathcal{C} \mapsto UR \times IN$$

Ein Reaktor u , auf dem n Chargen gefertigt werden, hat die Bearbeitungsreihenfolge $\Pi_u = (p_u(1), \dots, p_u(n))$.

Eine genauere Beschreibung von \mathcal{S} , bei der auch die weiteren Anlagekomponenten von \mathcal{U} in der Reihenfolgeplanung der Chargen verwendet werden, wird bei der Modellierung in Kapitel 4.2 gegeben.

Basierend auf dieser Definition kann eine Veränderungsvorschrift (oder Transformationsvorschrift) dann folgendermaßen beschrieben werden.

Definition 3.7 *Eine Transformationsvorschrift op beschreibt die Veränderung (Permutation), die an einer bestehenden Lösung \mathcal{S} durchgeführt wird, um eine neue Lösung zu erhalten.*

$$op : UR \times IN \rightarrow UR \times IN$$

Die Lösung $\mathcal{S}' := op(\mathcal{S})$ wird als Transformation oder auch Mutation bezeichnet.

In der Anwendung von Verbesserungsheuristiken werden im allgemeinen zwei unterschiedliche Transformationsvorschriften verwendet.

Definition 3.8 *Eine Move-Transformation op_{move} entfernt eine Charge an der Position $p_u(j)$ des Reaktor u und fügt sie an der Stelle $p_{u'}(i)$ des Reaktors u' wieder ein. Diese Veränderung⁴² bewirkt eine Verschiebung auf demselben Reaktor (falls $u = u'$) oder auf einen anderen Reaktor (wenn $u \neq u'$ ist).*

Eine Swap-Transformation op_{swap} führt zwei Move-Operationen aus. Hierbei werden zwei Chargen an den Positionen $p_u(j)$ und $p_{u'}(i)$ der Reaktoren u und u' entfernt und gegenseitig an den Stellen $p_{u'}(i)$ und $p_u(j)$ wieder eingefügt.

³⁹Die Menge \mathcal{C} entspricht der Menge der einzuplanenden Produktionsaufträge (Jobs) \mathcal{J} aus Kapitel 1.2.2.

⁴⁰Die Produktionsanlage \mathcal{U} bzw. die Teilmenge UR entsprechen der Notation der Maschinen in Kapitel 1.2.2. Anlagekomponenten werden nachfolgend immer mit u für (engl.) **unit** bezeichnet.

⁴¹ \mathcal{S} für (engl.) *Scheduling*.

⁴²Da die Stellen $p_u(j)$ $p_{u'}(i)$ nicht unbedingt benachbart sein müssen, d.h. für $u = u'$ muß nicht zwingend $i = j \pm 1$ sein, wird diese Permutation auch als *Jump-Transformation* bezeichnet.

Mit einer Transformation können nun die Nachbarschaft und weitere Eigenschaften der Nachbarschaft definiert werden ([Schu03], S. 81 ff.).

Definition 3.9 Die Nachbarschaft $\mathcal{N}(\mathcal{S})$ einer Lösung \mathcal{S} ist die Menge aller zulässigen Lösungen, die aus \mathcal{S} durch Anwendung einer Transformationsvorschrift op entsteht.

$$\mathcal{N}(\mathcal{S}) := \{\text{op}(\mathcal{S})\} \quad (3.5)$$

Eine Nachbarschaft heißt symmetrisch, wenn für jede Lösung $\mathcal{S}' \in \mathcal{N}(\mathcal{S})$ auch gilt, daß $\mathcal{S} \in \mathcal{N}(\mathcal{S}')$ ist.

Eine Nachbarschaft heißt schwach optimal zusammenhängend, wenn es für zwei beliebige Lösungen \mathcal{S} und \mathcal{S}' eine endliche Abfolge von Nachbarschaften $\mathcal{S}_1, \dots, \mathcal{S}_n$ gibt, für die gilt $\mathcal{S}_1 \in \mathcal{N}(\mathcal{S})$, $\mathcal{S}_{i+1} \in \mathcal{N}(\mathcal{S}_i)$ für $i = 1, \dots, n-1$ und $\mathcal{S}' \in \mathcal{N}(\mathcal{S}_n)$.

Da in der Anwendung von Verbesserungsheuristiken i.d.R. jeweils nur *eine* Veränderung durchgeführt und bewertet wird, werden Verbesserungsheuristiken auch als *lokale Suchverfahren* bezeichnet.

Bei der Implementierung einer Meta-Heuristik in einem Planungssystem muß also zunächst die Veränderungsvorschrift zur Erzeugung der Nachbarschaft bestimmt werden. In einem weiteren Schritt werden dann die Parameter festgelegt, mit denen die Mutation (anhand von Zufallsvariablen) erzeugt wird.

Diese Eigenschaften bestimmen insgesamt das Verhalten eines Verfahrens in der Lösungsfindung.

Simulated Annealing

Das Verfahren des Simulated Annealing (*simulierte Abkühlung*, SA) wurde unabhängig voneinander von KIRKPATRICK ET AL. ([KGV82], [KGV83]), sowie ČERNÝ [Cer85] entwickelt und basiert auf dem 1953 von METROPOLIS ET AL. [MR+53]⁴³ vorgestellten Algorithmus zur Simulation des physikalischen Verhaltens von Festkörpern. Die Idee des Simulated Annealing entstammt der Metallurgie. Ein Metall (Kristall) soll in einen stabilen, belastbaren Zustand (möglichst reiner Kristallstruktur) gebracht werden. Dazu wird dieses zunächst erhitzt und anschließend während einer kontrollierten *Abkühlung* (*annealing*) in die entsprechende Form gebracht. Der Energiezustand der Atome wird dabei von einem höheren Energiezustand in einen Zustand minimaler Energie überführt.

Das Simulated Annealing zählt zu den heuristischen Verfahren mit stochastischer Komponente. Ausgehend von einer vorhandenen Lösung wird durch eine Transformationsvorschrift eine neue Lösung erzeugt. Die neue Lösung wird anschließend mit einer gewissen Wahrscheinlichkeit akzeptiert ([LA92], S. 7 ff.). Die *Akzeptanzwahrscheinlichkeit* P ist dabei abhängig von der *Temperatur* T_i in dem jeweiligen Iterationsschritt i .

⁴³Alle Literaturquellen zitiert nach [LA92], S. ix.

Algorithmus 3.1 (Simulated Annealing)

- Gegeben ist eine Startlösung \mathcal{S} mit der Zielfunktion $f(\mathcal{S})$ und eine Transformationsvorschrift op .
- Definiere einen Iterationszähler $i := 0$, eine Ausgangstemperatur T_0 , eine Vorschrift $\vartheta(\cdot)$ zum Anpassen der Temperatur und ein Abbruchkriterium $\mathbf{X}(\cdot) := \text{true}$.
- *Rückgabewert:* Die beste Lösung⁴⁴ $\mathcal{S}_{\text{best}}$ anhand der obigen Parameter.

Solange ($\mathbf{X} == \text{true}$) *ist:*

1. *Permutation der bestehenden Lösung:*
Erstelle eine neue Lösung $\mathcal{S}' := \text{op}(\mathcal{S})$.
2. *Akzeptanzentscheidung:* Wenn $f(\mathcal{S}') \leq f(\mathcal{S})$ ist,
 - dann ist $\mathcal{S}_{\text{best}} := \mathcal{S}'$ und neue Ausgangslösung $\mathcal{S} := \mathcal{S}'$;
 - ansonsten wenn $p < \exp\left(-\frac{f(\mathcal{S}') - f(\mathcal{S})}{T_i}\right)$, für eine Zufallszahl $p \in [0, 1]$, dann ist \mathcal{S}' neue Ausgangslösung $\mathcal{S} := \mathcal{S}'$.
3. *Vorbereitung weiterer Iterationen:*
Setze $i := i + 1$. Reguliere die Temperatur $T_i := \vartheta(T_{i-1}, \cdot)$.
Bestimme das Abbruchkriterium $\mathbf{X} := \mathbf{X}(T_i, i, \cdot)$.

■ A(3.1)

Zur Permutation der bestehenden Lösung \mathcal{S} in Schritt 1 wird eine Lösung aus der Nachbarschaft $\mathcal{N}(\mathcal{S})$ gewählt. Für diese Auswahl können nach MEYR ([Me99], S. 97) oder FINK UND VOSS [FV02] unterschiedliche *Abarbeitungsstrategien* verwendet werden. Die neue Lösung \mathcal{S}' bei einer *First Fit*-Auswahl ist die stochastisch oder deterministisch erste Auswahl einer (zulässigen) Lösung. Bei der Auswahlregel *Best Fit* wird stattdessen die beste Lösung aus der Nachbarschaft ausgewählt. Dies bedingt allerdings eine Evaluation aller Lösungen in der Nachbarschaft. Da die Anzahl der Nachbarschaftslösungen, d.h. die Anzahl aller Transformationen der bestehenden Lösung mit wachsender Anzahl der miteinander zu kombinierenden Elemente schnell ansteigt, wird hierbei stattdessen auch eine Kandidatenliste $\mathcal{K}(\mathcal{S}) \subset \mathcal{N}(\mathcal{S})$ verwendet.

Die Akzeptanzregel in Schritt 2 beschreibt die Wahrscheinlichkeit, mit der eine neue Lösung akzeptiert wird. Die sog. *Boltzmann*-Wahrscheinlichkeitsverteilung

$$\mathcal{P}(\text{akzeptiere } \mathcal{S}') := \begin{cases} 1 & \text{falls } f(\mathcal{S}') \leq f(\mathcal{S}) \\ \exp\left(-\frac{f(\mathcal{S}') - f(\mathcal{S})}{T_i}\right) & \text{sonst} \end{cases} \quad (3.6)$$

basiert auf der Idee der Abkühlung des geschmolzenen Materials zu einem Festkörper. Bei hoher Temperatur T_i werden auch Verschlechterungen mit höherer Wahrscheinlichkeit akzeptiert. Weiterhin ist das Maß der Akzeptanz abhängig von dem Grad der Verschlechterung $f(\mathcal{S}') - f(\mathcal{S})$ einer neuen Lösung.

⁴⁴Der Algorithmus beschreibt hier die Suche nach einer besten Lösung anhand eines Minimierungsproblems.

Wesentliches Kriterium für die Lösungsgüte eines SA-Algorithmus ist die Auswahl des Abkühlungsschemas, d.h. der Starttemperatur und der Veränderung von T_i . Hier ist zu unterscheiden zwischen einer *homogenen* und *inhomogenen* Variante des Abkühlungsschemas⁴⁵. Bei der homogenen Variante wird die Temperatur T_i durch die Vorschrift $\vartheta(\cdot)$ nur nach Erreichen einer besseren Lösung reduziert. Bei der inhomogenen Variante wird die Temperatur in jedem Iterationsschritt reduziert. Entsprechend sind der Funktion $\vartheta(\cdot)$ die Werte des Iterationszählers i , der aktuellen Temperatur T_i , der besten bekannten Lösung $f(\mathcal{S}_{best})$ und der aktuellen Lösung $f(\mathcal{S}')$ zu übergeben. Um in akzeptabler Rechenzeit eine Lösung zu erreichen, die nahe am globalen Optimum liegt, werden die folgenden Werte für ein Abkühlungsschema propagiert ([Pu04], S. 23):

- Die *Starttemperatur* wird vor dem Start des Simulated Annealing extern berechnet. Für die Transformationen sollte eine anfängliche Akzeptanzwahrscheinlichkeit p von 75% bis 95% gegeben sein. Dazu werden zunächst mehrere Transformationen durchgeführt und die maximale Differenz der Zielfunktionswerte $\Delta f_{max} := \max\{|f(\text{op}(\mathcal{S})) - f(\mathcal{S})|\}$ bestimmt. Die Starttemperatur ist dann

$$T_0 := -\frac{\Delta f_{max}}{\ln(p)}. \quad (3.7)$$

Wie leicht ersichtlich ist, folgt diese Vorschrift aus der Umwandlung von (3.6) für einen festen Wert p .

- Die Temperatur wird multiplikativ verringert⁴⁶ durch

$$T_i := \alpha \cdot T_{i-1}. \quad (3.8)$$

Dabei hat sich für α ein Wert $\alpha \in [0,9; 0,99]$ bewährt.

Die Bestimmung dieser Werte wird von BRÜSSAU als *Dynamisches Simulated Annealing* bezeichnet und für die einfachere Anwendung durch einen Produktionsplaner in dem von ihm entwickelten Verfahren zur Losgrößen- und Reihenfolgeplanung in der Fertigungsindustrie implementiert ([Brü02], S. 55).

Als letztes ist das Abbruchkriterium X zu bestimmen. Entweder ist dazu eine *Endtemperatur* vorzugeben, ab der keine Veränderung zugelassen wird, oder es wird ein Maximalwert für die Gesamtanzahl der Iterationen oder der Iterationen seit der letzten Verbesserung des Zielfunktionswertes angegeben ([Me99], S. 101; [SNT96], S. 280).

NEMHAUSER UND WOLSEY weisen darauf hin, daß das Simulated Annealing gegen ein globales Optimum konvergiert, wenn die Nachbarschaft schwach optimal zusammenhängend und symmetrisch ist und jede Lösung mit gleicher Wahrscheinlichkeit ausgewählt wird ([NW88], S. 407). BOHACHEVSKY ET AL. untersuchten, ob die Akzeptanzwahrscheinlichkeit des Simulated Annealing Algorithmus durch eine andere Wahrscheinlichkeitsverteilung ersetzt werden könnte ([BJS95], S. 6). Doch hier hat sich ihren Aussagen zufolge die Wahrscheinlichkeitsverteilung aus (3.6) als überlegen erwiesen.

⁴⁵Eine Übersicht und Klassifikation der Abkühlungsschemata geben OSMAN UND CHRISTOFIDES [OC94].

⁴⁶Generell wäre hier z.B. auch die Subtraktion $T_i := T_{i-1} - \tau$ eines festen Wertes τ möglich.

Threshold Accepting

Basierend auf dem Simulated Annealing wurde 1990 von DUECK UND SCHEUER das Threshold Accepting (TA) eingeführt [DuS90]⁴⁷. Die Vorgehensweise des Threshold Accepting Algorithmus ist analog zu der des Algorithmus 3.1. Lediglich im Schritt 2 wird eine andere Akzeptanzregel verwendet:

Algorithmus 3.2 (Threshold Accepting – Veränderung des SA)

- Definiere einen Schwellenwert T_i .
- Ersetze Schritt 2 (*Akzeptanzregel*) im Verfahren des Simulated Annealing (Algorithmus 3.1) durch die folgende

2. *Akzeptanzentscheidung*: Wenn $f(\mathcal{S}') < f(\mathcal{S})$ ist,

- dann ist $\mathcal{S}_{best} := \mathcal{S}'$ und neue Ausgangslösung $\mathcal{S} := \mathcal{S}'$;
- ansonsten wenn $f(\mathcal{S}') \leq f(\mathcal{S}) + T_i$ ist, dann ist \mathcal{S}' neue Ausgangslösung $\mathcal{S} := \mathcal{S}'$.

■ A(3.2)

Anstelle der stochastischen Akzeptanzentscheidung des Simulated Annealing werden hier schlechtere Lösungen immer akzeptiert, sofern die Differenz einen gewissen Schwellenwert T_i (*threshold*) nicht übersteigt. Entsprechend ist hier das Verfahren zur Anpassung der Temperatur $\vartheta(\cdot)$ aus Algorithmus 3.1 als ein Verfahren zur Anpassung des Schwellenwerts zu verstehen.

Zur Bestimmung des Start-Schwellenwertes T_0 sind nach NISSEN ([Nis97], S. 255 f.)⁴⁸ zunächst mehrere Startlösungen \mathcal{S}_i zu erzeugen. Der Startwert T_0 wird dann bestimmt als

$$T_0 := \gamma \cdot \frac{1}{n} \sum_{i=1}^n f(\mathcal{S}_i). \quad (3.9)$$

Dabei ist γ ein beliebiger Wert. BRÜSSAU wandelt den Vorschlag für das *Dynamische Threshold Accepting* ab ([Brü02], S. 48 ff.). Er weist darauf hin, daß für $(\gamma \ll 0,3)$ nur schlechte Lösungen akzeptiert werden und das Verfahren schnell in ein lokales Optimum gelangt. Für $(\gamma \gg 2)$ entspricht die Suche einer Zufallssuche ohne optimierende Komponente.

Der Schwellenwert kann im Laufe des Verfahrens mit der gleichen Vorschrift wie beim Simulated Annealing angepaßt werden. Sofern das von BRÜSSAU entwickelte Dynamische Threshold Accepting auf einem gewissen Niveau stagniert, wird der Schwellenwert neu errechnet bzw. angepasst. PUCHTA bezeichnet dieses Vorgehen auch als *Reheating* ([Pu04], S. 25).

⁴⁷Vgl. OSMAN UND KELLY ([OK96b], S. 12)

⁴⁸Vgl. BRÜSSAU ([Brü02], S. 38)

Sintflut-Algorithmus

Als eine weitere Variante eines lokalen Suchverfahrens wurde 1993 von DUECK der *Great Deluge* Algorithmus (GD) entwickelt [Due93]. Dieser ist auch unter dem Namen *Sintflut-Algorithmus* bekannt. Dies spiegelt sich in der entsprechenden Akzeptanzregel wieder. Eine schlechtere Lösung wird im Gegensatz zu dem Threshold Accepting nicht unter Begrenzung der Verschlechterung akzeptiert, sondern nur dann akzeptiert, wenn (bei Maximierungsproblemen) die neue Lösung einen vorgegebenen minimalen Zielfunktionswert nicht unterschreitet. Diese untere Schwelle des Zielfunktionswertes entspricht einer Sintflut, die langsam ansteigt. Es werden also nur Lösungen akzeptiert, die oberhalb des Wasserspiegels liegen.

Algorithmus 3.3 (Sintflut-Algorithmus – Veränderung des SA)

- Definiere einen Sintflutwert T_i .
- Ersetze Schritt 2 (*Akzeptanzregel*) im Verfahren des Simulated Annealing (Algorithmus 3.1) durch die folgende

2. *Akzeptanzentscheidung*: Wenn $f(\mathcal{S}') < f(\mathcal{S})$ ist,

- dann ist $\mathcal{S}_{best} := \mathcal{S}'$ und neue Ausgangslösung $\mathcal{S} := \mathcal{S}'$;
- ansonsten wenn $f(\mathcal{S}') \leq T_i$ ist, dann ist \mathcal{S}' neue Ausgangslösung $\mathcal{S} := \mathcal{S}'$.

■ A(3.3)

Auch für diese Heuristik muß eine Vorschrift $\vartheta(\cdot)$ entwickelt werden, um den Sintflutwert in den einzelnen Iterationen entsprechend anzupassen.

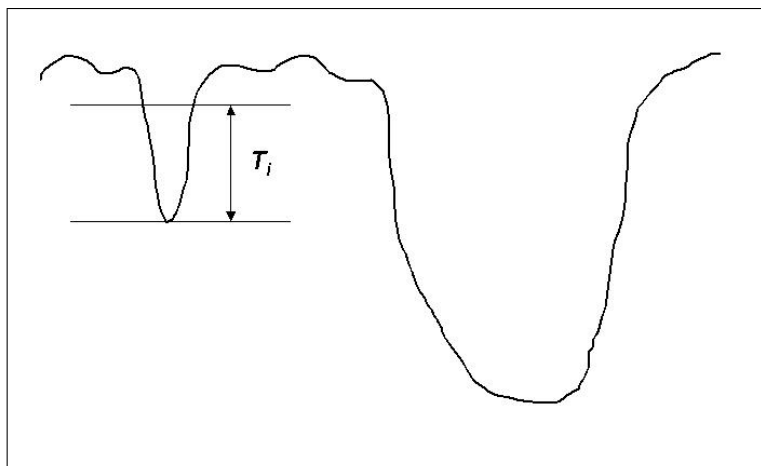
Für das Verfahren des Threshold Accepting, wie auch für den Sintflut-Algorithmus ist jedoch anzumerken, daß sie im Gegensatz zu dem Simulated Annealing nicht immer gegen die optimale Lösung konvergieren. Nach SCHNEIDER kann ein solcher Lösungsalgorithmus bei Optimierungsproblemen mit einem *Golfloch*-Optimum (vgl. Abb. 3.6) in diesem steckenbleiben ([Schn99], S. 24). Bei einer Lösungssuche mit dem Simulated Annealing besteht dagegen aufgrund der Wahrscheinlichkeitsentscheidung die Möglichkeit, diese Lösung wieder zu verlassen.

3.3.2 Literaturübersicht

Bei der Vorstellung des Threshold Accepting Verfahrens schrieben DUECK UND SCHEUER, daß dieses gegenüber dem Simulated Annealing bessere Ergebnisse in kürzerer Rechenzeit liefere [DuS90]. Ihre Untersuchungen haben sie jedoch anhand von Tourenplanungsproblemen durchgeführt. Diese Erfahrungen können daher nicht ohne weiteres auf Optimierungsprobleme der Reihenfolgeplanung übertragen werden.

Für das Job Shop-Problem stellen VAESSENS ET AL. eine bessere Performance des Simulated Annealing Verfahrens fest [VAL96]⁴⁹. Auch KÄSCHEL ET AL. stellen in einem

⁴⁹Vgl. MEYR ([Me99], S. 103)



Ein *Golfloch*-Optimum ist ein lokales Optimum, dessen Funktionswert stark von den benachbarten Lösungen abweicht. Wenn die Abweichung im Verfahren des Threshold Accepting größer als der Schwellenwert T_i ist, dann bleibt die Optimierung in diesem lokalen Optimum stecken.

Abbildung 3.6: Lösungsraum mit *Golfloch*-Optimum

Vergleich der, in dieser Arbeit zuvor beschriebenen Heuristiken (Verbesserungsverfahren SA, TA, GD und Genetische Algorithmen) eine Dominanz des Simulated Annealing fest [KT+99]. Eine Untersuchung von LEE ET AL. scheint dagegen die bessere Performance für ein Threshold Accepting Verfahren mit Prioritätsregeln für Probleme mit *no wait* Bedingungen zu bestätigen [LVP02]. Eine Übersicht über weitere Vergleiche dieser beiden Verfahren sind z.B. bei BRÜSSAU ([Brü02], S. 42) und KIESSWETTER ([Kie99], S. 77) zu finden.

Insgesamt sind diese Heuristiken zu stark verwandt, um einem Verfahren eine eindeutig bessere Performance nachzuweisen. FINK UND VOSS haben das System *HotFrame* entwickelt, in dem verschiedene Optimierungsverfahren in einem Baukastenprinzip erstellt werden können [FV02]. Anhand der Vielzahl der Einstellungen und Parameter, die hierbei berücksichtigt werden können, wird ersichtlich, daß nicht nur die Komplexität eines Planungsproblems die Ergebnisse und Rechengeschwindigkeit eines heuristischen Optimierungsverfahrens beeinflußt, sondern auch die Auswahl der Parameter entscheidend ist.

Das Verfahren des Simulated Annealing wurde erfolgreich von KARIMI UND HASEBE zur Reihenfolgeplanung in der Chemischen Industrie eingesetzt [KH95]. Auch RAAYMAKERS UND HOOGEVEN haben ein Job Shop-Problem aus der Prozeßindustrie unter besonderer Berücksichtigung von *no wait*-Bedingungen⁵⁰ mit dem Simulated Annealing gelöst [RaH00].

Verschiedene Veränderungen an dem Verfahren des Simulated Annealing haben bei den Untersuchungen verschiedener Autoren zu einer weiteren Verbesserung des Simulated Annealing geführt. So haben SCHRIMPF ET AL. [Sch+00] mit der Methode des *Ruin & Recreate* vor allem Veränderungen an der Transformationsvorschrift durchgeführt. Eine bestehende Lösung wird in größerem Umfang zerstört und wieder aufgebaut. Die Suche ist somit nicht mehr auf eine lokale Nachbarschaft beschränkt und kann schnell andere Be-

⁵⁰Vgl. Parameter β_2 in Kapitel 1.2.2

reiche des Lösungsraums aufsuchen. Ein weiteres Verfahren zum Verlassen lokaler Optima haben DATTAL ET AL. mit der Erweiterung zum *Multicanonical Jump Walk Annealing* entwickelt [DJV01].

3.4 Operative Produktionsplanung in der verfahrenstechnischen Industrie

Die operative Produktionsplanung verfahrenstechnischer Prozesse ist aufgrund der Anzahl der Produkte und der Anlagenstruktur einer Produktionsanlage i.d.R. so komplex, daß ein Produktionsplaner nicht mehr alle Anforderungen zufriedenstellend berücksichtigen kann. Hier ist eine Unterstützung durch entsprechende computergestützte Verfahren notwendig. Bei der Modellierung und Entwicklung solcher Planungsverfahren ist zu unterscheiden zwischen den Rahmenbedingungen, die korrekt eingehalten werden müssen, und den bedingten Beschränkungen, die soweit als möglich eingehalten werden sollten⁵¹. Die Erstellung eines Planungsverfahrens ist dabei immer auch an die Verfügbarkeit der Informationen gebunden ([Ge95], S. 49). Für die operative Produktionsplanung ist daher immer zu prüfen, welche Informationen benötigt und welche vernachlässigt werden können. Zur Entwicklung einer entsprechenden Optimierungsmethode werden nachfolgend einige Modelle und Verfahren zur Produktionsplanung in der Prozeßindustrie vorgestellt und anschließend die technischen Aspekte für die Entwicklung eines Planungsverfahrens beschrieben.

3.4.1 Literaturübersicht zu Modellen der Mengen- und Reihenfolgeplanung

Die Notwendigkeit der speziellen Betrachtung und Entwicklung von Modellen und Verfahren zur Planung in der Prozeßindustrie wird von verschiedenen Autoren beschrieben. ALLWEYER ET AL. [ALS96] weisen darauf hin, daß zwar viele Forschungsarbeiten zur Reihenfolgeplanung im allgemeinen durchgeführt werden. Hierbei werden jedoch i.d.R. nur spezielle Problemklassen betrachtet, die nicht die Komplexität der Anforderungen der Realität abdecken. Da auch die Modelle und Verfahren aus der Fertigungsindustrie nicht auf die verfahrenstechnische Industrie übertragbar sind, müssen hier spezielle Verfahren entwickelt werden. Ein erstes Modell, das als allgemeines Modell für die kurzfristige Reihenfolgeplanung in der Prozeßindustrie entwickelt wurde, ist das *State-Task-Network*⁵² von KONDILI ET AL. [KPS93]. Dieses und seine Weiterentwicklung, das *Resource-Task-Network*⁵³, werden nachfolgend für die Verwendung in dieser Arbeit diskutiert.

Die verschiedenen Anforderungen der Prozeßindustrie wurden erstmals in einer internen Fallstudie der Bayer AG und der BASF AG von WESTENBERGER UND KALLRATH zu-

⁵¹GEORGI ([Ge95], S. 47 ff.) bezeichnet diese Beschränkungen als imperative und admissive Beschränkungen. Erstere sind „Muß“-Faktoren, die unbedingt erfüllt sein müssen, damit eine zulässige Lösung erstellt werden kann. Letztere sind „Soll“-Faktoren, deren Nicht-Einhaltung zwar höhere Kosten verursachen, aber die Bestimmung einer Lösung nicht verhindern.

⁵²Aus sprachlichen Gründen wird in dieser Arbeit von einem State-Task-Netzwerk gesprochen.

⁵³Aus sprachlichen Gründen wird in dieser Arbeit anstelle des Resource-Task-Network die Bezeichnung Resource-Task-Netzwerk verwendet.

sammenfassend beschrieben⁵⁴ [WK94]. Bei dieser Studie werden alle aus der Literatur bekannten Anforderungen berücksichtigt. Daher ist sie auch Basis für die Untersuchungen verschiedener Autoren geworden. So untersucht AHLEFF [Ah95] die Möglichkeiten, mathematische Methoden zur Lösung zu verwenden. Basierend auf der Modellierung des State-Task-Netzwerkes entwickelt BLÖMER [Blö99] ein heuristisches Verfahren. Eine Übersicht über weitere Forschungsarbeiten zu diesem Testproblem wird von KALLRATH [Kr02] gegeben.

Maßgeblich für die Entwicklung eines möglichst allgemein einsetzbaren Modells ist jedoch auch die Bestimmung der Chargengrößen. Bei einer alleinigen Betrachtung der Reihenfolgeplanung werden diese immer als gegeben vorausgesetzt. Dies resultiert in einer hierarchischen Herangehensweise der Planung. In dieser Arbeit werden solche *Fertigungsaufträge*⁵⁵ allerdings nicht als gegeben vorausgesetzt. Stattdessen werden der Planung nur die „Kundenaufträge“, d.h. die täglich zu fertigenden Bedarfe der Endprodukte übergeben. Das hieraus für die Planung resultierende Problem wird von TRAUTMANN [Tra01] als *Batching-Problem* bezeichnet. Voraussetzung für die Lösung von Batching-Problemen ist die informationstechnische Darstellung und die Berechnung der Sekundärbedarfe.

Berechnung der Sekundärbedarfsmengen

Die Produktionsstruktur kann graphisch anhand von Gozinto-Graphen⁵⁶ dargestellt werden. In ERP-Systemen werden diese Graphen in einer Datenbank anhand einer Material-Tabelle und einer Aufschlüsselung in einer Bill-of-Material Tabelle gespeichert. LOOS [Loo01] weist darauf hin, daß zyklische Produktionsprozesse in Gozinto-Graphen zwar abgebildet werden können, dies aber aufgrund des Zyklus zu Problemen bei der algorithmischen Berechnung des Materialbedarfs führt. Ebenso können analytische (umgruppierende) Produktionsprozesse, bei denen mehrere Hauptprodukte, oder ein Hauptprodukt und mehrere Nebenprodukte erzeugt werden, nicht durch Gozinto-Graphen abgebildet werden. Für beide Fälle hat LOOS eine Erweiterung des ursprünglichen Gozinto-Graphen entwickelt und eine Implementierung für ERP-Systeme, die auf Basis von MRP-Algorithmen arbeiten, vorgeschlagen.

Eine andere Methode zur Berechnung der Sekundärbedarfsmengen in einer zyklischen Produktionsstruktur hat ANSCHÜTZ mit der sog. *Wasserfallmethode* beschrieben ([Ans01], S. 164 ff.). Dieses Verfahren erhöht sukzessive das Volumen der eingesetzten Rohstoffe in den frühesten Produktionsstufen der Produktionsfolge eines Endproduktes, so daß in dem jeweils nachfolgenden Produktionsschritt eine neue Charge erzeugt werden kann. Das Verfahren stoppt, sobald der Gesamtbedarf des Endprodukts erfüllt wird. ANSCHÜTZ weist darauf hin, daß dieses Verfahren nicht zwingend die optimale Lösung findet. Weiterhin berücksichtigt dieses Verfahren nicht die Interdependenz zwischen der Mengen- und der Reihenfolgeplanung.

⁵⁴Die Daten dieses Arbeitspapiers wurden zuerst von GÜNTHER UND TRAUTMANN [GTr01] und später auch von KALLRATH [Kr02] publiziert.

⁵⁵Der Begriff „Fertigungsauftrag“ wird bei der Planung von Produktionsprozessen in der fertigungstechnischen Planung verwandt. Er kann nur bedingt mit dem Begriff einer Charge gleichgesetzt werden und wird daher nicht weiter verwendet.

⁵⁶Zur Beschreibung von Gozinto-Graphen siehe S. 18 ff.

Modelle zur Reihenfolgeplanung

State-Task-Netzwerk (STN): Zur eindeutigen Abbildung verfahrenstechnischer Produktionsprozesse haben KONDILI ET AL. das *State-Task-Netzwerk* für die Reihenfolgeplanung entwickelt [KPS93].

In einem State-Task-Netzwerk werden zwei verschiedene Arten von Knoten betrachtet. Die *state*-Knoten repräsentieren die Rohstoffe, Zwischen- und Endprodukte. Die *task*-Knoten werden verwendet, um den eigentlichen Produktionsprozeß abzubilden. Diese Knoten werden dann entsprechend der Herstellungsreihenfolge miteinander verknüpft. Basierend auf diesem Netzwerk entwickeln KONDILI ET AL. ein MILP-Modell⁵⁷ zur Reihenfolgeplanung.

Wichtig für dieses deterministische Modell ist eine konstante, volumenunabhängige Bearbeitungsdauer für die einzelnen Chargen (*tasks*). Zu den weiteren Annahmen zählt die Nicht-Unterbrechbarkeit der Produktionsprozesse, und die sofortige Überführung der Materialien und Produkte aus einem *task* (Produktionsprozeß) in einen *state* (Materialbestand) und umgekehrt. Die Anlagen dürfen also nicht für eine Zwischenlagerung genutzt werden. Auf Basis eines äquidistanten Zeitraster wird dann die Reihenfolgeplanung zur Minimierung der Gesamtdauer modelliert. Der Vorteil dieses Modells liegt in der äußerst flexiblen Abbildung komplexer Netzwerkstrukturen ([PSP94] S. 3170).

Zur Modellierung der Reihenfolgeplanung muß allerdings zunächst die Intervalllänge für das Zeitraster bestimmt werden. KONDILI ET AL. verwenden hierzu einfach den größten gemeinsamen Teiler aller Prozeßzeiten [KPS93]. Dies kann im Extremfall dann die Dauer der verwendeten Zeiteinheit sein. Für eine 8-Stunden-Schicht sind schon bei einem Zeitintervall von 5 Minuten 96 Binärvariablen je Ressource zu modellieren, die darstellen, ob die Ressource verwendet wird oder für eine Produktion zur Verfügung steht. Diese Eigenschaft kann dazu führen, daß das Optimierungsproblem sehr umfangreich wird und nicht mehr in akzeptabler Zeit lösbar ist. Zur Lösung hat BLÖMER daher ein auf einer LP-Relaxation basierendes, zweistufiges heuristisches Verfahren entwickelt [BG00]. KIESSWETTER modelliert eine starke Verknüpfung der Binärvariablen, die je Task und Zeitraster definiert werden ([Kie99], S. 92 ff.). Dies erzeugt einen sog. *Domino-Effekt* bei der Fixierung einzelner Variablen, der in der Lösungssuche durch Branch & Bound-Verfahren auch bei mehr als 5.000 Binärvariablen noch zu optimalen Lösungen führt.

Daneben finden sich in der Modellierung weitere Schwächen:

- Die *tasks* bilden einen Produktionsprozeß ab. Zwischen den einzelnen Prozessen können jedoch zusätzliche Rüst- und Reinigungsvorgänge stattfinden. Diese haben jedoch keinen Bezug zu dem Material und können daher nicht modelliert werden. Eine Lösung dieser Problematik geben MÉNDEZ ET AL. ([MC+05], S. 13).
- Die Reihenfolgeplanung berücksichtigt zunächst nicht die Anlagen. Es wird angenommen, daß die *tasks* auf genau einer vorher bestimmten Anlage durchgeführt werden. Eine Anwendung der Modellierung auf Produktionsanlagen mit mehreren Ressourcen gleicher Funktionalität, Kapazität und Leistung kann nur durch Einfügen weiterer Binärvariablen abgebildet werden. Ebenso können mit dem vorliegenden Modell keine Prozesse abgebildet werden, die auf mehrere Ressourcen zeitgleich zugreifen.

⁵⁷MILP = Mixed Integer Linear Programming; Siehe Kapitel 3.2.1 zur *Gemischt-Ganzzahligen Programmierung*

Resource-Task-Netzwerk (RTN): Eine alternative Möglichkeit der Modellierung bietet das *Resource-Task-Netzwerk*, das von PANTELIDES entwickelt wurde [Pan94].

Das Resource-Task-Netzwerk kann als Verallgemeinerung des State-Task-Netzwerks betrachtet werden. Unter dem Begriff der *Ressource* werden hier alle benötigten Produktionsmittel zusammengefasst. Dies umfaßt die Materialien (Rohstoffe, Zwischen- und Endprodukte), die Produktionsanlage mit allen Reaktoren und Tanks, sowie weitere „Hilfsmittel“ (Mitarbeiter, Dampf, Energie). Der Vorteil des Resource-Task-Netzwerks liegt in der Betrachtung der Anlagen. Für Prozesse, die auf verschiedenen Anlagen durchgeführt werden können, muß lediglich eine Binärvariable in dem MILP-Modell definiert werden.

Die beschriebene MILP-Formulierung anhand eines STN oder auch eines RTN hat den Nachteil, daß bei Annahme eines festen Zeitrasters (insb. bei einem langen Planungshorizont und kurzen Intervallängen) eine große Anzahl von Binärvariablen betrachtet werden muß.

Verfahren zur operativen Produktionsplanung

Viele in der Literatur vorgestellte Verfahren zur Reihenfolgeplanung verwenden eine Modellierung mit gemischt-ganzzahliger Formulierung. Diese werden dann mit Optimierungssystemen⁵⁸ wie Xpress-MP und den darin implementierten Branch- & Bound-Verfahren (AHLEFF, [Ah95]) oder durch eigene Heuristiken gelöst. So entwickelt BLÖMER [Blö99] verschiedene Heuristiken, bei denen durch Einfügen zusätzlicher Restriktionen die Anzahl der Binärvariablen reduziert wird. BURKHARD ET AL. [BKR98] betrachten in ihrem Lösungsansatz die Möglichkeit ausgehend von der Lösung der LP-Relaxation⁵⁹, durch Auf- oder Abrunden einen Produktionsplan zu erstellen. In einer weiteren Arbeit entwickeln BURKHARD ET AL. [BH+98] einen auf Greedy-Heuristiken⁶⁰ basierenden Lösungsansatz.

In jüngeren Lösungsansätzen werden dagegen andere Formulierungen oder Optimierungsverfahren verwendet. TRAUTMANN [Tra01] beschreibt das Problem der Reihenfolgeplanung mit Hilfe der ressourcenbeschränkten Projektplanung. Dieser Ansatz basiert auf der Klassifikation von BRUCKER ET AL. [BD+99], in der die verschiedenen, bisher isoliert betrachteten Komponenten der Projektplanung berücksichtigt werden. Bei dieser Vorgehensweise werden alle Betriebsmittel und Materialien als erneuerbare oder nicht-erneuerbare Ressourcen beschrieben. Diese Formulierung ermöglicht es TRAUTMANN, auch den Einsatz der Mitarbeiter, für Reinigungen und Umrüstungen, etwa in der Planung zu berücksichtigen [Tra01].

LÖHL [Lö04] verwendet Genetische Algorithmen, um eine Feinplanung der Chargen durchzuführen. Auch in den Arbeiten von ANSCHÜTZ [Ans01] und KIESSWETTER [Kie99] wird das Problem der Reihenfolgeplanung mit Meta-Heuristiken gelöst. Sie verwenden das Simulated Annealing und das Threshold Accepting als Verbesserungsverfahren innerhalb einer Optimierung mit Evolutionären Algorithmen. Das Modell selbst wird dabei als gemischt-ganzzahliges Problem als State-Task- bzw. Resource-Task-Netzwerk formuliert.

⁵⁸Im Abschnitt *Software und Tools zur Optimierung* auf S. 68 wird eine Übersicht von „state-of-the-art“ Optimierungssystemen gegeben.

⁵⁹ Bei der LP-Relaxation eines als MIP formulierten Modells wird eine Lösung unter Mißachtung der Ganzzahligkeitsbedingungen gesucht.

⁶⁰Die Vorgehensweise von Greedy-Heuristiken wird in Fußnote 33 auf S. 55 beschrieben.

Die Durchführung einer Reihenfolgeplanung setzt jedoch immer voraus, daß die für die Reihenfolgeplanung erforderlichen Chargen in einem hierarchischen Planungsansatz zuvor oder in einem simultanen Planungsansatz gleichzeitig berechnet wurden. Die in der Literatur vorgestellten Modelle und Verfahren sind daher danach zu unterscheiden, ob eine feste Anlagenkonfiguration betrachtet wird und damit die Chargengrößen vorgegeben sind, oder ob das Modell die Chargengrößen zusätzlich berechnet. Die Vorgabe bestimmter Chargenzahlen kann zu zeitlich suboptimalen Ablaufplänen führen, während die Festlegung der Chargenvolumina sogar in der Unlösbarkeit der Reihenfolgeplanung resultieren kann ([Ans01], S. 167).

Eine feste Anlagenkonfiguration zur Herstellung von Polystyrol wird in der von SCHULZ [Schz02] beschriebenen Modellierung einer Mehrproduktanlage beschrieben. In dem MILP-Ansatz von CERDA ET AL. [CHG97] werden dagegen die Chargen und Chargengrößen berechnet. Das Modell ist jedoch stark eingeschränkt auf Problemstellungen, bei denen das Auftragsvolumen auf ein ganzzahliges Vielfaches einer maximalen Chargengröße festgelegt wird.

Das von KONDILI ET AL. [KPS93] entwickelte State-Task-Netzwerk beinhaltet kontinuierliche Entscheidungsvariablen, die das Volumen der Chargen in Abhängigkeit von dem Lagerbestand der benötigten Zutaten zum Produktionsstart jeder Charge bestimmen. Bei Anwendung dieser Modellformulierung treten aber die bereits in den vorherigen Abschnitten genannten Schwächen auf.

Ein weiterer Aspekt, der nur in wenigen Modellen explizit berücksichtigt wird, ist die Planung der Abfüllung nach dem letzten Produktionsschritt in der Herstellung eines Endproduktes. Hier gibt lediglich KIESSWETTER [Kie99] Beispiele von Produktionsprozessen an, bei denen zusätzlich die Abfüllung der Produkte in der Reihenfolgeplanung berücksichtigt wird. In den weiteren untersuchten Modellen werden die Endprodukte jeweils in einem Silo mit ausreichender Kapazität zwischengelagert, so daß die Abfüllung separat geplant und durchgeführt werden kann.

3.4.2 Technische Aspekte der Entwicklung eines Planungsverfahrens

Bei der Entwicklung eines Planungsverfahrens zur operativen Produktionsplanung sind neben den inhaltlichen Anforderungen auch die technischen Aspekte einer Umsetzung zu berücksichtigen. Diese sollen in diesem Abschnitt näher betrachtet werden.

Software und Tools zur Optimierung

Die Erstellung eines gemischt-ganzzahligen Optimierungsmodells erfordert hohe Präzision bei der mathematischen Formulierung. Schon eine kleine Änderung von Daten der Produktionsanlage oder in einer Rezeptur kann zu weitreichenden Änderungen in dem Modell führen. Ebenso ist zu prüfen, ob ein entwickeltes Lösungsverfahren sich stabil gegenüber Änderungen verhält.

Zum einfacheren Beschreiben und Erstellen von Optimierungsmodellen der Prozeßindustrie haben ZENTNER ET AL. die Sprache RCSpec entwickelt⁶¹ [ZE+98]. Dafür werden die

⁶¹Dem Autor sind allerdings keine Weiterentwicklungen und Anwendungen der Modellierungssprache

drei Hauptkomponenten eines Reihenfolgeproblems (Produktionsanlage, Produktionsprozesse und Materialien) entwickelt. Ein Parser⁶² übersetzt das Modell dann in die entsprechende mathematische Formulierung.

Weitere Verbreitung hat allerdings die Modelliersprache OPL⁶³ gefunden, die von der Softwarefirma ILOG in Verbindung mit dem mathematischen Optimierungspaket CPLEX angeboten wird [Ski01]. OPL bietet zugleich die Möglichkeit, Probleme des Constraint Programming (siehe Kapitel 3.2.2) zu formulieren. Für Reihenfolgeprobleme hat ILOG zusätzlich den ILOG Scheduler entwickelt. Die Software von ILOG wird u.a. von den Anbietern für E-Business Software Siebel und i2 Technologies verwendet [ILOG05].

Eine Solver-unabhängige Modellierungssprache ist die von FOURER ET AL. entwickelte Sprache AMPL zur Erstellung linearer und nichtlinearer Optimierungsmodelle [FGK90] und für das Constraint Programming [FG02]. Diese übersetzt das Optimierungsmodell in das allgemein verwendete Format MPS. Eine direkte Anbindung von AMPL wurde u.a. für die Solver ILOG CPLEX, IBM OSL und Xpress-MP von Dash geschaffen [AMPL05].

Das Optimierungspaket Xpress-MP besitzt mit Mosel eine eigene Modellierungssprache [Hei02]. Der Vorteil solcher Sprachen liegt in der Trennung zwischen Modell und Daten. Diese können dann aus speziell formatierten Textdateien oder Datenbanken problemspezifisch eingelesen werden. Ein Interpreter übergibt dem Solver des Optimierungspakets dann die Matrix und Zielfunktion des Optimierungsproblems. Xpress-MP wird zur Optimierung von den Anbietern für Supply Chain Management Software Intentia, Aspen Technology und Process Systems Enterprise (PSE) verwendet [Dash05]. Die beiden letzteren haben sich dabei mit dem Aspen Plant Scheduler bzw. OSS Scheduler von PSE besonders auf den Bereich der Prozeßindustrie spezialisiert [MC+05]. Der OSS Scheduler verwendet die im vorigen Abschnitt beschriebene STN- und RTN-Formulierung eines MILP-Modells.

Modellierung der Reihenfolgeplanung

Die in Kapitel 3.2 beschriebenen Möglichkeiten der Reihenfolgemodellierung sollen nun für die weitere Verwendung evaluiert werden.

Gemischt-ganzzahlige Programmierung: Modelle der gemischt-ganzzahligen Programmierung sind weit verbreitet und können von verschiedensten Standard-Solvern gelöst werden. Allerdings gewährleistet die Existenz von Standardalgorithmen nicht, daß eine Lösung in schneller Zeit gefunden wird. Dies hängt insbesondere von der Anzahl der Entscheidungsvariablen ab.

Ein Reihenfolgeproblem kann in den verschiedensten Varianten modelliert werden. Diese sind häufig nicht leicht nachvollziehbar, da je nach Variante zusätzliche Hilfsvariablen definiert werden müssen, die die Reihenfolgebeziehungen darstellen. Sie können daher bei Änderungen oder Erweiterungen des Ausgangsproblems nur mit großem Aufwand angepaßt werden.

von ZENTNER ET AL. von bekannt. Daher kann hier keine weitere Aussage über deren Praktikabilität getroffen werden.

⁶²Ein Parser ist ein Programm, das einen, in einer Programmier- oder Modellierungssprache geschriebenen Text in eine Struktur übersetzt, die eine Weiterverarbeitung der gegebenen Daten und Anweisungen ermöglicht.

⁶³In Abb. 3.3 auf S. 48 wird ein Job Shop-Problem unter Verwendung von OPL modelliert [VM+99].

Constraint Based Programming: Um ein Reihenfolgeproblem anhand des Constraint Based Programming zu modellieren und zu lösen, ist dem Autor bisher nur das kommerzielle System CPLEX mit der Modellierungssprache OPL bekannt.

Die Ergebnisse und Laufzeiten des Constraint Based Programming sind noch nicht von gleicher Qualität wie bei Verfahren der mathematischen Optimierung. Hier ist aber zu erwarten, daß in Kombination mit bewährten heuristischen Suchverfahren für Probleme der Reihenfolgeplanung akzeptable Ergebnisse erreicht werden. Ein besonderer Vorteil des CP liegt allerdings in der leichten Nachvollziehbarkeit für einen Produktionsplaner und in der Implementierung, wenn die Modellierungssprache OPL verwendet wird.

Graphentheoretische Modellierung: Bei einer Modellierung des Reihenfolgeproblems mit gerichteten Graphen kann eine Lösung einfach in ein GANTT-Diagramm übernommen und dort leicht nachvollziehbar überprüft werden. Bei der Implementierung können dabei in beliebiger Weise benutzerspezifische Anforderungen hinzugefügt werden.

Zur Lösung von graphenbasierten Modellen werden keine kommerziellen Optimierungsprogramme benötigt. Stattdessen kommen hier heuristische Verfahren zur Anwendung, die getrennt von der Modellierung entwickelt werden können.

Nach diesen Betrachtungen wird eine Modellierung des Reihenfolgeproblems mit Hilfe der Graphentheorie bevorzugt. Dazu wird mit Hilfe der Programmiersprache C++ ein entsprechendes Programm entwickelt.

Kapitel 4

Reihenfolgeplanung in der Chargenproduktion

In der Reihenfolgeplanung wird für eine vorgegebene Menge von Operationen eine Abfolge gesucht, so daß das gegebene Planungsziel¹ optimiert wird. Die *Tagesplanung* bezeichnet nun das Problem der Reihenfolgeplanung für eine vorgegebene Menge von Chargen an einem Tag des Planungszeitraums. Da hierbei nur eine einzelne Planungsperiode betrachtet wird, wird als Optimierungsziel die Minimierung des **Makespan** verfolgt.

Die verfahrenstechnischen Anforderungen des Optimierungsproblems wurden in Kapitel 2 beschrieben. Diese Anforderungen werden in der Modellierung der Tagesplanung entsprechend berücksichtigt. Eine zulässige (aber nicht unbedingt optimale) Lösung des verfahrenstechnischen Problems der Ablaufreihenfolge wird dabei als *Instanz* bezeichnet.

Nachfolgend wird daher zunächst die Modellierung einer Instanz als Graph vorgestellt und das Verfahren zum Erzeugen einer Ausgangslösung beschrieben. Für die vorliegende Modellierung wird anschließend ein **Makespan**-Algorithmus entwickelt, der die Startzeitpunkte der einzelnen Operationen der Produktionsprozesse berechnet. Eine Verbesserung der Ausgangslösung kann dann durch Verschieben in der Reihenfolge der Prozesse oder Einfügen auf anderen Reaktoren erreicht werden.

4.1 Ein Optimierungsproblem der Reihenfolgeplanung

Nachfolgend wird das in dieser Arbeit betrachtete Optimierungsproblem der Reihenfolgeplanung in der Chargenproduktion näher beschrieben. Dieses wird basierend auf dem bereits in Kapitel 1.2.2 beschriebenen Klassifikationsschema von GRAHAM ET AL. [GL+79] als sog. *Connected Job Shop-Problem*² beschrieben.

¹Hierzu wurden in Kapitel 1.2.2 bereits verschiedene Optimierungsziele beschrieben.

²Der Begriff des *Connected Job Shop-Problem* wird eingeführt, um die Reihenfolgebeziehungen der Chargenproduktion dieser Arbeit definieren zu können.

Anlagen-Charakteristik: Gegeben ist eine Mehrzweckanlage, von der die folgenden Anlagekomponenten³ in der Modellierung berücksichtigt werden.

- Die Reaktoren einer Produktionsanlage, in denen die Produktion erfolgt.
- Ein Pool von Lagerkapazitäten (*Silos*) zur Lagerung von Zwischenprodukten: Die Silos sind nicht einzelnen Reaktoren zugeordnet, sondern durch ein Rohrleitungssystem mit allen Reaktoren zur Einlagerung und Entnahme verbunden.
- Die Anlagekomponenten zur Abfüllung der Endprodukte: Eine Abfüllanlage hat mehrere Abfüll-Stutzen, die unabhängig voneinander einzelne Abfüllaufträge durchführen können. Jeder Abfüll-Stutzen kann aufgrund des Rohrleitungssystems von jedem Reaktor zur Abfüllung verwendet werden. Die Abfüllgeschwindigkeit aller Abfüll-Stutzen einer Abfüllanlage ist gleich.

Eine Abfüllanlage kann nur Abfüllungen eines Typs durchführen. Es sind die beiden Abfüllarten möglich: Abfüllung in einen LKW oder Abfüllung in ein Gebinde.

Eine flexible Verrohrung der Mehrzweckanlage ermöglicht einen Wechsel zwischen den einzelnen Produktionsstufen für alle Reaktoren, Silos und Abfüll-Stutzen. Die Produktion kann daher in Misch-Fahrweise mit seriellen und parallelen Produktionsabläufen geplant werden.

Die Auswahl des Reaktors, der für die Produktion einer Charge verwendet werden kann, wird durch die Funktionalitäten bestimmt, die in der Herstellungsanleitung des jeweiligen Urrezeptes gefordert werden. Eine Charge mit vorgegebenem Volumen kann daher auf allen Reaktoren gefertigt werden, deren Kapazitäten das gegebene Volumen verarbeiten können.

Einschränkungen bezüglich der Fertigungsgeschwindigkeiten der einzelnen Reaktoren werden in diesem Modell nicht betrachtet. Die (eentlichen) Produktionszeiten sind also insbesondere maschinen- und volumenunabhängig.

Eine Charge kann bis zur weiteren Verarbeitung in dem herstellenden Reaktor verbleiben, sofern dadurch keine weitere Produktion blockiert wird. Ansonsten muß eine Zwischenlagerung in einem Silo stattfinden. Sofern es die Rezeptur einer Charge zulässt, können mehrere Chargen des gleichen (Zwischen-)Produktes zeitgleich in einem Silo gelagert werden.

In dieser Arbeit wird ein Modell entwickelt, das für eine Reihenfolgeplanung mit beliebig vielen produzierenden Anlagekomponenten eingesetzt werden soll. Ebenso ist die Anzahl der Produktionsstufen nicht weiter festgelegt⁴. Der Begriff der Produktionsstufe ist dabei für diese Arbeit folgendermaßen definiert:

³Da die ganze Produktionsanlage betrachtet wird, wird hier die Bezeichnung *Anlagen-Charakteristik* und nicht *Maschinen-Charakteristik* verwendet. Zur Beschreibung der Anlagekomponenten vgl. Kapitel 2.2.2.

⁴Das Modell wurde ursprünglich zur Planung maximal 3-stufiger Produktionsprozesse entwickelt. Bei diesen tritt i.d.R. keine Produktion von Nebenprodukten durch Kuppelproduktion oder zyklische Materialflüsse auf.

Definition 4.1 *Die Herstellung einer einzelnen Charge ohne die Produktion von Zwischen- oder Vorprodukten wird als Produktionsstufe bezeichnet.*

Die Produktionstiefe einer Charge zählt die Anzahl der Produktionsstufen, die nachfolgen bis das hergestellte Produkt in einem Endprodukt verarbeitet wird.

Ein n-stufiger Produktionsprozeß bezeichnet den um eins erhöhten Wert der maximalen Produktionstiefe aller Chargen, die als Vor- oder Zwischenprodukt benötigt werden.

Eine Endprodukt-Charge hat demnach die Produktionstiefe 0. Wenn keine weiteren Zwischenprodukte für die Herstellung benötigt werden, dann ist dies ein einstufiger Produktionsprozeß. Die Werte von Produktionstiefe und n-stufigem Produktionsprozeß werden also rekursiv von der Herstellung einer Endprodukt-Charge bestimmt. Sie zählen lediglich in umgekehrter Richtung.

Als wichtige Besonderheit dieses Modells ist zu berücksichtigen, daß nicht nur der eigentliche Produktionsprozeß geplant und optimiert wird, sondern auch die Abfüllung der Endprodukte berücksichtigt wird. Dies setzt eine rein auftragsorientierte⁵ Produktionsplanung voraus.

Auftrags-Charakteristik: Gegeben ist eine Menge von Produktionsaufträgen zur Herstellung von Endprodukten, für die im Rahmen der Feinplanung (*Tagesplanung*) eine optimale Reihenfolge gefunden werden soll. Diese Auftragsmenge \mathcal{J} ist gleich zu setzen mit der Menge der Endprodukt-Chargen und zugleich Teilmenge aller Chargen CH . Basierend auf diesem Betrachtungsstand ist die Menge der Operationen O_i eines Auftrags $i \in \mathcal{J}$ somit die Menge aller Vor- und Zwischenprodukt-Chargen, sowie die Endprodukt-Charge selber.

Alle Produktionsaufträge (Chargen) werden in einem vorgelagerten Algorithmus zur Volumenplanung aus den vorhandenen Endprodukt-Aufträgen zusammengestellt. Diese Planung bestimmt die Anzahl und das Volumen der einzelnen Chargen. Im Falle von Endprodukt-Chargen werden in Abhängigkeit der verwendeten Abfülltypen (LKW, Gebinde) in der Reihenfolgeplanung zusätzlich entsprechende Abfüllaufträge erzeugt.

Die gesamte Herstellungs- und Lebensdauer einer einzelnen Charge kann (wiederum) in einzelne Operationen oder Prozeßschritte aufgesplittet werden, die in der Reihenfolgeplanung berücksichtigt werden müssen:

- Die eigentliche Produktionsdauer der Charge. Diese ist volumen- und maschinenunabhängig. Eine Unterbrechung der Produktion ist nicht möglich.
- Die Wartezeit bis zur Weiterverarbeitung in einem Nachfolgeprodukt oder bis zur Abfüllung als Endprodukt. Diese oder auch die Lagerdauer sind abhängig von den Produktionszeitpunkten der Nachfolgeprodukte. Dieser Wert ist nicht selber eine Eingangsgröße, sondern ergibt sich aus der bestehenden Reihenfolge.
- Die Transferdauer zwischen den einzelnen Anlagekomponenten zur Weiterverarbeitung in einem Nachfolgeprodukt oder zur Abfüllung. Ein Transfer zwischen den Anlagekomponenten darf nicht unterbrochen werden.

⁵Zu einer *Produktion auf Lager* siehe den Abschnitt *Praxis: Implementierung in einem Planungssystem* über die sog. FTL-Abfüllung auf S. 84.

Die Transferdauer ist volumenabhängig. Sie bestimmt sich aus der Transfer- und Abfüllgeschwindigkeit. Diese ist maschinenunabhängig.

- Eine Liegezeit des Produktes wie in Kapitel 2.2.3 beschrieben wird nicht benötigt.

In dem betrachteten Optimierungsmodell wird eine reine Batchproduktion angenommen. In dem Prozeßschritt der Produktion einer Charge wird daher nur ein Reaktor verwendet. Er ist dann jeweils für die Bearbeitung durch andere Chargen *gesperrt*. Bei Transferprozessen sind dagegen jeweils zwei Anlagekomponenten für die Bearbeitung durch eine Charge gesperrt.

Die Herstellungsabfolge der einzelnen Chargen bedingt sich aus der Beziehung der Chargen untereinander (Vorprodukte, Zwischenprodukte, Endprodukte). Eine Charge wird also von den direkt vorgeschalteten Zwischenprodukt-Chargen „*bedient*“ und ist zugleich Zwischenprodukt für die direkt nachgeschalteten Chargen. Die Produktionszeitpunkte dieser Chargen sind durch Ende-Start-Beziehungen mit möglichen Offset-Zeiten miteinander verknüpft. Im Falle von JiT-Produktionen⁶ sind dies sogar feste Beziehungen, bei denen keine Lagerung stattfindet. Die Transfer- und Abfüllprozesse sind durch Start-Start-Beziehungen miteinander verknüpft.

Aufgrund der Volumenplanung kann eine Charge als Zwischenprodukt für **mehrere** weitere Chargen dienen. Es besteht daher keine eindeutige Reihenfolgebeziehung in Form eines gerichteten Baumes auf eine Endprodukt-Charge hin. Daher wird das Problem auch als *Connected Job Shop-Problem* bezeichnet.

Zwischen den einzelnen Herstellungsprozessen sind reihenfolgeabhängige Reinigungen der Anlagekomponenten durchzuführen. Im Gegensatz zu den Herstellungsprozessen, für die eine (im wesentlichen) automatische Produktionssteuerung durch ein zentrales Prozeßleitsystem angenommen wird, werden bei der Reinigung Mitarbeiter des Produktionsbetriebes eingesetzt und damit zeitlich gebunden. Das Optimierungsproblem berücksichtigt daher den Einsatz der Mitarbeiter als weitere Restriktion:

- *Es dürfen zeitgleich nicht mehr Reinigungsprozesse stattfinden, als Mitarbeiter hierfür zur Verfügung stehen.*

Dabei wird für die Entwicklung von Optimierungsmodell und -verfahren zu Grunde gelegt, daß eine Reinigung jeweils durch einen Mitarbeiter durchgeführt wird.

Zielsetzung: Die Entscheidung über die Durchführung eines Produktionsprogramms erfolgt mit dem Ziel der Gewinnmaximierung bzw. Kostenminimierung ([DoS03], S. 100). Hierzu müssen die Kostentreiber und -faktoren des Optimierungsproblems ermittelt werden. Da diese jedoch nur schwer zu quantifizieren sind, werden oftmals zeitbezogene Ersatzziele verwendet.

Das beschriebene Optimierungsproblem beschreibt die Feinplanung, um die Reihenfolge für jeweils einen Tag zu ermitteln. Innerhalb dieser Planung müssen keine Termine zur Auftragsfreigabe oder Fertigstellung der Chargen berücksichtigt werden. Es erscheint daher plausibel, daß es in der Reihenfolgeplanung zunächst genügt, für die vorhandenen

⁶ Abweichend von der Bezeichnung *no wait*-Prozeß wird in dieser Arbeit der Begriff JiT-Produktion (= just-in-time) der Logistik verwendet, um auch den durch eine Offset-Zeit verschobenen Zeitpunkt einer weiteren Verarbeitung zu berücksichtigen.

Aufträge eine zulässige Lösung zu finden. Die verfügbare Produktionszeit (inkl. möglicher Überstunden) darf dabei nicht überschritten⁷ werden. Terminorientierte Ziele müssen auf dieser Ebene der Produktionsplanung daher ebensowenig betrachtet werden wie durchlaufzeitbezogene⁸ Ziele. Für die Reihenfolgeoptimierung kommen daher die folgenden Zielfunktionen in Betracht.

Definition 4.2 Der lokale Makespan $\text{MKS}(u)$ einer Anlagekomponente u ist der Fertigstellungszeitpunkt des letzten Prozeßschritts, der auf dieser Anlagekomponente durchgeführt wird.

Der (globale) Makespan oder auch die Zykluszeit einer Anlage mit den Anlagekomponenten \mathcal{U} ist dann der Wert

$$\text{Makespan} := \max \{ \text{MKS}(u) \mid u \in \mathcal{U} \}$$

Es ist leicht ersichtlich, daß diese Definition des Makespan lediglich eine anlagenbezogene Formulierung der in Kapitel 1.2.2 gegebenen Definition ist⁹. Für das vorliegende Optimierungsproblem wird aufgrund der obigen Überlegungen eine *multi-kriterielle* Zielfunktion definiert ([DD95], S. 48 ff.).

Hauptziel des Optimierungsproblems ist die

- *Minimierung des Makespan*

Zusätzlich wird das folgende **Nebenziel** verwendet:

- *Minimierung der Summe der Endzeitpunkte je Reaktor:*

$$\min \sum_{u \in UR} \text{MKS}(u)$$

Hier ist UR die Menge aller Anlagekomponenten für die Produktion (d.h. Reaktoren) der Produktionsanlage.

Das Hauptziel dient vornehmlich dazu, eine zulässige Lösung innerhalb der verfügbaren Zeit zu finden¹⁰. Bei einer durchgängigen Produktion im 3-Schicht-Betrieb ist hierbei jedoch keine Unterscheidung zwischen zwei Lösungen möglich. Hier gewährleistet das Nebenziel, daß zwei Lösungen miteinander verglichen werden können. Ebenso dient dieses Nebenziel dazu, die Leerlaufzeiten zu minimieren und damit eine bessere Auslastung der Reaktoren in der simultanen Volumen- und Reihenfolgeplanung zu erreichen. Diese Problematik wird später in Kapitel 5 genauer erläutert.

⁷Eine Ausnahme bildet hier der 3-Schicht-Betrieb mit durchgängiger Produktion.

⁸Die Gesamtdurchlaufzeit eines Auftrags muß in einer rollierenden Planung z.B. dann berücksichtigt werden, wenn ein Auftrag eine Bestellung für einen Rohstoff erzeugt, und damit eine Kapitalbindung erfolgen würde.

⁹Siehe dazu die Beschreibung zur *Zykluszeit* auf S. 13 f.; Für die nachfolgende Modellierung mit einem Graphen wird auch auf den Zusammenhang zwischen dem Makespan und dem kritischen Pfad in einem Graphen hingewiesen; vgl. hierzu die Anmerkungen nach Definition 3.3 auf S. 52 f.

¹⁰Wenn der Personaleinsatz im Rahmen eines Stundenkontingents des jeweiligen Mitarbeiters variabel ist, können hier durch Minimierung des Makespan auch die Personalkosten gesenkt werden.

4.2 Modellierung des Reihenfolgeproblems

Zur Modellierung und Optimierung werden in dieser Arbeit die Eigenschaften des disjunktiven Graphenmodells verwendet. Wenn eine Instanz des Reihenfolgeproblems eine zulässige Lösung ist, so kann der Ablaufplan dann als ressourcenorientiertes oder auftragsorientiertes GANTT-Diagramm dargestellt werden (vgl. Kapitel 3.2.3).

Für die Modellierung des betrachteten Problems wurde eine ressourcenorientierte Darstellung gewählt, da in dem Ablaufplan mit den Reinigungsaufträgen zwischen einzelnen Produktionsschritten weitere Operationen gegeben sind, die in einem auftragsorientierten Diagramm nicht dargestellt werden können.

Zu der weiter gefassten Menge der Ressourcen werden neben den Reaktoren auch die Silos für die Zwischenlagerung, die Abfüllanlagen (hier im besonderen die Abfüllstutzen) und die Mitarbeiter für die Reinigungsanlagen gezählt. In der weiteren Beschreibung des Algorithmus wird hier auch von *Scheduling-Units* gesprochen.

Aufgrund der in den Rezeptvorgaben gespeicherten verfahrenstechnischen Anforderungen können die Chargen nur auf vorgegebenen Reaktoren gefertigt und auf bestimmten Silos zwischengelagert werden. Dagegen kann eine Reinigung von jedem Mitarbeiter durchgeführt werden und jedes in ein Gebinde (in einen LKW) abzufüllende Volumen auf einem Abfüllstutzen für Gebinde (für LKW) abgefüllt werden.

In der Modellierung und der Berechnung des Reihenfolgeproblems wird daher unterschieden zwischen Ressourcen, bei denen die Operationen einer Instanz fest zugeordnet sind und Ressourcenpools, auf denen die Zuordnung und Reihenfolge erstmalig in der Berechnung des **Makespan** festgelegt wird.

4.2.1 Modellierung der Reihenfolge auf Reaktoren

Auf einem Reaktor finden zwei verschiedene Operationsarten statt:

- *auftragsbezogene Operationen* (Produktionsaufträge), also die Herstellung, Zwischenlagerung und der Transfer von Chargen zwischen den Maschinen, die in dieser Arbeit mit [PROD] gekennzeichnet werden *und*
- *planungsbezogene Operationen*, also Rüst- und Reinigungszeiten zwischen den einzelnen Produktionsaufträgen, die nachfolgend [CLEAN] bezeichnet werden.

Zusätzlich zu den Produktionsaufträgen mit echten Inputdaten werden auf jedem Reaktor zwei Dummy-Aufträge \circ für den Produktionsbeginn und $*$ für das Produktionsende als weitere Knoten in der Reihenfolge eingefügt. Somit ist auf jedem Reaktor die Abfolge

$$\circ \rightarrow [\text{CLEAN}] \rightarrow [\text{PROD}] \rightarrow [\text{CLEAN}] \rightarrow \dots \rightarrow [\text{CLEAN}] \rightarrow *$$

vorhanden. [CLEAN] und [PROD] repräsentieren hierbei *Hyperknoten*, die in entsprechende Knotenketten expandiert werden.

Da in dem hier beschriebenen Modell keine Rüstzeiten berücksichtigt werden, wird [CLEAN] nur durch den Knoten kn_{CLEAN} ersetzt. kn_{CLEAN} repräsentiert hierbei den Knoten zur Reinigung des Reaktors.

Abhängig von der Produktionsstufe¹¹ der Charge, wird der Knoten [PROD] in unterschiedliche Knotenketten expandiert. Dazu werden die folgenden Knotentypen definiert: kn_{PROD} repräsentiert die eigentliche Produktion der Charge, kn_{WARTE} die maximale Dauer einer zwischengelagerten (Teil-)Menge der Charge auf dem Reaktor vor einer weiteren Verarbeitung oder dem Transfer in ein Silo zur Zwischenlagerung und kn_{TRANSFER} beschreibt den Transfer der Charge auf eine andere Ressource. Die Lagerung auf dem Reaktor kn_{WARTE} wird nachfolgend auch als *Liegezeit auf dem Reaktor*¹² bezeichnet.

Die Dauer $t(kn_{\text{PROD}})$ wird hierbei durch die Rezeptur vorgegeben, die Dauer von kn_{WARTE} in dem **Makespan**-Algorithmus berechnet und $t(kn_{\text{TRANSFER}})$ ist abhängig vom Volumen und der Pumpleistung der Produktionsanlage (vgl. Kapitel 2.2.2).

Modellierung bei Zwischenprodukt-Chargen

Wenn mit der Charge ein Zwischenprodukt hergestellt wird, wird [PROD] in die Knotenkette $kn_{\text{PROD}} \rightarrow kn_{\text{WARTE}} \rightarrow kn_{\text{TRANSFER}}$ expandiert.

Modellierung bei Endprodukt-Chargen

Wenn mit der Charge ein Endprodukt hergestellt wird, dann muß zusätzlich die Abfüllung in Gebinde oder LKW berücksichtigt werden. Die Auswahl der Abfüllung richtet sich nach den Kundenaufträgen, die in einer Charge zusammengefasst sind. Alternativ zu einer Abfüllung aus einem Reaktor heraus könnte eine Charge auch in einem Silo zwischengelagert und von dort aus abgefüllt werden. Diese zusätzliche Entscheidungsvariante wird aber in der nachfolgenden Modellierung nicht weiter berücksichtigt, um die Anzahl der Lösungen durch Erhöhung der Freiheitsgrade in der Auswahl einer Abfüllvariante nicht weiter zu vergrößern.

Die Abfüllungen der beiden Abfüllarten können parallel stattfinden¹³. Dementsprechend muß das in Abb. 4.1 dargestellte Netzwerk implementiert werden.

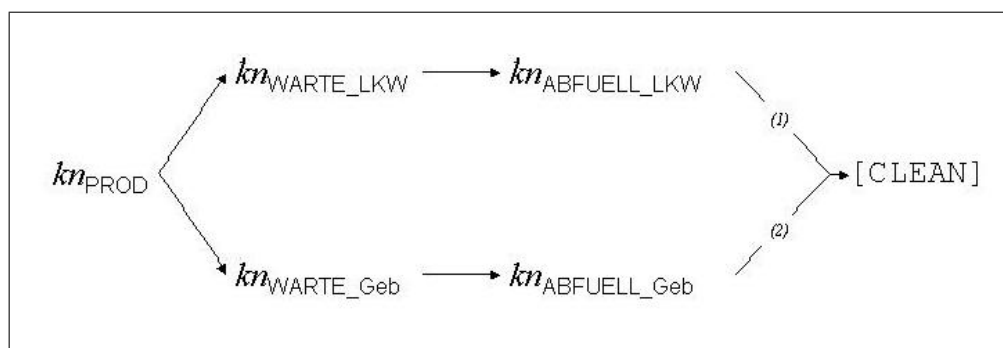


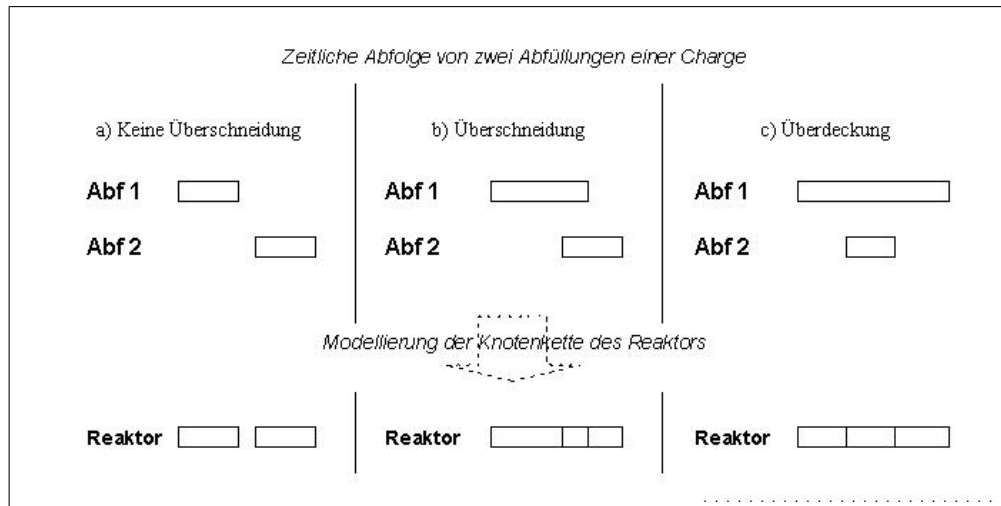
Abbildung 4.1: Parallele Operationen nach einer Endprodukt-Produktion

¹¹Endprodukt oder Zwischenprodukt

¹²Die *Liegezeit auf dem Reaktor* ist nicht zu verwechseln mit der allgemeinen *Liegezeit einer Charge* im Produktionsverlauf wie sie in Kapitel 2.2.3 auf S. 30 f. beschrieben wurde. Letztere wird in dieser Arbeit nicht betrachtet.

¹³Vergleiche *Annahmen für das Optimierungsmodell bei der Abfüllung* in Kapitel 2.2.2

Da in einem maschinenorientierten GANTT-Diagramm jedoch keine parallelen Prozesse auf einer Maschine dargestellt werden können, muß dieses Netzwerk in eine entsprechende Knotenkette umgewandelt werden. Die beiden Prozesse der Abfüllung können zeitlich nacheinander stattfinden, sich überschneiden, oder ein Abfüllprozeß kann komplett während der Dauer einer länger dauernden Abfüllung stattfinden (vgl. Abb. 4.2).



Abf 1 bezeichnet die zuerst startende Abfüllung. Bei gleichem Startzeitpunkt für beide Abfüllungen ist dies die Abfüllung, die zuerst beendet wird. Die zeitliche Abfolge von zwei Abfüllungen einer Charge wird dann in der Knotenkette des produzierenden Reaktors mit drei Knoten abgebildet. Im Fall (a) wird der dritte Knoten $kn_{ABFUELL}$ mit einer Dauer $t(kn_{ABFUELL}) = 0$ nicht dargestellt.

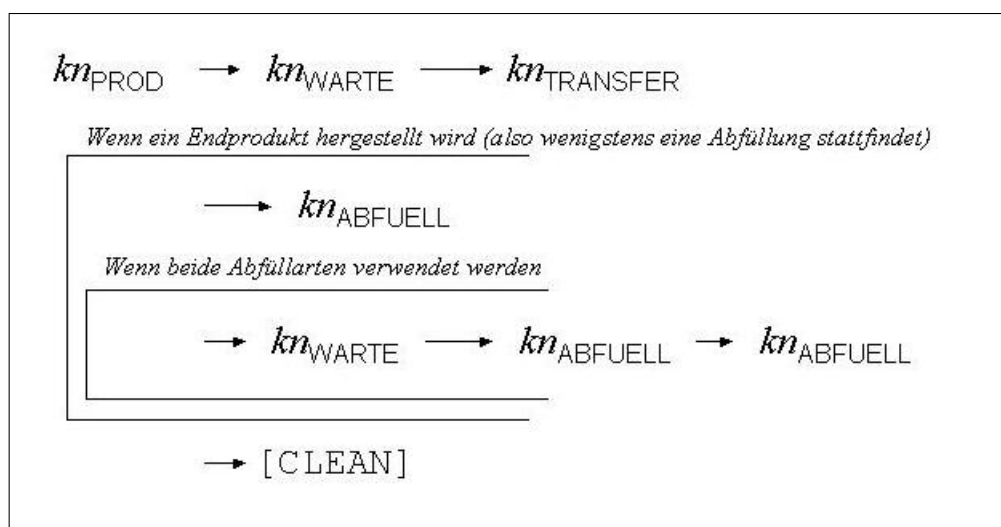
Abbildung 4.2: Umwandlung paralleler Prozesse in eine Knotenkette: Fallunterscheidung bei der Abfüllung

Bei der Einplanung einer Endprodukt-Charge in einer Scheduling-Instanz sind die jeweils benötigten Abfüllarten und damit die Anzahl der Abfüllungen aus der Zusammenstellung der Kundenaufträge zu einer Charge bereits bekannt. Der Knoten [PROD] (mit anschließender Reinigung [CLEAN]) wird unter Berücksichtigung des Netzwerkes aus Abb. 4.1 und in Abhängigkeit der Abfüllarten in die Knotenkette in Abb. 4.3 expandiert¹⁴.

Einplanung einer Charge

Generell kann ein Produktionsauftrag an jeder Stelle der Knotenkette eines Reaktors zwischen zwei Hyperknoten vom Typ [CLEAN] eingeplant werden. Wenn eine Charge direkt zu Beginn nach dem Knoten \circ oder direkt vor dem letzten Knoten $*$ eingeplant werden soll, muß hier evtl. eine zusätzliche Reinigung [CLEAN] durchgeführt werden. Ebenso muß berücksichtigt werden, daß die Dauer der reihenfolgeabhängigen Reinigungszeit nach dem Einfügen eines Produktionsauftrages vor und hinter dem Produktionsauftrag aktualisiert wird.

¹⁴Diese Vorgehensweise ist aufgrund der geringen Anzahl von höchstens zwei parallelen Abfüllungen in der Erstellung des Netzwerkes und bei der anschließenden Berechnung im Aufwand vertretbar. Zudem bietet die beschriebene Transformation die Möglichkeit, auch die Zeitpunkte der Überschneidungen der parallelen Prozesse zu bestimmen.



Abhängig von der Art des Produktes (Zwischenprodukt oder Endprodukt) und der Anzahl der benötigten Abfüllarten wird ein *Hyperknoten* [PROD] bei der Implementierung in unterschiedlich lange Knotenketten umgewandelt.

Abbildung 4.3: Umwandlung des Produktionsnetzwerkes aus Abb. 4.1 in eine Knotenkette

4.2.2 Modellierung der Reihenfolge auf Silos

Auf einem Silo finden wieder - wie bei einem Reaktor - zwei unterschiedliche Operationsarten statt:

- *auftragsbezogene Operationen*, also die Zwischenlagerung der Chargen, (gekennzeichnet mit [LAGER]) und
- *ressourcenbezogene Operationen*, also die Reinigung des Lagers (gekennzeichnet mit [CLEAN]).

Entsprechend ist auf jedem Silo die Abfolge

$$\circ \rightarrow [\text{CLEAN}] \rightarrow [\text{LAGER}] \rightarrow [\text{CLEAN}] \rightarrow \dots \rightarrow [\text{CLEAN}] \rightarrow *$$

gegeben mit \circ und $*$ als Dummy-Operationen (wie bei einem Reaktor).

Vor dem Einfügen eines Produktionsknotens auf einem Reaktor wird überprüft, ob das Chargenvolumen aufgrund der Reaktorvolumina¹⁵ *Sumpfmenge* und *Maximalvolumen* auch produziert werden kann. Bei der Zwischenlagerung von Chargen in einem Silo müssen dagegen zwei zusätzliche Bedingungen überprüft werden:

- Kann die Charge aufgrund von möglicherweise bereits gelagerten Chargen zusätzlich eingelagert werden? Hier ist das bereits lagernde Material zu ermitteln und bei Gleichheit anhand des Produktionsrezeptes zu entscheiden, ob ein Batch-Mix zulässig ist.

¹⁵Vgl. Abschnitt *Produktion* in Kapitel 2.2.2 auf S. 22 f.

- Ist bei einer Zulagerung ein Batch-Mix erlaubt, muß zusätzlich zu jedem Zeitpunkt das belegte bzw. das noch belegbare Volumen ermittelt werden.

Um die Zeitpunkte und Bestände der Ein- und Auslagerungen bestimmen zu können, wird daher für jeden Knoten $[\text{LAGER}]$ eine Knotenkette $kn_{\text{LAGER}}(1) \rightarrow \dots \rightarrow kn_{\text{LAGER}}(n)$ eingefügt. Die Anzahl n der Knoten vom Typ kn_{LAGER} ist dann abhängig von der Anzahl der Nachfolger-Chargen, die von der eingelagerten Charge bedient¹⁶ werden. Dies sei zunächst an einem einfachen Beispiel erläutert.

Beispiel 4.1 (Lagerung ohne Batch-Mix) *Wenn eine Charge drei Nachfolger-Chargen bedient, dann wird bei einer Zwischenlagerung der Hyperknoten $[\text{LAGER}]$ in die folgende Knotenkette expandiert:*

$$[\text{CLEAN}] \rightarrow kn_{\text{LAGER}}(1) \rightarrow kn_{\text{LAGER}}(2) \rightarrow kn_{\text{LAGER}}(3) \rightarrow [\text{CLEAN}]$$

Zur Startzeit von $kn_{\text{LAGER}}(1)$ wird die Charge komplett eingelagert. Zu den Endzeitpunkten aller Knoten $kn_{\text{LAGER}}(i)$ wird jeweils der Bedarf für eine der Nachfolger-Chargen ausgelagert. Die Dauer der Zwischenlagerung und Reihenfolge der Auslagerung wird in der Makespan-Berechnung bestimmt.

Bei einer Lagerung mit Batch-Mix entspricht die Anzahl der Knoten jedoch nicht der Anzahl der Auslagerungen. Die Expansion eines Knotens $[\text{LAGER}]$ in eine Knotenkette modelliert die Bestandsänderungen, die zu den einzelnen Zeitpunkten der Einlagerungen und Auslagerungen entstehen (vgl. Abb. 4.4).

Die eingelagerten Chargen chg_i haben, resultierend aus den bekannten Einlagerungszeitpunkten $st(chg_i)$, eine Ordnung $chg_1 < \dots < chg_n$. Eine Charge chg_i hat m_i nachfolgende Produktionen, die bedient werden. Abbildung 4.4.a stellt die parallel stattfindenden Einlagerungen und Auslagerungen mit jeweils eigener zeitlicher Abfolge dar.

Entsprechend der Anzahl der Produktionen, die bedient werden, bestehen zwischen den parallelen Knotenketten $\sum_{i=1}^n m_i$ Einlager-Auslager-Beziehungen über das Volumen der jeweils zwischengelagerten Chargen (Abb. 4.4.b).

Da die Zeitpunkte der Auslagerungen erst im Rahmen der Makespan-Berechnung bestimmt werden und (aufgrund des erlaubten Batch-Mix) somit Überschneidungen der Lagerdauer mehrerer Chargen bestehen können, sind die Bestandsbeziehungen zwischen den Auslagerungen einer Charge und jeder später eingelagerten Charge zu berücksichtigen (Abb. 4.4.c). Eine später eingelagerte Charge chg_k hat somit $\sum_{i=1}^{k-1} m_i$ zusätzliche Auslager-Einlager-Beziehungen.

Aus $[\text{LAGER}]$ wird somit eine Knotenkette $kn_{\text{LAGER}}(1) \rightarrow \dots \rightarrow kn_{\text{LAGER}}(n')$ mit $n' = \sum_{i=1}^n \sum_{k=1}^i m_k$ Knoten expandiert, die alle möglichen Bestandsbeziehungen darstellen (Abb. 4.4.d). Die Dauer jedes Knotens wird erst in der Makespan-Berechnung bestimmt. Aufgrund mehrerer Auslagerungen oder Überschneidungen zwischen Einlagerungen und Auslagerungen kann die Dauer eines einzelnen Knotens geringer als die Gesamtdauer der Lagerung sein bzw. die Dauer 0 haben.

¹⁶Man spricht davon, daß eine Charge chg_1 eine andere (nachfolgende) Charge chg_2 „bedient“, wenn chg_1 ein Zwischenprodukt in der Herstellung von Charge chg_2 ist.

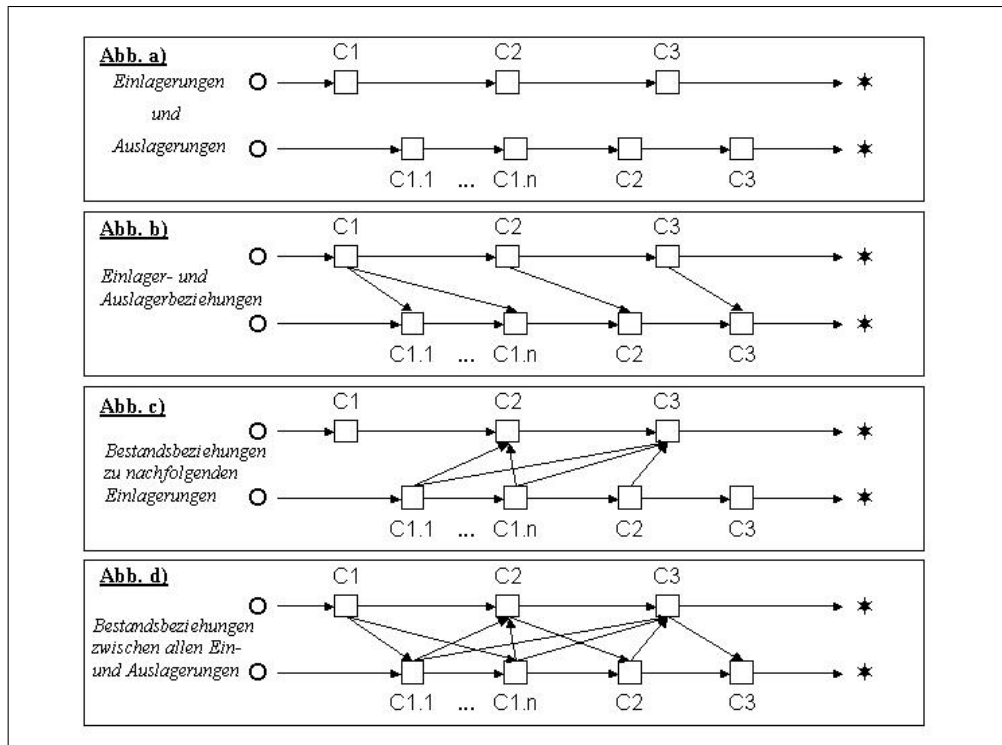


Abbildung 4.4: Silo Bestandskette

Dies sei an dem nachfolgenden Beispiel noch einmal näher erläutert.

Beispiel 4.2 (Lagerung mit Batch-Mix) Es werden zwei Chargen A und B mit 4 bzw. 3 Nachfolger-Chargen eingelagert.

Dementsprechend gibt es insgesamt 7 Einlager-Auslager-Beziehungen $A \rightarrow a_1, \dots, A \rightarrow a_4, B \rightarrow b_1, \dots, B \rightarrow b_3$ und weitere 4 Auslager-Einlager-Beziehungen $B \leftarrow a_1, \dots, B \leftarrow a_4$. Die Entnahmen für die beiden Nachfolger-Chargen a_3 und a_4 von A finden später statt als die erste Entnahme für eine der Nachfolgerchargen von B. Daher bestehen zwei Bestandsbeziehungen mit einer Dauer >0 Zeiteinheiten zwischen der Einlagerung von B und den Auslagerungen von a_3 und a_4 .

Die Lagerknoten repräsentieren entsprechend nur die Dauer bis eine erneute Einlagerung beginnt oder eine weitere Auslagerung durchgeführt wird (vgl. Abb. 4.5):

- $A \rightarrow a_1$ ist der Silo-Bestand zwischen der Einlagerung von A und der Entnahme durch die Nachfolger-Charge a_1 ,
- $A' \rightarrow a_2$ ist der Silo-Bestand zwischen der Auslagerung durch die Nachfolger-Charge a_1 und der Entnahme durch die Charge a_2 ,
- $A'' \rightarrow a_3$ ist der Bestand zwischen der Auslagerung durch die Charge a_2 und der Einlagerung der Charge B (Da die Einlagerung von B früher stattfindet als die Entnahme durch a_3 hat der Bestandsknoten eine verkürzte Dauer).
- Knoten mit der Dauer 0 werden nicht dargestellt: Dies sind die Überschneidungen zwischen der früher stattfindenden Einlagerung von B und der später stattfindenden

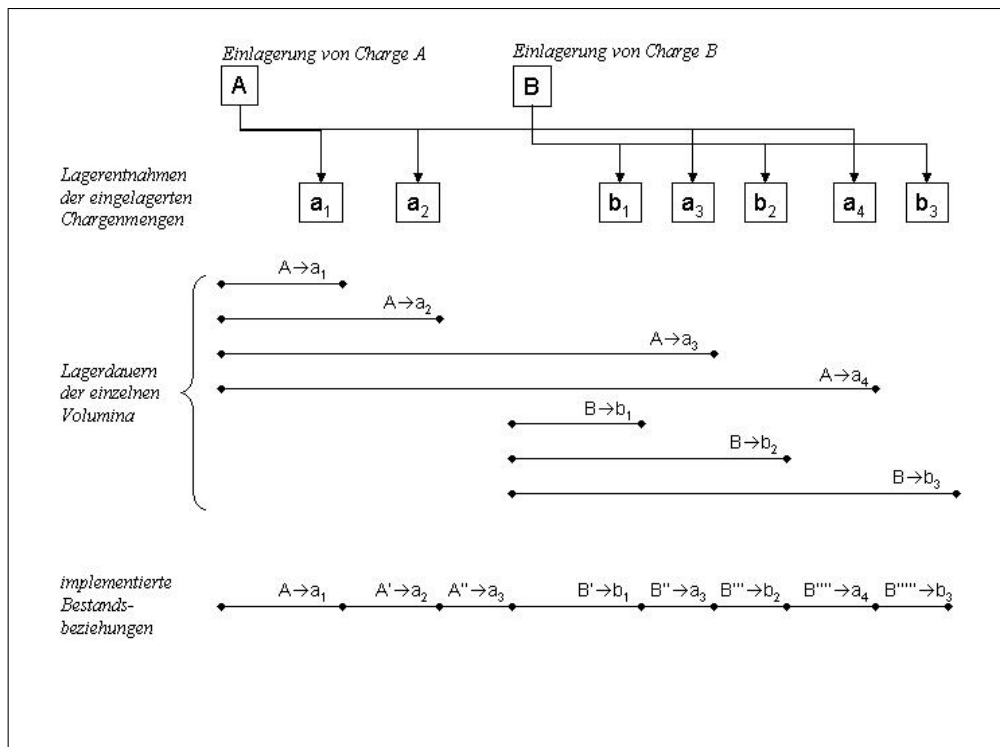


Abbildung 4.5: Lagerung mit Batch-Mix (Beispiel 4.2)

Auslagerung von a_4 . Ebenso haben die Bestandsbeziehungen $B \leftrightarrow a_1$ und $B \leftrightarrow a_2$ die Dauer 0, da die Auslagerungen vor der Einlagerung von B stattfinden.

- $B' \rightarrow b_1$ ist der Silo-Bestand, der durch die zusätzliche Einlagerung von B entsteht und durch Auslagerung für die Charge b_1 verringert wird.
- $B'' \rightarrow a_3$ ist der Silo-Bestand nach der Auslagerung für b_1 , der durch die Entnahme für die Charge a_3 weiter verringert wird.
(Die Bezeichnung $B'' \rightarrow a_3$ folgt aus der zeitlichen Abfolge, da die Entnahme durch die Charge a_3 nach der Einlagerung von der Charge B stattfindet.)
- Nachfolgend wird der Silo-Bestand in den Knoten $B''' \rightarrow b_2$, $B'''' \rightarrow a_4$ und $B''''' \rightarrow b_3$ jeweils durch eine Auslagerung für die jeweils genannte Charge verringert.

Lagerung einer Charge

Beim Zwischenlagern einer Charge im Silo ist in der Modellierung dieselbe Vorgehensweise wie im Abschnitt *Einplanung einer Charge* (S. 78) anzuwenden. Wenn zusätzlich die Möglichkeit des Batch-Mix gegeben ist, dann muß hierbei die Reihenfolge der Knoten kn_{LAGER} , die aus den Zeitpunkten der Einlagerung und Auslagerung resultiert berücksichtigt werden und dieser Abfolge angepaßt werden.

4.2.3 Modellierung von Ressourcenpools

Unter *Ressourcenpools* wird eine Menge von Scheduling-Units mit gleichen Funktionalitäten und Kapazitäten verstanden. Im Rahmen der hier betrachteten Modellierung werden als Ressourcenpools die Abfüllstutzen eines Typs (d.h. Gebinde-Abfüllung oder TKW-Abfüllung) und die Mitarbeiter für die Reinigung¹⁷ verstanden.

Für jede Ressource in einem Ressourcenpool wird eine eigene Scheduling-Unit mit Start \circ und Ende $*$ modelliert¹⁸. Ansonsten sind zu Beginn der Planung zunächst keine weiteren Prozesse eingeplant. Diese werden erst im Rahmen der **Makespan**-Berechnung erstellt und bei der **Makespan**-Optimierung in Zuordnung und Reihenfolge optimiert.

Die Prozesse, die auf den einzelnen Scheduling-Units eines Ressourcenpools stattfinden, sind gebunden an parallel stattfindende Prozesse. Wenn also ein Reaktor, ein Silo oder ein Abfüllstutzen gereinigt wird, muß dieser Prozeß zeitgleich einer Scheduling-Unit vom Typ *Mitarbeiter* zugeordnet und in dessen Reihenfolge eingeplant werden. Eine analoge Vorgehensweise ist bei der Planung der Abfüllungen zu berücksichtigen und in Abb. 4.6 für zwei Abfüllungen (Gebinde und TKW) auf Basis der Knotenkette eines Reaktors schematisch dargestellt.

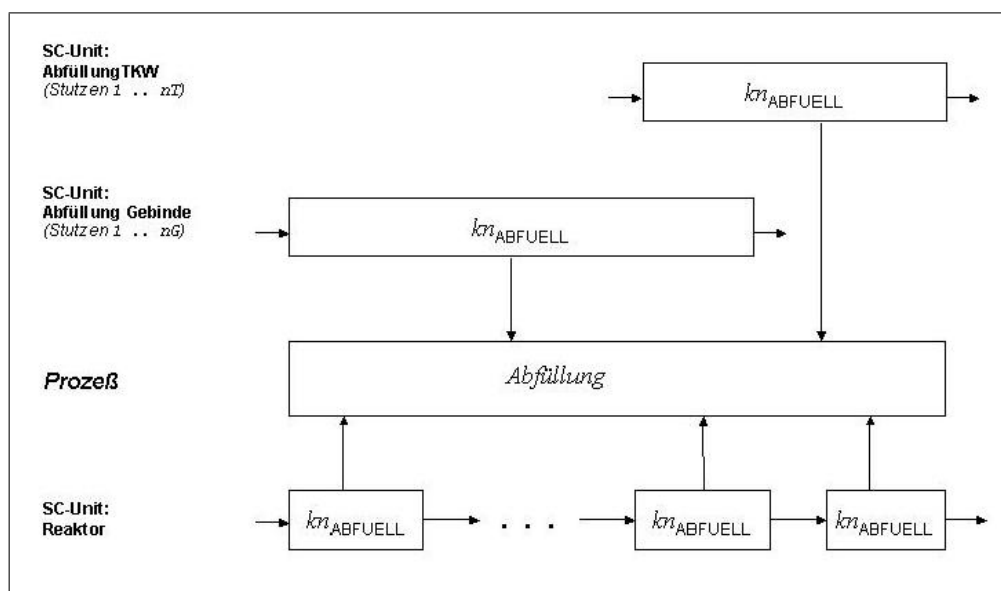


Abbildung 4.6: Modellierung mit Ressourcenpool Abfüllung

In dieser Arbeit werden die beiden nachfolgenden Begriffe zu weiteren Unterscheidung verwendet:

Definition 4.3 Ein Prozeß der Abfüllung oder Reinigung, der durch die Reihenfolgeplanung bedingt in der Knotenkette eines Reaktors oder Silos auftritt, erzeugt parallel dazu den gleichen Prozeß in der Knotenkette des entsprechenden Ressourcenpools. Der erzeugende Prozeß auf dem Reaktor oder Silo wird daher auch als Poolprozeß bezeichnet. Der

¹⁷Zur Berücksichtigung der Nebenbedingung *Es dürfen zeitgleich nicht mehr Reinigungsprozesse stattfinden, als Mitarbeiter hierfür zur Verfügung stehen.*

¹⁸Daher darf hier der Begriff *Ressourcenpool* nicht mit der Zusammenfassung mehrerer Maschinen in der *Poolresource* in Fußnote 36 auf S. 37 verwechselt werden

neu eingeplante Prozeß, der zeitgleich (parallel) dazu als Knoten in der Scheduling-Unit eines Ressourcenpools eingeplant wird, wird als Parallelprozeß bezeichnet.

Da die Abfüllung in der vorliegenden Modellierung bereits mittels eines Ressourcenpools implementiert ist, wird zur Reduzierung der Modellkomplexität zwischen den Abfüllungen keine Reinigung durchgeführt. Dies führte sonst zu weiteren Freiheitsgraden, die in der Lösungssuche berücksichtigt werden müssten.

Praxis: Implementierung in einem Planungssystem

Wenn die Charge eines Endproduktes nicht an einen Kundenauftrag gebunden ist oder für einen Kundenauftrag noch nicht die Abfüllart bekannt ist, muß das Endprodukt nach der Fertigstellung bis zur eigentlichen Abfüllung in einem Silo zwischengelagert werden. Diese *Produktion auf Lager* wird auch als *Abfüllung in ein Fertigtanklager (FTL)* bezeichnet. In diesem Fall ist nicht bekannt, wie lange das Fertigtanklager belegt ist, und wann eine Abfüllung erfolgt. Diese Art der Abfüllung wird in dem Modell dieser Arbeit daher nicht weiter betrachtet.

Um mit dem vorliegenden Modell eine Produktion auf Lager berücksichtigen zu können, sind die Verfügbarkeiten und Bestände der Fertigtanklager dem Modell für jeden Tag als zusätzliche, extern verwaltete Parameter mitzuteilen. Die Produktion wird durch einen „internen Kundenauftrag“ geplant. Eine spätere Abfüllung benötigt dann ein weiteres Produktionsrezept mit der Dauer $t = 0$, damit die Reaktoren nicht belegt werden, aber die Verwendung der Abfüllstutzen in der Optimierung berücksichtigt werden kann. Dieses Produktionsrezept verwendet dann nur das im Silo gelagerte Endprodukt als Rohstoff.

4.2.4 Zusammenfassung und Initial-Instanz

Die in den vorhergehenden Abschnitten beschriebene Modellierung beschreibt die Situation, wenn die einzelnen Prozesse in der Ablaufreihenfolge eingeplant werden. Hierbei wird noch keine Zuordnung der Ressourcen zu den Ressourcenpools (*Mitarbeiter* und *Abfüllung*) durchgeführt. Das resultierende Modell wird in Abb. 4.7 im Überblick dargestellt.

Für die weitere Verwendung und in Anlehnung bzw. Erweiterung der Definition 3.6 werden in dieser Arbeit die folgenden Begrifflichkeiten verwendet.

Definition 4.4 *Eine bestehende, nicht notwendigerweise optimale Lösung des Reihenfolgeproblems wird Instanz oder Scheduling genannt und mit \mathcal{S} bezeichnet.*

Wenn für die Reinigungs- und Abfüllprozesse einer Instanz noch keine Ressourcen aus den Ressourcenpools zugeordnet sind, wird auch von einer Initial-Instanz \mathcal{S}_0 gesprochen (vgl. Abb. 4.7). Wenn (noch) kein Produktionsauftrag eingeplant ist, dann wird die vorliegende Modellierung als leere Instanz \mathcal{S}_{empty} bezeichnet.

Eine Initial-Instanz entsteht also immer, wenn eine oder mehrere Produktionsaufträge (Chargen) einem bestehenden Scheduling \mathcal{S} hinzugefügt wurden, aber eine Berechnung des *Makespan* noch nicht durchgeführt wurde und somit keine Zuordnung der parallelen Prozesse Reinigung und Abfüllung besteht. Wenn alle zu berücksichtigenden Produktionsaufträge dem Scheduling \mathcal{S} in zulässiger, nicht blockierender Reihenfolge hinzugefügt

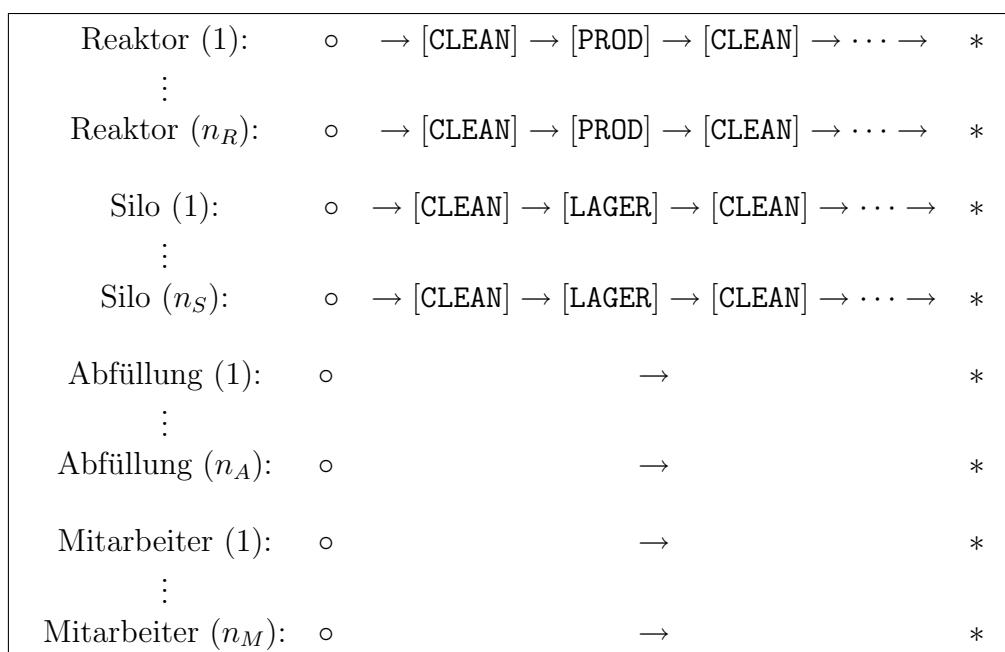


Abbildung 4.7: Initial-Instanz der Reihenfolgeplanung

wurden und eine erste Berechnung des **Makespan** durchgeführt wurde, erhält man eine *Startlösung*.

Praxis: Implementierung in einem Planungssystem

Zyklische Materialflüsse: Die beschriebene Modellierung beruht auf der Basis eines gerichteten Graphenmodells. Dies muß zur Darstellung einer Lösung zyklensfrei sein. Eine solche Forderung widerspricht jedoch nicht der Anforderung der verfahrenstechnischen Industrie, zyklische Materialflüsse zu berücksichtigen.

Das entwickelte Netzwerk beschreibt die Abfolge der einzelnen Produktionsschritte. Bei der zyklischen Produktion fließt ein Nebenprodukt wieder in den Herstellungsprozeß eines Vorproduktes ein. Dieser Herstellungsprozeß des Vorproduktes findet jedoch zeitlich später statt. Die Berücksichtigung der zyklischen Produktion muß daher nicht in der Reihenfolgeplanung, sondern bei der Erstellung der Chargen und ihrer Beziehungen untereinander in der Volumenplanung erfolgen. Alternativ dazu kann ein solches Nebenprodukt in der Rezepturverwaltung und bei der Chargenplanung auch als Rohstoff betrachtet werden.

Rollierende Planung bzw. Neuplanung: Bisher wurde nur beschrieben, wie die Aufträge in der Modellierung bei einer Planung ohne weitere Vorgaben in einer leeren Instanz \mathcal{S}_{empty} abgebildet werden. Bei einer Implementierung dieses Verfahrens in einem Planungssystem ist jedoch zu berücksichtigen, daß bereits Produktionen durchgeführt werden. Hierbei sind zwei Fälle zu unterscheiden.

- **Planung im 3-Schicht-Betrieb** (Abb. 4.8): Wenn durch einen 3-Schicht-Betrieb die Möglichkeit der durchgängigen Produktion gegeben ist, dann können Produktionsaufträge periodenübergreifend geplant werden.

Bei der Erstellung und Einplanung von Aufträgen kann die maximal zur Verfügung stehende Zeit der aktuellen Periode überschritten werden. Dann muß in der nachfolgenden Planungsperiode allerdings berücksichtigt werden, daß die verfügbare Zeit zur Einplanung neuer Aufträge um die bereits am Vortag zusätzlich verbrauchte Produktionszeit verringert wird.

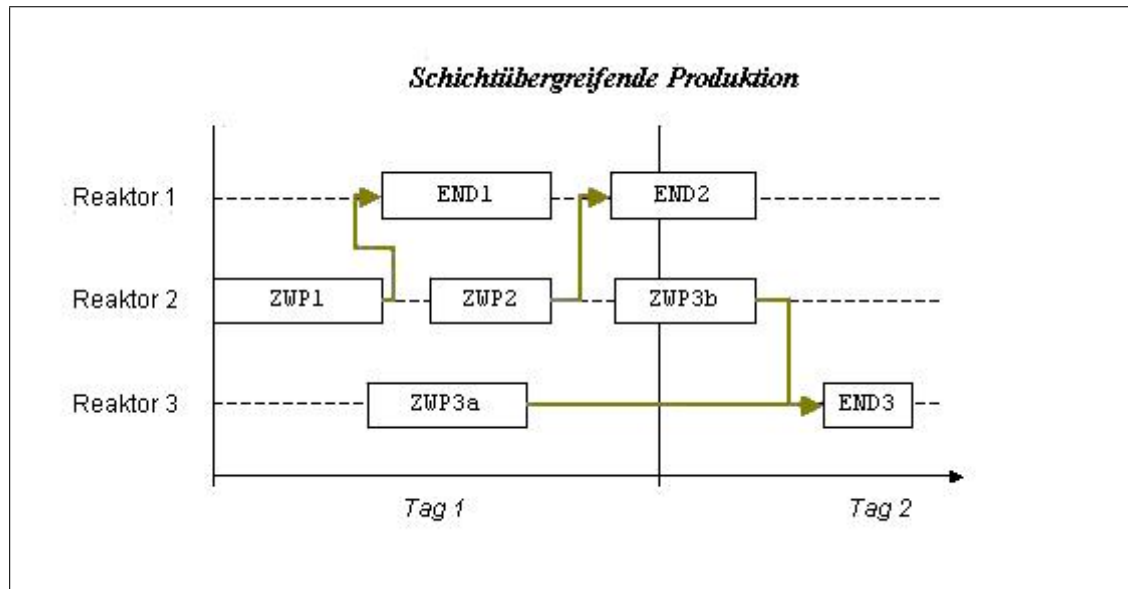


Abbildung 4.8: Schichtübergreifende Produktion

- **(Neu-)Planung bei rollierender Planung** (Abb. 4.9): Durch Veränderung der Auftragsituation (d.h. neue Kundenaufträge, Veränderung der Auftragsvolumina von Kundenaufträgen oder Stornierung von Kundenaufträgen) kann eine Neuplanung erforderlich werden. Ebenso kann im Rahmen der rollierenden Planung für einen veränderten Planungshorizont eine Neuplanung erforderlich werden. Je nach Dringlichkeit muß es auch möglich sein, die aktuelle Planungsperiode (d.h. deren Produktionsaufträge, die zum Zeitpunkt der Neuplanung bereits gefertigt werden) in der Neuplanung zu berücksichtigen.

Nach einer Untersuchung von MÉNDEZ ET AL. wurden trotz der hohen Relevanz der Neuplanung in der Praxis nur wenige Verfahren entwickelt, die sich speziell mit der Neuplanung befassen ([MC+05], S. 58). Meistens werden neue Aufträge hinzugefügt und eine erneute Optimierung mit den bestehenden Verfahren wie Vertauschen der Reihenfolge oder Zuweisen auf einen anderen Reaktor durchgeführt. Wichtig ist hierbei, daß die bestehende Reihenfolge für eine Akzeptanz durch den Anwender nicht mehr stark verändert wird und die Dauer der Neuoptimierung kürzer als ein kompletter Optimierungslauf ist.

In beiden Fällen muß vor der Erzeugung einer neuen Startlösung geprüft werden, welche Produktionen oder Prozesse zu Beginn der aktuellen Planungsperiode bzw. zum Zeitpunkt der Neuplanung noch andauern. Hierbei ist zunächst zu prüfen, ob die Herstellung eines Produktes bereits begonnen wurde. Diese Prüfung schließt auch die Produktion von Vor- und Zwischenprodukten mit ein.

Wenn eine erneute Planung aufgrund neuer Auftragsdaten oder einer Veränderung der Verfügbarkeit der Produktionsanlage notwendig wird, kann diese durch eine *inkrementelle*

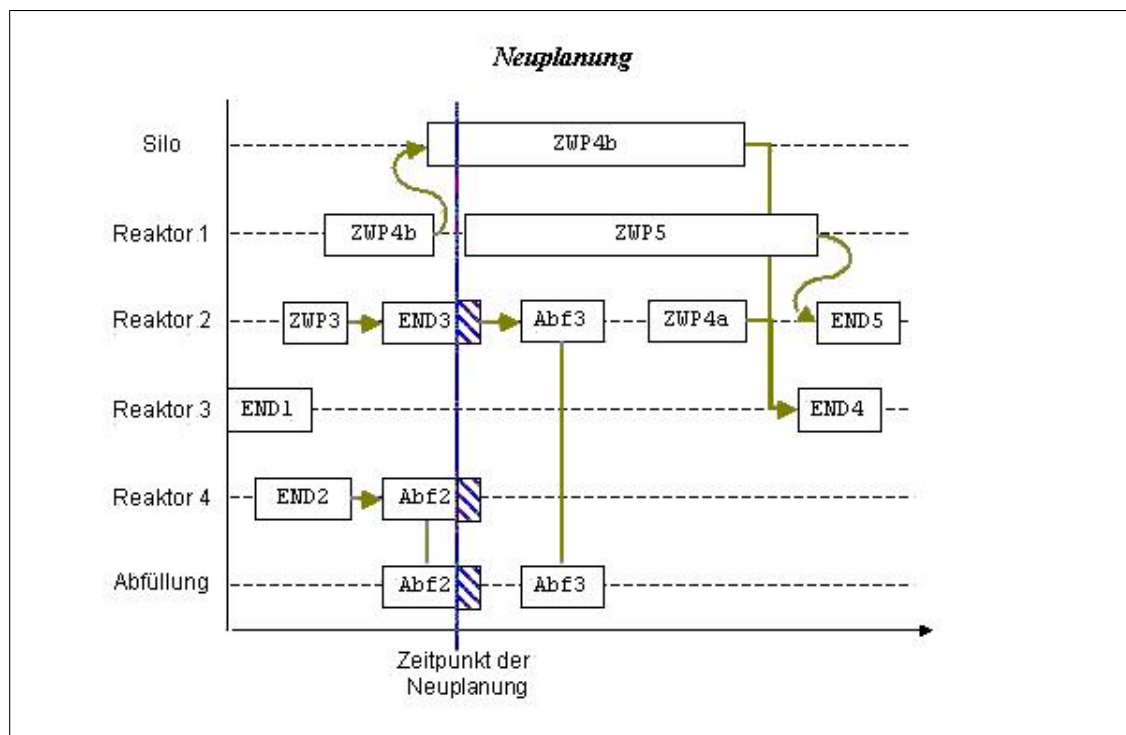


Abbildung 4.9: Neuplanung (ohne Reinigung)

Planung ([Sta00a], S. 158) oder eine *komplette Neuplanung* erfolgen. Bei der inkrementellen Planung werden neue Aufträge zu einer bestehenden Reihenfolge hinzugefügt und eine Optimierung des Reihenfolgeproblems durchgeführt.

Bei einer kompletten Neuplanung werden die *Endprodukt-Chargen* und alle mit diesen direkt oder indirekt verbundenen Zwischen- und Vorprodukt-Chargen aus der Planung entfernt. Die einzelnen, in dem Produktionsauftrag einer Charge zusammengefassten Kundenaufträge werden für die erneute Planung freigegeben¹⁹ und in der Volumenplanung (Kapitel 5) gemeinsam mit den neu einzuplanenden Aufträgen zu neuen Chargen zusammengestellt. Diese Vorgehensweise ist besonders bei einer Veränderung der Auftragslage durch Eilaufträge zu empfehlen, da hiermit sichergestellt wird, daß die Aufträge entsprechend ihrer Prioritäten genügend Rohstoffe zur Verfügung haben.

Die verbleibenden Produktionsaufträge, deren Herstellung bereits in einer Vorstufe, der Endproduktion oder einem abschließenden Prozeß (Abfüllung) durchgeführt wird, müssen bei der Neuplanung in dem bestehenden Produktionsvolumen übernommen werden. Die Prozesse, die zum Zeitpunkt der Neuplanung oder zum Ende der vorherigen Planungsperiode noch durchgeführt werden, müssen in der neuen Reihenfolge fixiert eingeplant werden. Erst die nachfolgenden Prozesse können bei einer Reihenfolgeoptimierung verändert werden.

Die Durchführung einer Neuplanung wird nachfolgend noch einmal näher anhand eines Beispiels erläutert.

¹⁹Sofern in der Volumenplanung der Chargen die Bestände der Rohstoffe und Abfüllbehälter verwaltet werden, müssen die Reservierungen der Bedarfe der aufgelösten Chargen entsprechend rückgängig gemacht werden.

Beispiel 4.3 (Neuplanung) In Abb. 4.9 ist eine bestehende Reihenfolge mit fünf zu produzierenden Endprodukt-Chargen gegeben.

Zum Zeitpunkt der Neuplanung sind Produktionen von Charge END1 und END2 bereits abgeschlossen. Allerdings ist der Prozeß der Abfüllung von END2 noch nicht abgeschlossen. Dieser kann bei einer Neuplanung nicht mehr geändert werden. Ebenso wenig kann bei einer Neuplanung die Reihenfolge von END3 verändert werden. Allerdings kann hier die Abfüllung zeitlich noch verschoben werden.

Die Herstellung von END4 wurde bereits begonnen, da das Zwischenprodukt ZWP4b bereits hergestellt wurde und im Silo gelagert wird. Hier muß also die Einlagerung von ZWP4b und der Bestand im Silo berücksichtigt werden.

Lediglich die Herstellung von END5 kann unberücksichtigt bleiben. Die Chargen END5 und ZWP5 können aufgelöst, und die darin eingeplanten Kundenaufträge können in einer Volumenplanung komplett neu eingeplant werden.

4.2.5 Schritte der Reihenfolgeoptimierung

In den vorherigen Abschnitten wurde die Modellierung der Reihenfolgeplanung für die Anforderungen und Eigenschaften der verfahrenstechnischen Industrie in einem Graphenmodell beschrieben. Darauf aufbauend wird in den nachfolgenden Abschnitten erläutert, wie dieses Modell verwendet, und welche Schritte in der Optimierung durchgeführt werden.

Ausgehend von einer leeren Instanz \mathcal{S}_{empty} (oder einem bestehendem Scheduling \mathcal{S}) müssen zunächst für die neu einzuplanenden Chargen \mathcal{C} die notwendigen Produktions-, Reinigungs- und Transferprozesse eingefügt werden. Dadurch erhält man eine Initial-Instanz \mathcal{S}_0 . Mit diesem ersten Schritt erhält man eine *Startlösung*. Die Vorgehensweise der Einplanung dazu wird in Kapitel 4.3 näher beschrieben und anhand eines Beispiels erläutert.

Nach der Erzeugung einer Startlösung ist allerdings zunächst nur die Abfolge der Prozesse festgelegt. Die einzelnen Startzeitpunkte müssen noch berechnet werden. Auch ist bei einer Startlösung noch keine Zuordnung der Reinigungs- oder Abfüllprozesse zu den jeweiligen Ressourcenpools gegeben. Diese beiden Aufgaben werden durch Ausführung des **Makespan-Algorithmus** (4.6) gelöst, der in Kapitel 4.4.2 beschrieben wird. Bei jeder nachfolgenden Veränderung in der Reihenfolge muß dieses Verfahren erneut durchgeführt werden. Hierbei kann die bestehende Zuordnung zu den Ressourcenpools übernommen werden oder neu ermittelt werden.

Eine erste Reduzierung des **Makespan** kann erreicht werden, wenn Chargen mit einer langen Wartezeit in einem Silo anstelle in dem Reaktor zwischengelagert werden. Hierzu ist ein separates Verfahren entwickelt worden, das testet, welche Chargen in Silos gelagert werden können (Kapitel 4.5.6). Dieser Algorithmus kann direkt nach der Erzeugung einer Startlösung, wie auch nach der Veränderung der Reihenfolge durchgeführt werden.

Da die Zuordnung der Prozesse zu den Ressourcenpools immer nachgeordnet erfolgt, sind bei der eigentlichen Reihenfolgeoptimierung zwei Ebenen zu betrachten (Kapitel 4.5). In der äußeren Ebene wird die Reihenfolge und Zuordnung der Chargen optimiert. Die innere Ebene der Optimierung wird in jeder äußeren Permutation durchgeführt. Dabei wird jeweils die Reihenfolge und Zuordnung der Abfüllung und Reinigungen optimiert.

Eine Übersicht über die Abfolge der einzelnen Schritte ist in Abb. 4.10 dargestellt.

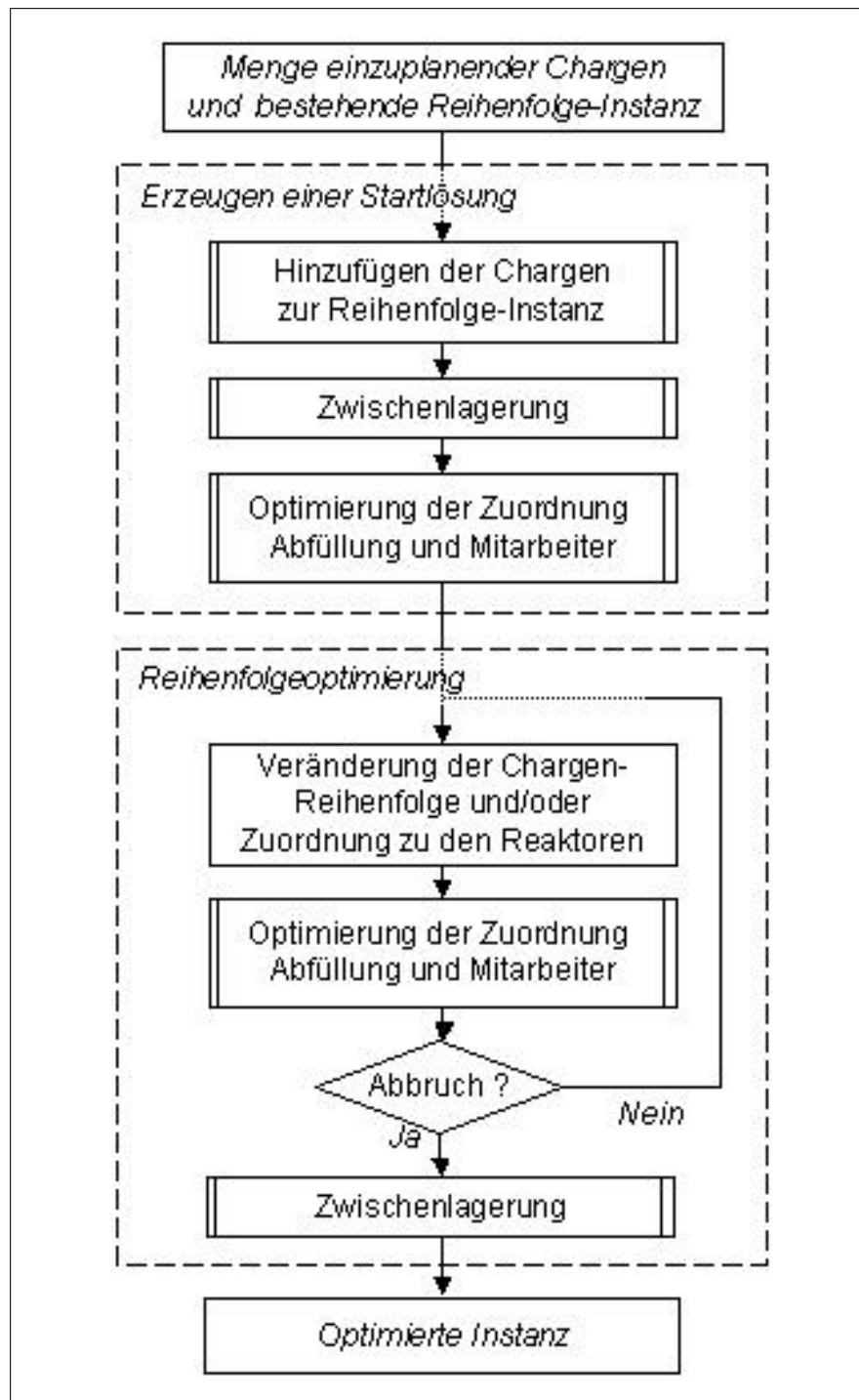


Abbildung 4.10: Ablaufschema der Reihenfolgeoptimierung

4.3 Erzeugen einer Startlösung

Mit dem folgenden Verfahren zum Erzeugen einer Startlösung wird eine Zuordnung der Chargen zu den Reaktoren durchgeführt und deren Reihenfolge auf den einzelnen Reaktoren festgelegt. Da durch die Modellierung mit einem Graphenmodell jedoch nicht die Startzeitpunkte der einzelnen Operationen²⁰ festgelegt werden, sind diese anschließend separat zu berechnen. Diese Berechnung wird mit dem Algorithmus zur **Makespan**-Berechnung (4.6) durchgeführt, der im Anschluß an diesen Abschnitt vorgestellt wird.

Das Ziel des vorliegenden Verfahrens ist es, mit geringem Aufwand eine zulässige Reihenfolge zu generieren, die anschließend weiter verbessert werden kann. Daher werden hier Prioritätsregeln verwendet. Die Planung anhand von Prioritätsregeln wird auch als *List Scheduling* bezeichnet, deren Vorgehensweise sich wie folgt schematisieren läßt:

Algorithmus 4.1 (List Scheduling)

1. Sortiere die Elemente einer Liste nach einer vorgegebenen Prioritätsregel.
2. Verwende die Elemente in der Reihenfolge der Sortierung für die Planung.

■ A(4.1)

Wie im Abschnitt *Einplanung einer Charge* auf S. 78 beschrieben wurde, kann eine Charge prinzipiell an jeder Stelle nach einem [CLEAN]-Hyperknoten eingeplant werden. Für die weitere Betrachtung in dieser Arbeit werden jedoch nur noch zwei mögliche Varianten betrachtet:

- *Einplanung am Ende einer Knotenkette:*
Bei dieser Variante werden neu einzuplanende Chargen immer am Ende einer Knotenkette vor dem Endknoten * eingeplant.
- *Einplanung am Anfang einer Knotenkette:*
Bei dieser Variante werden neu einzuplanende Chargen immer am Anfang einer Knotenkette nach dem Startknoten o eingeplant.

Diese beiden Varianten stellen sicher, daß ein zulässiges Scheduling, das so erweitert wird, keine „blockierenden Restriktionen“²¹ in Interaktion mit den bereits eingeplanten Chargen erzeugt. Bei der Einplanung am Ende einer Knotenkette kann dabei zusätzlich eine Schätzung²² des **Makespan** der neuen Lösung durchgeführt werden.

Der Algorithmus zum Erzeugen der Startlösung ist ein rekursiv konstruktivistisches Verfahren. In einer Tiefensuche werden zunächst die Chargen der Vorstufen einer Endprodukt-Charge eingeplant. Hierbei wird die Einplanungsvariante verwendet, eine Charge immer

²⁰Vergleiche hierzu die Möglichkeiten der *Modellierung des Zeitverlaufs* auf S. 42.

²¹Die Problematik einer gegenseitig blockierenden Reihenfolge, die aufgrund der Berücksichtigung verfahrenstechnischer Anforderungen entsteht, ist bspw. dann gegeben, wenn im Produktionsverlauf nicht genügend Zwischenlagermöglichkeiten vorhanden sind. Dies wird im Beispiel 4.9 auf S. 119 näher betrachtet.

²²Bei dem Einfügen einer neuen Chargen können die Restriktionen und Zuordnungen der Ressourcenpools nicht ausreichend berücksichtigt werden. Daher kann hier nur eine Schätzung durchgeführt werden.

am Ende einer Knotenkette einzufügen. Bei dieser Vorgehensweise wird dann zuletzt die Endprodukt-Charge selber eingeplant.

Dieses Konstruktionsverfahren verwendet das List Scheduling in der Auswahl der nächsten einzuplanenden Charge (**Regel (ch)**) und der Zuordnung des Reaktors der aktuell einzuplanenden Charge (**Regel (rk)**). Die Kriterien der beiden Auswahlregeln werden im Anschluß an den Algorithmus beschrieben.

Algorithmus 4.2 (Erzeugen einer Startlösung)

- Sei \mathcal{C} die nichtleere Menge aller neu einzuplanenden Chargen mit der Teilmenge \mathcal{C}_E der Endprodukt-Chargen ($\mathcal{C}_E \neq \emptyset$).

Jede Charge $chg \in \mathcal{C}$ hat eine (möglicherweise leere) Menge der direkt vorgeschalteten Chargen $CH_{scr}(chg)$.

Sei \mathcal{C}' eine Auswahl aktuell einzuplanender Chargen mit der Startmenge $\mathcal{C}' := \mathcal{C}_E$ und \mathcal{C}_S die Liste der bereits im Scheduling eingeplanten Chargen mit $\mathcal{C}_S := \emptyset$. Außerdem ist $\mathcal{L} := \emptyset$ eine Liste der in den verschiedenen Rekursionsschritten erzeugten Auswahllisten von \mathcal{C}' .

- Sei UR die Menge aller Produktionsanlagen (Reaktoren). Aufgrund der Vorgaben der Rezeptur und dem Volumen der Charge gibt es eine nichtleere Menge $UR(chg)$ von Reaktoren, auf denen eine Charge chg gefertigt werden kann.

Sei $UR'(chg)$ die Teilmenge der Reaktoren, die für den aktuellen Planungsschritt mit der Charge chg betrachtet werden. Für alle $chg \in \mathcal{C}$ ist $UR'(chg)$ initialisiert mit $UR'(chg) := UR(chg)$.

- Sei \mathcal{S}_0 eine bestehende Instanz der Reihenfolgeplanung (*vor der ersten Einplanung einer Charge ist $\mathcal{S}_0 := \mathcal{S}_{empty}$*) und \mathcal{S}' eine aktuelle Instanz mit dem Anfangswert $\mathcal{S}' := \mathcal{S}_0$.

Mit $\mathcal{S}' \oplus chg$ sei das Scheduling bezeichnet, das entsteht, wenn die Charge chg in dem Scheduling \mathcal{S}' eingeplant wird (bzw. ist). Analog ist $\mathcal{S}' \ominus chg$ das Scheduling, aus dem die Charge chg gelöscht wurde.

1. *Abbruch Algorithmus (Auftragsmenge):*

Wenn $\mathcal{C}' = \emptyset$ und

- $\mathcal{L} \neq \emptyset$, dann bestimme \mathcal{C}' neu als die zuletzt der Liste \mathcal{L} hinzugefügte Chargenmenge und setze $\mathcal{L} := \mathcal{L} \setminus \{\mathcal{C}'\}$.

(Der Algorithmus geht in der rekursiven Tiefensuche einen Schritt zurück.)

- $\mathcal{L} = \emptyset$, dann

- Ende des Algorithmus mit der erzeugten Startlösung \mathcal{S}' , wenn $\mathcal{C} = \emptyset$.

(Die oberste Ebene der rekursiven Tiefensuche ist erreicht und alle Chargen wurden erfolgreich eingeplant.)

- Abbruch des Algorithmus ohne eine zulässige Lösung, wenn $\mathcal{C} \neq \emptyset$.

2. *Auswahl Charge:*

Bestimme $chg \in \mathcal{C}'$ nach **Regel (ch)**.

3. *Rekursion:*

Wenn $CH_{scr}(chg) \cap \mathcal{C} \neq \emptyset$, dann

- füge \mathcal{C}' der Liste \mathcal{L} hinzu und
(Speichere die Auftragsmenge für die Rückschritte in der Rekursion.)
- setze $\mathcal{C}' := CH_{scr}(chg) \cap \mathcal{C}$.
(Bestimme die Auftragsmenge für die nächste Rekursionsstufe.)
- Gehe zu Schritt 2.
(Hierdurch wird zunächst rekursiv die Charge der frühesten Vorproduktstufe gesucht.)

4. *Abbruch Tiefensuche (Charge):*

Wenn $UR'(chg) = \emptyset$, dann

bestimme die zuletzt der Liste \mathcal{C}_S hinzugefügte Charge chg_S und gehe in der Lösungssuche einen Schritt zurück. Setze dazu

- $\mathcal{S}' := \mathcal{S}' \ominus chg_S$
(Lösche aus dem Scheduling die zuletzt hinzugefügte Charge, damit die Lösungssuche einen Schritt zurückgehen kann.)
- $\mathcal{C}_S := \mathcal{C}_S \setminus \{chg_S\}$ und $\mathcal{C} := \mathcal{C} \cup \{chg_S\}$.
(Die zuletzt hinzugefügte Charge muß wieder eingeplant werden.)
- Füge \mathcal{C}' der Liste \mathcal{L} hinzu und setze $\mathcal{C}' := \{chg_S\}$.
(Speichere die aktuelle Auftragsmenge für den nächsten Aufruf in der Rekursion und erzeuge eine neue Auftragsmenge.)
- Setze $UR'(chg) := UR(chg)$.
($UR'(chg)$ war eine leere Menge. Bei dem nächsten Versuch chg einzuplanen stehen wieder alle Reaktoren zur Verfügung.)
- Setze $chg := chg_S$ und prüfe erneut Schritt 4.
(Eventuell wird zunächst eine weitere Charge aus dem Scheduling entfernt, oder die Charge chg_S wird auf einem anderen Reaktor eingeplant.)

5. *Auswahl Reaktor und Einplanung Charge:*

Bestimme für chg den Reaktor $u(chg) \in UR'(chg)$ nach Regel (rk) und setze $UR'(chg) := UR'(chg) \setminus \{u(chg)\}$.

Plane die Charge chg in dem Scheduling \mathcal{S}' auf der Knotenkette des Reaktors $u(chg)$ ein und berechne den **Makespan** mit dem Algorithmus (4.6).

(Hier muß überprüft werden, ob nach der Einplanung wegen der verfahrenstechnischen Anforderungen eine „blockierende Reihenfolge“ in der Abfolge²³ entsteht, etwa indem zwei Chargen auf dem gleichen Reaktor gefertigt werden müssen, ohne daß Möglichkeiten zur Zwischenlagerung der zuerst produzierten bestehen. Wenn eine solche Situation vorhanden ist, bricht der Algorithmus 4.6 zur **Makespan-Berechnung** mit einer ungültigen (Zwischen-) Lösung ab.)

6. *Zwischenlagerung anderer Chargen;*

Wenn $\mathcal{S}' \oplus chg$ keine zulässige Lösung ist, dann prüfe, ob die in der Knotenkette des Reaktor $u(chg)$ zeitlich zuletzt vor der Charge chg herzustellende Charge chg_{prev} in einem Silo gelagert werden kann. Plane eine Zwischenlagerung für die Charge chg_{prev} und berechne den **Makespan** anhand des Algorithmus (4.6) erneut.

²³Diese Situation wird auch als *Deadlock* bezeichnet, weil zwei oder mehr Prozesse (in dem Graphenmodell: Knoten) bei der Berechnung der Startzeitpunkte gegenseitig auf die Fertigstellung der Berechnung warten, bevor die eigene Berechnung abgeschlossen wird.

7. Gültigkeit:

Wenn durch $\mathcal{S}' \oplus chg$ eine gültige (Zwischen-) Lösung erreicht wird, setze

- $\mathcal{C}_S := \mathcal{C}_S \cup \{chg\}$ und $\mathcal{S}' := \mathcal{S}' \oplus chg$
(Die Charge chg wurde erfolgreich eingeplant. Arbeite mit der gültigen Zwischenlösung weiter.)
- $\mathcal{C} := \mathcal{C} \setminus \{chg\}$
(Wenn chg Vorläufer von mehreren Chargen ist, dann genügt es, wenn die Charge nur einmal ausgewählt und eingeplant wird.)
- $\mathcal{C}' := \mathcal{C}' \setminus \{chg\}$
(Die Charge chg wurde erfolgreich eingeplant und muß nicht weiter in der aktuellen Auswahl berücksichtigt werden.)
- Gehe zu Schritt 1.
(Versuche, die nächste Charge einzuplanen.)

Ansonsten gehe zu Schritt 4.

(Gehe in der Lösungssuche einen Schritt zurück.)

■ A(4.2)

Für die Beschreibung des Algorithmus 4.3, der von dem vorigen Verfahren verwendet wird, wird die Definition 4.1 der *Produktionstiefe* folgendermaßen erweitert.

Definition 4.5 Die maximale rekursive Produktionstiefe zählt die maximal benötigte Anzahl von Produktionsschritten, die in der rekursiven Auflösung einer Charge in die einzelnen Vorprodukte benötigt werden.

In dem nachfolgendem Beispiel 4.4 (siehe auch Abb. 4.11 auf S. 95) hat die Endprodukt-Charge END die maximale rekursive Produktionstiefe 2. Die Zwischenprodukt-Chargen ZWP2.a und ZWP2.b haben den Wert 1, während die Charge ZWP1 den Wert 0 hat.

Dann kann die Auswahlregel für die Chargen mit den folgenden Abstufungen definiert werden.

Algorithmus 4.3 (Regel (ch) für Algorithmus 4.2)

1. Wähle die Charge(n) mit der größten maximalen rekursiven Produktionstiefe aus. Wenn für eine Charge bereits Zwischenprodukte eingeplant wurden, bevorzuge diese.
2. Bei mehreren Chargen in Schritt 1, wähle die Charge(n) aus, die den direkt nachfolgenden Produktionsschritt als JiT-Produkt bedient.
3. Bei mehreren Chargen in Schritt 2, wähle die Charge(n) aus, für die $|UR(chg)|$ den kleinsten Wert hat.
4. Bei Gleichheit in Schritt 3 wähle eine beliebige Charge aus.

■ A(4.3)

Diese Auswahlregel kann beliebig verfeinert werden. Der wesentliche Gedanke der dargestellten Vorgehensweise ist es, eine Startlösung so zu erzeugen, daß die “frühesten“ Vorprodukte im Scheduling zuerst eingeplant werden und nicht unnötig viele Rückwärtsschritte in der Lösungssuche durchgeführt werden müssen. Daher erhalten JiT-Chargen eine besondere Berücksichtigung. Aus dem gleichen Grund werden zuerst die Chargen mit dem geringsten Wert $|UR(chg)|$ eingeplant.

Für die Prioritätsregel (rk) im Algorithmus 4.2 wird zunächst ein weiteres Kennzeichen definiert.

Definition 4.6 *Wenn eine Charge chg an dem Ende der Knotenkette eines Reaktor u zu dem einem frühestmöglichen Startzeitpunkt $st_{src}(chg) > \text{MKS}(u)$ eingeplant werden soll, dann wird die Differenz $\text{MKS}(u) - st_{src}(chg)$ als Stillstandszeit des Reaktors u bezeichnet.*

Der lokale Makespan $\text{MKS}(u)$ (siehe Definition 4.2 auf S. 75) entspricht dem Endzeitpunkt der Dummy-Operation im Endknoten $*$.

Die Auswahlregel für die Reaktoren ist dann wie folgt definiert.

Algorithmus 4.4 (Regel (rk) für Algorithmus 4.2)

1. Wähle den Reaktor u mit dem geringsten lokalen Makespan $\text{MKS}(u)$ aus.
2. Bei mehreren Reaktoren in Schritt 1, wähle den Reaktor mit der geringsten Stillstandszeit aus.
3. Bei Gleichheit in Schritt 2 wähle einen beliebigen Reaktor aus.

■ A(4.4)

Das folgende Beispiel zeigt nun die Anwendung des Algorithmus.

Beispiel 4.4 (Erzeugen einer Startlösung) *Für eine Endprodukt-Charge des Rezeptes END mit allen Zwischen- und Vorprodukten ist eine Startlösung zu erstellen.*

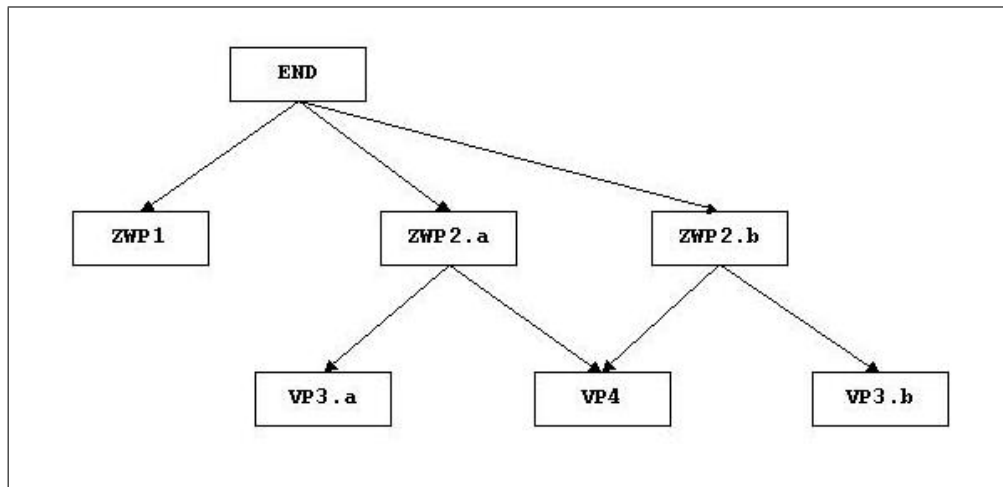
Die Chargen und ihre Beziehungen sind in Abb. 4.11 wiedergegeben. Dabei ist zu beachten, daß für das Rezept ZWP2 zwei Chargen ZWP2.a und ZWP2.b eingeplant werden müssen. Das Volumen des Vorproduktes VP4, das von diesen beiden Chargen benötigt wird, kann aber mit einem Produktionsauftrag (der Charge VP4) erstellt werden.

Die Konstruktion dieser Chargen wird in Beispiel 5.7 auf S. 164 beschrieben.

Die für die Chargen möglichen, zu verwendenden Reaktoren sind in Tabelle 4.1 aufgeführt. Die Volumina und Produktionszeiten der Chargen und Reaktoren werden in diesem Beispiel nicht weiter berücksichtigt.

Ausgangssituation für die Planung ist die leere Instanz \mathcal{S}_{empty} . Silos für eine Zwischenlagerung sind nicht vorhanden.

Der Algorithmus erzeugt dann die in Abb. 4.12 dargestellte Lösung. Die Reihenfolge, in der die (zulässigen und unzulässigen) Zwischenlösungen erzeugt werden, ist in Tabelle 4.2 wiedergegeben. Die einzelnen Zwischenschritte, die bei der Einplanung einer einzelnen Charge durchgeführt werden, werden im Anhang A genauer beschrieben.



Das Endprodukt **END** benötigt eine Charge des Zwischenproduktes **ZWP1** und zwei Chargen des Zwischenproduktes **ZWP2**. Die Produktion des Zwischenproduktes **ZWP2** erfordert wiederum die Produktion der Vorprodukte **VP3** und **VP4**. Die Charge **VP4** bedient dabei beide Chargen des Zwischenproduktes **ZWP2**.

Abbildung 4.11: Einzuplanende Chargen (Beispiel 4.4)

<i>chg</i>	<i>UR(chg)</i>
END	R1
ZWP1	R2
ZWP2 . a, ZWP2 . b	R2, R3, R4
VP3 . a, V3 . b	R5, R6
VP4	R7

Tabelle 4.1: Zugeordnete Reaktoren (Beispiel 4.4)

Nr.	# CHG	chg	$u(chg)$ und Kommentar
1*	1	VP3.a	R5
2*	2	VP4	R7
3	3	ZWP2.a	R2
4	4	VP3.b	R6
5	5	ZWP2.b	R3
6	6	ZWP1	R2
7	7	END	R1; Abbruch, da ZWP2.a und ZWP1 beide auf R2
8	6	ZWP1	aus \mathcal{S} gelöscht
9	5	ZWP2.b	R4; verschoben von R3 (Schritt 5)
10	6	ZWP1	R2; erneut eingeplant
11	7	END	R1; Abbruch, da ZWP2.a und ZWP1 beide auf R2
12	6	ZWP1	aus \mathcal{S} gelöscht
13	5	ZWP2.b	R2; verschoben von R4 (Schritt 9)
14	6	ZWP1	R2; erneut eingeplant
15	7	END	R1; Abbruch, da ZWP2.a, ZWP2.a und ZWP1 auf R2
16	6	ZWP1	aus \mathcal{S} gelöscht
17	5	ZWP2.b	aus \mathcal{S} gelöscht
18	4	VP3.b	R5; verschoben von R6 (Schritt 4)
19	5	ZWP2.b	R3; erneut eingeplant
20	6	ZWP1	R2; erneut eingeplant
21	7	END	R1; Abbruch, da ZWP2.a und ZWP1 beide auf R2
22	6	ZWP1	aus \mathcal{S} gelöscht
23	5	ZWP2.b	R4; verschoben von R3 (Schritt 19)
24	6	ZWP1	R2; erneut eingeplant
25	7	END	R1; Abbruch, da ZWP2.a und ZWP1 beide auf R2
26	6	ZWP1	aus \mathcal{S} gelöscht
27	5	ZWP2.b	R2; verschoben von R4 (Schritt 23)
28	6	ZWP1	R2; erneut eingeplant
29	7	END	R1; Abbruch, da ZWP2.a, ZWP2.a und ZWP1 auf R2
30	6	ZWP1	aus \mathcal{S} gelöscht
31	5	ZWP2.b	aus \mathcal{S} gelöscht
32	4	VP3.b	aus \mathcal{S} gelöscht
33*	3	ZWP2.a	R3; verschoben von R2 (Schritt 3)
34*	4	VP3.b	R6; erneut eingeplant
35	5	ZWP2.b	R2; erneut eingeplant
36	6	ZWP1	R2; erneut eingeplant
37	7	END	R1; Abbruch, da ZWP2.b und ZWP1 beide auf R2
38	6	ZWP1	aus \mathcal{S} gelöscht
39*	5	ZWP2.b	R4; verschoben von R2 (Schritt 35)
40*	6	ZWP1	R2; erneut eingeplant
41*	7	END	R1; gültige Lösung gefunden

Nr. gibt die laufende Nr. der Zwischenlösung an, # CHG die Anzahl Chargen, die aktuell eingeplant wurden. chg gibt an, welche Charge in der Zwischenlösung eingeplant wurde und $u(chg)$ gibt den Reaktor an, auf dem die Charge eingeplant wurde.

Die in dem endgültigen Scheduling \mathcal{S} eingeplanten Chargen sind **fett** und mit einem * markiert.

Tabelle 4.2: Erzeugte Zwischenlösungen (Beispiel 4.4)

Anhand der Tabelle 4.2 wird die rekursive Vorgehensweise des Einplanungsverfahrens sichtbar. In dem Zwischenschritt 3 wird die Charge ZWP2.a dem Reaktor R2 zugeordnet. Da aber die Charge ZWP1, für die die Zwischenlösung 6 ebenfalls auf diesem Reaktor eingeplant wird, ohne daß die Charge ZWP2.a in einem Silo zwischengelagert werden kann, kann anschließend keine gültige Lösung im Schritt 7 erzeugt werden. In den nachfolgenden Schritten werden die bereits eingeplanten Chargen zunächst entfernt und auf anderen Reaktoren eingeplant.

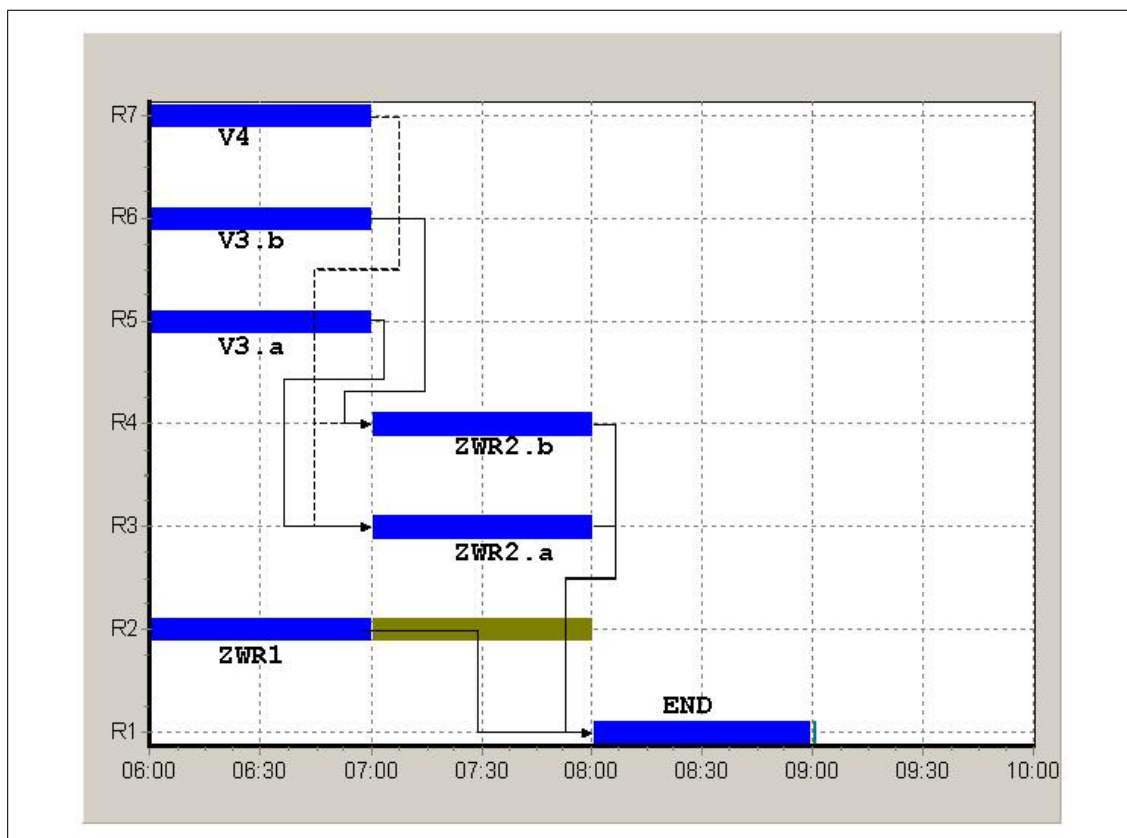


Abbildung 4.12: Erzeugte Startlösung (Beispiel 4.4)

Wie aus dem Beispiel weiterhin ersichtlich wird (vgl. Abb. 4.12), kann in Einzelfällen eine Zwischenlagerung im Silo erforderlich werden. Wenn etwa für die Chargen V3.a und V3.b nur ein Reaktor zur Produktion zur Verfügung stünde, dann müßte die zuerst eingeplante Charge V3.a in einem Silo zwischengelagert werden, bevor die Produktion der Charge V3.b durchgeführt werden könnte. Dieser Sonderfall wird im Schritt 6 des Algorithmus 4.2 berücksichtigt.

4.4 Algorithmus zur Makespan-Berechnung

Für einen bestehenden Ablaufplan müssen nachfolgend die Zeitpunkte der einzelnen Prozesse und somit letztendlich der **Makespan** einer Lösung bestimmt werden. Für die weitere Betrachtung dieser Arbeit wird gefordert, daß der resultierende Reihenfolgeplan einer solchen Feinplanung²⁴ semi-aktiv²⁵ ist. Dies kann etwa durch eine einfache Vorwärtsberechnung anhand des DIJKSTRA-Algorithmus durchgeführt werden, da die Länge des kürzesten Weges von einem Startknoten \circ zu dem Knoten einer Operation den frühestmöglichen Startzeitpunkt bestimmt. BRUCKER ET AL. bezeichnen diesen Schritt daher auch als eine eher technische Angelegenheit [BHH03]. Aufgrund der Häufigkeit mit der ein solches Verfahren jedoch durchgeführt wird, ist es notwendig, daß die Berechnung schnell und speichersparend durchgeführt wird. Hier ist laut BRUCKER ET AL. besonders die Reihenfolge zu beachten, in der die Knoten des Graphenmodells berechnet werden [BHH03].

Neben der bekannten Zuordnung der Chargen zu den Reaktoren muß in der vorliegenden Modellierung jedoch auch berücksichtigt werden, daß ein Zwischenprodukt etwa in einem Silo gelagert wird, während es auf seine weitere Verarbeitung wartet. Diese in dem Lösungsgraph modellierten Wartezeiten sind nicht bekannt und werden erst mit der Berechnung der Knotenkette ermittelt. Weiterhin muß u.a. auch die Zuordnung der Abfüllungen zu den Abfüll-Stutzen noch bestimmt werden. Der DIJKSTRA-Algorithmus als Basis zur Berechnung der Startzeitpunkte wurde daher im Rahmen dieser Arbeit zu dem **Makespan-Algorithmus 4.6** für die Berechnung der Wartezeiten weiterentwickelt und um diese Zuordnungsschritte erweitert.

4.4.1 Der DIJKSTRA-Algorithmus

Zum einfacheren Verständnis der **Makespan**-Berechnung sei hier zunächst der DIJKSTRA-Algorithmus wiedergegeben, wie er z.B. bei VAHRENKAMP ([Vah03], S. 74) zu finden ist.

Gegeben ist ein gerichtetes, zusammenhängendes Netzwerk $G = (V, E)$ mit einer Knotenmenge V und einer Kantenmenge $E \subset V \times V$. Weiterhin ist für jede Kante $(u, v) \in E$ eine (positive) Entfernung $d(u, v)$ gegeben.

Für einen Startknoten $\circ \in V$ berechnet der DIJKSTRA-Algorithmus die kürzeste Entfernung $d(kn) = d(\circ, kn)$ zwischen \circ und jedem Knoten $kn \in V \setminus \{\circ\}$.

In dem DIJKSTRA-Algorithmus werden ausgehend vom Startknoten \circ alle weiteren Knoten besucht und mit der kürzesten Entfernung „markiert“²⁶.

²⁴Da hier lediglich die Zeitpunkte innerhalb der Tagesplanung berechnet werden, ist der in der Fußnote 4 auf S. 7 gegebene Begriffe der *Terminplanung* für eine solche Planung eher irreführend.

²⁵Ein Reihenfolgeplan heißt semi-aktiv, wenn keine Operation früher starten kann, ohne daß dadurch die Abfolge oder eine Nebenbedingung verletzt wird; vgl. auch die Definition 3.4 auf S. 52

²⁶Der DIJKSTRA-Algorithmus wird daher auch als „Knoten-Label“-Algorithmus bezeichnet, da jedem Knoten die Entfernung als Markierung (engl.: label) hinzugefügt wird.

Algorithmus 4.5 (DIJKSTRA-Algorithmus)

1. *Initialisierung*

- Setze $d(o) := 0$ und $\forall kn \in V \setminus \{o\} \ d(kn) := \infty$
- Setze $V' := V$

2. *Verarbeitung*: Solange $V' \neq \emptyset$

- Bestimme den Knoten $v \in V'$ mit der kleinsten Entfernung $d(v)$, d.h. mit $d(v) = \min_{kn \in V'} d(kn)$.
- Berechne für alle benachbarten Knoten $kn \in V'$ von v , d.h. $(v, kn) \in E$, die neue Entfernung $d(kn) := \min(d(kn); d(v) + d(v, kn))$.
- Reduziere die Menge der zu berechnenden Knoten $V' := V' \setminus \{v\}$.

■ A(4.5)

4.4.2 Die Makespan-Berechnung

Der für die vorliegende Problemstellung entwickelte **Makespan-Algorithmus** berechnet die Startzeitpunkte aller Operationen (Knoten) in einer Instanz des Reihenfolgeproblems. Er ist wie der DIJKSTRA-Algorithmus ein “Knoten-Label“-Algorithmus.

Da zu Beginn der Berechnung noch nicht alle Inputdaten bekannt sind, wird hier eine zusätzliche Zwischenmarkierung verwendet, um diese Daten zu berechnen. In jedem Iterationsschritt des Algorithmus werden die folgenden Mengen betrachtet:

- Sei V_s die Menge der zu berechnenden Knoten²⁷.
- Sei V_p die Menge der noch nicht endgültig berechneten Knoten²⁸.
- Sei V_k die Menge der endgültig berechneten Knoten²⁹.

In einigen Fällen werden die Knoten als *nicht endgültig berechnet* gekennzeichnet. Die Dauer der Lagerung $t(kn)$ im Reaktor bzw. im Silo endet erst mit dem Start des nachfolgenden Produktionsprozesses und muß dann rückwirkend berechnet werden. Der Endzeitpunkt von JiT-Produktionen ist ohne weiteren Puffer an den Startzeitpunkt der nachfolgenden Produktion gekoppelt.

Die Operationen der Ressourcenpools (Reinigung und Abfüllung) sind noch nicht zugeordnet. Hier sind weitere Zwischenschritte durchzuführen. Für jeden Ressourcenpool wird daher eine Menge der in einem Iterationsschritt aktuell zugeordneten Operationen definiert. Dies wird genauer in dem Abschnitt zur *Kalkulation bei Parallelprozessen* auf Seite 106 erläutert.

²⁷s für Startknoten

²⁸p für pending; engl.: schwebend, unentschieden

²⁹k für kalkuliert

Berechnung der Startzeit

Die Startzeit eines Knotens wird aus den direkten Vorgängern und vorgeschalteten Produktionsschritten berechnet³⁰. Hierzu werden zwei Variablen st_{prev} und st_{src} definiert und daraus resultierend der Startzeitpunkt $st(kn)$ berechnet als

$$st(kn) := \max(st_{prev}(kn), st_{src}(kn)). \quad (4.1)$$

Entsprechend ist der Endzeitpunkt $e(kn)$ als $e(kn) := st(kn) + t(kn)$ definiert. Dabei bezeichnet $t(kn)$ die Dauer des Prozesses im Knoten kn .

Direkter Vorgänger: Jeder Knoten kn des Netzwerkes hat einen direkten Vorgänger³¹ kn_{prev} und einen direkten Nachfolger³² kn_{next} auf der gleichen Scheduling-Unit (Anlagekomponente). Der Startzeitpunkt st_{prev} wird berechnet als

$$st_{prev}(kn) := st(kn_{prev}) + t(kn_{prev}) \quad (4.2)$$

Vorgeschaltete und nachfolgende Produktion(en): Die vorgeschalteten Produktionsschritte resultieren aus der Volumenplanung, in der die Chargen für End- und Zwischenprodukte erstellt werden. Jede Charge chg hat eine Menge $CH_{src}(chg)$ der direkt vorgeschalteten Chargen und eine Menge $CH_{trg}(chg)$ der direkt nachgeschalteten Chargen.

Über den einer Charge zugeordneten Produktionsknoten kn_{PROD} kann so für jeden Knoten kn die vorgeschalteten und nachfolgenden Produktionsschritte und deren Knoten V_{kn}^{SRC} bzw. V_{kn}^{TRG} im Netzwerk³³ ermittelt werden. Sofern der aktuell ausgewählte Knoten nicht vom Typ kn_{PROD} ist, muß hier zunächst der nächste vorgeschaltete Produktionsknoten desselben Hyperknotens $[PROD]$ ermittelt werden.

Die Startzeit st_{src} berechnet sich dann als

$$st_{src}(kn) := \max_{src \in V_{kn}^{SRC}} (st(src) + t(src)) \quad (4.3)$$

Startzeitpunkte der Nachfolger: In jedem Iterationsschritt muß für den direkten Nachfolger kn_{next} der Startzeitpunkt $st_{prev}(kn_{next})$ anhand von 4.2 neu berechnet werden.

Wenn kn ein Produktionsknoten, d.h. vom Typ kn_{PROD} ist, müssen auch die Startzeitpunkte st_{src} der Nachfolger berechnet werden:

$$\forall trg \in V_{kn}^{TRG} \quad st_{src}(trg) := \max(st_{src}(trg), st(kn) + t(kn)) \quad (4.4)$$

Hier wird die Maximumsfunktion verwendet, da ein Nachfolger durchaus mehrere Vorgänger als nur den aktuell ausgewählten Knoten haben kann. Wegen (4.3) berücksichtigt $st_{src}(trg)$ bereits die Startzeitpunkte der weiteren Vorgänger.

³⁰NOWICKI UND SMUTNICKI bezeichnen den direkten Vorgänger auch als Maschinen-Vorgänger und die vorgeschalteten Produktionsschritte als Auftrags-Vorgänger [NS05].

³¹mit Ausnahme der Startknoten \circ

³²mit Ausnahme der Endknoten $*$

³³Die Produktion eines Vorgängers oder Nachfolgers kann durchaus auch auf einen anderen Tag terminiert sein. In diesem Fall wird der entsprechende Produktionsknoten für die Berechnung der Startzeit nicht in der Menge V_{kn}^{SRC} bzw. V_{kn}^{TRG} aufgenommen, da sich dieser in einem anderen Netzwerk befindet.

Der Algorithmus

Zunächst wird hier der **Makespan**-Algorithmus erklärt. Die einzelnen Berechnungen werden dann in den nachfolgenden Abschnitten beschrieben.

Algorithmus 4.6 (Makespan-Algorithmus)

1. *Initialisierung:*

- Fülle V_s mit der Menge der Startknoten \circ der Scheduling-Units.
- Setze $V_p := \emptyset$ und $V_k := \emptyset$
- Setze für $kn \in V \setminus V_s$: $st_{src}(kn) := 0$ und $st_{prev}(kn) := \infty$
- Setze für $kn \in V_s$: $st_{src}(kn) := 0$ und $st_{prev}(kn) := 0$
- Setze für alle Knoten der Typen kn_{WARTE} , kn_{TRANSFER} , kn_{ABFUELL} und kn_{LAGER} die Dauer $t(kn) := 0$

2. *Verarbeitung:* Solange $V_s \neq \emptyset$

- (a) Bestimme den Iterationsknoten kn mit Hilfe des Algorithmus 4.7.
- (b) Abbruch des Algorithmus mit einer ungültigen Lösung, wenn kein Knoten gefunden wurde (und $V_s \neq \emptyset$ ist).
- (c) Aktualisiere die Zeiten der Vorgänger mit Algorithmus 4.8.
Prüfe für alle Vorgänger, ob sie Element der Menge V_p sind. Wenn dies der Fall ist, prüfe, ob sie endgültig berechnet sind³⁴ und aktualisiere gegebenenfalls $V_p := V_p \setminus \{kn_{src}\}$ und $V_k := V_k \cup \{kn_{src}\}$
- (d) Berechne die Zeiten der parallelen Aktionen mit Algorithmus 4.9.
- (e) Aktualisiere den Startzeitpunkt $st_{prev}(kn_{next})$ des direkten Nachfolgers kn_{next} anhand von (4.2) und die Startzeitpunkte der nachfolgenden Produktionen anhand (4.4).
- (f) Setze $V_s := V_s \setminus \{kn\}$.
Wenn kn_{next} existiert, setze $V_s := V_s \cup \{kn_{next}\}$
- (g) Wenn kn ein Knoten vom Typ kn_{WARTE} , kn_{TRANSFER} , kn_{ABFUELL} oder kn_{LAGER} ist, oder als Knoten vom Typ kn_{PROD} genau eine nachfolgende Produktion als JiT-Zwischenprodukt beliefert, dann markiere kn als vorläufig berechnet, d.h. füge ihn der Menge V_p hinzu.
Ansonsten markiere kn als endgültig markiert, indem er der Menge V_k hinzugefügt wird.

3. *Ergebnis:*

Der **Makespan** ist der maximale Wert $e(kn)$ aller Knoten $kn \in V_k$.

■ A(4.6)

³⁴Ein Knoten ist endgültig berechnet, wenn alle seine Nachfolger berechnet sind. Dies wird in Definition 4.8 auf S. 104 genauer erläutert.

Auswahl des Iterationsknotens

In jedem Iterationsschritt ist aus der Menge V_s ein Knoten für die Berechnung auszuwählen. Dies wird in dem Algorithmus 4.7 genauer beschrieben. Aufgrund der Beziehungen eines Knotens kn zu der Vorläufer-Menge V_{kn}^{SRC} ist dabei nicht jeder Knoten zulässig. Für das Verfahren zur Auswahl des Iterationsknotens wird hier daher die folgende Eigenschaft definiert.

Definition 4.7 *Ein Knoten kn heißt frei einplanbar in einem Iterationsschritt, wenn er Element der Menge V_s ist und keiner seiner Vorgänger $src \in V_{kn}^{SRC}$ Element der Menge V_p ist.*

Bei der Auswahl eines Knotens kn sind die Auswirkungen auf die Startzeitpunkte anderer Knoten zu berücksichtigen. Ein Knoten hat *parallele Auswirkungen*, wenn die damit verbundene Aktion mehrere Ressourcen betrifft. Die Aktion verbindet dann zwei Knoten mit einer *Start-Start-Beziehung*. Dies ist bei allen Knoten der Fall, die mit Aktionen auf den *Ressourcenpools* verknüpft sind, also Reinigungen und Abfüllungen. Ebenso ist der Beginn einer Lagerung im Silo an den Start des Knotens $kn_{TRANSFER}$ geknüpft. Alle anderen Knoten haben nur *nachfolgende Auswirkungen*.

Algorithmus 4.7 (Auswahl des Iterationsknotens)

1. *Initialisierung*

Überprüfe für alle *frei einplanbaren* Knoten der Menge V_s , ob sie *nachfolgende* oder *parallele Auswirkungen* haben. Erstelle auf Basis dieser Eigenschaft die disjunkten Teilmengen V_s^p der Knoten mit *parallelen Auswirkungen* und V_s^n der Knoten mit *nachfolgenden Auswirkungen*.

2. *Auswahl*

Sei V'_s die Teilmenge, in der nach dem Iterationsknoten gesucht wird. Wenn $V_s^n \neq \emptyset$, setze $V'_s := V_s^n$, ansonsten verwende $V'_s := V_s^p$

Bestimme den Knoten $kn \in V'_s$ mit dem frühesten Produktionsende $e(kn)$, d.h. mit $e(kn) = \min \{e(kn) \mid kn \in V'_s\}$.

■ A(4.7)

In der Bestimmung des Iterationsknotens werden also Knoten mit nachfolgenden Auswirkungen bevorzugt ausgewählt, da hier nur die Zeiten für Knoten *einer* Scheduling-Unit festgelegt werden. Nach einer Zuordnung der Prozesse für Reinigung und Abfüllung können diese in der **Makespan**-Berechnung nicht mehr verändert werden. Dies geschieht erst durch eine entsprechende Optimierung³⁵. Durch die möglichst spät im Verlauf des Algorithmus stattfindende Zuordnung und Berechnung dieser Prozesse kann hier die Auswahl möglichst groß gehalten werden. Entsprechend werden dadurch möglichst wenige Schritte in der Reihenfolgeoptimierung der Poolprozesse³⁶ benötigt.

³⁵Siehe hierzu den Abschnitt *Nachbarschaften der Poolprozesse Abfüllung und Reinigung* auf S. 116 in Kapitel 4.5.3

³⁶Zur Definition eines Poolprozesses siehe Definition 4.3 auf S. 83.

Anstelle des frühesten Startzeitpunkt $st(kn)$ wie im DIJKSTRA-Algorithmus wird das früheste Ende $e(kn)$ als Entscheidungskriterium ausgewählt. In dem DIJKSTRA-Algorithmus wird eine (positive) Entfernung $d(u, v)$ zwischen den einzelnen Knoten vorausgesetzt. In der Modellierung des Reihenfolgeproblems sind dagegen die einzelnen Knoten ohne Bewertung der Kanten *direkt aufeinander folgend*. Zur Minimierung des **Makespan** ist in Analogie daher die *Dauer* des Knotens $t(kn)$ zu berücksichtigen und damit das Ende $e(kn)$ zu verwenden.

Nachkalkulation von Dauer und Zeitpunkten

Bei Knoten, die die Liegezeit im Reaktor oder eine Zwischenlagerung im Silo repräsentieren, ist die Dauer $t(kn)$ zu Beginn der **Makespan**-Berechnung nicht bekannt. Diese kann erst berechnet werden, wenn der Startzeitpunkt des nachfolgenden verarbeitenden Produktionsprozesses berechnet worden ist.

Ebenso kann der Startzeitpunkt von Chargen, die als JiT-Zutat in der Rezeptur einer nachfolgenden Produktion verwendet werden, erst dann endgültig berechnet werden, wenn die Startzeit dieser Produktion bekannt ist.

Wenn also in dem **Makespan**-Algorithmus ein Produktionsknoten berechnet wird, müssen zusätzlich auch noch einmal die Vorläuferknoten betrachtet werden, um eventuell deren Dauer oder Startzeitpunkt anzupassen. Mit dem Verfahren 4.8 werden diese Vorläufer bestimmt, deren Dauer oder Zeitpunkte neu berechnet bzw. aktualisiert werden müssen.

Algorithmus 4.8 (Berechnung der Vorläuferknoten)

Wenn der Iterationsknoten kn ein Knoten vom Typ kn_{PROD} ist, dann sind ausgehend von der zugehörigen Charge chg die Prozesse der Vorgänger von kn folgendermaßen zu berechnen:

1. *Direkte Vorläuferknoten:*

- Wenn eine Zwischenprodukt-Charge $chg_{src} \in CH_{src}(chg)$ als JiT-Zutat gekennzeichnet ist, dann berechne den Startzeitpunkt des zugehörigen Knotens kn_{PROD} anhand der Formel (4.8). Eine Liegezeit im Reaktor oder Lagerung im Silo findet in diesem Fall nicht statt.
- (Ansonsten:) Wenn eine Charge nur im Reaktor gelagert wird, dann berechne die Liegezeit $t(kn_{\text{WARTE}})$ anhand der Formel (4.11).
- (Ansonsten:) Wenn eine Zwischenprodukt-Charge in einem Silo zwischengelagert wird, dann berechne die Lagerdauer im Silo anhand der Formel (4.12).

2. *Knotenkette der Vorläuferknoten:*

Aktualisiere den Startzeitpunkt st_{prev} (kn_{next}) des direkten Nachfolgers kn_{next} der oben berechneten Knoten anhand von (4.2).

■ A(4.8)

Die Fallunterscheidungen dieses Verfahrens werden in den nachfolgenden Abschnitten näher erläutert. Im Anschluß an die Berechnung der hier genannten Vorläuferknoten ist in dem Algorithmus 4.6 im Schritt 2c noch die folgende Eigenschaft zu prüfen.

Definition 4.8 *Ein Knoten kn mit seiner zugehörigen Charge chg ist endgültig berechnet, wenn die Produktionsknoten (vom Typ kn_{PROD}) aller Chargen $chg_{\text{trg}} \in CH_{\text{trg}}(chg)$ in der Menge V_k sind.*

Der Vorläuferknoten kann also noch Element der Menge V_p sein und damit als noch nicht endgültig berechnet markiert sein. Wenn er aber als endgültig berechnet erkannt wird, ist er in die entsprechende Menge V_k zu verschieben.

JiT-Produktion: Wenn eine Charge als JiT-Zutat in der Rezeptur des nachfolgenden Produktionsauftrages benötigt wird, muß der Endzeitpunkt mit dem Startzeitpunkt der nachfolgenden Produktion übereinstimmen³⁷, d.h. für die beiden Knoten kn_{JiT} als JiT-Produktionsknoten zu der nachfolgenden Produktion kn gilt:

$$st(kn_{\text{JiT}}) + t(kn_{\text{JiT}}) = st(kn) \quad (4.5)$$

Wenn hier eine (positive) Offset-Zeit t_{offset} berücksichtigt wird, so ist

$$st(kn_{\text{JiT}}) + t(kn_{\text{JiT}}) = st(kn) + t_{\text{offset}} \quad (4.6)$$

Zur Berücksichtigung der JiT-Beziehung wird daher die Berechnung der Startzeit erweitert um $st_{\text{jit}}(kn)$. Diese Variable wird mit dem Wert $st_{\text{jit}}(kn) := 0$ initialisiert. Entsprechend berechnet sich die Startzeit $st(kn)$ dann als

$$st(kn) := \max(st_{\text{jit}}(kn), st_{\text{prev}}(kn), st_{\text{src}}(kn)) \quad (4.7)$$

Nach der ersten Betrachtung einer JiT-Charge wird der entsprechende Produktionsknoten kn_{JiT} in die Menge der noch nicht endgültig berechneten Knoten V_p verschoben. Erst nach der Berechnung der Startzeit der nachfolgenden Produktion im Knoten kn wird der Startzeitpunkt für alle Knoten trg_{jit} der vorhergehenden JiT-Chargen aus der Menge V_{kn}^{TRG}

$$st_{\text{jit}}(trg_{\text{jit}}) := st(kn) + t_{\text{offset}} - t(trg_{\text{jit}}) \quad (4.8)$$

neu berechnet und aus der Menge V_p in die Menge V_k verschoben.

Liegezeit eines Zwischenprodukts im Reaktor: Wenn eine Zwischenprodukt-Charge chg_{src} nicht direkt weiterverarbeitet werden kann, wird sie entweder im Reaktor oder in einem Silo zwischengelagert. Die Dauer der Lagerung im Reaktor wird auch als *Liegezeit* bezeichnet; die Dauer der Lagerung im Silo dagegen als (echte) *Lagerzeit*.

Eine Charge kann über die ganze Dauer bis zur weiteren Verarbeitung in einem Reaktor liegen oder nur so lange, bis sie in ein entsprechendes Silo umgelagert wird. Dementsprechend berechnet sich die Liegezeit entweder aus dem Zeitpunkt der Umlagerung kn_{TRANSFER} oder dem Zeitpunkt der letzten (Teil-) Menge der Charge in einem Nachfolgeprodukt.

Die Dauer eines Knotens vom Typ kn_{WARTE} wird zu Beginn der **Makespan**-Berechnung mit dem Wert $t(kn_{\text{WARTE}}) = 0$ initialisiert. Wenn der nachfolgende Knoten kn_{TRANSFER}

³⁷Bei der Erstellung der Chargen muß daher berücksichtigt werden, daß die Menge CH_{trg} nur aus einem Element besteht, da ansonsten aufgrund mehrerer zeitlicher Beziehungen vom Typ (4.6) eine zulässige Lösung nicht garantiert werden kann.

der Knotenkette eine Umlagerung in ein Silo durchführt, dann wird die Dauer berechnet als

$$t(kn_{\text{WARTE}}) := st(kn_{\text{TRANSFER}}) - st(kn_{\text{WARTE}}). \quad (4.9)$$

Wenn die Charge chg_{src} dagegen auf dem Reaktor verbleibt, ist zu prüfen, ob bereits der Produktionsbeginn $st(kn_{\text{PROD}})$ aller Chargen $chg_{trg} \in CH_{trg}(chg)$, in denen die Charge weiter verarbeitet wird, berechnet worden ist. Dieser als $st(kn_{trg})$ bezeichneter Zeitpunkt ist

$$st(kn_{trg}) := \max \{st(chg_{trg}) | chg_{trg} \in CH_{trg}(chg_{src})\} \quad (4.10)$$

Wenn nun das Zwischenprodukt nicht sofort zu Produktionsbeginn, sondern erst zu einem späteren (Offset-) Zeitpunkt t_{offset} hinzugefügt wird, dann ist die gesamte Liegezeit auf dem Reaktor

$$t(kn_{\text{WARTE}}) := st(kn_{trg}) + t_{offset} - st(kn_{\text{WARTE}}). \quad (4.11)$$

Lagerung eines Zwischenproduktes im Silo: Wenn eine Zwischenprodukt-Charge in einem Silo gelagert wird, dann sind für die Knoten kn_{LAGER} der Knotenkette [LAGER] der jeweilige Startzeitpunkt und die Lagerdauer zu berechnen. Wie in dem Kapitel 4.2.2 definiert wurde, beschreibt ein Knoten kn_{LAGER} nicht die Lagerdauer der Charge selber, sondern die Dauer einer Bestandssituation für ein Volumen des gelagerten Produktes.

Die Lagerdauer für eine Bestandssituation kann unabhängig davon, ob die Lagerung mit oder ohne Batch-Mix stattfindet, mit dem gleichen Verfahren berechnet werden. Die Knoten der Knotenkette [LAGER] sind in der Reihenfolge ihrer Einlagerung sortiert.

Für jede eingelagerte Charge chg_{src} in dieser Knotenkette ist der Zeitpunkt der Auslagerung³⁸ $st(kn_{trg}) + t_{offset}$ für die nachfolgende Produktion in der Charge $chg_{trg} \in CH_{trg}(chg_{src})$ zu bestimmen. Da ein Knoten kn_{LAGER} auch eine Bestandsbeziehung für eine früher eingelagerte Charge darstellen kann, sind zusätzlich auch deren Zeitpunkte der Auslagerung zu bestimmen.

Diese Zeitpunkte sind in der zeitlichen Reihenfolge ihres Auftretens zu sortieren und beginnend bei dem ersten Knoten der (Teil-) Knotenkette den Bestandsknoten kn_{LAGER} zuzuordnen. Eine Teil-Knotenkette ist dabei die Folge aller Knoten kn_{LAGER} , beginnend bei dem ersten Knoten, der zeitgleich mit der Einlagerung zum Zeitpunkt $st(kn_{\text{TRANSFER}})$ der Umlagerung startet, und endend mit dem letzten Knoten vor der nächsten Einlagerung.

Jedem Knoten kn_{LAGER} sind damit zwei Bestandsaktionen zugeordnet: Die zuletzt durchgeführte Einlagerung einer Charge chg_{src} mit dem entsprechenden Volumen und der Einlagerungszeit $st(kn_{src})$ und die nächste durchzuführende Auslagerung für eine Charge chg_{trg} mit dem benötigten Volumen und dem Zeitpunkt der $st'(kn_{trg}) := st(kn_{trg}) + t_{offset}$.

Aus der Modellierung ist bekannt, daß nicht unbedingt $st(kn_{src}) < st'(kn_{trg})$ gilt. Weiterhin ist für jeden Knoten der Teil-Knotenkette der Zeitpunkt der nachfolgenden Einlagerung $st(kn_{nE})$ zu bestimmen. Wenn es keine nachfolgende Einlagerung gibt, dann ist hier der Wert $st(kn_{nE}) = \infty$ zu setzen.

Die Dauer der einzelnen Bestandssituationen in einem Knoten kn_{LAGER} berechnet sich - beginnend bei dem ersten Knoten in der Teil-Knotenkette - dann als

$$t(kn_{\text{LAGER}}) := \max(0; \max(st'(kn_{trg}), st(kn_{nE})) - st(kn_{\text{LAGER}})). \quad (4.12)$$

³⁸Auch hier ist eine Offset-Zeit t_{offset} zu berücksichtigen.

Diese Berechnung wird iterativ bis zum letzten Knoten der Teil-Knotenkette fortgesetzt, wobei vor jeder Berechnung der Startzeitpunkt $st(kn_{\text{LAGER}})$ anhand von (4.2) aktualisiert wird.

Kalkulation bei Parallelprozessen

Die Zeiten paralleler Aktionen sind zu berücksichtigen und zu berechnen, wenn der Iterationsknoten entweder ein Reinigungsknoten vom Typ kn_{CLEAN} ist oder ein Abfüllknoten vom Typ kn_{ABFUELL} . In beiden Fällen ist der Startzeitpunkt abhängig von der Zuordnung zu den Ressourcenpools. Dazu sei zunächst noch einmal auf die Modellierung in Abschnitt 4.2.3 (S. 83) verwiesen.

Jedem Knoten in dem Netzwerk ist eine vom Datentyp abhängige Datenstruktur zugeordnet. Bei Parallelprozessen verweisen zwei oder mehr Knoten (vgl. Abb. 4.6 auf S. 83) auf dasselbe Datenfeld. Vor der ersten Berechnung des **Makespan** ist dabei auf den Scheduling-Units noch kein Knoten eingeplant (vgl. Abb. 4.7 auf S. 85) und dementsprechend noch keine Verknüpfung für die Parallelprozesse erstellt. Die Berechnung der Startzeitpunkte für einen Parallelprozeß auf einem Reaktor oder in einem Silo erfolgt dann über die Möglichkeit, einen entsprechenden Knoten in einer Scheduling-Unit des entsprechenden Ressourcenpools einzuplanen.

Für eine Abfüllung oder Reinigung ist daher mit dem Verfahren 4.9 zunächst zu bestimmen, ob in der aktuellen Reihenfolge ein Parallelknoten zugeordnet ist. Wenn dies nicht der Fall ist, dann muß mit dem Algorithmus 4.10 erst ein neuer Knoten für die parallele Aktion in dem Graphenmodell eingefügt werden. Im Falle einer Abfüllung müssen dann zusätzlich mit dem Verfahren 4.11 die Dauer und Startzeitpunkte in der Knotenkette des Reaktors aktualisiert werden.

Algorithmus 4.9 (Berechnung paralleler Aktionen)

Für den zu berechnenden Iterationsknoten kn in dem Scheduling \mathcal{S} prüfe, ob über die zugeordnete Datenstruktur bereits ein Knoten kn_{PARALLEL} auf einer Scheduling-Unit des entsprechenden Ressourcenpools verknüpft ist.

1. *Startzeitpunkt:*

- Wenn eine Verknüpfung besteht, dann bestimme den zugeordneten Knoten kn_{PARALLEL} und darauf aufbauend den zugehörigen Startzeitpunkt $st(kn_{\text{PARALLEL}})$ nach (4.14).
- Ansonsten erzeuge den Knoten kn_{PARALLEL} zu einem frühestmöglichen Zeitpunkt mit Algorithmus 4.10.

2. *Berechnung:*

Aktualisiere den Startzeitpunkt $st(kn)$ und die aller verknüpften Knoten.

- Wenn kn vom Typ kn_{CLEAN} ist, dann aktualisiere den Startzeitpunkt von kn anhand von (4.15).
- (Ansonsten:) Wenn kn vom Typ kn_{ABFUELL} ist, dann aktualisiere den Startzeitpunkt von kn anhand von Algorithmus 4.11.

Startzeitpunkt und Dauer bei Reinigungen: Die Dauer einer Reinigung $t(kn_{\text{CLEAN}})$ ist ein konstanter Vorgabewert der Planung und reihenfolgeabhängig von den Produkten die vorhergehend und nachfolgend auf einem Reaktor produziert oder in einem Silo gelagert werden. Wenn keine Reinigung benötigt wird (d.h. $t(kn_{\text{CLEAN}}) = 0$ ist), dann muß im Algorithmus 4.9 keine weitere Zuweisung und Berechnung durchgeführt werden.

Wenn dem Iterationsknoten kn kein Parallelknoten kn_{PARALLEL} zugewiesen ist, dann muß zunächst ein solcher Knoten gesucht werden.

Algorithmus 4.10 (Bestimmung eines Parallelknotens)

Bestimme den Parallelknoten kn_{PARALLEL} für eine Reinigung oder eine Abfüllung.

- Sei \mathcal{U}_{Res} die nichtleere Menge aller Scheduling-Units, die die Arbeitsreihenfolge einer „Ressource“ des Ressourcenpools *Reinigungsmitarbeiter* oder *Abfüllung* (eines Abfülltyps) abbilden.
- Es ist kn der einzuplanende Prozeß (Reinigung oder Abfüllung) mit der Dauer $t(kn)$ und dem frühestmöglichen Startzeitpunkt³⁹ $st_{\text{prev}}(kn)$.

1. *Suche:*

Für alle $u \in \mathcal{U}_{\text{Res}}$:

Durchlaufe die Knotenkette von u und prüfe, ob ein Knoten kn_u eingefügt werden kann, der die Bedingungen (4.13) und (4.14) erfüllt.

2. *Neuer Knoten:*

Der zu erzeugende Parallelknoten kn_{PARALLEL} ist der Knoten mit

$$st(kn_{\text{PARALLEL}}) = \min\{st(kn_u) | kn_u \text{ ist möglicher Knoten in } u \in \mathcal{U}_{\text{Res}}\}$$

■ A(4.10)

Wenn in einer Knotenkette $u \in \mathcal{U}_{\text{Res}}$ also ein Platz zum Einfügen des neuen Knotens kn_{PARALLEL} gefunden wurde, dann verschiebt er dadurch nicht den Startzeitpunkt des direkt nachfolgenden Knotens kn_{next} (und entsprechend aller weiteren nachfolgenden Knoten). Es gilt also

$$st(kn_{\text{PARALLEL}}) + t(kn_{\text{PARALLEL}}) \leq st_{\text{prev}}(kn_{\text{next}}) \quad (4.13)$$

Weiterhin beginnt die Reinigung oder Abfüllung nicht früher, als es durch den Iterationsknoten kn vorgegeben ist.

$$st_{\text{src}}(kn_{\text{PARALLEL}}) := st(kn) \quad (4.14)$$

Der tatsächliche Startzeitpunkt berechnet sich dann aus (4.1) (auf S. 100). Dementsprechend muß abschließend über die Verknüpfung der Datenstruktur der Startzeitpunkt des Iterationsknotens kn aktualisiert werden.

$$st_{\text{src}}(kn) := st(kn_{\text{PARALLEL}}) \quad (4.15)$$

³⁹Vgl. Formel (4.2) auf S. 100.

Startzeitpunkt und Dauer bei Abfüllungen: Die Vorgehensweise zur Bestimmung der Parallelknoten für die Abfüllung erfolgt analog zu der Suche bei Reinigungsprozessen. Anstelle einer fest vorgegebenen Zeit ist hier die Dauer der Abfüllung $t(kn_{\text{ABFUELL}})$ jedoch volumenabhängig von dem Abfülltyp (Gebinde oder TKW). Weitere Parameter, etwa *unterschiedliche* Abfüllgeschwindigkeiten der Abfüll-*Stutzen*⁴⁰, werden in der Berechnung der Abfülldauer nicht berücksichtigt.

Anders als bei der Reinigung können bei der Abfüllung jedoch *zwei* Abfüllprozesse zeitgleich stattfinden. Da das Volumen einer Charge aus den Volumina mehrerer Kundenaufträge zusammengestellt werden kann, können hier unterschiedliche Gebindetypen benötigt werden. Entsprechend kann die Knotenkette [PROD] auch unterschiedlich expandiert⁴¹ werden. Daher sind hier auch ein oder zwei Parallelknoten⁴² zu bestimmen.

Das Volumen des abzufüllenden Produkts ist also nach Abfülltyp getrennt zu ermitteln. Entsprechend ist für jedes Volumen eine eigene Abfülldauer zu berechnen.

Insbesondere wenn also zwei verschiedene Abfüllungen stattfinden, ist eine zusätzliche Berechnung in dem *Makespan*-Algorithmus notwendig. Der Hyperknoten [PROD] wird in die längstmögliche Knotenkette aus Abb. 4.3 expandiert. Für die Abfüllung wird dabei die Teilkette

$$\dots \rightarrow kn_{\text{ABFUELL}} \rightarrow kn_{\text{WARTE}} \rightarrow kn_{\text{ABFUELL}} \rightarrow kn_{\text{ABFUELL}} \rightarrow \dots$$

benötigt. Es werden also jeweils drei Knoten vom Typ kn_{ABFUELL} in der Knotenkette des Reaktors erzeugt. Dies resultiert aus der folgenden Fallunterscheidung (siehe hierzu Abb. 4.2 auf S. 78).

- **Die Abfüllungen finden nicht zeitgleich statt** (Abb. 4.2 a))

Nach der ersten Abfüllung verbleibt eine Restmenge im Reaktor, was zunächst eine weitere Wartezeit begründet. Dies begründet die Liegezeit kn_{WARTE} des Produktes.

- **Die Abfüllungen finden mit zeitlicher Überschneidung statt**
(Abb. 4.2 b) und c))

In diesem Fall findet während der gesamten Dauer wenigstens eine Abfüllung statt. Die Liegezeit wird mit $t(kn_{\text{WARTE}}) = 0$ berechnet.

Die Berechnung der Abfüllzeiten muß dann eine zeitliche Überschneidung der beiden Abfüllprozesse berücksichtigen. Diese Überschneidung der Knoten wird mit dem folgenden Verfahren 4.11 berechnet.

Algorithmus 4.11 (Berechnung der Abfüllzeiten)

Dieses Verfahren berechnet und aktualisiert die Dauer und Startzeitpunkte aller Knoten in der Knotenkette [PROD] nach der Herstellung eines Endproduktes.

- Gegeben ist der Iterationsknoten kn vom Typ kn_{ABFUELL} und je nach Zusammensetzung der Charge aus den Kundenaufträgen und den benötigten Abfülltypen der Kundenaufträge ein oder zwei Parallelknoten $kn_{\text{PARALLEL}}(i)$ $i \in \{1, 2\}$ mit ihren jeweiligen Startzeitpunkten und der volumenabhängigen Dauer.

⁴⁰Vgl. hierzu die Abschnitte *Abfüllung* auf S. 27 und *Anlagen-Charakteristik* auf S. 71.

⁴¹Vgl. Abb. 4.3 auf S. 79

⁴²Vgl. Abb. 4.6 auf S. 83

- Aus dem Makespan-Algorithmus 4.6 sind die Mengen V_s der zu berechnenden Knoten und V_k der endgültig berechneten Knoten gegeben.

1. *Startzeit:*

Bestimme die frühestmögliche Startzeit $st(kn_{\text{ABFUELL}})$ einer der beiden Abfülltypen als

$$st(kn_{\text{ABFUELL}}) := \min(st(kn_{\text{PARALLEL}}(1)), st(kn_{\text{PARALLEL}}(2))).$$

2. *Liegezeit des Endproduktes:*

Bestimme den zu kn vorhergehenden Knoten vom Typ kn_{WARTE} und berechne die Liegezeit analog zu (4.9) als

$$t(kn_{\text{WARTE}}) := st(kn_{\text{ABFUELL}}) - st(kn_{\text{WARTE}}).$$

Wegen der Modellierung ist noch ein Transferknoten kn_{TRANSFER} der Dauer $t(kn) = 0$ zu berücksichtigen. Aktualisiere dessen Startzeitpunkt mittels (4.2).

3. *Abfüllung:*

Wenn nur *eine* Abfüllart verwendet wird, dann ist aufgrund der Modellierung nur ein Knoten vom Typ kn_{ABFUELL} vorhanden. Die Dauer des Knotens ist gegeben durch $t(kn) = t(kn_{\text{PARALLEL}}(1))$. Der Startzeitpunkt wurde bereits in Schritt 1 berechnet und wird mittels (4.15) aktualisiert.

Wenn *zwei* Abfüllarten verwendet werden, dann sind Dauer und Startzeitpunkt von drei Abfüllknoten zu berechnen (vgl. Abb. 4.2 auf S. 78). Sofern dabei keine zeitliche Überschneidung zwischen den beiden Abfüllprozessen stattfindet, ist zusätzlich eine Liegezeit für das Restvolumen zu berechnen. Die Dauer des letzten Abfüllknotens ist dann auf $t(kn) = 0$ zu setzen.

4. *Markierung:*

Verschiebe alle Knoten, deren Dauern und Startzeitpunkte in den Schritten 1 bis 3 aktualisiert und berechnet wurden, in die Menge V_k der endgültig berechneten Knoten.

5. *Reinigung:*

Aktualisiere den Startzeitpunkt der nachfolgenden Reinigung kn_{CLEAN} in der Knotenkette von $[\text{PROD}]$ mittels (4.2) und nimm diesen Knoten neu in die Menge V_s der zu berechnenden Knoten auf.

■ A(4.11)

Zusammenfassung

Mit dem Algorithmus 4.6 und den weiteren Verfahren dieses Abschnitts, mit denen die Berechnung der einzelnen Teilschritte durchgeführt werden, existiert nun ein Verfahren, den Makespan zu berechnen, der durch die Modellierung aus Abschnitt 4.2 in dem Graphenmodell gegeben ist. Diese Weiterentwicklung des DIJKSTRA-Algorithmus ist dabei nicht nur eine einfache Vorwärtsberechnung der Startzeitpunkte, sondern es können auch

zeitlich rückwirkende Auswirkungen kalkuliert werden. Damit werden die Anforderungen einer JiT-Produktion in verfahrenstechnischen Herstellungsprozessen berücksichtigt. Hierzu wurde eine zusätzliche Eigenschaft der Knoten eingeführt bzw. gegenüber dem DIJKSTRA-Verfahren die weitere Knotenmenge V_p der noch nicht endgültig berechneten Knoten hinzugefügt.

Bei parallel auszuführenden Prozessen (Reinigung und Abfüllung), deren Zeit (im Falle der Abfüllung) sogar volumenabhängig sein kann, wird bei der Bestimmung der Reihenfolge zunächst noch nicht die Ressource bestimmt, die für diesen Prozeß benötigt wird. Hier kann die Aufgabenstellung der Reihenfolgeplanung sogar als ein Problem der ressourcenbeschränkten Projektplanung verstanden werden⁴³. Daher wurde der **Makespan**-Algorithmus so erweitert, daß eine solche Zuordnung der Ressourcen Abfüll-Stutzen oder Reinigungsmitarbeiter im Rahmen der Berechnung der Startzeitpunkte der einzelnen Knoten durchgeführt werden kann. Hierbei wird allerdings immer die frühestmöglich verfügbare Ressource ausgewählt. Dies kann insgesamt bei der Betrachtung des **Makespan** zu suboptimalen Lösungen führen. Auf diese Problematik wird in dem nachfolgenden Abschnitt näher eingegangen.

4.5 Reihenfolgeoptimierung

Wie in Kapitel 3.3 beschrieben wurde, kann eine Optimierung durchgeführt werden, wenn eine Ausgangslösung gegeben ist, indem darauf aufbauend (lokale) Verbesserungsverfahren angewendet werden. Mit der Erstellung einer Startlösung durch Algorithmus 4.2 und der Berechnung der Startzeitpunkte mit dem **Makespan**-Algorithmus 4.6 wird eine solche Ausgangslösung für die weitere Reihenfolgeoptimierung geschaffen.

In der weiteren Betrachtung dieser Arbeit werden nun drei Schritte in der Optimierung des **Makespan** angewandt. Zwei dieser Verfahren sind Verbesserungsheuristiken zur Optimierung der Reihenfolge. Dazu wird in Abschnitt 4.5.1 zunächst die Notwendigkeit der unterschiedlichen Betrachtung und damit der Entwicklung von zwei Optimierungsverfahren begründet. Da die Ergebnisse in Bezug auf die Reihenfolge aufeinander aufbauen, wird hier auch von einer

- *inneren* Optimierung, bei der die Abfolge der Abfüllungen und Reinigungen verändert wird, und einer
- *äußeren* Optimierung, bei der die Reihenfolge der Chargen betrachtet wird, gesprochen.

Dies wird in dem nachfolgenden Abschnitt näher erläutert. Für jede dieser beiden Ebenen wird anschließend (Kapitel 4.5.2 - 4.5.4) eine entsprechende Transformationsvorschrift auf Basis des Graphenmodells entwickelt. Die entsprechenden Akzeptanzvorschriften der heuristischen Optimierung werden in Kapitel 4.5.5 näher beschrieben.

Das dritte entwickelte Verfahren betrachtet die Zwischenlagerung als Möglichkeit, den **Makespan** zu reduzieren. Dies wird in Kapitel 4.5.6 näher beschrieben. Hierbei wird allerdings keine Veränderung in der Reihenfolge durchgeführt.

⁴³Zur Beschreibung der ressourcenbeschränkten Projektplanung siehe Fußnote 17 auf S. 49.

4.5.1 Optimierungsaufgaben

Wenn eine Reihenfolgeoptimierung durchgeführt wird, wird die Zuordnung der Chargen zu den Reaktoren und die Abfolge der Chargen auf diesen verändert. In dem vorliegenden Modell müssen jedoch neben der Herstellung auch andere Operationen berücksichtigt werden. Die Abfolge dieser Prozesse (durch Erzeugung der Parallelknoten im Algorithmus 4.10 zu einem frühestmöglichen Zeitpunkt) wird jedoch ohne Betrachtung der Auswirkungen auf die weitere Reihenfolge durchgeführt. Es ist daher nicht gewährleistet, daß hierbei eine **Makespan**-minimale Zuordnung erfolgt.

Die Reihenfolgeoptimierung wird daher in zwei Ebenen durchgeführt (vgl. Kasten *Reihenfolge-Optimierung* in Abb. 4.10 auf S. 89). In einer *inneren* Optimierungsebene wird die Zuordnung und Reihenfolge der Ressourcenpools optimiert. Diese Optimierung wird für jede Permutation der *äußeren* Optimierung durchgeführt, in der die Zuordnung der Chargen zu den Reaktoren oder die Reihenfolge der Chargen auf den Reaktoren verändert wird.

Die Durchführung der inneren Optimierung ist nur dann notwendig, wenn ein Block⁴⁴ eines kritischen Pfades auf der Scheduling-Unit eines Ressourcenpools vorhanden ist. Dies ist allerdings nur möglich, wenn die Anzahl der parallel stattfindenden Prozesse größer als die Anzahl der Scheduling-Units des jeweiligen Ressourcenpools ist. Vor der Durchführung einer inneren Optimierung ist daher zu prüfen, ob durch die Anwendung des Verfahrens überhaupt eine Verbesserung erreicht werden kann.

Das nachfolgende Beispiel beschreibt die Notwendigkeit der Optimierung in der Zuordnung und Abfolge der Abfüllprozesse.

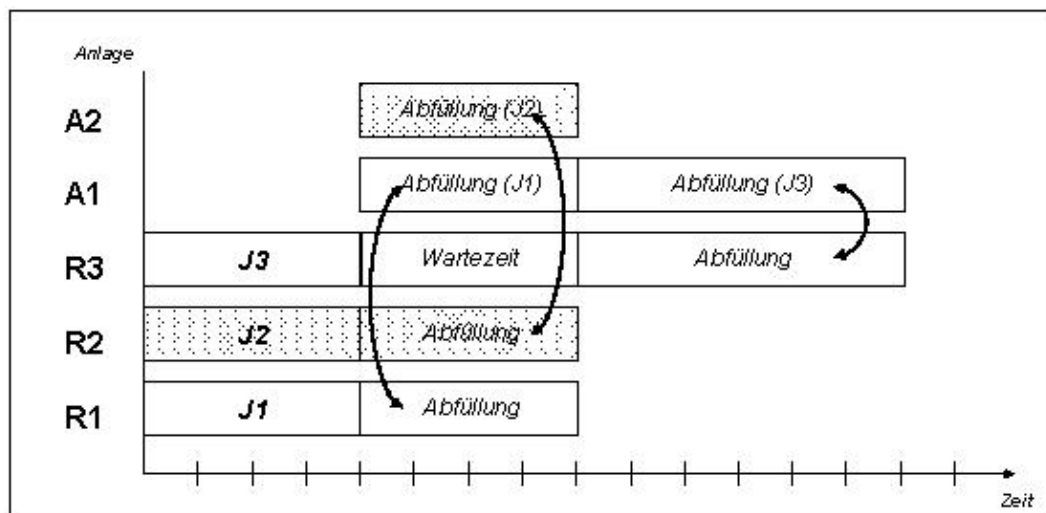
Beispiel 4.5 (Optimierung der Abfüllung) *Es ist eine optimale Reihenfolge für drei einstufige Aufträge J1, J2 und J3 mit gleicher Produktionsdauer zu finden. Dafür stehen drei Reaktoren, aber nur zwei Abfüll-Stutzen A1 und A2 zur Verfügung. Die Aufträge J1 und J2 haben das gleiche Volumen; das Volumen von Auftrag J3 ist 50% größer als das Volumen von J1. Demzufolge dauert die Abfüllung von J3 auch 50% länger.*

Bei der Berechnung des Makespan werden die Zeiten der Abfüllungen von J1 und J2 zuerst berechnet, da ihre Dauer geringer ist und der Prozeß der Abfüllung daher früher enden kann. Dementsprechend werden den Abfüllungen die jeweils frühest verfügbaren Abfüll-Stutzen A1 und A2 zugeordnet. Die Abfüllung von J3 muß daher so lange warten, bis einer der beiden Abfüll-Stutzen verfügbar ist. Nach der Berechnung des Makespan ist daher eine Reihenfolge wie in Abb. 4.13 gegeben.

Durch Vertauschen der Zuordnung und der Reihenfolge der Abfüllungen zu den Abfüll-Stutzen A1 und A2 kann jedoch eine optimale Reihenfolge wie in Abb. 4.14 erreicht werden.

Entsprechend ist auch die Zuordnung der Mitarbeiter zu den Reinigungsaufträgen und die Arbeitsabfolge für jeden einzelnen Mitarbeiter zu berücksichtigen, wie anhand des folgenden Beispiels ersichtlich wird.

⁴⁴Vgl. Definition 3.5 auf S. 53: Da die Reihenfolge anhand der *Scheduling-Units* jeweils für die gesamte Produktionsanlage modelliert wird, können Blöcke eines kritischen Pfades auch auf anderen „Maschinen“ als den Reaktoren bestehen.



Die Pfeile zwischen den Prozessen geben an, auf welchem Abfüll-Stutzen (A1, A2) die Chargen (J1, J2, J3) abgefüllt werden, die auf den Reaktoren (R1, R2, R3) gefertigt werden. Reinigungszeiten werden hier nicht dargestellt.

Abbildung 4.13: Ausgangssituation von Beispiel 4.5 zur Optimierung der Abfüllung

Beispiel 4.6 (Optimierung der Reinigung) Es ist eine optimale Reihenfolge für vier einstufige Aufträge J1, J2, J3 und J4 mit jeweils gleicher Produktionsdauer zu finden. Dafür stehen drei Reaktoren, aber nur zwei Mitarbeiter M1 und M2 zur Reinigung zwischen den Produktionsprozessen zur Verfügung (Der Einfachheit des Beispiels wegen wird die Abfüllung nicht betrachtet). Die Dauer der Reinigung ist reihenfolgeabhängig. Die Reinigung zum Abschluß einer Produktion beträgt 2 Zeiteinheiten (ZE). Die Dauer der Reinigung zwischen zwei Produktionsprozessen beträgt 3 ZE.

Es ist eine Startlösung gegeben, bei der die Zuordnung der Reinigungsprozesse zu den Mitarbeitern als Engpaß-Ressource im Makespan-Algorithmus noch bestimmt werden muß. Aufgrund der kürzeren Dauer der Reinigung werden zuerst die Zeiten der Reinigungsprozesse nach der Produktion von J1 und J2 berechnet und den Mitarbeitern M1 und M2 zu dem jeweils frühestmöglichen Zeitpunkt zugeordnet. Die Reinigung J3 → J4 muß daher so lange warten, bis einer der beiden Mitarbeiter verfügbar ist. Nach der Berechnung des Makespan ist daher eine Reihenfolge wie in Abb. 4.15 gegeben.

Durch Vertauschen der Zuordnung und der Reihenfolge der Reinigungsprozesse zu den Mitarbeitern kann jedoch eine optimale Reihenfolge wie in Abb. 4.16 erreicht werden.

4.5.2 Transformationen im Graphenmodell

Aus dem Abschnitt *Eigenschaften zur Optimierung* auf S. 51 ist bekannt, daß es bei einer Reihenfolgeoptimierung auf Basis eines Graphenmodells genügt, Transformationen einer bestehenden Lösung mit Knoten durchzuführen, die auf dem kritischen Weg liegen. Daher wird hier kurz die Vorgehensweise beschrieben, wie in dem vorliegenden Modell die kritischen Pfade ermittelt werden.

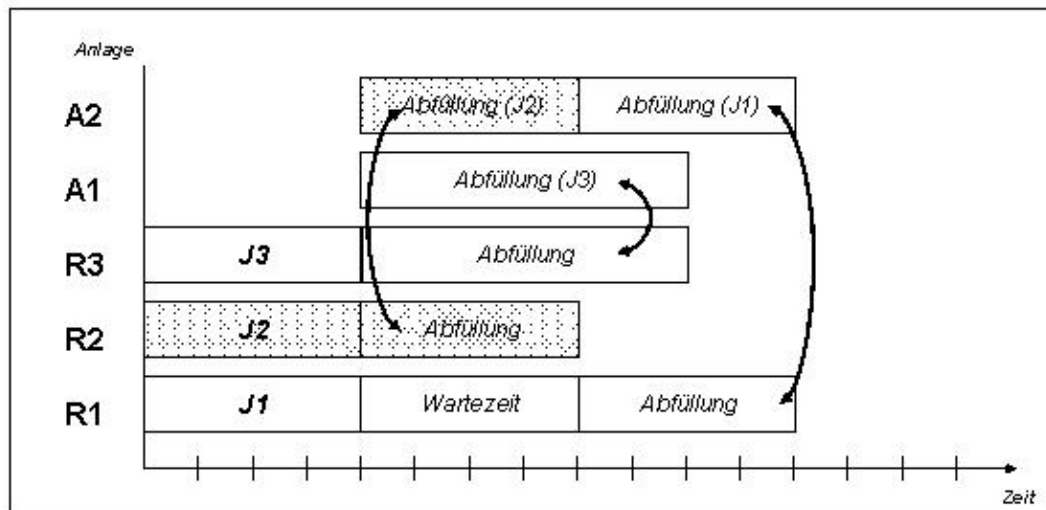


Abbildung 4.14: Optimierte Reihenfolge der Abfüllung in Beispiel 4.5

Bestimmung der kritischen Pfade

In einer Lösung \mathcal{S} werden zunächst die Scheduling-Units u mit dem größten lokalen Makespan $\text{MKS}(u)$, d.h. mit $\text{Makespan} = \max\{\text{MKS}(u) \mid u \in UR\}$, bestimmt. Ausgehend von dem jeweiligen Endknoten $*$ dieser Anlagekomponenten werden dann die kritischen Pfade gesucht.

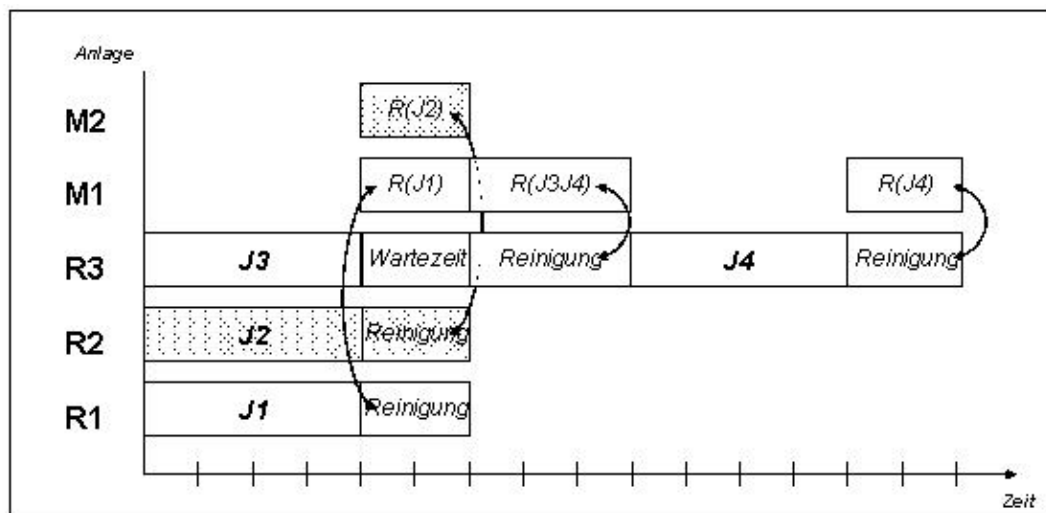
Hierbei sind die folgenden Punkte zu berücksichtigen, an denen es zu einer Verzweigung des kritischen Pfades oder zu einer Weiterführung auf einer anderen Scheduling-Unit kommen kann:

- Bei einer Produktion sind der direkt vorhergehende Prozeß und bestehende Zwischenprodukt-Chargen zu berücksichtigen.
- Bei den Poolprozessen Abfüllung oder Reinigung ist eine Weiterführung auf derselben Scheduling-Unit oder auf der Scheduling-Unit des Ressourcenpools zu prüfen.

Fortführung oder Verzweigung bei Produktionen: Wenn auf dem kritischen Weg ein Produktionsknoten kn_{PROD} vorhanden ist, dann ist zu prüfen, ob zwischen der vorhergehenden Reinigung kn_{CLEAN} und der Produktion eine Wartezeit stattfindet, d.h. ob $st(kn_{\text{PROD}}) > e(kn_{\text{CLEAN}})$ ist.

- Wenn keine Wartezeit besteht, dann wird die Bestimmung des kritischen Pfades auf der aktuell betrachteten Scheduling-Unit in Richtung auf den Startknoten \circ fortgeführt.

Wenn die Charge chg , deren Produktion durch den Knoten kn_{PROD} repräsentiert wird, zusätzlich noch die Produktion von Zwischenprodukten erfordert, dann ist hier zu prüfen, ob eine Verzweigung notwendig ist.



Die Pfeile zwischen den Prozessen geben an, welcher Reinigungsmitarbeiter (M1, M2) für eine Reinigung verantwortlich ist, die in der Abfolge der Chargen (J1, J2, J3, J4) auf den Reaktoren (R1, R2, R3) notwendig wird.

Abfüllungen werden hier nicht dargestellt.

Abbildung 4.15: Ausgangssituation in Beispiel 4.6 zur Optimierung der Reihenfolge

- Eine Verzweigung des kritischen Pfades wird für jede Zwischenprodukt-Charge chg_{src} durchgeführt, deren Produktionsende zeitnah zu dem Produktionsbeginn chg liegt, d.h. wenn $e(st_{src}) \geq st(chg)$ ist. In diesem Fall wird eine Kopie des bestehenden kritischen Pfades erstellt und der Pfad auf der Scheduling-Unit des zugehörigen Produktionsknotens kn_{PROD} von chg_{src} in Richtung auf den Startknoten \circ fortgesetzt.

Eine Fortsetzung des kritischen Weges durch Verzweigung oder Weiterführung auf einer anderen Scheduling-Unit, die durch eine Zwischenprodukt-Charge bedingt ist, findet also nur statt, wenn eine Zwischenprodukt-Charge direkt weiterverarbeitet wird. Dementsprechend ist in einem kritischen Pfad keine Liegezeit für Zwischenprodukt-Chargen oder gar eine Lagerung im Silo⁴⁵ vorhanden.

Fortführung oder Verzweigung bei Poolprozessen: Wenn auf dem kritischen Weg ein Poolprozeß, d.h. eine Abfüllung oder eine Reinigung, vorhanden ist, dann muß geprüft werden, ob zwischen dem vorhergehenden Prozeß und diesem Prozeß eine Wartezeit stattfindet.

- Wenn eine Wartezeit besteht, dann wird der kritische Weg über den Parallelprozeß auf der zugeordneten Scheduling-Unit des Ressourcenpools weitergeführt.

Wenn die betrachtete Abfüllung oder Reinigung bereits selber der Parallelprozeß auf der Scheduling-Unit eines Ressourcenpools ist, dann muß geprüft werden, ob eine Wartezeit

⁴⁵Die Zwischenlagerung im Silo wird näher in Kapitel 4.5.6 betrachtet.

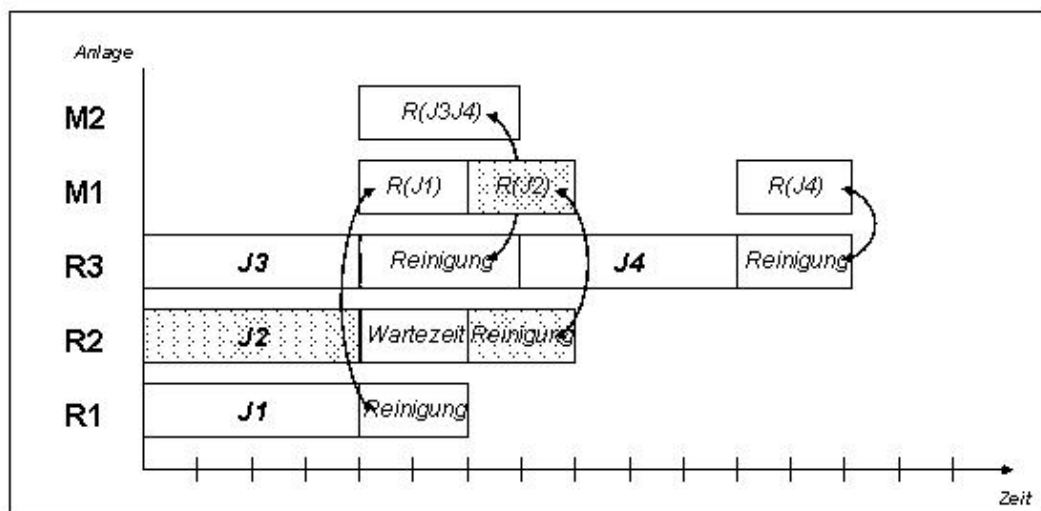


Abbildung 4.16: Optimierte Reihenfolge der Reinigung in Beispiel 4.6

zu dem vorhergehenden Prozeß im Knoten kn_{prev} derselben Scheduling-Unit bzw. zu dem vorhergehenden Prozeß auf dem Reaktor oder Silo besteht.

- Der kritische Pfad wird bei Parallelprozessen auf Scheduling-Units der Ressourcen-pools über die Knoten fortgeführt bzw. verzweigt, bei denen keine Wartezeit besteht.

Erzeugung der Nachbarschaften

Für das Graphenmodell können nun die Transformationsvorschriften entwickelt werden, die eine bestehende Lösung verändern. Die Gesamtheit aller möglichen Transformationen einer Lösung bilden die Nachbarschaft, in der nach einer neuen Lösung gesucht wird.

Die Anzahl der Nachbarschaftslösungen kann insgesamt eingeschränkt werden, indem nur Transformationen mit Knoten vorgenommen werden, die auf einem kritischen Pfad der Ablaufreihenfolge liegen. Hierzu werden üblicherweise Move- oder Swap-Transformationen verwendet⁴⁶. SCHUSTER gibt einen Überblick über die Verwendung dieser beiden Verfahren für das klassische Job Shop-Problem und auch für den Sonderfall der *no wait*-Bedingung⁴⁷ ([Schu03], S. 81). Demnach ist in beiden Fällen die Anwendung einer Move-Operation der Swap-Operation in der Optimierung überlegen. Für die weitere Entwicklung der Verfahren zur inneren und äußeren Optimierung werden daher nur Move-Operationen verwendet.

⁴⁶Siehe hierzu auch Definition 3.8 auf S. 57.

⁴⁷Die *no wait*-Bedingung wird in Parameter β_2 in Abschnitt *Auftrags-Charakteristik* auf S. 11 beschrieben und entspricht der JiT-Eigenschaft in der Abfolge der Knoten in dem vorliegenden Graphenmodell.

4.5.3 Nachbarschaften der inneren Optimierung

Wie aus den Beispielen 4.5 und 4.6 ersichtlich wird, muß eine Optimierung nur dann gestartet werden, wenn vor einem Poolprozeß (Abfüllung oder Reinigung) eine Wartezeit besteht. Der entsprechende Ressourcenpool ist damit ein Engpaß in der bestehenden Reihenfolge. Dieses Kriterium ist allerdings nicht hinreichend, um zu prüfen, ob tatsächlich eine bessere Lösung existiert. Daher muß eine Optimierung immer gestartet werden, wenn eine solche Situation besteht.

Die Nachbarschaft, die bei der Optimierung der Ressourcenpools betrachtet wird, wird durch die Menge der möglichen Move-Operationen definiert. Bei dieser Transformation wird der zugeordnete Parallelprozeß eines Poolprozesses in der Reihenfolge der aktuell zugeordneten Scheduling-Unit verschoben oder einer anderen Scheduling-Unit desselben Ressourcenpools zugeordnet. Nach Durchführung einer Move-Operation müssen anschließend die Startzeitpunkte der verschobenen und aller direkt und indirekt nachfolgenden Prozesse neu berechnet werden. Hierzu wird der **Makespan**-Algorithmus neu ausgeführt, ohne daß eine neue Zuordnung der Parallelknoten (Algorithmus 4.9) durchgeführt wird.

Diese Nachbarschaft kann durch eine geeignete Auswahl der Parallelprozesse, die zu verschieben sind, und eine entsprechende Auswahl der Einfügapunkte für die neue Reihenfolge, die durch eine Move-Operation entsteht, reduziert werden. Diese Reduktion muß so durchgeführt werden, daß optimale Lösungen nicht aus dem Lösungsraum entfernt werden. In der Bestimmung der Nachbarschaft werden daher die folgenden Parameter für eine Move-Operation verwendet.

Auswahl der Parallelprozesse: Ausgehend von den Poolprozessen, denen eine Wartezeit vorausgeht, werden die entsprechenden Parallelprozesse bestimmt. Für jeden dieser Parallelprozesse wird anschließend der Block der mit diesem Parallelprozeß zusammenhängenden Poolprozesse bestimmt. Abweichend von der Definition 3.5 muß ein solcher Block dabei nicht zwingend auf einen kritischem Pfad liegen.

Definition 4.9 *Ein Block von Parallelprozessen ist hier also die Kette der Knoten vom Typ kn_{ABFUELL} (oder kn_{CLEAN}) auf einer Scheduling-Unit des Ressourcenpools Abfüllung (oder Reinigung), für die gilt, daß $e(kn) = st(kn_{\text{next}})$ ist.*

Dazu wird die Ausgangssituation bereits bekannter Beispiele betrachtet.

Beispiel 4.7 (Anwendung auf Beispiele 4.5 und 4.6) *In dem Beispiel 4.5 umfaßt in der Ausgangssituation von Abb. 4.13 der Block also die Abfüllungen der Chargen J1 und J3. Es genügt also, nur die Abfüllung einer dieser beiden Chargen zu verschieben.*

In dem Beispiel 4.6 ist der Block der Reinigungen in der Ausgangssituation von Abb. 4.15 die Endreinigung von J1 und die Reinigung zwischen den Herstellungsprozessen von J3 und J4. Auch hier kann nur wieder die Reihenfolge von zwei Operationen verändert werden.

Auswahl der Einfügapunkte: Der frühestmögliche Startzeitpunkt eines Poolprozesses $st_{\text{src}}(kn_{\text{prev}})$ bestimmt sich aus dem direkt vorgelagerten Prozess kn_{prev} (auf dem Reaktor oder dem Silo). Weiterhin folgt aus der aktuellen Reihenfolgeplanung der Endzeitpunkt $e(kn)$.

Der Parallelprozeß wird daher in einer Move-Operation vor oder hinter einen Knoten auf derselben oder einer anderen Scheduling-Unit desselben Ressourcenpools verschoben, so daß für den neuen Startzeitpunkt $st(kn_{\text{PARALLEL}})$ gilt $st_{\text{src}}(kn_{\text{prev}}) \leq st(kn_{\text{PARALLEL}}) \leq e(kn)$.

Beispiel 4.8 (Fortsetzung von Beispiel 4.7) *In der Ausgangssituation von Beispiel 4.5 kann die Abfüllung von Charge J1 alternativ auf dem Abfüll-Stutzen A2 durchgeführt werden oder in der Reihenfolge von A1 verschoben werden. Die Abfüllung von J1 kann also nur vor oder hinter die Abfüllung von J2 oder hinter die Abfüllung von J3 verschoben werden. Analog kann die Abfüllung der Charge von J3 nur vor oder hinter die Abfüllung von J2 oder vor die Abfüllung von J1 verschoben werden. Dabei ist zu beachten, daß das Verschieben von J1 hinter J3 die gleiche Reihenfolge erzeugt wie das Verschieben von J3 vor J1. Die Anzahl der Move-Operationen kann hier also weiter eingeschränkt werden. In der Ausgangssituation von Abb. 4.13 (a) umfaßt die Nachbarschaft nur die Instanzen, die aus 5 verschiedenen Move-Operationen erzeugt werden.*

Entsprechend können im Beispiel 4.6 die Reinigungen $R(J1)$ und $R(J3, J4)$ nur vor oder hinter die Reinigung $R(J2)$ verschoben werden, sowie $R(J1)$ und $R(J3, J4)$ in der Reihenfolge vertauscht werden. Ein Verschieben hinter die Reinigung $R(J4)$ wird nicht betrachtet, da dadurch nur eine schlechtere Lösung entstünde.

Berücksichtigung der Optimierungsziele: Das *Hauptziel* der Optimierung des Reihenfolgeproblems ist die Minimierung des **Makespan**. Daneben wird das *Nebenziel* der Minimierung der Summe der Endzeitpunkte je Anlagekomponente (Scheduling-Unit) verfolgt. Bei der Bestimmung der Nachbarschaften muß daher berücksichtigt werden, welches dieser beiden Ziele in dem jeweils aktuellen Optimierungsschritt verfolgt wird.

Wenn lediglich eine Minimierung des **Makespan** gesucht wird, wie dies in den Beispielen 4.5 und 4.6 möglich ist, dann kann die Nachbarschaft so eingeschränkt werden, daß bei der *Auswahl der Parallelprozesse* nur diejenigen Poolprozesse betrachtet werden, die auch auf einem *kritischen Pfad* liegen. Wenn dagegen die Summe der Endzeitpunkte je Scheduling-Unit minimiert werden soll, dann muß die *Auswahl der Parallelprozesse* wie in dem oben beschriebenen Abschnitt auch mit den Blöcken erfolgen, die nicht auf einem kritischen Pfad liegen.

Bewertung

Die Nachbarschaft, die durch die hier definierten Move-Operationen erzeugt wird, ist insgesamt sehr klein, da eine Abfüllung in der Reihenfolge eines Abfüllstutzens bzw. eine Reinigung in der Arbeitsabfolge eines Mitarbeiter nur um jeweils einen Platz nach vorne oder hinten verschoben wird. Demnach kann *eine* solche Operation nur an maximal $2 \cdot (n - 1) - 1$ andere Stellen der aktuellen Abfolge verschoben werden; dabei ist n die Anzahl der jeweiligen Parallelprozesse überhaupt. Es ist zu berücksichtigen, daß ein Verschieben *hinter* (bzw. *vor*) eine benachbarte Abfüllung (bzw. Reinigung) im Ergebnis gleich ist mit dem Verschieben dieser benachbarten Abfüllung (bzw. Reinigung) *vor* (bzw. *hinter*) den betrachteten Parallelprozeß. Bei m Abfüllungen oder Reinigungen, die in einem Block von Parallelprozessen liegen, umfaßt eine Nachbarschaft einer Move-Transformation in der inneren Optimierung also maximal $m \cdot (2(n - 1) - 1)$ Lösungen.

Wie weiterhin leicht erkennbar ist, ist die hier definierte Nachbarschaft symmetrisch und schwach optimal zusammenhängend⁴⁸. Diese Eigenschaften werden bei der Anwendung einer Verbesserungsheuristik verwendet.

4.5.4 Nachbarschaften der äußeren Optimierung

Bei der *eigentlichen* Reihenfolgeoptimierung der Chargen, die durch Änderung der Zuweisung der Chargen zu den Reaktoren oder Änderung der Reihenfolge der Chargen auf den Reaktoren stattfindet, wird lediglich das Hauptziel der Optimierung, die Minimierung des **Makespan**, betrachtet. Entsprechend muß eine Transformationsvorschrift nur auf die Chargen angewandt werden, die auf einem kritischen Pfad der aktuellen Lösung liegen.

Für die Reihenfolgeoptimierung der Chargen wurden zwei Transformationsvorschriften zur Bestimmung der Nachbarschaften verwendet:

- Neu-Einplanung einer Endprodukt-Charge
- Verschieben einer einzelnen Charge

Neu-Einplanung einer Endprodukt-Charge: Bei diesem Nachbarschafts-Verfahren wird eine zufällig ausgewählte Endprodukt-Charge mitsamt der verknüpften Vor- und Zwischenprodukt-Chargen aus der bestehenden Reihenfolge gelöscht. Dabei muß berücksichtigt werden, daß eine Charge als Zwischenprodukt für mehrere weitere Chargen dienen kann. Dementsprechend sind nicht nur alle Chargen der vorhergehenden Produktionsstufen, sondern auch alle verknüpften Chargen, die die gleichen Chargen als Zwischen- oder Vorprodukt benötigen, aus der Reihenfolgeplanung zu löschen. Anschließend wird die Endprodukt-Charge mit allen verknüpften Chargen mit dem Verfahren zum *Erzeugen einer Startlösung* (Algorithmus 4.2) wieder eingeplant.

In diesem Verfahren wird die Charge eines Endprodukts und aller verknüpften Chargen eingeplant. Hier werden ausgehend von einer Startlösung die *Blöcke der miteinander verknüpften Charge* permutiert. Dabei genügt es, nur die Endprodukt-Chargen aus der bestehenden Reihenfolge zu entfernen und neu einzuplanen, die selber oder für die wenigstens eine der Vor- oder Zwischenprodukt-Chargen auf dem kritischen Weg liegen. Da in dem Algorithmus 4.2 die Chargen immer nur am Ende einer bestehenden Reihenfolgeplanung eingeplant werden, erzeugt diese Vorgehensweise, im Gegensatz zu den bisher durch Move-Operationen vorgestellten Nachbarschaften, keine symmetrische Nachbarschaft.

Das entwickelte Verfahren kann daher nur zur Lösung von Permutations Job Shop-Problemen eingesetzt werden. Diese Nachbarschafts-Transformation kann besonders dazu eingesetzt werden, um bei der Optimierung schnell ein lokales Optimum zu verlassen.

Verschieben einer einzelnen Charge: Anders als bei dem vorherigen Verfahren kann auch nur eine einzelne Charge verschoben werden. Analog zu der Definition einer Nachbarschaft bei den Poolprozessen durch Move-Operationen werden auch hier die Chargen in der Reihenfolge des aktuellen Reaktors verschoben bzw. in der Reihenfolge eines anderen Reaktors neu eingeplant.

⁴⁸Siehe hierzu Definition 3.9 auf S. 58

Bei der Reihenfolgeplanung in der verfahrenstechnischen Industrie führt aber nicht jede Veränderung einer Lösung durch eine Move-Operation zu einer zulässigen Lösung. Dies sei an einem Beispiel näher erläutert.

Beispiel 4.9 (Unzulässige Lösungen durch Move-Operationen) Gegeben ist eine Produktionsanlage ohne Zwischenlagermöglichkeiten und zwei Reaktoren. Zwei jeweils zweistufige Aufträge J1 und J2 sind in der Reihenfolge wie Abb. 4.17 eingeplant.

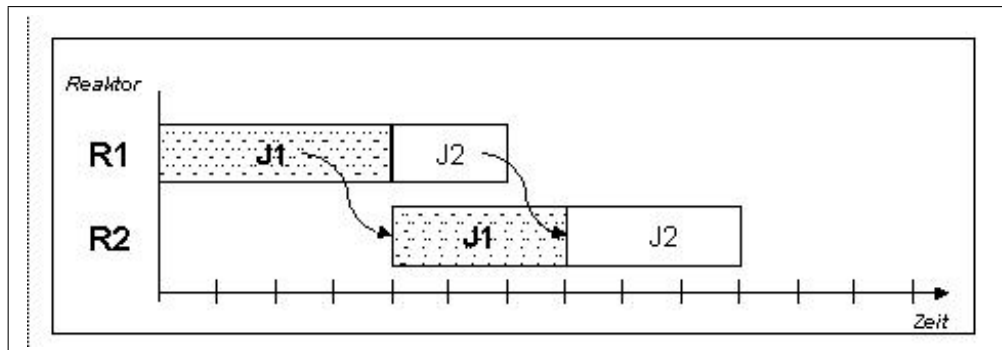


Abbildung 4.17: Ausgangssituation in Beispiel 4.9

Die bestehende Lösung kann durch keine Move-Operation einer einzelnen Charge verändert werden, so daß eine zulässige Lösung entsteht. Eine Reihenfolge, die durch ein Vorziehen von Auftrag J2 auf Reaktor R2 vor den Auftrag J1 wie in Abb. 4.18 entsteht, ist nicht zulässig, da der Auftrag J1 von Reaktor R1 nicht zwischengelagert werden kann.

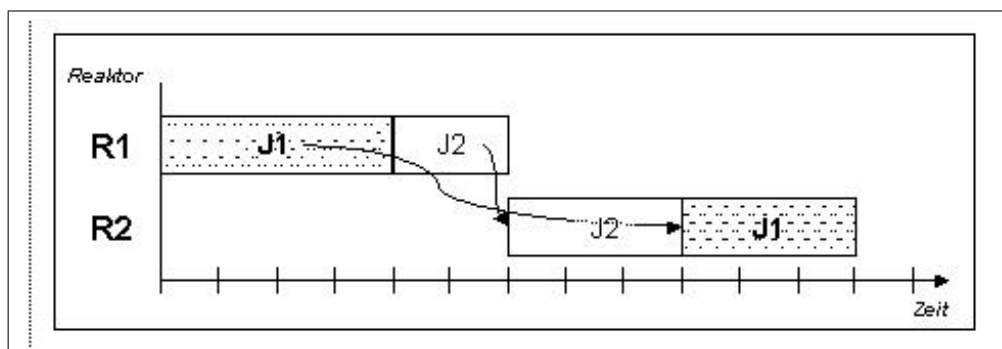


Abbildung 4.18: Unzulässige Zwischenlösung in Beispiel 4.9

Erst ein weiteres Vorziehen von Auftrag J2 vor Auftrag J1 auf Reaktor R1 wie bei Abb. 4.19 erzeugt dann wieder eine zulässige Lösung.

Für die Bestimmung einer Nachbarschaft wird daher eine *erweiterte Move-Operation* verwendet. Dabei wird zunächst bestimmt, an welche Stelle eine Charge *chg* verschoben werden soll. Diese Verschiebung kann in der Reihenfolge des aktuellen Reaktors erfolgen, oder die Charge kann in der Reihenfolge eines anderen Reaktors neu eingeplant werden. Diese Verschiebung erfolgt dann jedoch schrittweise, indem die Charge um jeweils einen Platz in der entsprechenden Richtung verschoben wird.

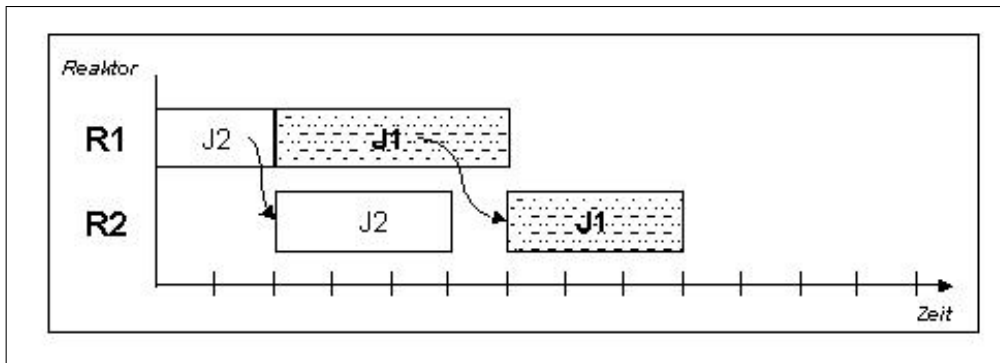


Abbildung 4.19: Abgeschlossene Move-Operation (Beispiel 4.9)

Sofern die so erzeugte Zwischenlösung zulässig ist, kann die Verschiebung der Charge fortgesetzt werden. Wenn allerdings durch eine solche Verschiebung um einen Platz eine unzulässige Zwischenlösung erzeugt wird, muß durch Verschiebung der verknüpften Chargen zunächst wieder eine zulässige Zwischenlösung erzeugt werden. Wenn die Charge chg in Richtung auf den Startknoten \circ verschoben wird, müssen daher evtl. die Chargen aus der Menge $CH_{trg}(chg)$ in der Reihenfolge des jeweils eigenen Reaktors in derselben Richtung verschoben werden. Wenn die Charge chg in Richtung auf den Endknoten $*$ verschoben wird, sind bei Erreichen einer unzulässigen Lösung entsprechend die Chargen aus der Menge $CH_{src}(chg)$ in derselben Richtung zu verschieben. Bei dem Verschieben einer Endprodukt-Charge muß entsprechend auch die Abfüllung neu eingeplant werden.

4.5.5 Anwendung eines heuristischen Suchverfahrens

In Kapitel 3.3 wurde bereits die Notwendigkeit beschrieben, daß im Verlauf der Lösungssuche auch schlechtere Ergebnisse akzeptiert werden müssen, um lokale Optima wieder zu verlassen. Die entsprechenden Heuristiken, die diese Notwendigkeit berücksichtigen, wurden bereits dort beschrieben.

Für die beiden Aufgaben der inneren und äußeren Optimierung wird im Rahmen dieser Arbeit das Simulated Annealing verwendet. Diese Heuristik ist in Algorithmus 3.1 auf S. 59 beschrieben. Da dieses Verfahren nicht mengenbasiert ist, ist auch sein Implementierungsaufwand geringer als bei dem Tabu Search oder evolutionsbasierten Verfahren. Zudem konvergiert das Simulated Annealing (bei ausreichender Laufzeit) gegen die beste Lösung⁴⁹ eines Optimierungsproblems.

Für die weitere Beschreibung des in dieser Arbeit entwickelten Verfahrens sind daher die Parameter zu beschreiben, die von dem Simulated Annealing verwendet werden. Dies sind die Starttemperatur T_0 , die Vorschrift zur Anpassung der Temperatur $\vartheta(\cdot)$ und das Abbruchkriterium $X(\cdot)$.

Starttemperatur: Zur Bestimmung der Starttemperatur wird die Vorschrift aus (3.7) verwendet, nach der

$$T_0 := -\frac{\Delta f_{max}}{\ln(p)}$$

⁴⁹Die Bedingungen hierfür werden im Abschnitt *Simulated Annealing* auf S. 60 beschrieben.

ist. Es ist also die Δf_{max} zu bestimmen und die anfängliche Akzeptanzwahrscheinlichkeit p festzulegen.

Wie im Abschnitt 4.5.3 festgestellt wurde, umfaßt die Nachbarschaft bei der *inneren Optimierung*, die aufgrund von Move-Transformationen erzeugt wird, nur wenige Lösungen. Daher kann die Differenz Δf_{max} relativ einfach berechnet werden. Ausgehend von einer Startlösung \mathcal{S} werden alle Transformationen $\text{op}(\mathcal{S})$ erzeugt. Anhand der in dieser Nachbarschaft gefundenen besten Lösung \mathcal{S}_{best} und schlechtesten Lösung \mathcal{S}_{worst} wird in dieser Arbeit der Wert

$$\Delta f_{max} := \text{Makespan}(\mathcal{S}_{best}) - \text{Makespan}(\mathcal{S}_{worst}) \quad (4.16)$$

verwendet.

Bei der *äußeren Optimierung* ist der Umfang der Nachbarschaft abhängig von der gewählten Transformationsvorschrift. Bei dem Verfahren zur *Neu-Einplanung einer Endprodukt-Charge* wird *genau eine* Endprodukt-Charge, die selber oder für die wenigstens eine ihrer Vor- oder Zwischenprodukt-Chargen auf einem kritischen Weg liegen, mitsamt diesen verknüpften Chargen aus der Reihenfolge entfernt und neu eingeplant. In der durch dieses Verfahren erzeugten Nachbarschaft sind daher höchstens n_{END} Lösungen enthalten (wobei n_{END} die Anzahl der Endprodukt-Chargen ist). Das hierbei verwendete Verfahren zum Erzeugen einer Startlösung berücksichtigt mit der Regel (ch) in Algorithmus 4.3 nur jeweils einen Reaktor, auf dem die Produktion eingeplant werden kann. Es ist damit sehr restriktiv. Insgesamt werden durch diese Transformationsvorschrift grobe Änderungen in der Reihenfolge erzeugt.

Eine feine Veränderung der Reihenfolge wird dagegen durch das *Verschieben einer einzelnen Charge* erzeugt. Die Nachbarschaft umfaßt hierbei eine weitaus größere Zahl von Lösungen⁵⁰, deren Evaluation somit sehr zeitaufwendig wird.

Unabhängig von dem gewählten Transformationsverfahren wird für den Wert Δf_{max} daher vorgeschlagen

$$\Delta f_{max} := \text{Makespan}(\mathcal{S}_{worst}) - \text{Makespan}(\mathcal{S}) \quad (4.17)$$

zu verwenden. Hierbei ist \mathcal{S} die aktuelle Lösung. Die schlechteste Lösung \mathcal{S}_{worst} und ihr Funktionswert $\text{Makespan}(\mathcal{S}_{worst})$ ist dagegen nicht bekannt, da nicht die gesamte Nachbarschaft evaluiert wird. Er ist daher zu ersetzen durch eine obere Schranke einer (zwischenzeitig) schlechteren Lösung. Dazu wird *innerhalb der bestehenden Reihenfolge* des betrachteten Planungstages die längste Durchlaufzeit $t_{DLZ}^{max}(\mathcal{S})$ einer Produktion, d.h. die Differenz zwischen dem Ende der Abfüllung einer Endprodukt-Charge und dem frühesten Beginn der Herstellung einer zugehörigen Vor- oder Zwischenprodukt-Charge bestimmt. Als Wert für $\text{Makespan}(\mathcal{S}_{worst})$ wird dann

$$\text{Makespan}(\mathcal{S}_{worst}) := \text{Makespan}(\mathcal{S}) + t_{DLZ}^{max}(\mathcal{S}),$$

und somit

$$\Delta f_{max} := t_{DLZ}^{max}(\mathcal{S}) \quad (4.18)$$

in dieser Arbeit verwendet.

Ein Anwender muß daher in der Verwendung dieses Optimierungsverfahrens zur Bestimmung der Starttemperatur nur die Akzeptanzwahrscheinlichkeit p festlegen. Die weiteren

⁵⁰Es wird nur jeweils eine Charge verschoben. Da hierbei Vor- oder Zwischenprodukt-Chargen, die auf einem kritischen Weg liegen, separat verschoben werden, wird schon eine größere Auswahl an zu verschiebenden Chargen betrachtet. Jede dieser Chargen kann wiederum an wenigstens eine weitere Stelle verschoben werden.

Werte sind modell- und datenabhängig und werden zur Laufzeit des Algorithmus bestimmt.

Temperaturanpassung: Die Anpassung der Temperatur erfolgt bei beiden Optimierungsproblemen anhand der Vorschrift (3.8) mit $T_i := \alpha \cdot T_{i-1}$. Dabei wird die Temperatur nur dann verändert, wenn eine bessere Lösung gefunden wurde. Ansonsten wird die aktuelle Temperatur T_i weiterverwendet. Ebenso wie der Parameter p in der Vorschrift (3.7) ist der Parameter $\alpha < 1$ durch einen Produktionsplaner frei einstellbar. In der Literatur werden hier Werte zwischen 0,8 und 0,99999 als erfolgreich angegeben ([Pu04], [Schn99], [Brü02]). Die Bestimmung dieses Parameters ist wiederum modell- und datenabhängig.

Abbruchkriterium: Ein Abbruch des Simulated Annealing kann aufgrund verschiedener Kriterien erfolgen. Das Optimierungsverfahren bricht immer ab, wenn aufgrund der Temperaturabsenkung eine schlechtere Lösung nicht mehr akzeptiert wird. Darüber hinaus können auch benutzerdefinierte Abbruchkriterien definiert werden. So kann durch den Benutzer eine Maximalzahl an Iterationen vorgegeben werden, nach der das Verfahren beendet wird. Durch einen solchen erzwungenen Abbruch kann die Lösungsfindung beschleunigt werden. Eine solche Vorgehensweise ist insbesondere dann möglich, wenn die solchermaßen gefundenen suboptimalen Lösungen bereits eine für einen Produktionsplaner ausreichende Qualität besitzen.

Für die in dieser Arbeit entwickelten Methoden, die zwischen einer *inneren* und einer *äußeren Optimierung* unterscheiden, wurde zusätzlich für die Reihenfolgeoptimierung der Abfüll- und Reinigungsprozesse das Verfahren 4.12 entwickelt, um eine lokale untere Schranke zu bestimmen. Wenn eine Lösung gefunden wurde, die dem Wert der unteren Schranke entspricht, kann daher die weitere Optimierung abgebrochen werden.

Optimierung von Abfüllung und Reinigung

Die Optimierung der Reihenfolge der Poolprozesse Abfüllung und Reinigung wird genau dann gestartet, wenn in der Abfolge der Prozesse auf einem Reaktor oder einem Silo vor einem Poolprozeß eine Wartezeit besteht. Der entsprechende Ressourcenpool ist damit eindeutig ein Engpaß in der bestehenden Reihenfolge.

Über die Anzahl der Scheduling-Units eines Ressourcenpools kann hier eine *lokale untere Schranke* bei einer gegebenen Reihenfolge der Chargen bestimmt werden. Diese Schranke kann dann als ein Abbruchkriterium in der Reihenfolgeoptimierung der Poolprozesse verwendet werden.

Eine lokale untere Schranke der inneren Optimierung

Der Wert einer lokalen unteren Schranke bei einer gegebenen Reihenfolge der Chargen kann durch Variation der Anzahl der Scheduling-Units, d.h. der Abfüll-Stutzen (Gebinde oder LKW) bzw. der Anzahl Mitarbeiter für die Reinigung, bestimmt werden.

Algorithmus 4.12 (Bestimmung einer lokalen unteren Schranke)

Gegeben ist ein Scheduling \mathcal{S}_0 , das eine Produktionsanlage mit n_R Reaktoren, n_S Silos, n_{LKW} Abfüllstutzen für LKW und n_{Geb} Abfüllstutzen für Gebinde darstellt. Für die Reinigungsprozesse stehen n_M Mitarbeiter zu Verfügung.

1. *Initialisierung:*

- Füge der Produktionsanlage in dem Scheduling \mathcal{S}_0 „virtuelle“ Abfüllstutzen und Mitarbeiter hinzu. Es werden $(n_R + n_S) - n_M$ zusätzliche Scheduling-Units für Reinigungs-Mitarbeiter, $n_R - n_{LKW}$ weitere Scheduling-Units für LKW-Abfüllstutzen und $n_R - n_{Geb}$ Scheduling-Units für LKW-Abfüllstutzen hinzugefügt.
- Lösche die Verweise der Poolprozesse auf die jeweiligen Parallelprozesse und berechne den **Makespan** mit Algorithmus 4.6 neu.
- Sei $n'_M (= n_R + n_S)$ die neue Anzahl der Reinigungs-Mitarbeiter, $n'_{LKW} (= n_R)$ die neue Anzahl der Abfüllstutzen für LKW und $n'_{Geb} (= n_R)$ die neue Anzahl der Abfüllstutzen für Gebinde-Abfüllung.

2. *Solange $(n'_{Geb} > n_{Geb})$, $(n'_{LKW} > n_{LKW})$ oder $(n'_M > n_M)$ ist:*

- Finde für jeden dieser drei Typen (*LKW-Abfüllung*, *Gebinde-Abfüllung*, *Reinigungs-Mitarbeiter*), für die die aktuelle Anzahl der Scheduling-Units größer als in dem ursprünglichen Scheduling \mathcal{S}_0 ist, die Scheduling-Unit, die als nächste zu löschen ist.

Die folgenden Kriterien werden angewandt, um zu untersuchen, ob eine Scheduling-Unit Löschkandidat ist:

- (a) Auf der Scheduling-Unit wird die geringste Anzahl an Parallelprozessen ausgeführt.
 - (b) Wenn mehrere Scheduling-Units mit gleicher Anzahl an Parallelprozessen gefunden wurden, dann bestimme unter diesen die Scheduling-Unit, die die geringste Gesamtdauer aller Parallelprozesse hat.
 - (c) Bei Gleichheit in Schritt 2b wähle eine beliebige Scheduling-Unit aus.
- Lösche die gefundene Scheduling-Unit und alle Prozesse auf dieser, sowie die Verweise auf diese Parallelprozesse aus dem Scheduling. Berechne den **Makespan** mit Algorithmus 4.6 neu.

■ A(4.12)

Der beschriebene Algorithmus reduziert also schrittweise die Anzahl der Scheduling-Units für die einzelnen Ressourcenpools. Bei der maximalen Anzahl n_R bzw. $n_R + n_S$ ist kein Engpaß vorhanden. Der Engpaß wird schrittweise erzeugt, indem die Anzahl der zusätzlich hinzugefügten Scheduling-Units wieder reduziert wird.

4.5.6 Zwischenlagerung

Im Rahmen der Reihenfolgeoptimierung, die in den vorherigen Abschnitten beschrieben wurde, wird nur die Abfolge der Chargen auf den Reaktoren verändert. Die Möglichkeit der Zwischenlagerung wird bei den Move-Operationen nicht berücksichtigt.

In diesem Abschnitt wird die Zwischenlagerung als eine weitere Möglichkeit der Optimierung beschrieben. Der hier vorgestellte Algorithmus 4.13 ist ein zusätzliches Verfahren, um das Nebenziel der Optimierung⁵¹, die *Minimierung der Summe der Endzeitpunkte je Reaktor*, zu verbessern. Dabei wird nicht die Reihenfolge der Produktionen verändert, sondern nur die *Liegezeit* der Chargen auf den Reaktoren verringert, um die verfügbare Zeit für die Durchführung weiterer Produktionen zu erhöhen.

Algorithmus 4.13 (Zwischenlagerung im Silo)

- Gegeben ist ein Scheduling \mathcal{S} , in dem die Chargen \mathcal{C} eingeplant sind.
 - Für die Produktionsanlage wird eine einheitliche (Durchschnitts-) Geschwindigkeit angenommen, mit der die Chargen zwischen zwei Anlagekomponenten transferiert werden. Die Transferdauer einer Charge chg ist linear abhängig von dem Volumen $V(chg)$ und wird mit $t_{pump}(V)$ bezeichnet.
1. Bestimme in dem Scheduling \mathcal{S} den Startzeitpunkt $st(kn_{\text{WARTE}})$ und die Dauer $t(kn_{\text{WARTE}})$ der ersten stattfindenden Liegezeit (Wartezeit) einer Charge.
 2. Bestimme für diese Charge den Zeitpunkt st_{silo} der frühestmöglichen Einlagerung in ein Silo.

Hierzu müssen die zulässigen Silos für eine Einlagerung ermittelt werden. Die Menge der zulässigen Silos bestimmt sich aus der Schnittmenge der Silos, in denen das Produkt laut Rezeptur gelagert werden darf, und den Silos, die über die Verrohrung mit dem produzierenden Reaktor verbunden sind.

Als drittes Kriterium muß in der Auswahl des Silos das gelagerte Produkt (unter Berücksichtigung der Möglichkeit von Batch-Mix) und das Volumen, das aufgrund des aktuellen Bestandes noch hinzugefügt werden darf, mit dem Produkt und dem Volumen der einzulagernden Charge verglichen werden.

Der frühestmögliche Zeitpunkt der Einlagerung in den einzelnen Silos ist dann der früheste Zeitpunkt $\geq st(kn_{\text{WARTE}})$ nach einer möglicherweise notwendigen Reinigung des Silos, sofern Batch-Mix nicht erlaubt ist. Wenn ein Batch-Mix des zu lagernden Produktes mit anderen Chargen erlaubt ist, dann ist dies der früheste Zeitpunkt, zu dem die Charge vollständig eingelagert werden kann.

3. Die tatsächliche Lagerdauer $t'(kn_{\text{WARTE}})$ ist dann

$$t'(kn_{\text{WARTE}}) := t(kn_{\text{WARTE}}) - (st_{silo} - st(kn_{\text{WARTE}})).$$

Eine Einlagerung der Charge in das Silo findet dann statt, wenn

$$t'(kn_{\text{WARTE}}) \geq c + 2 \cdot t_{pump}(V)$$

ist. Dabei ist $c \geq 0$ eine konstante Mindestlagerzeit.

⁵¹siehe Abschnitt *Zielsetzung* auf S. 74

4. Berechne die Zeitpunkte und Dauer der Prozesse in \mathcal{S} mit dem Makespan-Algorithmus neu.
5. Wiederhole die Schritte 1 bis 4 so lange, bis in einem Durchlauf keine Einlagerung mehr durchgeführt werden kann.

■ A(4.13)

Die Ungleichung in Schritt 3 entscheidet hier, ob es sinnvoll ist, eine Charge für die Dauer der Wartezeit bis zur Weiterverarbeitung im Silo zwischenzulagern. Eine Zwischenlagerung ist nur dann sinnvoll, wenn die Dauer der Ein- und Auslagerung durch die Transferprozesse nicht länger dauert als die eigentliche Wartezeit. Die Mindestlagerzeit c ist dabei ein als Parameter von einem Produktionsplaner einstellbarer Wert. Für die Berechnungen in dieser Arbeit wurde hier der Wert $c := 0$ angesetzt.

Beispiel 4.10 Anhand der Reihenfolge in Abbildung 4.20 (a) kann die Durchführung des Algorithmus 4.13 zur Zwischenlagerung im Silo dargestellt werden:

Die in dem Reaktor $R3$ hergestellte Zwischenprodukt-Charge des Auftrags $J2$ wird so lange auf dem Reaktor $R3$ zwischengelagert, bis die Produktion der Charge $J1$ auf dem Reaktor $R4$ beendet ist und die Endprodukt-Charge von $J2$ hergestellt werden kann. Die Wartezeit auf dem Reaktor (= Liegezeit) beträgt $t(kn_{\text{WARTE}}) = 4$ Zeiteinheiten (ZE).

Die Produktion des Zwischenproduktes von $J2$ endet zum Zeitpunkt $t = 3$ ZE, ein Silo steht aber erst zum Zeitpunkt $t = 4$ ZE für eine Zwischenlagerung zur Verfügung. Die tatsächliche Wartezeit beträgt daher nur $t'(kn_{\text{WARTE}}) = 3$ ZE. Bei einer einfachen Transferdauer von $t = 1$ ZE ist eine Zwischenlagerung im Silo also sinnvoll.

Durch die Zwischenlagerung von $J2$ im Silo (siehe Abbildung 4.20 (b)) kann die Herstellung des Endproduktes von $J3$ früher begonnen werden. Die Produktion der beiden Zwischenprodukt-Chargen $J3.a$ und $J3.b$ gibt hier jedoch den frühestmöglichen Produktionsbeginn $t = 6$ ZE vor.

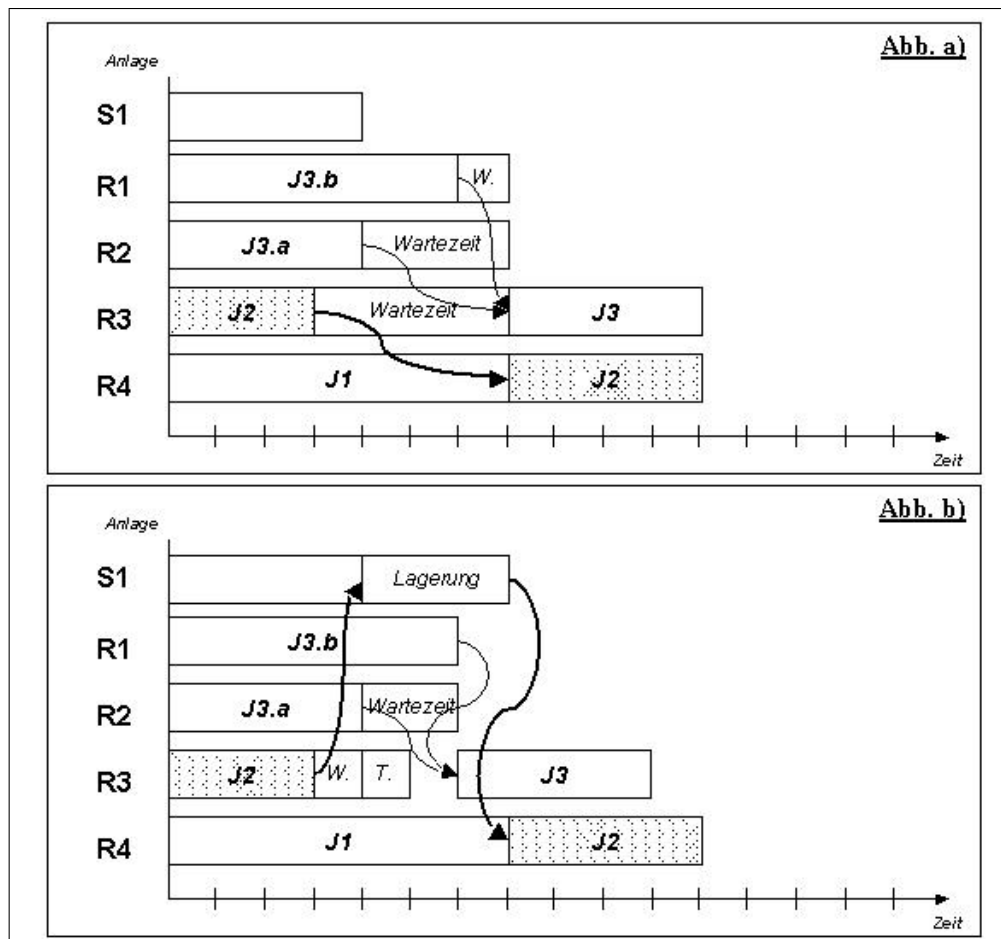


Abbildung 4.20: Zwischenlagerung von Chargen (Beispiel 4.10)

Praxis: Implementierung in einem Planungssystem

Der hier beschriebene Algorithmus zur Berücksichtigung der Zwischenlagerung im Rahmen der Reihenfolgeplanung geht davon aus, daß eine Charge immer vollständig in einem Silo eingelagert wird. Es werden keine Teilmengen gebildet, die in verschiedenen Silos gelagert werden. Dementsprechend ist bei der Anwendung dieses Verfahrens in der Produktionsplanung zu berücksichtigen, daß das Volumen der Silos, in denen die Produkte zwischengelagert werden sollen, mindestens so groß ist wie das Volumen der Reaktoren und somit der darin produzierten Chargen.

Kapitel 5

Kombinierte Chargen- und Reihenfolgeplanung

Im vorhergehenden Kapitel wurde die Reihenfolgeplanung vorgestellt, mit der für eine vorgegebene Menge von Chargen eine **Makespan**-optimale Belegung der Reaktoren für einen Tag gesucht wird. Zur Durchführung dieses Verfahrens müssen zuvor die Chargen erstellt werden.

In der Produktionsprozeßplanung erfolgt die Reihenfolgeplanung direkt nach einer zuvor durchgeführten Losgrößenplanung. In dieser Arbeit wird stattdessen die sog. *Volumenplanung der Chargen* oder einfacher *Chargenplanung* durchgeführt¹. Diese bestimmt die Volumina der Chargen und damit zugleich auch deren Anzahl. Damit ist die Chargenplanung der Losgrößenplanung gleich zu setzen, bei der terminierte Bedarfsmengen in terminierte Produktionsaufträge umgewandelt werden ([Dor90], S. 135).

In diesem Kapitel werden verschiedene Lösungsverfahren für die Erstellung der Chargen entwickelt. Hierzu werden im nachfolgenden Abschnitt zunächst die Planungsgrundlagen betrachtet (Kapitel 5.1) und in besonderer Weise die Vorgehensweise bei einem mehrperiodigen Planungshorizont vorgestellt. Kapitel 5.2 beschreibt genauer die Berechnung des Volumens einer Charge als Grundlage für die Volumenplanung. Abschließend werden die beiden Ebenen der operativen Produktionsplanung, die Volumen- und die Reihenfolgeplanung, in einem simultanen Optimierungsmodell miteinander verknüpft (Kapitel 5.3). Die verschiedenen Lösungsverfahren unterscheiden sich hierbei in der Bestimmung des einzuplanenden Volumens einer Charge. Für jedes dieser Verfahren wird zusätzlich ein Schritt zur Nachoptimierung und Anpassung an das Ergebnis der Reihenfolgeplanung durchgeführt.

5.1 Planungsgrundlagen

Ausgangslage einer Produktionsplanung sind die Informationen über die absatzseitig benötigten Bedarfsmengen. Bei einer Planung auf Lager sind dies die prognostizierten Verkaufsmengen, bei auftragsorientierter Planung die terminierten Kundenaufträge selber. Dabei ist für die Produktionsplanung im letzteren Fall anzunehmen, daß das Volumen der Kundenaufträge um die Menge reduziert wurde, die bereits als fertig produzierte Wa-

¹Siehe dazu die Beschreibung der Anforderungen einer *Volumenplanung* in Kapitel 3.1.

re im Lager frei verfügbar vorhanden ist. Weitere Grundlagen sind die Daten über die Kapazitäten und Verfügbarkeiten aller produktionstechnischen Ressourcen, d.h. der Anlagekomponenten, wie auch der Mitarbeiter. Auf der Input-Seite der Herstellung werden schließlich die Rohstoffe inkl. der Bestände und Lieferfristen berücksichtigt.

Die Planung wird dann in einzelne Teilschritte untergliedert:

- *Bedarfsermittlung*: Die absatzseitig benötigten Bedarfsmengen werden anhand der durch die Gozinto-Graphen dargestellte Produktionsstruktur in die zu produzierenden Bedarfsmengen der einzelnen Zwischenprodukte und Rohstoffe aufgeschlüsselt. Für die Produktionsmengen der Zwischenprodukte werden entweder Kundenaufträge über das Zwischenprodukt oder zusätzliche interne Aufträge zur Erzeugung von Halbzeugen erstellt.
- *Rohstoffprüfung*: Die erstellten Teilaufträge für die Zwischen- und Endprodukte sowie die Halbzeuge sind auf die Verfügbarkeit der benötigten Rohstoffe zu prüfen. Wenn die Rohstoffbestände nicht ausreichen, dann sind die Produktionstermine der Aufträge entsprechend anzupassen.
- *Chargenplanung*: Anhand der zeitlichen Auslastung der Reaktoren werden die Volumina der Chargen bestimmt und die Produktionsaufträge (Chargen) durch Zusammenfassen oder Splitten von Teilaufträgen über das jeweilige Produkt zusammengestellt.
- *Reihenfolgeplanung*: Die neu erzeugten Produktionsaufträge werden den Reaktoren zugeordnet und in der Reihenfolge optimiert. Die zeitliche Auslastung der Reaktoren verändert sich entsprechend zu dem Ergebnis der Reihenfolgeplanung.

Ein Planungsverfahren führt diese Teilschritte entweder *sequentiell* oder *simultan* durch. Bei einer simultanen Planung ist dabei gewährleistet, daß die zeitlichen Interdependenzen, die u.a. durch die reihenfolgeabhängigen Reinigungszeiten auftreten, ausreichend berücksichtigt werden ([Zäp82], S. 298). Der Informationsgrad aller Input-Daten bestimmt, ob eine Planung mit einem *deterministischen* oder einem *stochastischen Modell* durchgeführt wird ([DSV93] S. 64).

Die Bedarfsmengen definieren dabei mit ihren Terminen den *Planungshorizont*. Die über diesen Zeitraum zunehmenden Unsicherheiten können in einem Totalmodell nicht ausreichend berücksichtigt werden. Es können z.B. weitere (Eil-) Aufträge aufgenommen werden, oder bestehende Aufträge werden in Termin und/oder Menge geändert. Dieser Unsicherheit wird mit einer *rollierenden Planung* Rechnung getragen. Der Planungshorizont wird dazu in einzelne Planungsintervalle (*Perioden*) unterteilt, in denen das Optimierungsproblem jeweils für eine Teilmenge der Aufträge bearbeitet wird. Die Planung kann dabei periodenübergreifend erfolgen, d.h. der Lösungsraum des einzelnen Planungsintervalls ist nicht auf den Zeitraum der aktuellen Periode beschränkt.

Die Optimierung jedes Planungsintervalls wird dabei i.d.R. mit deterministischen Modellen durchgeführt. Die Ergebnisse der jeweils ersten Periode werden festgeschrieben und umgesetzt. Die nachfolgenden Perioden werden nach der Umsetzung der ersten Periode bei einer Änderung der Bedarfsmengen erneut berechnet.

Für die weitere Chargenplanung dieser Arbeit werden in diesem Abschnitt daher die Planungsdaten (*Kundenaufträge*) näher betrachtet. Für die Kombination der Planungsebenen

von Volumen- und Reihenfolgeplanung werden anschließend mögliche Optimierungskriterien diskutiert und ein Verfahren der rollierenden Planung vorgestellt.

5.1.1 Kundenaufträge

Für die Zusammenstellung der Chargen können bei der Auswahl der Kundenaufträge² verschiedene Eigenschaften verwendet werden.

- *ABC-Klassifizierung*: Die Kunden können nach ihrer Wichtigkeit in einer Prioritätsliste (ABC-Klassifizierung) geordnet sein.
- *Eilauftrag*: Einzelne Aufträge können aufgrund der Wichtigkeit des Kunden terminlich vor die Produktion aller anderen (auch verspäteter) Aufträge gezogen werden.
- *Sammellieferung*: Ein Kunde wünscht, verschiedene Produkte in einer Sendung geliefert zu bekommen.
- *Abfüllung*: Es können mehrere Aufträge über dasselbe Produkt mit unterschiedlichen Gebindeabfüllungen vorhanden sein.
- *Ertrag*: Produkte können unterschiedliche Deckungsbeiträge haben oder (abhängig vom Volumen) höhere oder niedrigere Umsätze erzielen.

Diese Eigenschaften können im Algorithmus 5.3 in Kapitel 5.3.1 alternativ zu den dort genannten Kriterien nach Vorgabe eines Produktionsplaners zur Auswahl eines Produktionsrezeptes verwendet werden.

5.1.2 Zielsetzungen

Für die Durchführung der Chargenplanung ist zu prüfen, anhand welcher Ziele diese Planung durchgeführt werden kann. Einerseits ersetzt die Volumenplanung der Chargen in ihren Ergebnissen, der „Zusammenstellung von terminierten Bedarfsmengen zu terminierten Produktionsaufgaben“ ([Dor90], S. 135), die Losgrößenplanung. Das Ziel der klassischen Losgrößenplanung ist es, die vorhandenen Aufträge so zusammenzufassen, daß die auftretenden Gesamtkosten aus Produktions-, Lagerhaltungs-, Rüst- und Reinigungskosten und evtl. Strafkosten minimiert werden (vgl. Kapitel 1.2.1).

Da solche Kostentreiber und Kostenfaktoren mit zunehmender Hierarchiestufe in der Produktionsplanung jedoch schwerer quantifizierbar werden, wird hier auf einfacher meßbare Leistungsgrößen als Ersatzziele der Planung zurückgegriffen ([Vah00], S. 96). Die Chargenplanung ist in der hierarchischen Produktionsplanung der Reihenfolgeplanung direkt vorgeschaltet. Daher wird hier diskutiert, welche Ersatzziele in der Volumenplanung verwendet werden können.

²Es sei darauf hingewiesen, daß im Rahmen dieser Arbeit ein Kundenauftrag als der Auftrag über die Menge eines *einzelnes* Produktes zu einem festen Termin verstanden wird; vgl. Definition 1.1 auf S. 6.

Optimierungskriterium Liefertermin

Eine Bewertung anhand der Produktions- oder Liefertermine kann für die einzelnen Chargen oder auch für die Kundenaufträge durchgeführt werden. Dazu wird zunächst die Abweichung zwischen den gewünschten und den tatsächlichen Produktions- bzw. Lieferterminen berechnet. Das Planungsproblem der Volumenplanung muß in der Zielfunktion dann die *Anzahl der Abweichungen* oder die *maximale Abweichung* aller Aufträge minimieren. Als Auftragsmenge werden hierbei die einzelnen Chargen (Produktionsaufträge), die Kundenaufträge oder die in Sammelaufträgen zusammengefassten Kundenaufträge betrachtet.

Für diese unterschiedlichen Auftragsstypen sind die Termine zuvor separat zu berechnen. Der *tatsächliche Liefertermin*³ eines *Sammelauftrags* ergibt sich als Maximalwert (d.h. der späteste Termin) der Liefertermine aller Kundenaufträge eines Sammelauftrages. Da das Volumen eines Kundenauftrags wiederum auf mehrere Chargen verteilt werden kann, ist der *tatsächliche Liefertermin eines Kundenauftrags* der späteste Produktionstermin aller Chargen, in denen das gewünschte Volumen hergestellt wird. Erst dann ist ein Kundenauftrag *vollständig produziert*. In umgekehrter Vorgehensweise wird der *gewünschte Liefertermin einer Charge* als Minimalwert von den Lieferterminen der Kundenaufträge berechnet, deren Volumen mit dieser Charge „bedient“ wird.

Optimierungskriterium Makespan

Im Ergebnis der Reihenfolgeplanung ist der **Makespan** möglichst gering zu halten. Ein Verfahren zur Volumenplanung der Chargen könnte daher so beschaffen sein, lediglich dieses Ziel zu unterstützen. Dies führt gleichzeitig dazu, daß eine möglichst gleichmäßige zeitliche Auslastung der Reaktoren erreicht wird.

Wenn in der Produktionsplanung allerdings eine Anlage gegeben ist, bei der ein Rezept auf Reaktoren unterschiedlicher Volumina gefertigt werden kann, kann die Minimierung des **Makespan** dazu führen, daß aufgrund der gleichmäßigen zeitlichen Belastung der Reaktoren das Produktionsvolumen auf eine größere Anzahl von Chargen verteilt werden muß. Diese Schwierigkeiten werden in dem Beispiel 5.1 am Ende dieses Abschnitts näher dargestellt.

Optimierungskriterium Chargenanzahl

Als dritte Möglichkeit kann bei der Optimierung die Anzahl der Chargen betrachtet werden, die in der Volumenplanung erstellt werden. Insbesondere wenn mit der Herstellung jeder Charge auch weitere Fixkosten entstehen, dann führt ein Ergebnis mit möglichst wenig Chargen zugleich zu einer kostenminimalen Lösung⁴.

Die minimale Anzahl benötigter Chargen n_{chg} ist gegeben durch das einzuplanende Bedarfsvolumen V_{bed} und das Maximalvolumen einer Charge \bar{V}_{chg} als $n_{chg} = \lceil V_{bed} / \bar{V}_{chg} \rceil$. Allerdings ist dabei zu berücksichtigen, daß das Maximalvolumen mit fortschreitender Einplanung weiterer Chargen geringer werden kann, weil durch die Reihenfolgeplanung die zeitliche Verfügbarkeit der entsprechenden Reaktoren verringert wird.

³Unter *Liefertermin* wird in allen Fällen der *Termin der Auslieferung* verstanden und mit dem Datum des Produktionsendes gleich gesetzt. Die Lieferzeit zu dem jeweiligen Kunden wird also nicht weiter berücksichtigt. Stattdessen muß der Kundenauftrag auf ein entsprechend früheres Datum terminiert werden.

⁴Hier müssen allerdings weitere Bedingungen erfüllt sein. Dazu zählt u.a., daß die Fixkosten unabhängig von dem herzustellenden Produkt immer gleich sind.

Zusammenfassung und Bewertung

Eine Volumenplanung wird in dieser Arbeit immer in Kombination mit einer Reihenfolgeplanung durchgeführt. Daher ist bei der Auswahl eines der drei zuvor genannten Optimierungskriterien darauf zu achten, daß eine ausreichende Verbindung zwischen den Aufgabenstellungen beider Planungsaufgaben besteht. Bei der Reihenfolgeplanung wird ein **Makespan**-optimaler Plan erstellt. Die Bestimmung der Chargenvolumina muß dieses Optimierungsziel also unterstützen.

Das zuerst betrachtete Optimierungskriterium der Abweichung zwischen den gewünschten und tatsächlichen Lieferterminen betrachtet vor allem Kundenaufträge eines mehrtägigen Planungshorizonts. Die Reihenfolgeplanung optimiert dagegen nur die Chargen eines einzelnen Tages. Eine Volumenplanung mit dieser Zielfunktion kann daher nicht mit der Reihenfolgeplanung kombiniert werden. Für die Problematik der Abweichung wird zusätzlich ein rollierendes Verfahren entwickelt. Dieses wird nachfolgend in Kapitel 5.1.3 beschrieben.

Die beiden weiteren Optimierungskriterien werden zunächst in dem folgenden Beispiel näher diskutiert. Dieses zeigt den Zusammenhang zwischen dem **Makespan** und der *Anzahl der Chargen* auf.

Beispiel 5.1 (Zielfunktionen der Chargenplanung) *Gegeben ist eine Produktionsanlage mit den Reaktoren $R1$, $R2$ und $R3$ mit den Maximalvolumina $\bar{V}(R1) = \bar{V}(R2) = 1000$ Mengeneinheiten (ME) und $\bar{V}(R3) = 2000$ ME. Einzuplanen ist ein Gesamtvolumen an Kundenaufträgen von 4000 ME für ein Rezept, das auf diesen drei Reaktoren gefertigt werden kann. Die Produktionsdauer ist nicht volumenabhängig und beträgt immer 4 Zeiteinheiten (ZE). Die insgesamt zur Verfügung stehende Produktionszeit beträgt 12 ZE.*

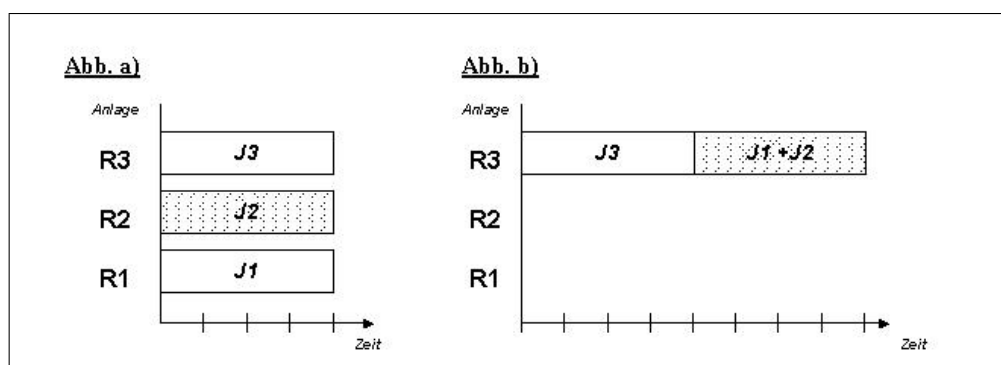
*Wenn die Minimierung des **Makespan** für die Chargenplanung vorgegeben ist, dann werden die drei Chargen $J1$, $J2$ und $J3$ so erstellt, daß der aktuelle **Makespan** einer Reihenfolgeplanung geringstmöglich überschritten wird (siehe Abb. 5.1 (a)).*

*Wenn die Minimierung der Chargenanzahl für die Chargenplanung vorgegeben ist, dann können die beiden Chargen $J1$ und $J2$ aus Abb. 5.1 (a) zusammengefasst werden. Die zeitliche Auslastung des Reaktors $R3$ (der lokale **Makespan**) steigt gegenüber den weiteren Reaktoren so lange an, soweit die insgesamt zur Verfügung stehende Produktionszeit nicht überschritten wird.*

Die Berücksichtigung des **Makespan** als Zielfunktion der Chargenplanung kann also dazu führen, daß das insgesamt herzustellende Volumen auf mehr Chargen verteilt wird. Die Anlagekomponenten werden zwar gleichmäßiger ausgelastet. Insgesamt wird die Produktionsanlage aber stärker belastet, da mehr Produktionsvorgänge gestartet werden müssen. Für die weitere Vorgehensweise wird daher die Minimierung der Anzahl herzustellender Chargen als Optimierungskriterium ausgewählt.

5.1.3 Rollierende Planung

Als weitere Optimierungsaufgabe kann nach der Reihenfolgeoptimierung und der Chargenplanung die *Einhaltung der Liefertermine* betrachtet werden. Dazu wird eine *rollierende Planung* durchgeführt. Da der Schwerpunkt dieser Arbeit aber in der Entwicklung der



Bei gleichem Volumen der Kundenaufträge werden in Abhängigkeit von der Zielfunktion unterschiedlich viele Chargen erstellt. Das Ergebnis der Reihenfolgeplanung ist abhängig von dem Ergebnis der Volumenplanung.

Abbildung 5.1: Ergebnis der Chargenplanung bei unterschiedlichen Optimierungskriterien (Beispiel 5.1)

simultanen Chargen- und Reihenfolgeplanung liegt, wird an dieser Stelle keine weitere Evaluation für das Zielkriterium der Liefertermine oder ein anderes Zielkriterium durchgeführt.

Die in diesem Abschnitt vorgestellte rollierende Planung stellt daher nur ein *äußeres Gerüst* für die Planung von Aufträgen eines mehrperiodigen Planungshorizonts dar. In jeder Planungsperiode wird hier der Algorithmus 5.2 zur simultanen Chargen- und Reihenfolgeplanung aus Kapitel 5.3 gestartet. Hierbei kann zuvor nach Vorgabe eines Produktionsplaners geprüft werden, ob für die jeweiligen Produktionsaufträge genügend Rohstoffe zur Verfügung stehen, bzw. in welchem Umfang sie verfügbar sind⁵.

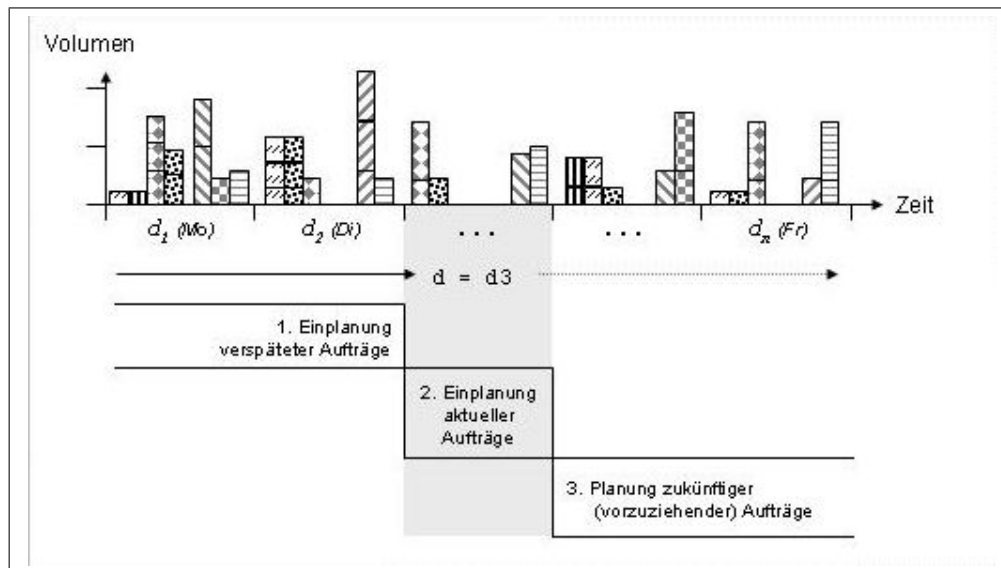
Algorithmus 5.1 (Rollierende Planung)

- *Input:*
Gegeben ist ein Planungshorizont $T = d_1, \dots, d_n$. Jedem Tag d ist eine Menge \mathcal{K}_d von Kundenaufträgen zugeordnet, die an dem jeweiligen Tag (spätestens) zu fertigen sind.
- *Ausführung:*
Beginnend beim ersten Tag des Planungshorizonts plane für den jeweils aktuellen Tag d anhand des Algorithmus 5.2
 1. die *verspäteten Aufträge* der Tage $d' < d$,
 2. die *aktuellen Aufträge* des Tages d und
 3. die *zukünftigen (vorzuziehenden) Aufträge* der Tage $d' > d$ ein.

■ A(5.1)

⁵Aufgrund der Reihenfolgeplanung müssen die Rohstoffe dann *zu Beginn* des jeweiligen Planungsintervalls zur Verfügung stehen. Wenn diese erst während der aktuellen Planungsperiode angeliefert werden, dann kann die Herstellung des Produkts erst in der darauffolgenden Periode geplant und durchgeführt werden.

Ziel der rollierenden Planung ist es, alle Aufträge möglichst zeitnah einzuplanen. Daher wurde jede Planungsperiode zusätzlich in drei *Planungsphasen* unterteilt, in der zuerst die verspäteten, dann die aktuellen und zuletzt die zukünftigen Aufträge eingeplant werden (vgl. Abb. 5.2).



Dargestellt sind die Kundenaufträge für verschiedene Produkte an den einzelnen Tagen des Planungshorizonts mit ihrer jeweiligen Auftragsmenge (Höhe der jeweiligen Säule). Je nach dem zu planenden Tag in der *rollierenden Planung* werden diese zusammengefasst.

Abbildung 5.2: Auftragsmengen in der rollierenden Planung

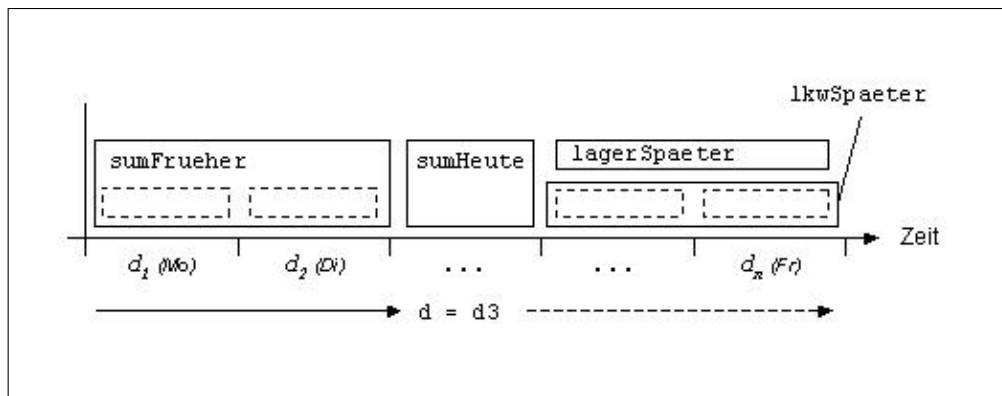
Bei der Einplanung zukünftiger Aufträge in Schritt 3 des Algorithmus 5.1 ist insbesondere der Typ der Abfüllung zu berücksichtigen (vgl. Abb. 5.3). Hier wird eine auftragsorientierte Planung, d.h. eine Vorgabe von Abfüllbehältern, angenommen. Aufträge, die in Tankkraftwagen abgefüllt werden⁶, können zeitlich nicht vorgezogen werden. Eine Abfüllung in Gebinde⁷ kann dagegen zeitlich vorgezogen werden, da ein Gebinde bis zur Auslieferung gelagert werden kann. Der Lagerort für die Gebindebehälter wird dabei nicht als Engpaß angesehen und daher in diesem Verfahren nicht weiter berücksichtigt.

Praxis: Implementierung in einem Planungssystem

Aufträge mit einer Abfüllung in Tankkraftwagen (TKW) können in der *rollierenden Planung*, wie auch bei der *Bestimmung des Produktionsvolumens* in Kapitel 5.3.2 zeitlich nicht vorgezogen werden. Dies resultiert aus der Eigenschaft, daß in dem Modell der Reihenfolgeplanung auch die Abfüllung berücksichtigt wird und die Chargen dabei direkt abgefüllt werden.

⁶Diese sind in Abb. 5.3 als Aufträge `1kwSpaeter` gekennzeichnet.

⁷Dies sind die in Abb. 5.3 als `lagerSpaeter` gekennzeichneten Aufträge.



In der rollierenden Planung werden die Aufträge in vier Mengen aufgeteilt: verspätete Aufträge (*sumFrueher*), aktuelle Aufträge (*sumHeute*), zukünftige Aufträge mit Gebindeabfüllung (*lagerSpaeter*) und TKW-Abfüllung (*lkwSpaeter*).

Abbildung 5.3: Zusammenfassung der einzuplanenden Aufträge

Unterschiedliche Aspekte müssen bei einer Lieferung in einem TKW von einem Produktionsplaner berücksichtigt werden.

- Wird das Auftragsvolumen von dem Kunden *just-in-time* benötigt?
- Das Auftragsvolumen ist so groß, daß es sinnvollerweise nicht in anderen Gebinden geliefert werden kann.
- Ein TKW steht nur zu dem bestimmten Termin zur Verfügung.
- Wie groß ist das Lagervolumen der Produktionsanlage und des Kunden?

Sofern für einen Auftrag mit TKW-Abfüllung *keine* just-in-time Eigenschaft besteht, kann die Produktion auch als *Produktion auf Lager vorgezogen* werden. Dabei muß seitens des Produktionsplaners lediglich gewährleistet werden, daß ausreichend Silokapazitäten zur Verfügung stehen und eine spätere, von dem Produktionsplaner zusätzlich im Produktionsablauf eingefügte Abfüllung nicht zeitgleich mit einer durch das Optimierungsverfahren geplanten Abfüllung stattfindet.

Ansonsten kann ein im TKW geliefertes Produkt früher produziert und geliefert werden, wenn nach Rücksprache mit dem Kunden und dem Spediteur genügend Transport- und Lagerkapazitäten zur Verfügung stehen.

Damit das in TKW abzufüllende Produktionsvolumen möglichst termingerecht produziert wird, wurde die rollierende Planung für eine Praxisanwendung dahingehend erweitert, daß vor dem Schritt 1, vor der Einplanung verspäteter und aktueller Aufträge, die Chargen aller TKW-Aufträge erstellt und eingeplant werden können.

In einer weiteren Erweiterung der rollierenden Planung für eine Praxisanwendung wird berücksichtigt, ob in der jeweils vorhergehenden Planungsphase Chargen erstellt wurden, die das Volumen eines Reaktors *nicht* vollständig ausfüllen. Hier wird dann vor der Erstellung neuer Chargen zunächst geprüft, ob Kundenaufträge der aktuellen Planungsphase zunächst zu bestehenden Chargen hinzugefügt werden können.

5.2 Volumenberechnung

Eine wesentliche Vorbedingung zur Einplanung einer Charge ist die technische Realisierbarkeit bezüglich des Chargenvolumens. Dieses ist durch die technischen Restriktionen der Produktionsanlage und die Rezepturen aller Zwischenprodukte bestimmt. Erst nach der erfolgreichen Einplanung einer Charge kann die bestehende Lösung anhand eines Kostenmodells oder anderer Zielfunktionen bewertet und evtl. verändert werden.

In diesem Abschnitt wird näher beschrieben, welche Volumengrenzwerte bei der Erstellung der Chargen berücksichtigt werden müssen. Dazu wird im folgenden immer eine konvergierende Produktionsstruktur vorausgesetzt. Zur Berechnung werden die folgenden Daten verwendet.

$V(chg)$	Das zu bestimmende Volumen der Charge chg . Einer Charge ist eindeutig ein Produktionsrezept rz zugeordnet.
UR_{rz}	Menge der Reaktoren, auf denen das Rezept rz gefertigt werden kann.
\underline{V}_u	Minimaler Füllstand (<i>Sumpfmenge</i>) des Reaktors u .
\overline{V}_u	Maximaler Füllstand des Reaktors u .
ZT_{rz}	Menge der Zutaten, die in der Herstellung des Rezeptes rz benötigt werden.
$d_{rz,zt}$	Prozentualer Anteil der Zutat zt am Rezept rz . Wenn das Rezept bekannt ist, wird im folgenden auch verkürzt d_{zt} geschrieben.
$V(chg_{zt})$	Volumen einer Zwischenprodukt-Charge chg_{zt} , die in der Produktion der Charge chg als Zutat benötigt wird:

$$V(chg_{zt}) := d_{zt} \cdot V(chg). \quad (5.1)$$

Das Volumen $V(chg)$ einer einzuplanenden Charge chg wird begrenzt durch das Volumen der Reaktoren, in denen die Charge produziert werden soll. Nach Zuordnung der Charge zu einem Reaktor muß das Volumen mindestens so groß sein wie der minimale Füllstand, die sog. *Sumpfmenge*. Dieser minimale Füllstand resultiert aus den technischen Gegebenheiten, daß z.B. ein Rührwerk im Reaktor bedeckt sein muß, damit ein Mischen der Materialien überhaupt möglich ist. Ebenfalls darf der maximale Füllstand \overline{V}_u nicht überschritten werden.

Somit muß nach der Zuordnung einer Charge chg zu einem Reaktor u durch die Reihenfolgeplanung gelten:

$$(chg \mapsto u) \implies \underline{V}_u \leq V(chg) \leq \overline{V}_u \quad (5.2)$$

Für jede Produktionsstufe errechnen sich die Volumengrenzwerte einer Charge anhand der Reaktoren, die dem jeweiligem Produktionsrezept rz der Charge chg zugeordnet sind,

$$\underline{V}_u(chg) \geq \min \{ \underline{V}_u \mid u \in UR_{rz} \} \quad (5.3)$$

und

$$\overline{V}_u(chg) \leq \max \{ \overline{V}_u \mid u \in UR_{rz} \}. \quad (5.4)$$

In dem Produktionsprozeß eines Endprodukts werden verschiedene Zwischen- und Vorprodukte gefertigt. Für jede Charge dieser Vorstufen müssen jeweils eigene minimale und maximale Füllstände berücksichtigt werden. Daraus folgt, daß bei der Erstellung einer Endprodukt-Charge in der Produktionsplanung auch die minimalen und maximalen Volumina der Reaktoren jeder Vorstufe rekursiv berücksichtigt werden müssen.

Für eine Charge chg , die eine Zwischenprodukt-Charge chg_{zt} zu einem prozentualen Anteil d_{zt} des eigenen Volumens $V(chg)$ benötigt, ist entsprechend zu berücksichtigen:

$$V(chg_{zt}) \geq \underline{V}_u(chg_{zt}) \quad (5.5)$$

und

$$V(chg_{zt}) \leq \overline{V}_u(chg_{zt}). \quad (5.6)$$

Für die weitere Entwicklung eines Verfahrens zur Volumenberechnung und die Beschreibung in dieser Arbeit wird die folgende Annahme über die Produktionsstruktur gemacht.

Annahme: Für die Entwicklung eines Verfahrens zur Volumenberechnung wird die folgende Forderung als erfüllt vorausgesetzt:

$$\begin{aligned} &\text{Das Verfahren zur Volumenberechnung wird für Rezepturen} \\ &\text{mit konvergierender Produktionsstruktur entwickelt.} \end{aligned} \quad (5.7)$$

Diese Forderung bedeutet, daß ein Zwischenprodukt nur in *einem* Produkt weiterverarbeitet wird. Es besteht nicht die Möglichkeit, daß verschiedene Zwischen- oder Endprodukte dasselbe Produkt als Zutat verwenden⁸.

Darüber hinaus wird für die weitere Entwicklung des Verfahrens zunächst einmal folgende Forderung gestellt.

Forderung: Für die Beschreibung der Volumenberechnung in diesem Abschnitt wird die folgende Situation in Bezug auf die weitere Verarbeitung angenommen.

$$\begin{aligned} &\text{Bei der Produktion einer Charge wird für jede} \\ &\text{Zwischenprodukt-Zutat genau eine Charge hergestellt.} \end{aligned} \quad (5.8)$$

Unter der Annahme (5.8) ergeben sich mit (5.1) und (5.2) dann die weiteren Grenzwerte für $V(chg)$

$$\underline{V}_{zt}(chg) \geq \max \left\{ \frac{1}{d_{zt}} \cdot \underline{V}_u(chg_{zt}) \mid zt \in ZT_{rz} \right\} \quad (5.9)$$

⁸Die Forderung (5.7) wird im weiteren Verlauf dieser Arbeit abgeschwächt bzw. kann bei der Implementierung in einem System zur Produktionsplanung umgangen werden.

und

$$\bar{V}_{zt}(chg) \leq \min \left\{ \frac{1}{d_{zt}} \cdot \bar{V}_u(chg_{zt}) \mid zt \in ZT_{rz} \right\} \quad (5.10)$$

Aus den Bedingungen (5.3) und (5.9) folgt für die Untergrenze

$$\underline{V}(chg) \geq \max (\underline{V}_u(chg) , \underline{V}_{zt}(chg)) \quad (5.11)$$

bzw. aus (5.4) und (5.10) für die Obergrenze

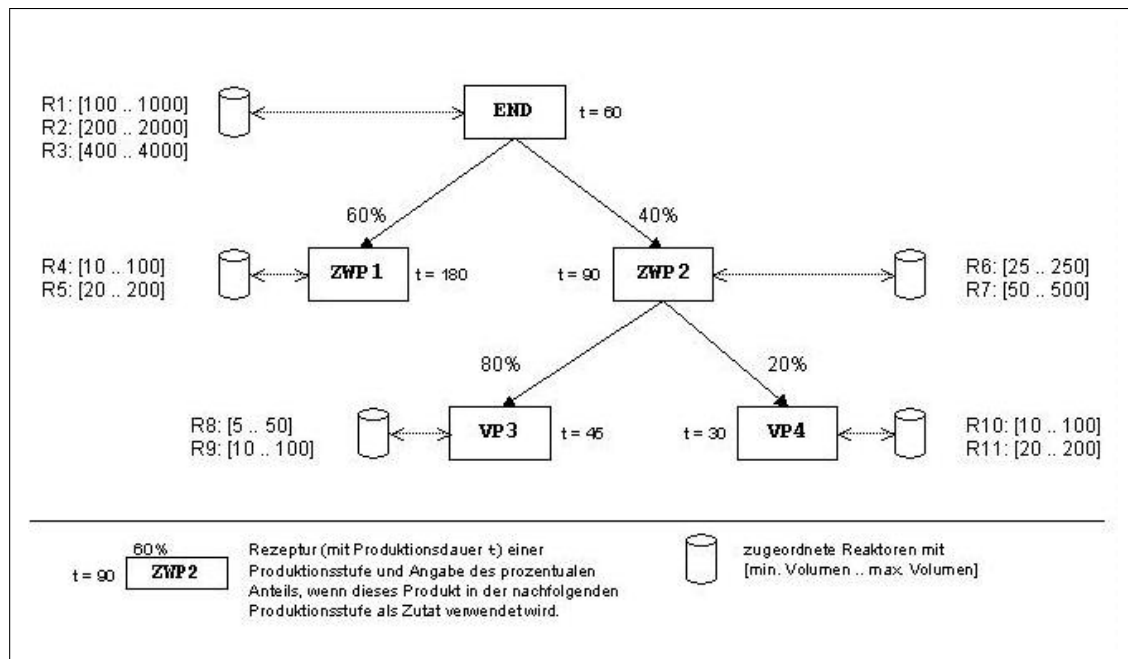
$$\bar{V}(chg) \leq \min (\bar{V}_u(chg) , \bar{V}_{zt}(chg)) \quad (5.12)$$

Diese Berechnung wird noch einmal an einem Beispiel anschaulich dargestellt.

Beispiel 5.2 Gegeben sei die dreistufige Produktion eines Produkts END (vgl. Abb. 5.4). Die Rezepturen können auf Reaktoren mit unterschiedlichen Volumengrenzwerten ausgeführt werden, damit ergeben sich folgende Volumengrenzwerte:

$$\begin{aligned} \text{VP4:} \quad & \underline{V}_u(\text{VP4}) = 10; \quad \bar{V}_u(\text{VP4}) = 200 \\ \text{VP3:} \quad & \underline{V}_u(\text{VP3}) = 5; \quad \bar{V}_u(\text{VP3}) = 100 \\ & \Rightarrow \underline{V}_{zt}(\text{ZWP2}) = \max \left(10 \cdot \frac{1}{0,2}; 5 \cdot \frac{1}{0,8} \right) = 50 \\ & \Rightarrow \bar{V}_{zt}(\text{ZWP2}) = \min \left(200 \cdot \frac{1}{0,2}; 100 \cdot \frac{1}{0,8} \right) = 125 \\ \text{ZWP2:} \quad & \underline{V}_u(\text{ZWP2}) = 25; \quad \bar{V}_u(\text{ZWP2}) = 500 \\ & \Rightarrow \underline{V}(\text{ZWP2}) = \max (\underline{V}_{zt}(\text{ZWP2}); \underline{V}_u(\text{ZWP2})) = 50 \\ & \Rightarrow \bar{V}(\text{ZWP2}) = \min (\underline{V}_{zt}(\text{ZWP2}); \bar{V}_u(\text{ZWP2})) = 125 \\ \text{ZWP1:} \quad & \underline{V}_u(\text{ZWP1}) = 10; \quad \bar{V}_u(\text{ZWP1}) = 200 \\ & \Rightarrow \underline{V}_{zt}(\text{END}) = \max \left(10 \cdot \frac{1}{0,6}; 50 \cdot \frac{1}{0,4} \right) = 125 \\ & \Rightarrow \bar{V}_{zt}(\text{END}) = \min \left(200 \cdot \frac{1}{0,6}; 125 \cdot \frac{1}{0,4} \right) = 312,5 \\ \text{END:} \quad & \underline{V}_u(\text{END}) = 100; \quad \bar{V}_u(\text{END}) = 4000 \\ & \Rightarrow \underline{V}(\text{END}) = \max (\underline{V}_{zt}(\text{END}); \underline{V}_u(\text{END})) = 125 \\ & \Rightarrow \bar{V}(\text{END}) = \min (\bar{V}_{zt}(\text{END}); \bar{V}_u(\text{END})) = 312,5 \end{aligned}$$

Hinweis: Die Forderung (5.8) wird in dem Verfahren 2 zur Bestimmung des Chargenvolumens und Nachoptimierung in Kapitel 5.3.4 nicht weiter berücksichtigt.



Gegeben sind 5 Rezepte rz mit den jeweiligen Zutaten ZT_{rz} und den zugeordneten Reaktoren UR_{rz} mit ihren Volumengrenzwerten $[min .. max]$.

Abbildung 5.4: Berechnung der Volumengrenzwerte (Beispiel 5.2)

Praxis: Implementierung in einem Planungssystem

Bei einer Anwendung der zuvor beschriebenen Volumenberechnung in einem Planungssystem kann die Forderung (5.7) nicht immer erfüllt werden.

Alternative Annahme: Die Forderung (5.7) kann für die Vorgehensweise der Berechnung abgeschwächt werden, indem gefordert wird, daß

$$\begin{aligned} &\text{eine Zwischenprodukt-Charge nur in einer Charge} \\ &\text{der nachfolgenden Produktionsstufe weiterverarbeitet wird.} \end{aligned} \quad (5.13)$$

Zur Unterscheidung dieser beiden Forderungen werden für diese Arbeit die folgenden Definitionen gegeben:

Definition 5.1 Wenn in der Abfolge der Produktionsstufen jede Charge nur in der Herstellung einer weiteren Charge als Zwischenprodukt verwendet wird, wird dies als eine konvergierende Produktionsreihenfolge bezeichnet.

Wenn jedes Zwischenprodukt nur einmal für die Herstellung eines weiteren Produktes verwendet wird, dann wird dies als konvergierende Produktionsstruktur bezeichnet.

In (5.7) wird also eine konvergierende Produktionsstruktur gefordert, während bei (5.13) nur die schwächere Forderung der konvergierenden Produktionsreihenfolge erfüllt sein muß.

Beide Forderungen können in den folgenden Situationen zu Schwierigkeiten in der Volumenberechnung führen:

- *Ein Zwischenprodukt wird nur in einer sehr geringen Menge in der weiteren Herstellung benötigt.*
- *Das Volumen eines Reaktors wird in der Herstellung einer Charge nur gering gefüllt.*
- *Ein Zwischenprodukt wird in der Herstellung verschiedener Produkte benötigt.*

Die beschriebenen Problemfälle werden nun mit ihrer Auswirkung auf die Chargenplanung näher betrachtet.

Produktion sehr geringer Mengen: Für die Betrachtung dieser Situation wird das Beispiel 5.2 ein wenig modifiziert und die Auswirkungen analysiert.

Beispiel 5.3 (Modifikation von Beispiel 5.2) *Die Rezeptur des Endprodukts END wird verändert, indem das Zwischenprodukt ZWP1 nur noch zu 59% des Chargenvolumens als Zutat benötigt wird. Dafür wird eine weitere Zutat ZWP3 zu 1 % in der Herstellung von END hinzugefügt. Das Produkt ZWP3 kann nur auf dem Reaktor R8 mit den Grenzwerten $\underline{V}_{R8} := 5$ und $\overline{V}_{R8} := 50$ gefertigt werden.*

Daraus ergibt sich als weiterer Wert für die untere Grenze

$$\underline{V}(\text{END}) \geq 500 = \left(5 \cdot \frac{1}{0,01} \right)$$

und für die obere Grenze

$$\overline{V}(\text{END}) \leq 5000 = \left(50 \cdot \frac{1}{0,01} \right).$$

Dementsprechend ergibt sich $\underline{V}(\text{END}) = 500$ und $\overline{V}(\text{END}) = 312,5$.

Wie also ersichtlich wird, kann schon eine geringe Veränderung der Rezept- oder Produktionsstruktur dazu führen, daß $\underline{V}(\text{chg}) \leq \overline{V}(\text{chg})$ nicht mehr gewährleistet werden kann.

Im folgenden wird angenommen, daß Produktionsstruktur und Rezepturen so aufeinander abgestimmt sind, daß diese Bedingung nicht verletzt wird. Sofern dennoch die Situation auftritt, daß Produkte wie in dem modifiziertem Beispiel 5.3 in sehr geringen Mengen in der Herstellung einer Charge benötigt werden, die zuvor in dem Produktionsbetrieb produziert werden müssen, so sind diese zuvor durch interne Aufträge herzustellen und in der Rezeptur als Rohstoffe zu kennzeichnen.

Chargen mit geringer Reaktorauslastung: Durch das anteilige Volumen, das eine Zwischenprodukt-Charge in der Herstellung einer weiteren Charge beisteuern muß, kann es sein, daß das Volumen des Reaktors, auf dem die Zwischenprodukt-Charge hergestellt wird, nur gering ausgelastet ist.

In dem Beispiel 5.2 wird bei einem Produktionsvolumen von 312,5 Mengeneinheiten (ME) des Endprodukts **END** genau eine Charge des Produkts **VP4** mit einem Volumen von 25 ME hergestellt. Je nach Auswahl des Reaktors (R10 mit Maximalvolumen 100 ME oder R11 mit Maximalvolumen 200 ME), auf dem diese Charge produziert werden kann, wird das Volumen des Reaktors nur zu maximal 25% ausgelastet.

Wenn nun ein doppelt so großes Volumen von 625 ME für das Endprodukt **END** hergestellt werden soll, werden bei gleicher Kalkulation und der *Beibehaltung einer konvergierenden Produktionsreihenfolge* u.a. zwei Chargen des Vorprodukts **VP4** und entsprechend zwei Chargen des Endprodukts **END** erstellt. Wie aus der Abb. 5.4 ersichtlich wird, ist die volumenbezogene Auslastung der Reaktoren, auf denen diese Chargen hergestellt werden können, dann entsprechend gering. Dagegen werden die Reaktoren in zeitlicher Hinsicht stark ausgelastet. Um eine solche Situation zu vermeiden, wurden verschiedene Verfahren zur Minimierung der Anzahl erzeugter Chargen entwickelt (Verfahren zum *Zusammenfassen von Chargen* bzw. zum *Umverteilen von Chargen* in Kapitel 5.3.4). Diese neu zusammengestellten Chargen haben auch keine konvergierende Produktionsreihenfolge mehr. Dies muß entsprechend bei der Erzeugung einer Startlösung, wie auch bei der weiteren Reihenfolgeplanung berücksichtigt werden⁹.

Anhand des Beispiels 5.2 kann nun ein weiterer Problemfall analysiert werden. Das Endprodukt **END** hat die Volumenuntergrenze $\underline{V}(\text{END}) = 125$ ME und die Volumenobergrenze $\overline{V}(\text{END}) = 312,5$ ME. Es zeigt sich, daß eine möglichst hohe Auslastung des Reaktorvolumens nicht immer ratsam ist. Bei einer Produktionsmenge von 400 ME kann nur eine Charge mit dem Volumen $V(\text{END}) = 312,5$ ME hergestellt werden. Das verbleibende Volumen von 87,5 ME könnte aufgrund der Volumenuntergrenze nicht hergestellt werden. Diese Problematik wird in dem Verfahren zur *Volumenreduktion* (Algorithmus 5.7) berücksichtigt.

Nicht konvergierende Produktionsstruktur: Wenn entgegen der Forderung (5.7) keine konvergierende Produktionsstruktur vorliegt, dann ist zu prüfen, welche Mengen eines Zwischenprodukts in den nachfolgenden Herstellungsschritten der Produkte benötigt werden. Sofern das benötigte Volumen der einzelnen Zwischenprodukte groß genug ist, daß eine konvergierende Produktionsreihenfolge vorliegt (d.h. Forderung (5.13) ist erfüllt), dann ist keine differenzierte Vorgehensweise notwendig.

Eine andere Vorgehensweise ist jedoch durchzuführen, wenn

- das Zwischenprodukt nur in geringen Anteilen den verschiedenen Produktionen hinzugefügt wird (vgl. hierzu die Vorgehensweise des vorherigen Abschnitts zur *Produktion sehr geringer Mengen*), oder
- bei geringer Auslastung des Reaktorvolumens das Gesamtvolumen der jeweiligen Endprodukt-Aufträge nicht ausreichend groß ist, so daß die Verfahren aus dem Abschnitt *Chargen mit geringer Reaktorauslastung* angewandt werden können, oder
- ein Produkt in verschiedenen Produktionsstufen der Herstellung eines Endprodukts zugleich als Zwischen- und Vorprodukt benötigt wird.

⁹Das Beispiel 4.4 zum Erzeugen einer Startlösung auf S. 94 berücksichtigt bereits die Problematik, wenn keine konvergierende Produktionsreihenfolge besteht, eine Vor- oder Zwischenprodukt-Charge also in mehreren Chargen weiterverarbeitet wird.

Die beiden ersten Fälle können durch die Lösungsansätze aus den vorherigen Abschnitten gelöst werden. In dem letzten Problem sind die betroffenen Zwischenprodukte in den Rezepturen als Rohstoffe zu kennzeichnen¹⁰. Sie werden der Planung zuvor als *interne Aufträge ohne Abfüllung* übergeben. Hierbei ist durch den Produktionsplaner sicherzustellen, daß ausreichende Kapazitäten zur Zwischenlagerung zur Verfügung stehen.

Eine solche Vorgehensweise bietet die zusätzliche Möglichkeit, daß in der Reihenfolgeplanung die Abfolge der Chargen untereinander entkoppelt wird und damit die Komplexität der Reihenfolgeoptimierung sinkt. Die entsprechend vorproduzierten Zwischenprodukte werden *Halbprodukte* oder *Halbzeuge* genannt.

5.2.1 Volumenberechnung und Reihenfolgeplanung

Bei der Erstellung der Chargen muß neben den Volumina der Chargen auch die zeitliche Auslastung der Reaktoren einer Produktionsanlage berücksichtigt werden. Als zusätzlicher Input für die Chargenplanung wird daher die bestehende Reihenfolge \mathcal{S} verwendet. Um eine neue Charge zu planen muß die benötigte Zeit der Produktion und die (noch) verfügbare Zeit der Reaktoren beachtet werden. Daher wird das Verfahren der Chargenplanung in einem simultanen Planungsprozeß von Volumen- und Reihenfolgeplanung durchgeführt.

In der Berechnung der Volumengrenzwerte werden die *verfügbaren* und *benötigten* Produktionszeiten berücksichtigt:

\mathcal{S}	Die bestehende Reihenfolgeplanung.
$t_u^{max}(\mathcal{S})$	Die maximal verfügbare Produktionszeit eines Reaktors u in der Planungsperiode \mathcal{S} .
$t_u^{frei}(\mathcal{S})$	Die (tatsächlich) verfügbare Produktionszeit eines Reaktors u für die Einplanung weiterer Chargen.
$t_u^{prod}(\mathcal{S})$	Die durch Schedulingprozesse (Produktion, Lagerung, Transferzeiten, und Abfüllung) genutzte Zeit eines Reaktors u .
$t^{arbeit}(chg)$	Die benötigte Zeit für alle Arbeitsschritte bis zur endgültigen Verarbeitung einer Charge chg (d.h. inkl. Lagerung, Weiterverarbeitung und Abfüllung).

Damit muß in der Berechnung anstelle der Menge UR_{rz} die Teilmenge

$$UR_{rz}^t := \left\{ u \in UR_{rz} \mid t_u^{frei}(\mathcal{S}) \geq t^{arbeit}(chg) \right\} \quad (5.14)$$

verwendet werden.

Die Schwierigkeit der Berechnung liegt nun in der Definition und Berechnung der Werte $t_u^{frei}(\mathcal{S})$ und $t^{arbeit}(chg)$. Dies wird nachfolgend näher diskutiert.

Die (tatsächlich) verfügbare Produktionszeit ist die Differenz aus $t_u^{max}(\mathcal{S})$ und $t_u^{prod}(\mathcal{S})$

$$t_u^{frei}(\mathcal{S}) := t_u^{max}(\mathcal{S}) - t_u^{prod}(\mathcal{S}) \quad (5.15)$$

¹⁰Die benötigten Rohstoffmengen müssen dann zu Beginn einer jeweiligen Planungsperiode zur Verfügung stehen.

Der Wert $t_u^{max}(\mathcal{S})$ kann aus den Inputdaten der Planung bestimmt werden (vgl. nachfolgende Definition und Abb. 5.5).

Definition 5.2 Die maximal verfügbare Produktionszeit $t_u^{frei}(\mathcal{S})$ eines Reaktors u in der Planungsperiode \mathcal{S} umfaßt die gesamte Dauer vom Beginn der Planungsperiode \mathcal{S} bis Produktionsende. Das Produktionsende kann in derselben oder (bei 3-Schicht-Betrieb) in einer nachfolgenden Planungsperioden erfolgen.

Es ist zu beachten, daß Produktionsanlagen an einzelnen Tagen des gesamten Planungshorizonts z.B. wegen Wartungsarbeiten oder Reparaturen nicht zur Verfügung stehen. Auch beim Einsatz der Mitarbeiter muß ein entsprechender Pausenplan berücksichtigt werden ([Tra01], S. 9 ff.). Für diese Arbeit werden unterschiedliche Verfügbarkeiten der Produktionsanlagen und Pausenkalender nicht weiter berücksichtigt.

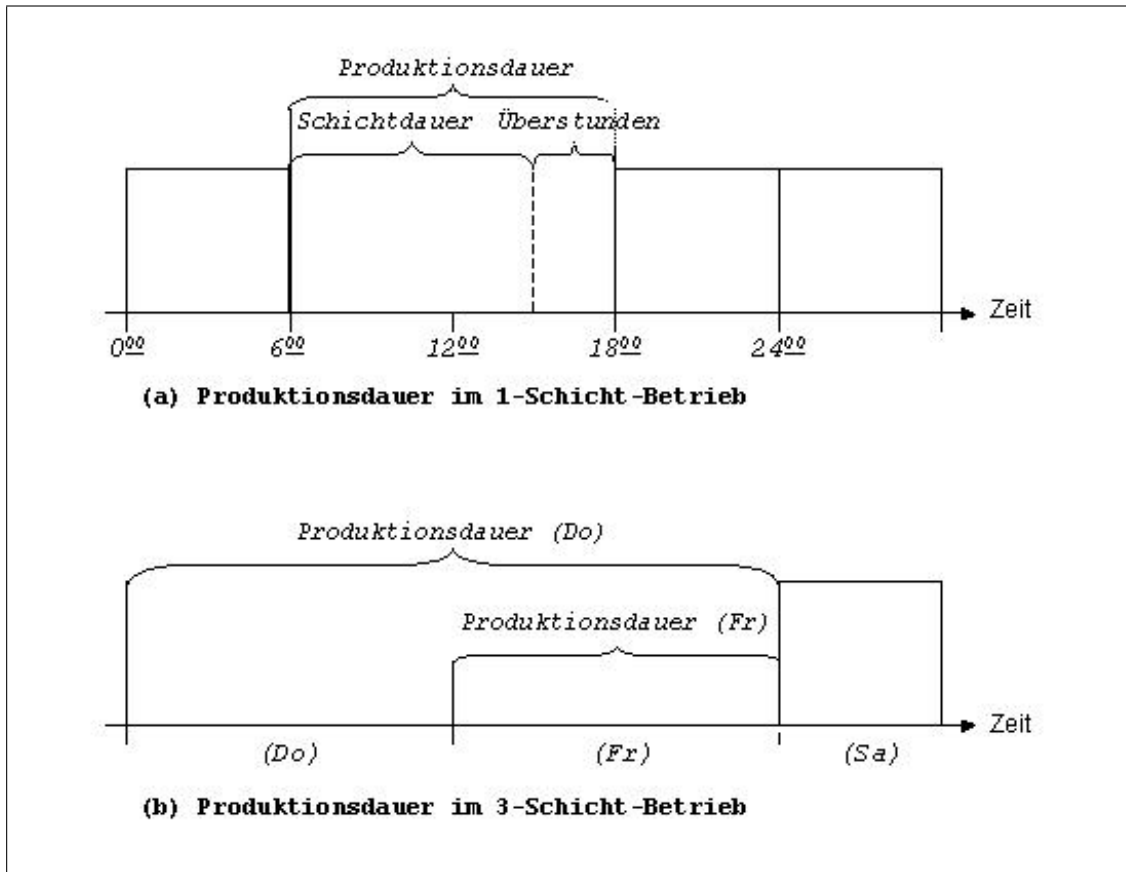


Abbildung 5.5: Gesamte Produktionszeit (Definition 5.2)

Bei der Einplanung der Chargenabfolge einer weiteren Produktion müssen die Reihenfolgebeziehungen des bestehenden Scheduling \mathcal{S} und der neu einzuplanenden Chargen berücksichtigt werden (siehe Beispiel 5.4). Erschwert wird diese Forderung noch durch mögliche JiT-Beziehungen in einzelnen Produktionsabfolgen. Mit $t_u^{prod}(\mathcal{S}) := \sum_{kn \in u} d(kn)$ kann hier nur eine Untergrenze bestimmt werden.

Daher wird in (5.14) statt $t_u^{frei}(\mathcal{S})$ der einfacher zu berechnende Wert

$$\tilde{t}_u^{frei}(\mathcal{S}) := \max(0, t_u^{max}(\mathcal{S}) - e(kn)) \quad (5.16)$$

verwendet. Dabei ist $e(kn)$ der Endzeitpunkt von kn , dem letzten Knoten in der Knotenkette [PROD], die dem End-Knoten $*$ in der Knotenkette des Reaktors u vorausgeht.

Da die Grenzwerte einer Charge chg eines Rezeptes rz bereits vor der Erstellung der Charge und der Zuweisung berechnet werden müssen, werden für die weitere Verwendung in dieser Arbeit die folgenden Bezeichnungen definiert.

Definition 5.3 Die aktuelle Sumpfmenge $\underline{V}_{rz}(\mathcal{S})$ eines Rezeptes rz in dem Scheduling \mathcal{S} gibt das Mindestvolumen einer neu zu erstellenden Charge chg vom Rezept rz an, das für (5.11) unter Berücksichtigung der Menge UR_{rz}^t aus (5.14) mit der vereinfachten Berechnung der freien Zeit aus (5.16) berechnet wird.

Die allgemeine Sumpfmenge \underline{V}_{rz} berücksichtigt dagegen die Verfügbarkeiten der Reaktoren in dem Scheduling \mathcal{S} nicht weiter. Es gilt dann $UR_{rz}^t = UR_{rz}$.

Die Schwierigkeit der Berechnung der verfügbaren Zeit wird anhand des folgenden Beispiels noch einmal näher dargestellt.

Beispiel 5.4 Es sei ein Scheduling \mathcal{S} mit einer Produktionszeit von 8 ZE gegeben, in dem bereits zwei 2-stufige JiT-Produktionen eingeplant sind. Nun soll eine weitere 1-stufige Charge der Produktionsdauer $t(chg) = 3ZE$ auf dem Reaktor R2 eingeplant werden (vgl. Abb. 5.6).

Die insgesamt für weitere Produktionen zur Verfügung stehende Zeit auf dem Reaktor R2 beträgt $t_u^{frei}(\mathcal{S}) = 8 - \sum_{kn \in RS} d(kn) = 6$. Demzufolge müssten sogar zwei Chargen der Produktionsdauer $t(chg) = 3ZE$ eingeplant werden können. Tatsächlich kann aber höchstens eine Charge der Produktionsdauer $t(chg) = 3ZE$ eingeplant werden.

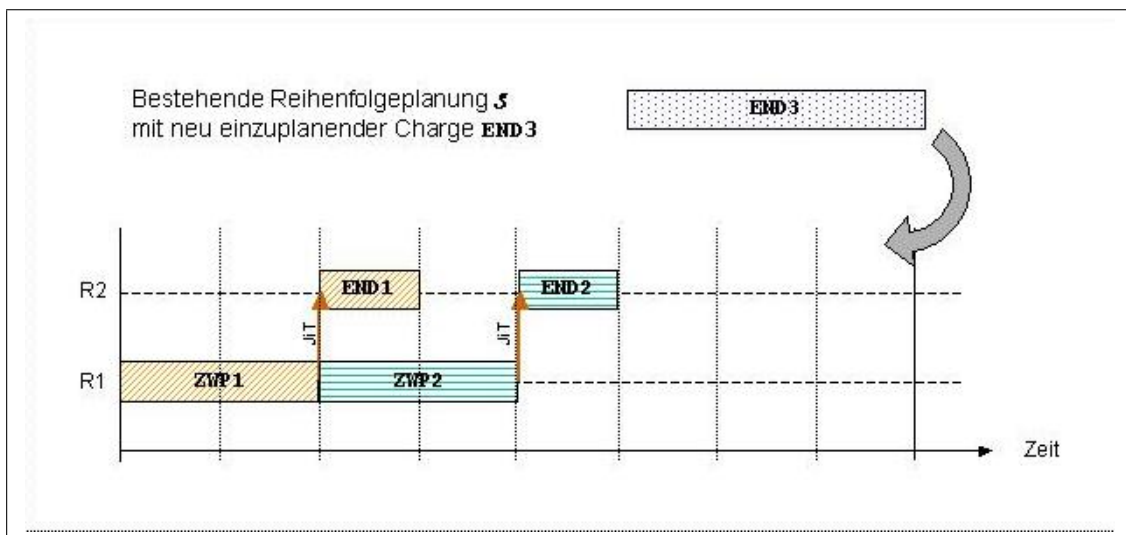


Abbildung 5.6: Zusätzlich einzufügende Charge (Beispiel 5.4)

Wie aus dem Beispiel ersichtlich wird, kann in einer Volumenberechnung, auf Basis einer bestehenden Reihenfolge die verfügbare Produktionszeit nur anhand von Richtwerten

berücksichtigt werden. Entsprechend muß der zu entwickelnde Algorithmus zur simultanen Chargen- und Reihenfolgeplanung auf Situationen reagieren, in denen eine *Überlast-Situation* entsteht, d.h. die Produktion oder damit verbundene Arbeitsschritte über das Ende der maximal verfügbaren Produktionszeit hinaus reichen.

Die benötigte Zeit für die endgültige Verarbeitung einer Charge $t^{arbeit}(chg)$ ist (in mehrstufigen) Prozessen ebenso schwierig zu ermitteln. Entweder wird die Charge im Anschluß an die eigentliche Produktion (nach eventueller Zwischenlagerung) als Zwischenprodukt in nachfolgenden Produktionsschritten verarbeitet oder als Endprodukt abgefüllt. In beiden Fällen kann abschließend noch eine Reinigung des Reaktors stattfinden, auf dem die Charge gefertigt wurde. Diese einzelnen Arbeitsschritte können damit zwar einzeln bestimmt werden und ihre Summe als unterer Grenzwert ermittelt werden. Zum Zeitpunkt der Volumenberechnung kann jedoch nicht bestimmt werden wie groß die tatsächlich benötigte Zeit nach Einplanung der Charge (und ihrer Vorgänger und Nachfolger) ist.

Daher wird hier der einfacher zu ermittelnde Wert $t^{prod}(chg)$, die Dauer für die Herstellung der Charge ohne Berücksichtigung nachfolgender Prozesse, verwendet. Wie bereits bei der Bestimmung des Wertes $t_u^{frei}(\mathcal{S})$ bzw. $\tilde{t}_u^{frei}(\mathcal{S})$ festgestellt wurde, muß in Anschluß an die Einplanung der neu erstellten Chargen im Falle einer Überlast-Situation eine Nachoptimierung durchgeführt werden. Eine genauere Berechnung des Wertes $t^{arbeit}(chg)$ würde hier also keine größere Genauigkeit für das Gesamtergebnis liefern.

5.3 Chargenplanung

Eine Charge ist der Auftrag zur Herstellung einer bestimmten Menge eines Produkts zu einem bestimmten Termin. Dieser wird unter Maßgabe der zuvor gewählten Zielsetzung und betrachteten Eigenschaften der Kundenaufträge geplant. Die Charge setzt sich in ihrem Volumen aus einem oder mehreren Kundenaufträgen zusammen. Bei der Erzeugung eines Produktionsauftrages sind daher die folgenden Bedingungen zu berücksichtigen:

- Das Volumen der Charge muß die technischen Restriktionen (5.11) und (5.12) bzgl. Unter- und Obergrenze des Volumens erfüllen.
- Wenn das Volumen eines Kundenauftrags kleiner als die Volumenuntergrenze (5.11) ist, müssen mehrere Kundenaufträge zu einer Charge zusammengefasst werden.
- Wenn das Volumen eines Kundenauftrags größer als die Volumenobergrenze (5.12) ist, dann muß das Volumen auf mehrere Chargen verteilt werden, d.h. der Auftrag wird *gesplittet*.
- Die Kundenaufträge sind möglichst nach der Art der Abfüllung (Gebinde oder TKW) zusammenzufassen.

Die Schritte der *Chargenplanung* und der *Reihenfolgeplanung* können bei der Produktionsplanung sukzessiv oder simultan durchgeführt werden. Wie leicht ersichtlich ist und in der *Volumenberechnung* des vorigen Abschnitts bereits berücksichtigt wird, besteht eine starke Interdependenz zwischen den Ergebnissen der Volumenberechnung und der Reihenfolgeplanung¹¹. Einem Scheduling \mathcal{S} können nur dann weitere Chargen hinzugefügt werden,

¹¹Es besteht vor allem eine zeitliche Interdependenz zwischen der Volumen- und der Reihenfolgeplanung. Diese wurde bereits in dem Beispiel 5.1 auf S. 131 näher beschrieben.

wenn für diese ausreichend Produktionszeit zur Verfügung steht. Sobald eine oder mehrere Chargen erfolgreich eingeplant wurden, verringert sich wiederum die Produktionszeit der Reaktoren.

Bei der Volumenberechnung kann jedoch nicht berücksichtigt werden, auf welchen Reaktoren die Chargen letztendlich eingeplant werden. Daher kann hier für die reihenfolgeabhängige Reinigungszeit a priori nur mit Schätzwerten gearbeitet werden. Ebenfalls ist vorher nicht bekannt, welchen Einfluß die Planung der Ressourcenpools (Mitarbeiter zur Reinigung und Abfüll-Stutzen) auf das Ergebnis der Reihenfolgeplanung haben wird. Eine nicht ausreichende Anzahl von Mitarbeitern für die Reinigung kann dazu führen, daß zusätzliche Stillstandszeiten eines Reaktors vor einer Reinigung und der nächsten Produktion entstehen. Auch durch eine Abfüllung kann in ähnlicher Weise eine Wartezeit entstehen, die wiederum die tatsächlich verfügbare Produktionszeit verringert. Hier kann aufgrund der volumenabhängigen Abfülldauer kein zuverlässiger Schätzwert ermittelt werden. Lediglich die Dauer der Produktion ist volumenunabhängig vorgegeben und kann bei der Berechnung der verfügbaren Produktionszeit zuverlässig verwendet werden.

Für die Produktionsplanung wurde in dieser Arbeit ein simultaner Lösungsansatz gewählt, der das in dem vorigen Kapitel beschriebene Verfahren zur Reihenfolgeplanung verwendet. Die kombinierte Chargen und Reihenfolgeplanung ist ein konstruktives Verfahren (Algorithmus 5.2), bei dem die Aufträge iterativ eingeplant werden. Dadurch wird gewährleistet, daß in jeder Iteration eine zulässige Lösung vorhanden ist. Zusätzlich wird in einem Rückkopplungsschritt geprüft, ob die Lösung *tagesgültig* ist, und dann gegebenenfalls eine Nachoptimierung durchgeführt werden muß.

Für die Beschreibung des Algorithmus zur Chargenplanung werden zunächst weitere, im Rahmen dieser Arbeit verwendete Begriffe definiert.

Definition 5.4 *Ein Scheduling \mathcal{S} heißt tagesgültig, wenn es eine zulässige Lösung des zugrundeliegenden Reihenfolgeproblems darstellt und der Makespan nicht größer als die verfügbare Produktionszeit des zugeordneten Tages ist.*

Die *Produktionszeit eines Tages*¹² resultiert dabei aus dem Schichtbetrieb und den möglichen Überstunden eines Produktionstages. Maximal (bei 3-Schicht-Betrieb) stehen also 24 Stunden Produktionszeit zur Verfügung. Dementsprechend kann diese Definition erweitert werden.

Definition 5.5 *Ein Scheduling \mathcal{S} heißt tagvoll, wenn der lokale Makespan jedes einzelnen Reaktors so groß ist, daß keine weitere Charge mehr eingeplant werden kann, bei dem das neu erstellte Scheduling tagesgültig ist.*

Ein tagvolles Scheduling ist also das maximale tagesgültige Scheduling, das mit den zu Chargen zusammengestellten Aufträgen erstellt werden kann. Wenn ein 3-Schicht-Betrieb geplant wird, kann allerdings eine überlappende Produktion in den nachfolgenden Produktionstag durchgeführt werden. Abweichend von der Definition heißt in diesem Fall ein Scheduling dann erst tagvoll, wenn der lokale Makespan jedes Reaktors mindestens so groß wie die verfügbare Produktionszeit ist, da hier die verfügbare Produktionszeit vollständig ausgenutzt werden kann.

¹²Vgl. hierzu auch Definition 5.2 auf S. 142.

Anhand dieser beiden Kriterien kann die simultane Chargen- und Reihenfolgeplanung durchgeführt werden, die die Ergebnisse der Volumenberechnung verwendet und nötigenfalls eine Nachoptimierung startet.

Algorithmus 5.2 (Simultane Chargen- und Reihenfolgeplanung)

Gegeben ist eine Menge von Kundenaufträgen \mathcal{K} , die zu Chargen \mathcal{C} zusammengefasst und in dem Scheduling \mathcal{S} eingeplant werden sollen.

1. *Abbruch Algorithmus:*
Wenn die Menge \mathcal{K} leer ist oder das Scheduling \mathcal{S} tagvoll ist, dann Abbruch des Algorithmus.
2. (a) Bestimme die Rezeptur eines Endprodukts, für die eine neue Charge mitsamt Zwischen- und Vorprodukt-Chargen erstellt werden soll, anhand des Algorithmus 5.3.
(b) Bestimme das maximale Produktionsvolumen¹³ und daraus resultierend das Volumen der Endprodukt-Charge¹⁴
(c) Erstelle die Endprodukt-Charge und alle benötigten Zwischen- und Vorprodukt-Chargen anhand von Algorithmus 5.4.
3. Erstelle eine Startlösung mit den neu erstellen Chargen anhand des Algorithmus 4.2.
4. Wenn das Scheduling *nicht* \mathcal{S} tagesgültig ist,
 - dann führe eine Reihenfolgeoptimierung durch,
 - ansonsten gehe zu Schritt 1.
5. Prüfe, ob das Scheduling \mathcal{S} *mehr als* tagvoll¹⁵ ist.
 - Wenn dies nicht der Fall ist, dann gehe zu Schritt 1.
 - Ansonsten führe eine Nachoptimierung¹⁶ in Bezug auf die Anzahl der neu erstellten Chargen oder das Volumen der neu erstellten Chargen durch.

■ A(5.2)

Die Erstellung der Chargen in Schritt 2 verwendet die Ergebnisse der Volumenberechnung, die in den nachfolgenden Abschnitten näher erläutert wird. Wenn die mit diesen Chargen erzeugte Lösung keine gültige Reihenfolge liefert, wird in Schritt 5 die Nachoptimierung gestartet. Hierzu wurden zwei verschiedene Vorgehensweisen entwickelt. Das Verfahren 1 der „Chargenminimierung“ faßt sukzessive Chargen kleinerer Volumina zusammen. Das

¹³Die Bestimmung des Produktionsvolumens wird in Kapitel 5.3.2 näher beschrieben.

¹⁴Das einzuplanende Volumen einer neuen Charge wird in Kapitel 5.3.3 allgemein und in verschiedenen Vorgehensweisen in Kapitel 5.3.4 genauer beschrieben.

¹⁵Wenn \mathcal{S} tagvoll ist, dann ist es zugleich auch tagesgültig. Daher muß nach einer Reihenfolgeoptimierung keine weitere Überprüfung auf Tagesgültigkeit durchgeführt werden.

¹⁶Die Auswahl des Algorithmus zur Nachoptimierung ist abhängig von der Vorgehensweise, anhand der im Schritt 2b das Volumen einer neu zu erstellenden Charge bestimmt wird.

Verfahren 2 der „*Volumenreduktion*“ reduziert iterativ das Maximalvolumen der geplanten Chargen.

Die in diesem Algorithmus vorgestellte Vorgehensweise mit einer entscheidungsabhängigen Reihenfolgeoptimierung in Schritt 4 kann auch verändert werden, indem dort *immer* eine Reihenfolgeoptimierung durchgeführt wird. Dies resultiert allerdings in insgesamt längeren Laufzeiten. Andererseits kann dadurch die verfügbare Produktionszeit $t_u^{frei}(\mathcal{S})$ (vgl. Formel (5.15)) bzw. $\hat{t}_u^{frei}(\mathcal{S})$ (vgl. Formel (5.16)) der einzelnen Reaktoren u erhöht werden und damit eventuell Chargen mit einem größerem Volumen erstellt werden.

Wenn in Schritt 2 das Verfahren 1 für die Bestimmung des Chargenvolumens verwendet wird, dann wird empfohlen, in Schritt 5 *immer* die Nachoptimierung der Chargenanzahl durchzuführen, unabhängig davon, ob $calS$ mehr als tagvoll ist oder nicht.

Der gesamte Algorithmus der simultanen Chargen- und Reihenfolgeplanung wird noch einmal schematisch in Abb. 5.7 dargestellt.

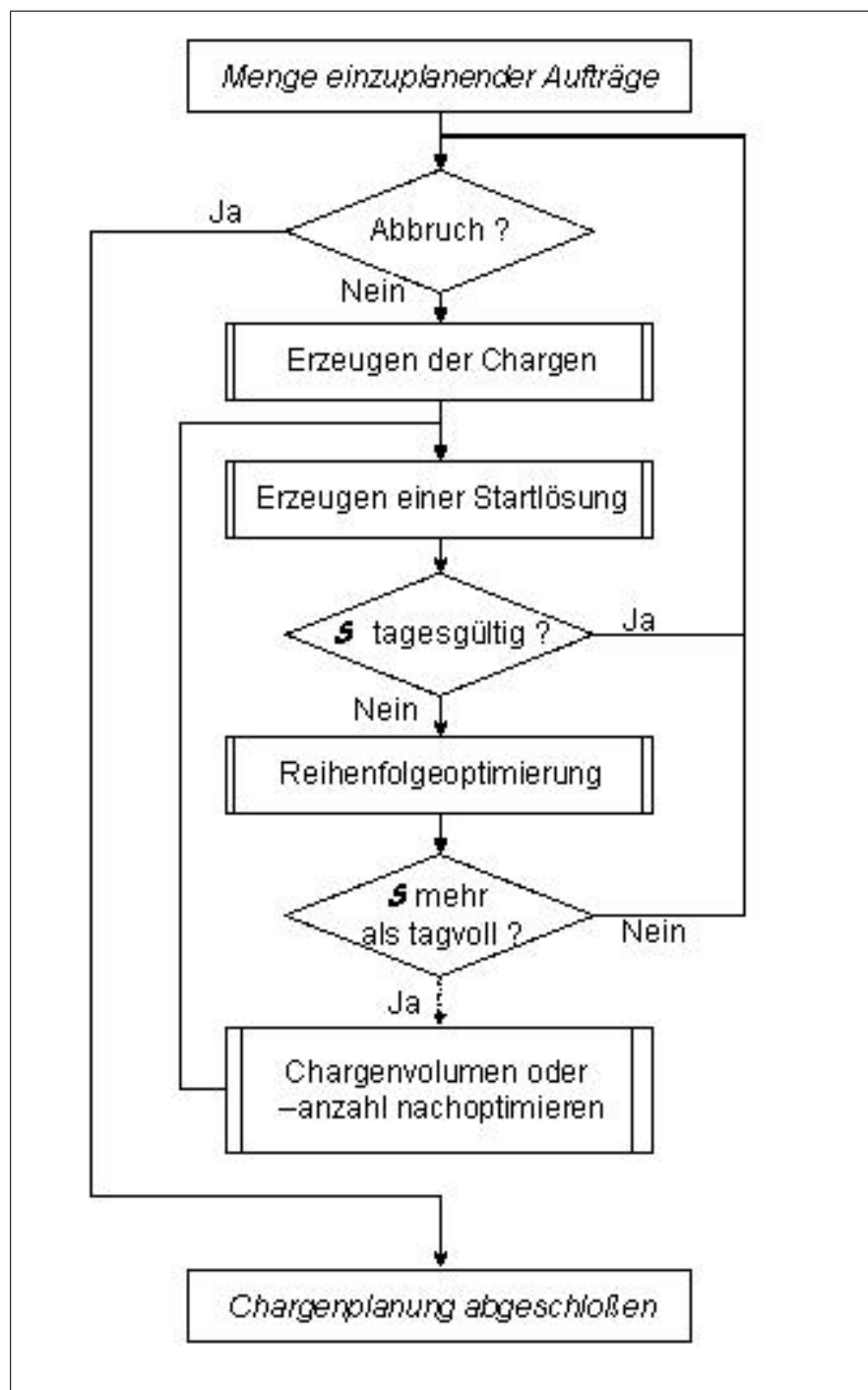


Abbildung 5.7: Simultane Planung

5.3.1 Auswahl des Produktionsrezeptes

Die Entscheidung über den ersten Schritt der Simultanplanung, die Erstellung der Chargen, erfolgt über die Auswahl eines Endprodukt-Rezeptes. Hierzu können die eigentlichen Eigenschaften einer Rezeptur und seiner Produktionsstruktur verwendet werden. Andererseits kann die Auswahl auch über die Eigenschaften der Kundenaufträge¹⁷ aus Abschnitt 5.1.1 erfolgen. Da einem (End-)Produkt eindeutig ein Produktionsrezept zugeordnet ist, können diesem Rezept dann die Eigenschaften der jeweiligen Kundenaufträge zugeordnet werden.

Für jedes Produktionsrezept werden aus der Produktionsstruktur nachfolgend zwei Auswahligenschaften abgeleitet.

Definition 5.6 *Die maximale rekursive Dauer einer Rezeptur ist die maximale Durchlaufzeit, die benötigt wird, wenn alle benötigten Zwischen- und Vorprodukte mit jeweils einer Charge hergestellt werden.*

Dieser Wert kann also rekursiv berechnet werden, indem zu der Dauer der Rezeptur immer die maximale rekursive Dauer der Rezepturen aller Zwischenprodukte addiert wird. Wichtig ist bei dieser Berechnung die Annahme, daß bei der konvergierenden Produktionsstruktur pro Zwischenprodukt jeweils nur eine Charge erzeugt werden muß und diese Chargen ohne Wartezeit oder Zwischenlagerung in einem Scheduling eingeplant werden können. Die tatsächliche und möglicherweise größere Anzahl einzuplanender Chargen eines Rezeptes ergibt sich erst aus dem einzuplanendem Volumen und der zeitlichen Verfügbarkeit der Reaktoren unterschiedlicher Volumina.

Anstelle der Maximalfunktion kann in der rekursiven Berechnung auch die Gesamtsumme der einzelnen Zeiten der jeweils vorgelagerten Produktionsstufen verwendet werden.

Definition 5.7 *Die gesamte (rekursive) Dauer einer Rezeptur ist die Herstellungsdauer, die benötigt wird, wenn alle benötigten Zwischen- und Vorprodukte mit jeweils einer Charge hergestellt werden.*

Die maximale rekursive Dauer gibt also die Zeit an, die zwischen dem Produktionsbeginn des frühesten Vorprodukts und dem Produktionsende des Endprodukts benötigt wird. Wenn für jedes Zwischen- und Vorprodukt nur jeweils eine Charge hergestellt wird und bei der Einplanung dieser Chargen in einem Scheduling keine weiteren Zeiten für die Reinigung oder Zwischenlagerung benötigt werden, entspricht dieser Wert der Durchlaufzeit. Ansonsten gibt dieser Wert eine Untergrenze für die Durchlaufzeit an. Die gesamte rekursive Dauer gibt dagegen die zeitliche Belastung der gesamten Produktionsanlage an.

Unter Beibehaltung der Forderung nach jeweils nur einer Charge je Rezeptur können diese entsprechend auch gezählt werden.

Definition 5.8 *Die Chargenanzahl einer Rezeptur ist die Gesamtzahl aller Chargen, die erzeugt werden, wenn für das Rezept und alle Zwischen- und Vorprodukte jeweils eine Charge erzeugt wird.*

Die Berechnung dieser Werte wird nun noch einmal anhand eines Beispiels dargestellt.

¹⁷Als Kundenauftrag wird hier die Bestellung über die Menge eines Produkts verstanden. Wenn ein Kunde mehrere Produkte für denselben Termin bestellt, ist dies ein Sammelauftrag.

Beispiel 5.5 Gegeben ist ein Rezept END (Herstellungsdauer $t = 60$ ZE) wie in Abb. 5.4 auf S. 138 mit zwei Zwischenprodukten ZWP1 ($t = 180$ ZE) und ZWP2 ($t = 90$ ZE). Das Zwischenprodukt ZWP2 benötigt wiederum die Herstellung der Vorprodukte VP3 ($t = 45$ ZE) und VP4 ($t = 30$ ZE).

Für das Rezept END müssen also (mindestens) 5 Chargen erzeugt werden. Die gesamte (rekursive) Dauer des Rezeptes END beträgt dann $t_{SUM} = 60 + 180 + 90 + 45 + 30 = 405$ ZE. Die maximale rekursive Dauer ist dagegen $t_{MAX} = 60 + \max(180, 90 + \max(45, 30)) = 240$ ZE.

Die Liefertermine der Kundenaufträge werden durch die rollierende Planung berücksichtigt. Die Auswahl der Rezeptur sollte dies entsprechend unterstützen, indem möglichst viele Aufträge termingerecht eingeplant werden. In dieser Arbeit wird jedoch keine differenzierte Betrachtung der Kundenaufträge vorgenommen. Stattdessen wird nur das Gesamtvolumen der Kundenaufträge je Produkt und Tag betrachtet.

Die Auswahl der Rezepturen erfolgt dann nach folgender Prioritätsregel:

Algorithmus 5.3 (Auswahl des Produktionsrezeptes)

Gegeben ist eine Liste der Endprodukt-Rezepturen einer Planungsperiode der rollierenden Planung, für die in der aktuellen Planungsphase (*verspätete, aktuelle oder zukünftige Aufträge*) ein einzuplanendes Volumen vorliegt.

1. Bestimme die Rezeptur mit dem kleinsten Wert für die maximale rekursive Dauer (nach Def. 5.6).
2. Bei Gleichheit mehrerer Rezepturen in Schritt 1 wähle die Rezeptur mit minimaler Chargenanzahl aus.
3. Bei Gleichheit mehrerer Rezepturen in Schritt 2 wähle die Rezeptur mit dem geringsten Auftragsvolumen aus.
4. Bei Gleichheit mehrerer Rezepturen in Schritt 3 wähle die Rezeptur mit dem größten Deckungsbeitrag aus.

■ A(5.3)

Nach der Auswahl der Rezeptur sind anschließend die neuen Chargen für das Scheduling zu erstellen.

5.3.2 Bestimmung des Produktionsvolumens

Nach der Bestimmung einer Endprodukt-Rezeptur werden die entsprechende Charge und alle benötigten Zwischen- und Vorprodukt-Chargen erstellt. Dazu ist zunächst das Volumen zu bestimmen, das eingeplant werden kann und soll. Hierzu werden die folgenden Berechnungen durchgeführt:

- Sofern in dem Planungsprozeß eine Rohstoffprüfung durchgeführt werden soll, ist das *maximale Volumen* \bar{V}_{roh} zu bestimmen, das aufgrund der Rohstoffbedarfe der Endprodukte und aller vorgelagerten Produktionsstufen produziert werden kann.

- Analog zu der Rohstoffprüfung ist eine Prüfung über das maximale Volumen \bar{V}_{beh} durchzuführen, das in die benötigten und verfügbaren Abfüllbehälter abgefüllt werden kann. Diese Prüfung ist separat von der Rohstoffprüfung durchzuführen, da ein Endprodukt nach Abschluß der Produktion in verschiedene Behälter abgefüllt werden kann.
- Bestimme die Sumpfmenge \underline{V}_{rz} , die aufgrund der Volumenberechnung in Abschnitt 5.2 aufgrund der Produktionsstruktur *bei zeitlicher Verfügbarkeit aller Reaktoren* benötigt wird, um eine Charge zu erstellen.
- Bestimme die Sumpfmenge $\underline{V}_{rz}(\mathcal{S})$, die *aufgrund der aktuellen zeitlichen Verfügbarkeit der Reaktoren in dem Scheduling \mathcal{S}* zur Erstellung einer Charge benötigt wird.
- Bestimme das Produktionsvolumen $V_{order}(d)$ aller Aufträge, die aufgrund der aktuellen Planungsphase der rollierenden Planung hergestellt werden sollen, d.h. bestimme das Volumen der verspäteten, aktuellen oder der zukünftigen (vorzuziehenden) Aufträge.
- Sofern in der Planungsphase die verspäteten oder aktuellen Aufträge hergestellt werden sollen, bestimme auch das Volumen der vorzuziehenden Aufträge $V_{order}(d+)$. Hierbei ist zu beachten, daß nur solche Aufträge vorgezogen werden können, die nicht in TKW, also nur in lagerbare Behälter abgefüllt werden können¹⁸.

$$V_{order}(d+) := V_{order}^{TKW}(d+) + V_{order}^{Geb}(d+) \quad (5.17)$$

Anhand dieser Werte wird anschließend das Volumen bestimmt, das bei der aktuellen Auswahl des Produktionsrezeptes eingeplant werden soll.

Für das Volumen V_{beh} der Aufträge, die in einen Behälter *beh* abgefüllt werden sollen, ergibt sich aufgrund des Maximalvolumens \bar{V}_{beh} der modifizierte Wert

$$\bar{V}'_{beh} := \min \{ V_{beh} , \bar{V}_{beh} \}$$

und das Gesamtvolumen

$$V'_{beh} := \sum_{beh \in \mathcal{B}} \bar{V}'_{beh} ,$$

wobei \mathcal{B} die Gesamtmenge aller Abfüllbehälter ist.

Entsprechend ist das Maximalvolumen der Aufträge, die aufgrund der verfügbaren Rohstoff- und Behälterbestände hergestellt werden können

$$\bar{V}_{bestand} := \min \{ V_{roh} , V'_{beh} \} . \quad (5.18)$$

Das Produktionsvolumen, das letztendlich für die Erstellung neuer Chargen zur Verfügung steht, muß für die aktuellen Aufträge die Bedingung

$$V_{order}(d) \geq \underline{V}_{rz}(\mathcal{S}) \quad (5.19)$$

und für die zukünftigen Aufträge die Bedingung

$$V_{order}(d+) \geq \underline{V}_{rz} \quad (5.20)$$

¹⁸Zur Unterscheidung sei hier noch einmal auf Abb. 5.3 auf S. 134 verwiesen.

erfüllen.

Damit diese technischen Restriktionen erfüllt werden können, ist entweder ein Volumen $V_{early}(d+)$ der zukünftigen Aufträge terminlich vorzuziehen oder ein Volumen $V_{late}(d)$ der aktuellen Aufträge zu verspäten. Dementsprechend kann (5.19) zu

$$V_{order}(d) + V_{early}(d+) - V_{late}(d) \geq \underline{V}_{rz}(\mathcal{S}) \quad (5.21)$$

und (5.20) zu

$$V_{order}(d+) - V_{early}(d+) + V_{late}(d) \geq \underline{V}_{rz} \quad (5.22)$$

umgewandelt werden.

Dabei muß lediglich einer der beiden Werte $V_{early}(d+)$ und $V_{late}(d)$ bestimmt werden:

$$V_{early}(d+) > 0 \quad \text{oder} \quad V_{late}(d) > 0 \quad (5.23)$$

Darüber hinaus müssen die folgenden Nebenbedingungen erfüllt sein.

$$V_{early}(d+) \leq V_{order}^{Geb}(d+) \quad (5.24)$$

$$V_{order}(d) + V_{early}(d+) - V_{late}(d) \leq \overline{V}_{bestand} \quad (5.25)$$

Praxis: Implementierung in einem Planungssystem

Prüfung der Rohstoffverfügbarkeit: Die Berücksichtigung oder Nicht-Berücksichtigung der Rohstoffbestände durch (5.25) dient einem Produktionsplaner vor allem dazu, in verschiedenen Szenarien zu evaluieren, welchen Einfluß die Rohstoffbestände auf das Ergebnis einer Planung haben können. Auswirkungen zeigen sich insbesondere dann, wenn Fehlbestände bis zur aktuellen Planungsperiode der rollierenden Planung aufgrund der Lieferfrist der betroffenen Rohstoffe nicht durch Nachlieferungen aufgefüllt werden können.

Wenn bei einer nicht-konvergierenden Produktionsstruktur einzelne Zwischenprodukte zunächst durch interne Aufträge als Rohstoffe vorproduziert werden müssen, dann ist auf jeden Fall eine Rohstoffprüfung notwendig.

Berücksichtigung weiterer Eigenschaften: Über die einfache Bestimmung des Produktionsvolumens je Planungsphase hinaus (*verspätete, aktuelle, und zukünftige Aufträge*), kann das einzuplanende Volumen dahingehend eingeschränkt werden, daß zunächst nur Aufträge mit gleicher Eigenschaft (vgl. Kapitel 5.1.1) für die Erstellung neuer Chargen in das einzuplanende Produktionsvolumen aufgenommen werden. Diese Eigenschaften sind von einem Produktionsplaner entsprechend festzulegen. In der vorliegenden Beschreibung werden die einzelnen Aufträge nur bei Endprodukten aufgrund der benötigten Abfüllung und des Abfüllbehälter weiter unterschieden.

5.3.3 Erstellen von neuen Chargen

Wenn das einzuplanende Produktionsvolumen

$$V_{prod}(d) := V_{vorder}(d) + V_{early}(d+) - V_{late}(d)$$

bestimmt wurde, wird eine Endprodukt-Charge und die benötigten Vor- und Zwischenprodukt-Chargen erstellt. Hier ist in ähnlicher Weise wie bei der Bestimmung des Produktionsvolumens eine Unter- und Obergrenze zu bestimmen.

Die Untergrenze für das Volumen einer Charge ist bereits bekannt als der Wert $\underline{V}_{rz}(\mathcal{S})$. Aus den vorherigen Berechnungen gilt dann

$$V_{prod}(d) \geq \underline{V}_{rz}(\mathcal{S}) \quad (5.26)$$

Die Obergrenze wird mit $\overline{V}_{chg}(\mathcal{S})$ bezeichnet; ihre Berechnung bei Endprodukten wird in den nachfolgenden Abschnitten näher beschrieben. Bei Zwischenprodukten wird der Wert folgendermaßen bestimmt¹⁹:

Rezeptur rz von chg stellt ein Zwischenprodukt her \Rightarrow

$$\overline{V}_{chg}(\mathcal{S}) := \min \{ \overline{V}_u \mid u \in UR_{rz}^t \}. \quad (5.27)$$

Die Größe einer neu zu erstellenden Charge ist dann gegeben als

$$V_{chg}(\mathcal{S}) := \begin{cases} \overline{V}_{chg}(\mathcal{S}) & , \text{ für } V_{prod}(d) \leq \overline{V}_{chg}(\mathcal{S}) \\ \overline{V}_{chg}^{-sumpf}(\mathcal{S}) & , \text{ für } \overline{V}_{chg}(\mathcal{S}) < V_{prod}(d) < \overline{V}_{chg}(\mathcal{S}) + \underline{V}_{rz}(\mathcal{S}) \\ \overline{V}_{chg}(\mathcal{S}) & , \text{ für } \overline{V}_{chg}(\mathcal{S}) + \underline{V}_{rz}(\mathcal{S}) \leq V_{prod}(d) \end{cases} \quad (5.28)$$

Dabei ist

$$\overline{V}_{chg}^{-sumpf}(\mathcal{S}) := \max \{ \underline{V}_{rz}(\mathcal{S}), V_{prod}(d) - \underline{V}_{rz}(\mathcal{S}) \}. \quad (5.29)$$

Analog zu der Bestimmung des Produktionsvolumens im vorigen Abschnitt muß auch hier das verbleibende Volumen nach der Bestimmung von $V_{chg}(\mathcal{S})$ berücksichtigt werden. Wenn $\overline{V}_{chg}(\mathcal{S}) < V_{prod}(d) < \overline{V}_{chg}(\mathcal{S}) + \underline{V}_{rz}(\mathcal{S})$ ist, dann reicht das verbleibende Volumen nicht aus, um eine weitere Charge zu erstellen. In diesem Fall muß das Volumen der Charge entsprechend reduziert werden. Dabei darf aber nicht die untere Schranke $\underline{V}_{rz}(\mathcal{S})$ aus (5.26) unterschritten werden, d.h. es muß $V_{chg}(\mathcal{S}) \geq \underline{V}_{rz}(\mathcal{S})$ gelten.

Nach der Berechnung des Chargenvolumens für das Endprodukt werden in einem rekursivem Verfahren alle weiteren Zwischen- und Vorprodukt-Chargen erstellt. Dabei wird der nachfolgende Algorithmus mit dem Wert $V_{prod}(d) := V_{chg}(\mathcal{S})$ für das Endprodukt gestartet. Die Vorgehensweise des Algorithmus wird später in Beispiel 5.7 auf S. 164 näher beschrieben.

¹⁹Dabei wird - wie für die Bestimmung dieses Wertes bei Endprodukten in dem nachfolgendem Abschnitt - die Eigenschaft (5.30) angenommen. Eine weitere Analyse des Verhaltens des Algorithmus bei Zwischenprodukten wurde in dieser Arbeit nicht durchgeführt.

Algorithmus 5.4 (Volumenbestimmung und Chargenerstellung)

Gegeben ist ein Rezept rz mit der Auftragsmenge \mathcal{K}_{rz} und dem Produktionsvolumen $V_{prod}(d)$. Für das Rezept wird aufgrund der zeitlichen Auslastung in dem Scheduling \mathcal{S} die Sumpfmenge $\underline{V}_{rz}(\mathcal{S})$ berechnet.

Solange $V_{prod}(d) > 0$ *ist*

1. Bestimme das Volumen $V_{chg}(\mathcal{S})$ einer neuen Charge anhand der Formel (5.28).
2. Setze $V_{prod}(d) := V_{prod}(d) - V_{chg}(\mathcal{S})$
3. Stelle eine neue Charge mit dem Volumen $V_{chg}(\mathcal{S})$ aus der Auftragsmenge \mathcal{K}_{rz} zusammen²⁰.
4. Für alle Zutaten zt des Rezeptes rz , die keine Rohstoffe sind:
 - (a) Bestimme das anteilige Volumen $V_{zt}(d) := V_{chg}(\mathcal{S}) \cdot d_{rz,zt}$.
 - (b) Erzeuge die Chargen der Zwischenprodukte chg_{zt} durch den rekursiven Start dieses Algorithmus 5.4 für das Volumen $V_{zt}(d)$ und die Rezeptur von zt .
5. Verknüpfe die neu erstellten Chargen untereinander, in dem die Chargen chg_{zt} der Zutaten zt zu der Menge $CH_{src}(chg)$ der direkt vorgeschalteten Chargen hinzugefügt werden, und die Charge chg zu den Mengen $CH_{trg}(chg_{zt})$ der direkt nachgeschalteten Produktionen der Zutaten-Chargen chg_{zt} hinzugefügt wird²¹.

■ A(5.4)

Da die Sumpfmenge einer Rezeptur aus den Sumpfmengen der Zutaten und der Reaktoren berechnet wird, ist in dem rekursivem Planungsprozeß gewährleistet, daß für jedes anteilige Produktionsvolumen $V_{zt}(d)$ die Bedingung (5.26) erfüllt ist. Für eine Zwischenprodukt-Charge ist bei der Berechnung des Wertes $V_{chg}(\mathcal{S})$ durch (5.28) dann die Obergrenze aus der Formel (5.12) in Verbindung mit der Menge UR_{rz}^t aus Formel (5.14) zu verwenden. Hier können sich lediglich Schwierigkeiten ergeben, wenn die Sumpfmenge eines Reaktors im Vergleich zu dem jeweiligem Maximalvolumen sehr groß²² ist. Da eine solche Eigenschaft jedoch als „pathologisch“²³ angesehen wird, wird sie in der weiteren Betrachtung dieser Arbeit nicht weiter analysiert.

Anders als bei den Zwischenprodukten wird für das Endprodukt jeweils nur eine Charge erstellt. Im Rahmen der simultanen Planung werden dann diese Chargen und alle weiteren erstellten Zwischen- und Vorprodukt-Chargen in dem Scheduling \mathcal{S} eingeplant. Hierbei

²⁰Die Zusammenstellung von Kundenaufträgen zu Chargen wird in dieser Arbeit nicht weiter beschrieben. Eine Unterscheidung der Aufträge findet nur bei den Endprodukten in Bezug auf den Abfüllbehälter statt.

²¹Vergleiche dazu in Kapitel 4.4.2 die Beschreibung *Vorgeschaltete und nachfolgende Produktion(en)* auf S. 100.

²²Dem Autor ist bekannt, daß i.d.R. mindestens $\underline{V}_u \leq 0,5 \cdot \bar{V}_u$ gilt.

²³Die Instanz eines Optimierungsproblems oder eine ganz bestimmte Eigenschaft der Inputdaten wird als *pathologisch* bezeichnet, wenn ein Optimierungsverfahren bewußt in ein lokales Optimum geführt werden kann oder keine zulässige Lösung mehr erzeugen kann. Wesentlich ist dabei, daß die vorgegebene Konstellation der Inputdaten bei Anwendungen, obwohl aufgrund der beschriebenen Nebenbedingungen möglich, in der Praxis nicht auftritt.

kann es passieren, daß mehrere Chargen auf demselben Reaktor eingeplant werden und dieser in der zeitlichen Auslastung überlastet wird. Die jeweiligen Sumpfmengen $\underline{V}_{rz}(\mathcal{S})$ beziehen sich jedoch nur auf die Möglichkeit, daß *eine* weitere Charge auf dem Reaktor eingeplant wird. Daher muß nach der Reihenfolgeplanung der neu erstellten Chargen evtl. eine Nachoptimierung durchgeführt werden. Entsprechend wird für das Endprodukt jeweils nur eine Charge erstellt, da nach der Einplanung aller Chargen der Wert $\underline{V}_{rz}(\mathcal{S})$ für das Endproduktrezept neu berechnet werden muß.

Praxis: Implementierung in einem Planungssystem

Wenn bei der Zusammenstellung der Chargen aus den Kundenaufträgen in dem Sinne verfahren werden soll, daß *möglichst viele* Aufträge termingerecht eingeplant und produziert werden, dann müssen hier die Aufträge mit dem kleinsten Auftragsvolumen zuerst eingeplant werden.

Im Rahmen dieser Arbeit wird die Zusammenstellung der einzelnen Chargen aus den Kundenaufträgen nicht näher betrachtet. Hier ist eine weitere Vertiefung des betrachteten Optimierungsproblems möglich. Dabei ist vor allem die Problematik von Sammelaufträgen, d.h. die gemeinsame Lieferung verschiedener Produkte zu einem Termin an einen Kunden zu berücksichtigen. Wenn z.B. ein Sammelauftrag nur komplett ausgeliefert werden darf, dann muß die Prüfung verfügbaren Rohstoffe gleichzeitig für alle Produkte durchgeführt werden.

5.3.4 Bestimmung des Chargenvolumens und Nachoptimierung

In dem vorhergehenden Abschnitt wurde die Berechnung von $\bar{V}_{chg}(\mathcal{S})$ für eine Endprodukt-Charge nicht weiter erläutert. Auf diese Problematik soll nun näher eingegangen werden.

Für die Berechnung von $\bar{V}_{chg}(\mathcal{S})$ wurden zwei Verfahren entwickelt, die jeweils in unterschiedlichen Verfahren der Nachoptimierung resultieren.

- Bestimmung von $\bar{V}_{chg}(\mathcal{S})$ als *kleinster Ganzfüllung*²⁴ und Nachoptimierung von Chargen und Reihenfolge durch *Zusammenfassen von Chargen* oder *Umverteilen von Chargen*.
- Bestimmung von $\bar{V}_{chg}(\mathcal{S})$ als *maximales Reaktorvolumen* und Nachoptimierung durch *Reduktion des Chargenvolumens*.

Die Nachoptimierung ist der Situation geschuldet, daß die Volumina der einzelnen Chargen festgelegt werden, ohne daß die Reaktoren bekannt sind, auf denen die Chargen letztendlich hergestellt werden.

²⁴Eine *Ganzfüllung* ist gegeben, wenn das Produktionsvolumen einer Charge chg , die auf einem Reaktor u gefertigt wird, genauso groß wie das Maximalvolumen des Reaktors ist, d.h. $V_{chg} = \bar{V}_u$. Wenn aufgrund der Rezeptur rz einer Charge mehrere Reaktoren $u \in UR_{rz}$ mit unterschiedlichen Maximalvolumina gegeben sind, dann ist die kleinste Ganzfüllung $\min \{ \bar{V}_u \mid u \in UR_{rz} \}$.

Verfahren 1: Chargenvolumen

In dem Scheduling \mathcal{S} stehen für das Endprodukt-Rezept rz die Reaktoren der Menge UR_{rz}^t zur Verfügung. Diese haben jeweils eine eigene Sumpfmengen \underline{V}_u und das Maximalvolumen \overline{V}_u . Durch die Volumenberechnung sind für $V_{chg}(\mathcal{S})$ der Minimalwert $\underline{V}_{rz}(\mathcal{S})$ und der Maximalwert $\overline{V}_{rz}(\mathcal{S})$ bekannt. Das Volumen $V_{chg}(\mathcal{S})$ einer neu zu erstellenden Charge kann daher zunächst jeden Wert zwischen diesen beiden Grenzwerten annehmen. Unter der Annahme, daß die Volumina der Reaktoren einer Produktionsanlage nicht stark divergieren und daß insbesondere

$$\max \{ \underline{V}_u \mid u \in UR_{rz}^t \} \leq \min \{ \overline{V}_u \mid u \in UR_{rz}^t \} \quad (5.30)$$

gilt²⁵, wird $\overline{V}_{chg}(\mathcal{S})$ dann definiert als

$$\overline{V}_{chg}(\mathcal{S}) := \min \{ \overline{V}_u \mid u \in UR_{rz}^t \}. \quad (5.31)$$

Diese Definition unter der Annahme (5.30) gewährleistet, daß eine neu erstellte Charge in der Reihenfolgeplanung auf jedem Reaktor $u \in UR_{rz}^t$ eingeplant werden kann.

Verfahren 1: Nachoptimierung

Wenn das ursprünglich einzuplanende Volumen $V_{prod}(d)$ für ein Endprodukt größer als das aus (5.31) und (5.28) berechnete Chargenvolumen $V_{chg}(\mathcal{S})$ ist, dann muß für das verbleibende Volumen $V_{prod}(d) - V_{chg}(\mathcal{S})$ mindestens je eine weitere Charge für das Endprodukt und die weiteren Zwischen- und Vorprodukte erstellt werden²⁶. Diese Vorgehensweise aus Algorithmus 5.4 resultiert in der Eigenschaft, daß jede Zwischenprodukt-Charge *nur eine* Charge der nachfolgenden Produktionsstufe bedient. Dies ist für die Abfolge der Produktionsprozesse jedoch nur notwendig im Falle von Zwischenprodukten, die als JiT-Zutat benötigt werden. In der Reihenfolgeplanung selber ist es möglich, daß eine Zwischenprodukt-Charge auch mehrere Chargen der nachfolgenden Produktionsstufe bedient²⁷.

Je nach Rezeptur und Produktionsanlage kann die Vorgehensweise, daß zunächst das Chargenvolumen bestimmt und erst anschließend die Chargen eingeplant werden, zu einer schlechten Auslastung in Bezug auf das Volumen eines Reaktors führen. Ein solches Planungsergebnis ist zugleich dadurch gekennzeichnet, daß das insgesamt einzuplanende Volumen auf mehr Chargen verteilt wird, als benötigt werden. Hieraus folgt, daß der **Makespan** der Reihenfolgeplanung durch diese zusätzlichen Chargen unnötig verlängert wird.

Zusammenfassen von Chargen: Nach der Einplanung der neu erstellten Chargen und der Reihenfolgeoptimierung kann der **Makespan** daher verringert werden, indem die Aufträge in anderer Weise zu Chargen zusammengestellt werden und somit eine bessere volumenbezogene wie auch zeitliche Auslastung der Reaktoren erreicht wird.

Für die Nachoptimierung des Chargenvolumens wurde der nachfolgende Algorithmus zum Zusammenfassen von Chargen entwickelt.

²⁵Wenn (5.30) zutrifft, dann kann eine neu erstellte Charge chg mit dem Volumen $V(chg) = \min \{ \overline{V}_u \mid u \in UR_{rz}^t \}$ auf jedem Reaktor eingeplant werden.

²⁶Für die Vorgehensweise in Verfahren 1 muß die Forderung (5.8) erfüllt sein, d.h. die Anzahl der erzeugten Vor- und Zwischenprodukt-Chargen ist linear abhängig von der Anzahl der Endprodukt-Chargen.

²⁷Vergleiche hierzu auch Beispiel 4.4 zur Erzeugung einer Startlösung auf S. 94.

Algorithmus 5.5 (Chargen zusammenfassen)

- Gegeben ist ein Scheduling \mathcal{S} mit den darin eingeplanten Chargen \mathcal{C} . Jede Charge hat ein Produktionsvolumen $V(chg)$ und einen zugeordneten Reaktor $u(chg)$ der Produktionsanlage mit dem Maximalvolumen \bar{V}_u .
- Bestimme die Menge \mathcal{C}' anhand der nachfolgenden Vorgehensweise: In jedem Schritt der Optimierung wird eine Teilmenge \mathcal{C}' der Chargen $chg \in \mathcal{C}$ bestimmt, deren Volumina zusammengefasst werden können:
 - Eine Charge bedient eine Charge der nachfolgenden Produktionsstufe *nicht* als JiT-Zutat.
 - Es sind wenigstens zwei Chargen desselben Rezeptes rz in dem Scheduling von \mathcal{S} vorhanden.
- In jedem Optimierungsschritt wird für jede Charge $chg \in \mathcal{C}'$ die Differenz der Volumina $V_{diff}(chg) := \bar{V}_{u(chg)} - V(chg)$ bestimmt.

Durchführung als while-Schleife:

Solange in der Menge \mathcal{C}' Chargen vorhanden sind, die zusammengefasst werden können:

1. *Bestimmung der Chargen zum Zusammenfassen:*

- (a) Bestimme die Chargen $chg \in \mathcal{C}'$ mit der geringsten Anzahl an Vorläufer-Chargen in der Menge $CH_{src}(chg)$.
Diese Auswahl bedingt, daß zuerst die Chargen zum Zusammenfassen ausgewählt werden, die in der Reihenfolgeplanung möglichst wenig Vorläufer-Chargen berücksichtigen müssen. Somit werden die Chargen ausgehend von den Vorprodukten zusammengefasst. Damit wird gewährleistet, daß die Reihenfolgeoptimierung nicht zu komplex wird aufgrund einer zu großen Anzahl von Nachbarschaftslösungen.
- (b) Bestimme aus den zuvor bestimmten Chargen von Schritt 1a die Charge chg_{best} mit dem geringsten Differenz $V_{diff}(chg)$.
- (c) Bestimme aus den zuvor bestimmten Chargen von Schritt 1a die Charge chg_{add} mit gleicher Rezeptur zu der in Schritt 1b bestimmten Charge chg_{best} , so daß das hinzufügende Volumen $V(chg_{add})$ maximal ist, d.h.
$$V(chg_{add}) = \max \{ V(chg) \mid chg \in \mathcal{C}' \}.$$
- (d) Wenn in dem Schritt 1c keine passende Charge zum Zusammenfassen gefunden wird, gehe zurück zu Schritt 1b, lösche chg_{best} aus der Liste von \mathcal{C}' und bestimme dort eine andere Charge chg_{best} .
- (e) Lösche die Chargen chg_{best} und chg_{add} aus der Liste \mathcal{C}' .

2. *Durchführung Nachoptimierung:*

- (a) Lösche chg_{best} und chg_{add} , sowie alle direkt oder über mehrere Produktions
- (b) Fasse das Volumen der Chargen chg_{best} und chg_{add} zusammen und aktualisiere die Verknüpfungen zu den direkt vorhergehenden Chargen
 $chg \in CH_{src}(chg_{best}) \cup CH_{src}(chg_{add})$ und den direkt nachfolgenden Chargen
 $chg \in CH_{trg}(chg_{best}) \cup CH_{trg}(chg_{add})$.

- (c) Plane die zusammengefasste Charge und alle in Schritt 2a entfernten Chargen in \mathcal{S} neu ein und starte eine Reihenfolgeoptimierung.

3. *Überprüfung des Optimierungsschrittes:*

- (a) Wenn die neue Lösung von \mathcal{S} keine zulässige Lösung erzeugt, gehe zurück zu Schritt 1b.
 (b) Speichere die neu erzeugte Lösung und die zugehörigen Chargen.
 (c) Bestimme die Menge \mathcal{C}' neu.

Überprüfung der Nachoptimierung:

Wenn die zuletzt erzeugte Lösung *nicht tagesgültig* ist, gehe zu der zuvor erzeugten Lösung in Schritt 3b zurück und suche in dem entsprechend vorhergehendem Schritt 1 nach anderen Chargen zum Zusammenfassen.

■ A(5.5)

Bei der Durchführung der Nachoptimierung mittels Zusammenfassen ist jedoch zu berücksichtigen, daß für eine Zwischenprodukt-Charge, die mehrere Chargen bedient, genügend Zwischenlagerkapazitäten zur Verfügung stehen müssen. Wenn hierbei die Chargen nicht in einem Batch-Mix gelagert werden dürfen, dann verringert sich das insgesamt zur Verfügung stehende Lagervolumen mit jeder Zwischenlagerung einer Charge nicht nur um das Chargenvolumen, sondern um das Volumen des Silos.

Wenn keine Silos zur Verfügung stehen, müssen die einzelnen Chargen in den herstellenden Reaktoren zwischengelagert werden. Dadurch werden die Reaktoren blockiert, und es können für nachfolgende Produktionen zusätzliche Wartezeiten entstehen.

Die möglichen Nachteile der unterschiedlichen Ergebnisse der Chargenplanung sind noch einmal in der Tabelle 5.1 zusammengefasst.

Geringe Volumenauslastung	Hohe Volumenauslastung
<ul style="list-style-type: none"> • hohe Anzahl von Chargen • unnötig großer Makespan durch zusätzliche Chargen 	<ul style="list-style-type: none"> • hohe Komplexität des Reihenfolgeproblems durch viele Reihenfolgebeziehungen zwischen den Chargen • unnötig großer Makespan durch zusätzliche Wartezeiten

Tabelle 5.1: Mögliche Nachteile bei unterschiedlicher Volumenauslastung

Ein weiterer Nachteil folgt aus der Anwendung von Algorithmus 5.5. Hierbei werden immer die Volumina zweier Chargen zusammengefasst. Der Algorithmus des Zusammenfassens wird nach jedem Erstellen einer neuen Endprodukt-Charge und dem Einplanen der Chargen in der Reihenfolge gestartet. Das Volumen einer neuen Charge des Rezeptes rz ist

daher höchstens so groß wie²⁸

$$mMV_{rz} := \min \{ \bar{V}_u \mid u \in UR_{rz}^t \}. \quad (5.32)$$

Wenn nun zwei Chargen zusammengefasst werden, wird zu einer bestehenden Charge jeweils das Volumen mMV_{rz} addiert. Der Reaktor mit dem Maximalvolumen $\bar{V}_u = mMV_{rz}$ steht anschließend für die Erstellung einer neuen Charge zeitlich wieder zur Verfügung. Wenn das Maximalvolumen der weiteren Reaktoren $u \in UR_{rz}$ dann *nicht* ein Vielfaches von mMV_{rz} ist, dann wird durch diese Vorgehensweise das Volumen eines Reaktors u mit $\bar{V}_u > mMV_{rz}$ nie vollständig ausgenutzt.

Umverteilen von Chargen: In Erweiterung von Algorithmus 5.5, bei dem jeweils nur zwei Chargen betrachtet werden, wurde das Verfahren des Umverteils von Aufträgen entwickelt, bei dem die Volumina von n_{rz} beliebig vielen Chargen neu bestimmt werden. Hier werden die Aufträge, die einer Charge zugeordnet sind, gesplittet und auf mehrere (neu erstellte) Chargen (um-)verteilt. Zuvor wurden sie lediglich einer anderen Charge zugeordnet.

Algorithmus 5.6 (Aufträge umverteilen)

- Gegeben ist ein Scheduling \mathcal{S} mit den Chargen $\mathcal{C} = \bigcup \{ \mathcal{C}_{rz} \mid rz \in RZ \}$ der Rezepturen $rz \in RZ$. Jede Charge hat ein Produktionsvolumen $V(chg)$ und einen zugeordneten Reaktor $u(chg)$ mit dem Maximalvolumen \bar{V}_u .
- Für jede Charge kann die Menge der Endprodukt-Chargen $CH_{end}(chg)$ bestimmt werden, in denen ein Teil des Produktionsvolumen der Charge chg direkt oder über weitere durchzuführende Produktionsstufen verarbeitet wird.
- Bestimme die Menge $\mathcal{C}' \subset \mathcal{C}$:
 - Eine Charge $chg \in \mathcal{C}'$ bedient eine Charge der nachfolgenden Produktionsstufe *nicht* als JiT-Zutat.
 - Es sind wenigstens zwei Chargen desselben Rezeptes rz in dem Scheduling von \mathcal{S} vorhanden, d.h. $|\mathcal{C}_{rz}| \geq 2$.
- In jedem Optimierungsschritt wird für jede Rezeptur rz anhand der Chargen in der Menge $\mathcal{C}'_{rz} := \mathcal{C}' \cap \mathcal{C}_{rz}$ die minimal benötigte Anzahl $n_{min}(rz)$ an Chargen ermittelt:

1. Bestimme das maximale Volumen aller Chargen

$$\bar{V}_{rz}(\mathcal{S}) := \max \{ \bar{V}_{u(chg)} \mid chg \in \mathcal{C}'_{rz} \}.$$

2. Bestimme das gesamte Produktionsvolumen der Chargen

$$V_{rz}(\mathcal{S}) := \sum_{chg \in \mathcal{C}'_{rz}} V(chg).$$

²⁸ mMV ist hier die Abkürzung für *minimales Maximalvolumen*. mMV_{rz} ist also die *kleinste Ganzfüllung*.

3. Bestimme die minimale Anzahl benötigter Chargen

$$n_{\min}(rz) := \left\lceil \frac{V_{rz}(\mathcal{S})}{\bar{V}_{rz}(\mathcal{S})} \right\rceil.$$

Dabei ermittelt $\lceil x \rceil$ die kleinste natürliche Zahl $\geq x$.

- Bestimme die maximale Anzahl benötigter Chargen

$$n_{\max}(rz) := |\mathcal{C}'_{rz}|.$$

Durchführung als while-Schleife:

Solange in der Menge \mathcal{C}' Chargen vorhanden sind, deren Volumen verteilt werden kann:

1. *Auswahl der Rezeptur:*

Bestimme die Rezeptur rz mit der geringsten Anzahl zu erzeugender Zwischenprodukte.

Analog zu Schritt 1a in Algorithmus 5.5 werden zunächst die Aufträge der Chargen umverteilt, die keine weiteren bzw. die wenigsten Zwischenprodukte in der Reihenfolgeplanung berücksichtigen müssen.

2. *Durchführung als for-Schleife:*

Für $n := n_{\min}(rz) \dots (n_{\max}(rz) - 1)$

- (a) Bestimme eine Teilmenge $\mathcal{C}_{rz}^n \subset \mathcal{C}'_{rz}$, von n Chargen, so daß die Anzahl der Chargen in der Menge

$$\mathcal{C}_{\text{end}}^n(rz) := \bigcup \{ chg_{\text{end}} \in CH_{\text{end}}(chg) \mid chg \in \mathcal{C}_{rz}^n \}$$

minimal ist.

Die Nebenbedingung bzgl. $\mathcal{C}_{\text{end}}^n(rz)$ stellt sicher, daß möglichst wenig Reihenfolgebeziehungen zwischen den Chargen erstellt werden.

- (b) *Lösche die bestehende Lösung:*

- Lösche die Chargen der Menge \mathcal{C}_{rz}^n und aller direkt oder indirekt verknüpften Chargen aus dem Scheduling \mathcal{S} .
- Lösche die Verknüpfungen zu allen vorhergehenden Chargen

$$chg_{\text{src}} \in CH_{\text{src}}(chg) \quad \forall chg \in \mathcal{C}_{rz}^n$$

und allen nachfolgenden Chargen

$$chg_{\text{trg}} \in CH_{\text{trg}}(chg) \quad \forall chg \in \mathcal{C}_{rz}^n.$$

- (c) *Erstelle eine neue Lösung:*

- Erstelle n neue Chargen mit dem Maximalvolumen

$$\bar{V}_{n,rz} := \frac{1}{n} \cdot V_{rz}(\mathcal{S}) \tag{5.33}$$

- Erstelle neue Verknüpfungen zu den vorhergehenden und nachfolgenden Chargen.

- Plane die neu erstellten Chargen mitsamt aller direkt und indirekt verknüpften in dem Scheduling \mathcal{S} ein und optimiere die Reihenfolge.

(d) *Abbruch:*

Wenn im Schritt 2c keine zulässige Reihenfolgeplanung erzeugt wurde, dann restauriere die Lösung von 2b, gehe zurück zu Schritt 2a und wähle eine andere Teilmenge aus oder erhöhe, sofern möglich, den Zähler von n .

3. *Abschluß für Rezeptur rz:*

Setze $\mathcal{C}' := \mathcal{C}' \setminus \mathcal{C}'_{rz}$.

■ A(5.6)

Die Notwendigkeit des Algorithmus wird exemplarisch anhand eines Testfalls von BLÖMER ([Blö99], S. 130 ff.) gezeigt. Zur besseren Nachvollziehbarkeit werden hier die Bezeichnungen von BLÖMER übernommen.

Beispiel 5.6 (Aufträge umverteilen) *Gegeben ist ein zweistufiger Produktionsprozeß. Ein Produkt 7 wird auf einem Reaktor Unit2 mit den Volumengrenzwerten $\underline{V}_{\text{Unit2}} = 1$ Mengeneinheiten (ME) und $\bar{V}_{\text{Unit2}} = 1.691$ ME hergestellt. Dazu wird ein Zwischenprodukt Produkt 4 benötigt, das auf einem Reaktor Unit1 mit den Volumengrenzwerten $\underline{V}_{\text{Unit1}} = 1$ ME und $\bar{V}_{\text{Unit1}} = 2.029$ ME hergestellt wird. Das Zwischenprodukt kann in einem Silo mit der Maximalkapazität $\bar{V}_{\text{Silo}} = 10.000$ ME gelagert werden (vgl. Abb. 5.8).*

Es soll ein Auftragsvolumen von 10.000 ME produziert werden. Die Produktionsdauer für Produkt 7 beträgt $t_{\text{Produkt7}} = 11$ Zeiteinheiten (ZE), die Reinigungszeit zwischen den Produktionsprozessen ebenfalls $t = 11$ ZE. Die Produktionsdauer für Produkt 4 beträgt $t_{\text{Produkt4}} = 20$ ZE mit anschließender Reinigungsdauer $t = 20$ ZE. Eine abschließende Reinigung nach dem Produktionsende der letzten Charge muß nicht durchgeführt werden.

Der Algorithmus 5.4 zur Volumenbestimmung und Chargenerstellung erzeugt dann 5 Chargen des Endproduktes Produkt 7 mit dem Volumen $V(\text{chg}) = 1.691$ ME und eine Endprodukt-Charge mit dem Volumen $V(\text{chg}) = 1.545$ ME. Entsprechend werden 5 Zwischenprodukt-Chargen von Produkt 4 mit dem Volumen $V(\text{chg}) = 1.691$ ME und eine Charge mit dem Volumen $V(\text{chg}) = 1.545$ ME erstellt. Die Lösung hat dann den Makespan = 231 ZE (vgl. Abb 5.9). Das Produkt 4 wird nicht zwischengelagert.

Das maximale Reaktorvolumen von Unit1 $\bar{V}_{\text{Unit1}} = 2.029$ ME wird durch die Chargen von Produkt 4 mit dem Volumen $V(\text{chg}) = 1.545$ ME bzw. $V(\text{chg}) = 1.691$ ME nicht voll ausgelastet. Tatsächlich genügt es hier für das Gesamtvolumen $V_{\text{Produkt4}}(\mathcal{S}) = 10.000$ ME 4 Chargen mit dem Volumen $V(\text{chg}) = 2.029$ ME und eine Charge mit dem Volumen $V(\text{chg}) = 1.884$ ME zu erstellen. Durch den Algorithmus 5.6 zum Umverteilen von Aufträgen wird diese geringere Anzahl an Chargen aus der vorhergehenden Lösung erzeugt. Das verbleibende Volumen einer Charge von Produkt 4, das nicht sofort in einer Charge von Produkt 7 verarbeitet wird, wird in dem Silo zwischengelagert. Somit kann zur Produktion der letzten Charge von Produkt 7 mit dem Volumen $V_{\text{Produkt7}} = 1.545$ ME die benötigte Menge des Zwischenproduktes Produkt 4 komplett aus dem Silo entnommen werden (vgl. Abb. 5.10).

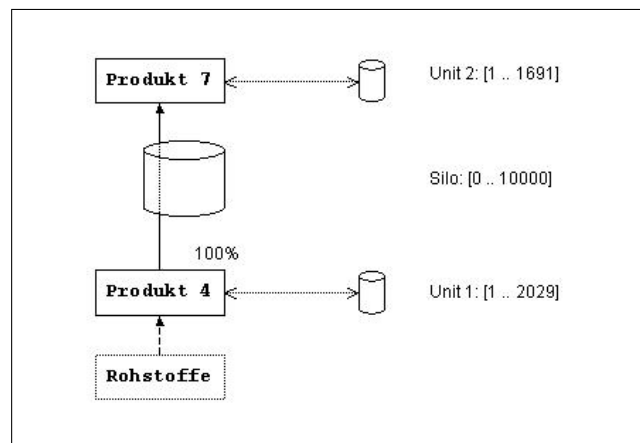
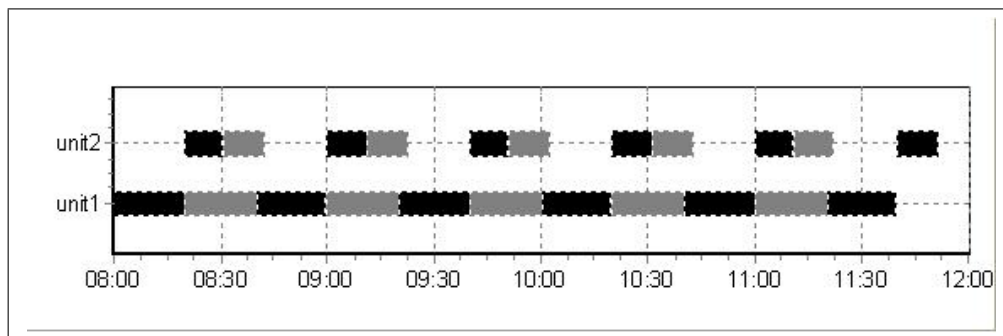
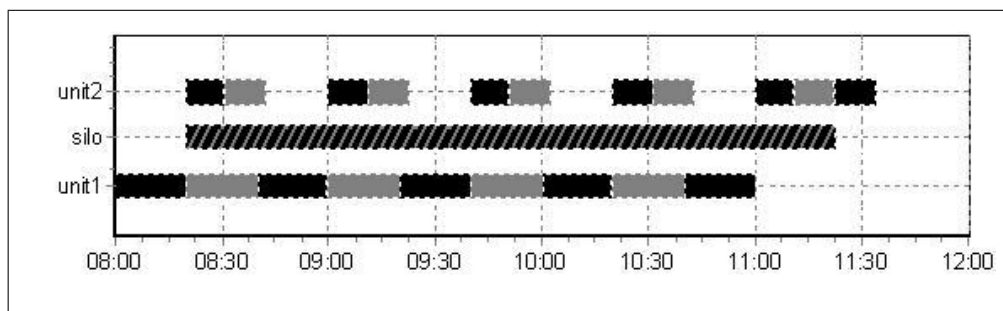


Abbildung 5.8: Aufträge umverteilen (Beispiel 5.6)



Produktionsprozesse sind schwarz dargestellt, Reinigungszeiten grau. Das Produktionsende ist 11:51 Uhr, also 231 Minuten nach Schichtbeginn um 8:00 Uhr.

Abbildung 5.9: Ausgangslösung (Beispiel 5.6)



Produktionsprozesse sind schwarz dargestellt, Reinigungszeiten grau; die Lagerung im Silo ist schraffiert. Das Produktionsende ist 11:33 Uhr, also 213 Minuten nach Schichtbeginn um 8:00 Uhr.

Abbildung 5.10: Lösung nach dem Umverteilen (Beispiel 5.6)

Bei der Anwendung des Algorithmus 5.6 muß darauf geachtet werden, daß mit steigender Anzahl der Verknüpfungen der Chargen zu den vorhergehenden und nachfolgenden Produktionsstufen die Komplexität der Reihenfolgeoptimierung steigt. Eine alternative Vorgehensweise in Bezug auf das Volumen der Endprodukt-Chargen bietet das *Verfahren 2* an, bei dem $\bar{V}_{chg}(\mathcal{S})$ ein maximal mögliches Reaktorvolumen bestimmt und eingeplant wird.

Verfahren 2: Chargenvolumen

Die zweite Vorgehensweise, die entwickelt wurde, bestimmt für die Endprodukt-Charge ein Maximalvolumen und reduziert dieses in der Nachoptimierung sukzessive.

Das Maximalvolumen einer Charge wird in Formel (5.12) unter Berücksichtigung der Maximalvolumina der Reaktoren und der berechneten Obergrenze des Volumens aufgrund der Zutaten beschrieben. Dieses Maximalvolumen wird jedoch eingeschränkt durch die Annahme (5.8), daß für jede Zutat einer Charge nur eine Zwischenprodukt-Charge hergestellt wird. Wie in Beispiel 4.4 gezeigt wurde, kann jedoch auch die parallele Produktion eines Zwischenprodukts stattfinden, das benötigte Volumen wird also in zwei oder mehr Chargen hergestellt.

In der weiteren Bestimmung der Chargenvolumina wird in Forderung (5.8) daher vernachlässigt; die Menge eines Zwischenprodukts kann dann in mehreren Chargen hergestellt werden. $\bar{V}_{chg}(\mathcal{S})$ wird dann definiert als

$$\bar{V}_{chg}(\mathcal{S}) := \max \{ \bar{V}_u \mid u \in UR_{rz}^t \}. \quad (5.34)$$

Es wird also eine Endprodukt-Charge für das technisch maximal realisierbare Volumen erzeugt und diese mitsamt aller weiteren erstellten Zwischen- und Vorprodukt-Chargen in der Reihenfolge eingeplant.

Verfahren 2: Nachoptimierung

Die Bestimmung des Chargenvolumens anhand von (5.34) und (5.28) gewährleistet, daß die Reaktoren der Endprodukte bis zum Maximalvolumen genutzt werden können. Die Reihenfolge, die sich nach der Einplanung einer solchermaßen erzeugten Endprodukt-Charge und der Chargen aller vorgelagerten Produktionsstufen ergibt, kann jedoch aus zwei Gründen nicht gültig sein.

- Es müssen mehrere Chargen auf einem Reaktor gefertigt werden. Die zunächst produzierten Chargen müssen dann entsprechend zwischengelagert werden. Wenn nicht ausreichend Kapazitäten zur Zwischenlagerung zur Verfügung stehen, dann kann *keine zulässige Reihenfolge* gefunden werden.
- Wie schon in Kapitel 5.2.1 beschrieben wurde, kann die zur Verfügung stehende Zeit eines Reaktors und die benötigte Zeit einer Produktionsabfolge vor der Erstellung der Reihenfolge nur geschätzt werden. Wenn in der betrachteten Vorgehensweise zusätzlich die Forderung (5.8) nicht erfüllt sein muß, kann das Ergebnis der Reihenfolgeplanung entsprechend noch stärker von diesen Schätzwerten abweichen. Auch wenn eine zulässige Reihenfolge für \mathcal{S} gefunden wird, kann es sein, daß \mathcal{S} *nicht mehr tageseigentlich* ist.

In beiden Fällen kann eine zulässige Lösung erreicht werden, indem das eingeplante Gesamtvolumen der Endprodukt-Charge (iterativ) reduziert wird. Dazu wird der Algorithmus 5.7 durchgeführt. Hierzu ist in jeder Iteration die Schrittweite der Volumenreduktion zu bestimmen. Diese wird in (5.39) aus den beiden Algorithmen 5.8 und 5.9 berechnet.

Volumina der Vorprodukte: Ein weiteres Problem ergibt sich - unabhängig von der Zulässigkeit einer Lösung - aus der Anwendung von Algorithmus 5.4. In der rekursiven Vorgehensweise des Algorithmus wird für die Erstellung einer neuen Zwischenprodukt-Charge nur das Volumen der rekursiv vorhergehenden Charge berücksichtigt. Es werden also nicht alle Chargen und damit das Gesamtvolumen der nachfolgenden Produktionsstufe betrachtet. Dies führt dazu, daß für ein (Zwischen-)Produkt, für das mehrere Chargen erstellt werden, eine mindestens gleich große Anzahl an (Vorprodukt-)Chargen erzeugt wird. Tatsächlich kann aber in Abhängigkeit von der Rezeptur und Produktionsanlage auch hier wieder eine geringe Auslastung der Reaktoren der Vorprodukt-Chargen in Bezug auf das Volumen entstehen.

Diese Problematik wird anhand des nachfolgenden Beispiels verdeutlicht.

Beispiel 5.7 (Chargenvolumen der Vorprodukte) *Gegeben ist die Rezeptur eines Endprodukts END, für das zwei Zwischenprodukte ZWP1 zu einem Anteil von 33,3 % und ZWP2 zu einem Anteil von 66,7 % benötigt werden. Für das Zwischenprodukt ZWP2 müssen wiederum die zwei Vorprodukte VP3 und VP4 mit den Anteilen 50 % und 50 % hergestellt werden. Die Zuordnung der Reaktoren zu den Rezepten, sowie deren Sumpfmengen und Maximalvolumina werden in Tabelle 5.2 aufgelistet.*

Anhand der nachfolgenden Berechnung werden die Chargen für das Beispiel 4.4 auf S. 94 erstellt. Die Produktionszeiten werden dabei nicht weiter berücksichtigt; alle Reaktoren stehen mit ausreichender Produktionsdauer zur Verfügung.

Es sind Chargen für das Produktionsvolumen $V_{\text{prod}}(\text{END}) = 900$ Mengeneinheiten (ME) herzustellen:

1. *Die Sumpfmenge für die Rezeptur END beträgt $\bar{V}_{\text{END}} = 300$ ME. Wegen $V_{\text{prod}} = 900 < 1000 = \bar{V}_{R1} = \bar{V}_{\text{END}}$ kann das gesamte Volumen für END in einer Charge erstellt werden. Für dieses Volumen müssen Chargen von ZWP1 im Umfang von $V_{\text{prod}}(\text{ZWP1}) = 300$ ME und von ZWP2 im Umfang von $V_{\text{prod}}(\text{ZWP2}) = 600$ ME hergestellt werden.*
2. *Das Produktionsvolumen $V_{\text{prod}}(\text{ZWP1})$ kann in einer Charge erstellt werden, für das Zwischenprodukt ZWP2 werden zwei Chargen ZWP2.a und ZWP2.b mit einem Volumen von jeweils 300 ME hergestellt.*
3. *Für jede der beiden Chargen von ZWP2 werden Chargen für die Vorprodukte VP3 und VP4 im Volumen von jeweils 150 ME hergestellt. Es sind also je zwei Chargen VP3.a und VP3.b, sowie VP4.a und VP4.b einzuplanen.*
4. *Das Maximalvolumen für eine Charge von VP3 beträgt $\bar{V}_{\text{VP3}} = 200$ ME; für VP4 ist dies $\bar{V}_{\text{VP4}} = 1000$ ME. Die beiden Chargen VP4.a und VP4.b können also zu einer Charge zusammengefasst werden.*

u	rz	\underline{V}_u	\overline{V}_u
R1	END	100	1000
R2	ZWP1,ZWP2	100	300
R3	ZWP2	100	300
R4	ZWP2	100	300
R5	VP3	100	200
R6	VP3	100	200
R7	VP4	100	1000

Tabelle 5.2: Volumengrenzwerte der Reaktoren (Beispiel 5.7)

Nach der Erstellung und Einplanung der neuen Chargen wird daher zunächst der Algorithmus 5.5 zum Zusammenfassen von Chargen oder Algorithmus 5.6 zum Umverteilen von Aufträgen durchgeführt, um hier eine bessere Auslastung der Reaktoren für Vorprodukte zu erreichen. Diese Algorithmen werden dabei nur auf die *neu erstellten Chargen* angewandt. Wenn durch das Zusammenfassen oder Umverteilen eine vorher zulässige Lösung anschließend unzulässig wird, müssen die Ergebnisse des durchgeführten Algorithmus wieder rückgängig gemacht werden.

Algorithmus zur Volumenreduktion: Der Algorithmus zur Volumenreduktion erstellt und löscht Chargen mit einem stetig kleinerem Volumen bis eine zulässige Lösung gefunden wird.

Algorithmus 5.7 (Volumen reduzieren)

Vorarbeiten und Einzelschritte:

Die folgenden Schritte werden einmalig oder innerhalb jeder Iteration der nachfolgenden Volumenreduktion durchgeführt.

- In der rollierenden Planung wird zu Beginn jeder Planungsphase einer Planungsperiode d für jedes Endprodukt-Rezept rz das Maximalvolumen $\overline{V}_{rz,d} := \max\{\overline{V}_u | u \in UR_{rz}\}$ bestimmt.
- Eine neu erstellte Endprodukt-Charge und alle Zwischen- und Vorprodukt-Chargen werden in der Menge \mathcal{C}_{rz} gespeichert und dem Scheduling \mathcal{S} hinzugefügt.
- Für die *neu* erstellten Zwischen- und Vorprodukt-Chargen der Menge \mathcal{C}_{rz} wird Algorithmus 5.5 oder Algorithmus 5.6 durchgeführt.
- Nach Durchführung eines Algorithmus zum Zusammenfassen von zwei oder mehr Chargen wird eine Reihenfolgeoptimierung durchgeführt.

Durchführung Volumenreduktion:

Solange durch das Einplanen der neuen Chargen \mathcal{C}_{rz} im Scheduling \mathcal{S} keine zulässige oder tagesgültige Reihenfolge erreicht wird:

1. Abbruch des Algorithmus, wenn \mathcal{S} *zulässig, aber nicht tagesgültig* ist und in der Menge \mathcal{C}_{rz} von jedem Produkt nur eine Charge vorhanden ist.
2. Aktualisiere $\bar{V}_{rz,d} := \min \{ \bar{V}_{rz,d}, V(chg) \}$.
Dabei ist $V(chg)$ das Volumen der Endprodukt-Charge in der Menge \mathcal{C}_{rz} .
3. Bestimme das Volumen V_{red} anhand von Formel (5.39).
 V_{red} ist das kleinste Volumen, das für eine Reduktion der Durchlaufzeit der Chargen \mathcal{C}_{rz} benötigt wird.
4. Löse die Chargen \mathcal{C}_{rz} auf.
Die Chargen werden aus dem Scheduling \mathcal{S} entfernt, und die zugeordneten Aufträge jeder Charge (vgl. Schritt 3 in Algorithmus 5.4) werden frei. Sie müssen später in einer anderen Charge zusammengestellt werden.
5. Bestimme das größtmögliche Maximalvolumen unterhalb $\bar{V}_{rz,d}$

$$\bar{V}_{red}(\mathcal{S}) := \max \{ \bar{V}_u \mid u \in UR_{rz}^t \text{ und } \bar{V}_u < \bar{V}_{rz,d} \} \quad (5.35)$$

Beachte, daß die Chargen $chg \in \mathcal{C}_{rz}$ aus dem Scheduling \mathcal{S} gelöscht wurden. Ohne die Einschränkung $\bar{V}_u < \bar{V}_{rz,d}$ erhält man mit (5.35) denselben Wert wie für $\bar{V}_{chg}(\mathcal{S})$ in (5.34).

6. Bestimme das neue Volumen einer Endprodukt-Charge des Rezeptes rz anhand von Formel (5.28). Verwende anstelle von $\bar{V}_{chg}(\mathcal{S})$ den folgenden Wert

$$\bar{V}_{rz,red}(\mathcal{S}) := \max \{ \bar{V}_{red}(\mathcal{S}), \bar{V}_{rz,d} - V_{red} \}. \quad (5.36)$$

7. *Plane eine neue Endprodukt-Charge:*

- (a) Erstelle eine neue Endprodukt-Charge und die jeweiligen Zwischen- und Vorprodukt-Chargen anhand des Algorithmus 5.4 für das Volumen $\bar{V}_{rz,red}(\mathcal{S})$ und füge sie der Menge \mathcal{C}_{rz} hinzu.
- (b) Plane die neu erstellten in dem Scheduling \mathcal{S} ein und führe den Algorithmus 5.5 oder den Algorithmus 5.6 für diese durch.
- (c) Führe eine Reihenfolgeoptimierung durch.

■ A(5.7)

Durch die iterative Reduktion des Volumens ist in dem letzten Durchlauf des Verfahrens das Volumen so gewählt, daß von jedem Produkt jeweils nur eine Charge erstellt wird. In diesem Fall ist dann auch die Annahme (5.8) erfüllt. Wenn nach dieser letzten Iteration zur Reduktion des einzuplanenden Volumens weiterhin keine zulässige Lösung in der Reihenfolgeplanung gefunden wird, dann sind Produktionsstruktur und Rezeptur nicht zur Herstellung aufeinander abgestimmt. In diesem Fall sind alle Aufträge für dieses Produkt aus der Planung zu entnehmen.

Wenn eine *zulässige, aber nicht tagesgültige* Lösung erstellt wird, dann kann dies auch aus der Dauer der einzelnen Produktionsstufen folgen, die insgesamt länger als die verfügbare Produktionszeit dauern. In diesem Fall sind die Chargen im Scheduling einzuplanen. Chargen, die über das Ende der verfügbaren Produktionszeit hinaus reichen, werden zu Beginn der nächsten Planungsperiode der rollierenden Planung als verspätete Chargen zuerst eingepplant.

Bestimmung des Reduktionsvolumens: Sofern die *Lösung nicht zulässig* ist und für wenigstens ein Produkt mehr als eine Charge erstellt und eingeplant wurde, kann die Anzahl der Chargen minimiert werden, damit eine zulässige Lösung erreicht wird. Die Anzahl der Chargen kann iterativ so weit reduziert werden, bis eine zulässige Lösung gefunden wird.

Der Betrag, um den das Volumen der Endprodukt-Charge reduziert werden soll, wird dann folgendermaßen bestimmt:

Algorithmus 5.8 (Bestimmung des Reduktionsvolumens, Chargen)

Gegeben ist die Menge \mathcal{C}_{rz} einer neu eingeplanten Endprodukt-Charge chg_{rz} des Rezeptes rz mit allen Zwischen- und Vorprodukt-Chargen. ZT ist die Menge der Zutaten, die nicht bereits als Rohstoffe verfügbar sind, d.h. der Zwischen- und Vorprodukte. Dann ist $\mathcal{C}_{rz} := \{chg_{rz}\} \cup \bigcup_{zt \in ZT} \mathcal{C}_{zt}$.

1. Bestimme für alle Zutaten $zt \in ZT$

- die Anzahl der Chargen der Zutat $n_{zt} := |\mathcal{C}_{zt}|$,
- das Volumen der kleinsten Charge $\underline{V}_{zt} := \min \{V(chg) \mid chg \in \mathcal{C}_{zt}\}$.
- und das Gesamtvolumen aller Chargen $V_{zt} := \sum_{chg \in \mathcal{C}_{zt}} V(chg)$.

2. Für alle Zutaten mit $n_{zt} > 1$ bestimme

- den Anteil der Zutat am Gesamtvolumen $f_{zt} := V_{zt}/V(chg_{rz})$,
- und den auf das Volumen bezogenen Beitrag der kleinsten Charge am Gesamtvolumen $V_{zt,rz} := \underline{V}_{zt}/f_{zt}$.

3. Für alle Zutaten aus Schritt 2 bestimme das Reduktionsvolumen

$$V_{red,rz} := \min \{V_{zt,rz} \mid zt \in ZT\}. \quad (5.37)$$

■ A(5.8)

Wenn dagegen *nach einer Reihenfolgeoptimierung* die Lösung *nicht tagesgültig* ist, dann muß eine Reduktion der benötigten Zeit durchgeführt werden. Dies kann zum einen durch eine Reduktion des Volumens mit Algorithmus 5.8 erfolgen. Dann wird die Chargenanzahl und entsprechend die benötigte Produktionszeit reduziert. Weiterhin beeinflusst das Volumen auch die Dauer der Abfüllung. Entsprechend kann das Volumen auch so reduziert werden, daß durch Reduktion der Abfülldauer die Lösung tagesgültig wird.

Dieses Volumen wird mit dem folgenden Algorithmus bestimmt.

Algorithmus 5.9 (Reduktionsvolumen, Abfüllung)

- Gegeben ist eine nicht tagesgültige Reihenfolgeplanung \mathcal{S} einer Planungsperiode d mit der verfügbaren Produktionszeit t_d und dem Makespan t_{MKS} . Das Ende der Reihenfolge ragt daher um die Dauer $t_{over} := t_{MKS} - t_d$ über das Ende der verfügbaren Produktionszeit hinaus.

- In dem Scheduling \mathcal{S} ist die Charge chg_{rz} mit dem Volumen $V(chg_{rz})$ eingeplant. Dieses wird in Gebinde bzw. TKW abgefüllt: $V(chg_{rz}) := V(chg_{rz,Geb}) + V(chg_{rz,TKW})$. Die Abfüllung der Charge chg_{rz} erzeugt in \mathcal{S} für eine Gebinde-Abfüllung den Parallelknoten $kn_{\text{PARALLEL}}(chg_{rz,Geb})$ und $kn_{\text{PARALLEL}}(chg_{rz,TKW})$ für eine TKW-Abfüllung.
- Die Dauer der Gebindeabfüllung der Produktionsanlage ist volumenabhängig von der Abfüllgeschwindigkeit v_{Geb} . Demnach dauert eine Abfüllung $t_{Geb}(V) := V/v_{Geb}$. Entsprechend ist für eine TKW-Abfüllung die Abfüllgeschwindigkeit v_{TKW} gegeben. Die Abfülldauer berechnet sich analog.

Durchführung:

1. Bestimme die kritischen Pfade \mathcal{P} in dem Scheduling \mathcal{S} . Wenn einer der beiden Knoten $kn_{\text{PARALLEL}}(chg_{rz,Geb})$ oder $kn_{\text{PARALLEL}}(chg_{rz,TKW})$ auf einem kritischem Pfad liegt, dann führt eine Verringerung der Dauer der Abfüllung zu einer Minimierung des **Makespan**. Wenn diese Knoten in *keinem* der kritischen Pfade enthalten ist, dann Abbruch des Verfahrens.
2. *Sofern* der jeweilige Parallelknoten der Abfüllung auf einem kritischem Pfad liegt, bestimme das Volumen der jeweiligen Abfüllmenge, das zu einer Reduzierung des **Makespan** führen kann:

$$V_{red,i} := \max \{ V(chg_{rz,i}), t_{over} \cdot v_i \} \quad \text{mit } i \in \{Geb, TKW\}$$

Wenn der Parallelknoten nicht auf einem kritischem Pfad liegt, setze $V_{red,i} := \infty$.

3. Bestimme das Reduktionsvolumen der Abfüllungen für die Abfüllarten, deren Parallelknoten (in Schritt 2) auf einem kritischem Pfad liegen.

$$V_{red,abf} := \min \{ V_{red,Geb}, V_{red,TKW} \} \quad (5.38)$$

■ A(5.9)

Das Reduktionsvolumen im Falle einer *nicht tagesgültigen* Lösung von \mathcal{S} ergibt sich dann aus den Ergebnissen (5.37) in Algorithmus 5.8 und (5.38) in Algorithmus 5.9

$$V_{red} := \min \{ V_{red,rz}, V_{red,abf} \}. \quad (5.39)$$

Praxis: Implementierung in einem Planungssystem

In dem *Verfahren 2* können für ein Produkt mehrere Chargen erstellt werden. Diese Vorgehensweise widerspricht der Auswahl des Produktionsrezeptes in Kapitel 5.3.1, die auf der Annahme (5.8) beruht. Hier können von einem Produktionsplaner aber andere Auswahlregeln definiert werden, um so die Ergebnisse unterschiedlicher Regeln in einer Simulation miteinander zu vergleichen.

Kapitel 6

Schlußbetrachtung

In dieser Arbeit wurden die Eigenschaften der chargenorientierten Produktion in der Prozeßindustrie dargestellt und die daraus entstehenden Anforderungen an ein Verfahren zur operativen Planung erarbeitet. Darauf aufbauend wurde eine Überprüfung der bestehenden Planungssysteme und eine Evaluation ihrer Schwachstellen durchgeführt. Im Ergebnis wurde in dieser Arbeit ein Verfahren entwickelt, das einem Produktionsplaner ermöglicht, für ein beliebiges Volumen von Kundenaufträgen die Volumina der zu erstellenden Chargen und deren Produktionsreihenfolge zu bestimmen.

Eines der wesentlichen Merkmale der Chargenproduktion ist der mehrstufige Herstellungsprozeß, bei dem die jeweils produzierte Menge geschlossen an die nächste Produktionsstufe weitergegeben wird. Aufgrund der Eigenschaften des Materials ist es erforderlich, daß für ein Zwischenprodukt immer ein Behältnis zu Lagerung vorhanden ist. Bei der Planung muß also insbesondere die Existenz von Zwischenlagermöglichkeiten im Silo oder der Verbleib des Materials für eine Wartezeit in dem produzierendem Reaktor überprüft bzw. geplant werden. Bei der Herstellung in Mehrproduktanlagen, die besonders in mittelständischen Unternehmen der Prozeßindustrie eingesetzt werden, erfordert die Produktion unterschiedlicher Produkte eine Reinigung zwischen den einzelnen Herstellungsschritten. Für diese reihenfolgeabhängige Reinigung muß der Einsatz entsprechenden Reinigungspersonals zusätzlich geplant werden. Das aus diesen Anforderungen resultierende Problem der Reihenfolgeplanung wird als Connected Job Shop-Problem bezeichnet (vgl. Kapitel 4.1).

Für die Modellierung eines solchen Reihenfolgeproblems wurden verschiedene, aus der Literatur bekannte Vorgehensweisen untersucht. Die Formulierung anhand der gemischt-ganzzahligen Programmierung wird in vielen Arbeiten angewandt. Wesentlich für die Performance eines Optimierungsverfahrens, das auf dieser Problemformulierung aufbaut, ist jedoch die Anzahl der Binärvariablen. Mit steigender Anzahl der Variablen wird das Auffinden einer guten oder auch nur zulässigen Lösung komplizierter und aufwendiger. Wesentlich ist hierbei die Entscheidung, ob die Zeitpunkte für die Reihenfolgeplanung anhand eines fest vorgegebenen Rasters bestimmt werden, oder ob beliebige Zeitpunkte innerhalb des Planungszeitraums für den Beginn eines Herstellungsprozesses möglich sind. In dieser Arbeit wurde eine graphentheoretische Modellierung für die Reihenfolgeplanung verwendet. Dies ermöglicht eine rasterunabhängige Planung für alle Prozeßschritte einer Reihenfolge. Diese Modellierung erlaubt auch eine einfache Programmierung von heuristischen Optimierungsverfahren.

In dem Modell zur Reihenfolgeplanung dieser Arbeit wird sowohl der Einsatz des Reini-

gungspersonals als Restriktion und damit als möglicher Engpaß der Planung, als auch die Abfüllung der Endprodukte berücksichtigt. Eine gleichzeitige Betrachtung dieser beiden Anforderungen ist aus der Literatur zur Planung in chargenorientierten Produktionsprozessen bisher nicht bekannt. Damit wird in dieser Arbeit ein Modell vorgestellt, das einem Produktionsplaner ermöglicht, das Ergebnis einer operativen Produktionsplanung direkt in eine Fertigungssteuerung zu übernehmen, die diese Anforderungen stellt.

Zur Optimierung der Reihenfolgeplanung (vgl. Kapitel 4.5) wurde eine zweistufiges Verfahren entwickelt mit einer *inneren Optimierung* und einer *äußeren Optimierung*. Bei der inneren Optimierung wird nur die Reihenfolge der Reinigungen und Abfüllprozesse variiert. Bei der äußeren Optimierung wird die Zuordnung und eigentliche Abfolge der Produktionsschritte auf den einzelnen Anlagen variiert. Zur Optimierung wird hier das Simulated Annealing angewandt. Durch Vorgabe weniger Parameter durch einen Produktionsplaner werden die notwendigen Optimierungsparameter innerhalb des Verfahrens ermittelt, ohne daß ein Anwender, die genaue Vorgehensweise des Verfahrens kennen muß.

Basierend auf diesem Verfahren zur Reihenfolgeplanung wurden anschließend verschiedene Vorgehensweisen entwickelt, um in einer rollierenden Planung die Volumina und Anzahl der Chargen zu bestimmen, die in der zuvor vorgestellten Reihenfolgeplanung eingeplant werden. Hierzu wird zunächst beispielhaft die Berechnung der Volumina der Vorprodukte bei einer Beschränkung auf jeweils eine Charge je Vorprodukt aufgezeigt. Nach Aufhebung dieser Restriktion können für ein Vor- oder Zwischenprodukt auch mehrere Chargen erstellt werden, um somit eine serielle oder parallele Produktion planen zu können.

In einem ersten Planungsansatz werden zunächst Chargen mit dem kleinstmöglichen Volumen erstellt und die Anzahl der Chargen anschließend reduziert, indem diese zusammengefasst werden. Eine solche Vorgehensweise erfordert aber eine wesentlich größere Anzahl von Planungsschritten. Zunächst müssen die Chargen mit kleinem Volumen erstellt und in der Reihenfolge eingeplant werden. Anschließend müssen die Chargen ausgewählt werden, die zusammengefasst werden können. Nach dem Zusammenfassen muß erneut eine Reihenfolgeplanung durchgeführt werden. Ein weitaus geringerer Planungsaufwand wird bei dem dem zweiten entwickelten Verfahren benötigt. Hierbei wird zunächst das zulässige Maximalvolumen einer Charge ermittelt. Entsprechend wird hierzu eine Endproduktcharge mit allen Vor- und Zwischenprodukt-Chargen erstellt und in der Reihenfolge eingeplant. Sofern die Reihenfolgeplanung eine unzulässige Lösung zurückgibt, wird sukzessive das Volumen der Charge reduziert, bis in der Reihenfolgeplanung eine zulässige Lösung gefunden wird.

Der in dieser Arbeit vorgestellte Ansatz zur Produktionsplanung bietet die Möglichkeit, die chargenorientierte Produktion in einem einzigen Schritt zu planen. Das Volumen und die Zusammenstellung der Chargen muß zuvor also nicht *separat* als Zwischenergebnis berechnet werden, um dies als Input für ein zweites Planungsverfahren, die Reihenfolgeplanung, zu verwenden. Die Volumina der Chargen werden stattdessen in Abhängigkeit der verfügbaren Zeit berechnet. Ausgehend von dem beschriebenen Problem und dem daraus entwickelten Modell sind weitere Untersuchungen in Bezug auf die Modellierung möglich, die wiederum entsprechende Veränderungen in dem Verfahren erfordern.

- In dem Planungsproblem und dem dazu entwickelten Modell wird die Verrohrung der Anlage nicht weiter berücksichtigt. Es wird angenommen, daß hier genügend Leitungen zur Verfügung stehen, so daß diese nicht als Engpaß die Produktion beeinträchtigen. Wenn jedoch z.B. aus baulichen Gründen auf eine komplette Verrohrung

der Anlage verzichtet wurde oder einzelne Leitungsstücke mehrere Anlagekomponenten miteinander verbinden, dann muß zusätzlich zu der zeitlichen Belegung der Produktions- und Lagerkomponenten auch der Stofffluß durch die Anlage geplant werden.

- Bei der Planung der Produktionsprozesse kann der Fall auftreten, daß die verfügbare Dauer eines Produktionstages überschritten wird, die Herstellung also nicht mehr an dem eingeplanten Tag fertiggestellt werden kann. Dann muß der komplette Produktionsschritt auf den nachfolgenden Tag verschoben werden. Wenn es dagegen möglich ist, daß ein Produktionsschritt an einem oder mehreren durch die Rezeptur festgelegten Zeitpunkten unterbrochen werden kann, dann kann die Herstellung eines Produktes an dem eingeplanten Tag noch begonnen und am nachfolgenden Tag entsprechend fortgesetzt werden.

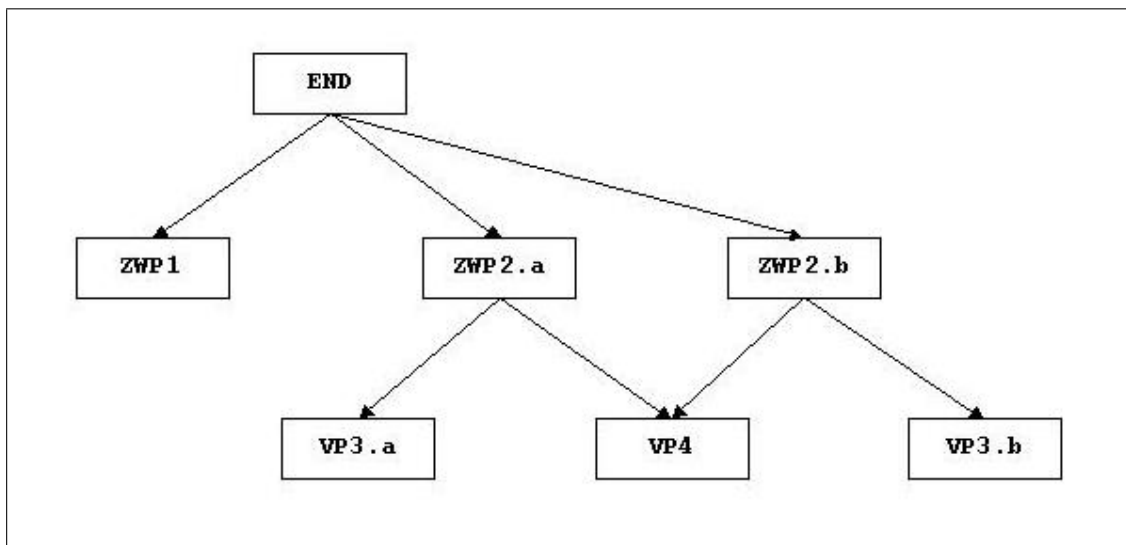
Das Modell und das Verfahren dieser Arbeit wurden in der Programmiersprache C++ programmiert. Die Modellierung und Reihenfolgeplanung können somit entsprechend einfach an weitere Anforderungen der verfahrenstechnischen Fertigungssteuerung angepasst werden. Hierzu muß die graphentheoretische Modellierung zur Reihenfolgeplanung erweitert werden.

Da die Optimierung anhand einer rollierenden Planung über einen vorgegebenen Planungshorizont erfolgt, kann ein Planungslauf über nur einen oder wenige Tage durchgeführt werden. Die Laufzeiten der Planung sind entsprechend kurz und erlauben somit zusätzlich die Möglichkeit, eine Variantenplanung durchzuführen. Die Produktionsplanung kann entsprechend flexibel auf Änderungen der Auftragslage reagieren.

Anhang A

Erzeugen einer Startlösung

In diesem Anhang werden die Schritte des Algorithmus (4.2) zum Erzeugen einer Startlösung anhand des Beispiel 4.4 von S. 94 beschrieben. Dazu wird zunächst noch einmal die Reihenfolgebeziehung (Abb. A.1) dargestellt und die Reaktoren aufgelistet, auf denen diese Produkte hergestellt werden können (Tabelle A.1).



Das Endprodukt **END** benötigt eine Charge des Zwischenproduktes **ZWP1** und zwei Chargen des Zwischenproduktes **ZWP2**. Die Produktion des Zwischenproduktes **ZWP2** erfordert wiederum die Produktion der Vorprodukte **VP3** und **VP4**. Die Charge **VP4** bedient dabei beide Chargen des Zwischenproduktes **ZWP2**.

Abbildung A.1: Einzuplanende Chargen (Beispiel 4.4)

chg	$UR(chg)$
END	R1
ZWP1	R2
ZWP2.a, ZWP2.b	R2, R3, R4
VP3.a, V3.b	R5, R6
VP4	R7

Tabelle A.1: Zugeordnete Reaktoren (Beispiel 4.4)

Die einzelnen Schritte des Algorithmus werden in der Tabelle A.2 dargestellt. Die Nummer der aktuellen Zwischenlösung wird in der Spalte **Z** angegeben. Wenn diese Zwischenlösung Teil des endgültig erzeugten Scheduling \mathcal{S} ist, dann wird diese zusätzlich mit einem * gekennzeichnet (vgl. Tabelle 4.2). In der Spalte **A** wird die Nummer des aktuellen Iterationsschrittes im Algorithmus (4.2) angegeben.

In der Spalte *Operationen* werden die Veränderungen der Variablen und Mengen des Algorithmus beschrieben: Gegeben ist die Menge \mathcal{C} der einzuplanenden Chargen, die in Reihenfolgeplan \mathcal{S} einzuplanen sind. Die Menge $\mathcal{C}' \subset \mathcal{C}$ ist dann die Teilmenge der Chargen, die in dem aktuellen Planungsschritt des Algorithmus dem Scheduling \mathcal{S} hinzugefügt werden können. Dazu wird zunächst eine einzuplanende Charge chg und der Reaktor $u(chg)$, auf dem diese eingeplant wird, ausgewählt. Das Scheduling wird dann in der Form $\mathcal{S} := \mathcal{S} \oplus \{u(chg) \rightarrow chg\}$ erweitert. Entsprechend wird chg aus der Menge \mathcal{C}' in die Menge $\mathcal{C}_{\mathcal{S}}$ der bereits eingeplanten Chargen verschoben und die Menge \mathcal{U}'_R der Reaktoren, auf denen die Charge eingeplant werden kann, reduziert.

z	A	Operationen
	1	$\mathcal{C}' = \{\text{END}\}$
	2	$\text{chg} = \text{END}, \mathcal{U}(\text{END}) = \{R1\}$
	3	$\mathcal{L} = \{\{\text{END}\}\}, \mathcal{C}' = \{\text{ZWP1}, \text{ZWP2.a}, \text{ZWP2.b}\}$
	2	$\text{chg} = \text{ZWP2.a}, \mathcal{U}(\text{ZWP2.a}) = \{R2, R3, R4\}$
	3	$\mathcal{L} = \{\{\text{END}\}, \{\text{ZWP1}, \text{ZWP2.a}, \text{ZWP2.b}\}\}, \mathcal{C}' = \{\text{VP3.a}, \text{VP4}\}$
	2	$\text{chg} = \text{VP3.a}, \mathcal{U}(\text{VP3.a}) = \{R5, R6\}$
	3	-/-
	4	-/-
	5	$u(\text{VP3.a}) = R5, \mathcal{U}(\text{VP3.a}) = \{R6\}$
1*	6	$\mathcal{S} = \{R5 \rightarrow \text{VP3.a}\},$ $\mathcal{C} = \{\text{END}, \text{ZWP1}, \text{ZWP2.a}, \text{ZWP2.b}, \text{VP3.b}, \text{VP4}\}, \mathcal{C}' = \{\text{VP4}\},$ $\mathcal{C}_S = \{\text{VP3.a}\}$
	1	-/-
	2	$\text{chg} = \text{VP4}, \mathcal{U}(\text{VP4}) = \{R7\}$
	3	-/-
	4	-/-
	5	$u(\text{VP4}) = R7, \mathcal{U}(\text{VP4}) = \emptyset$
2*	6	$\mathcal{S} = \{R5 \rightarrow \text{VP3.a}, R7 \rightarrow \text{VP4}\},$ $\mathcal{C} = \{\text{END}, \text{ZWP1}, \text{ZWP2.a}, \text{ZWP2.b}, \text{VP3.b}\}, \mathcal{C}' = \emptyset,$ $\mathcal{C}_S = \{\text{VP4}, \text{VP3.a}\}$

Tabelle A.2: Einzelschritte des Algorithmus (4.2) zur Erzeugung der ersten Zwischenlösungen aus Beispiel 4.4 (vgl. Tabelle 4.2 aus S. 96).

Literaturverzeichnis

- [AF02a] **Anschütz, S. und S. Friedrich** *Supply Chain Management in der Prozessindustrie - Ergebnisse einer bundesweiten Branchenstudie*. Hamburg: hansebuch Verlag, 2002
- [AF02b] **Anschütz, S. und S. Friedrich** *Supply Chain Management in der Prozessindustrie - Eine IT-Studie von initations AG und IWI Hamburg*. Download zu [AF02a], URL: http://www.initations.com/pdfs/PUB_Management_Summary.pdf; abgerufen am: 28.11.2002
- [Ah95] **Ahleff, J.** *Analyse von mathematischen Modellen zur Reihenfolgeplanung in der Prozeßindustrie*. RWTH Aachen, Masterarbeit (Lehrstuhl f. Unternehmensforschung), 1995
- [Aig93] **Aigner, M.** *Diskrete Mathematik*. Wiesbaden: vieweg Verlag, 1993
- [Ak56] **Akers, S.B.** *A Graphical Approach to Production Scheduling Problems*. Operations Research (1956) 4, S. 244-245
- [ALS96] **Allweyer, T.; Loos, P. und A.-W. Scheer** *Requirements and New Concepts for Production Planning and Scheduling in the Process Industries*. Vortrag: Conference I-CIMPRO96, Eindhoven (Niederlande), 1996 URL: <http://www.tu-chemnitz.de/wirtschaft/wi2/home/loos/cimpro96.ps> abgerufen am: 15.04.2005
- [Ans01] **Anschütz, S.** *Ablaufplanung bei Chargenproduktion - Ein Dekompositionsansatz auf der Basis Evolutionärer Algorithmen*. Wiesbaden: DUV, 2001
- [AMPL05] **AMPL** *Solvers that work with AMPL*. URL: <http://www.ampl.com/solvers.html>; abgerufen am: 20.10.2005
- [ASS86] **Axsätter, S.; Schneeweiss, Ch. und E. Silver (Hrsg.)** *Multi-Stage Production Planning and Inventory Control - Lecture Notes in Economics and Mathematical Systems*. Heidelberg: Springer Verlag, 1986.
- [BCT83] **Billington, P.J.; Mc Clain, J.O. und L.J. Thomas** *Mathematical Programming Approaches to Capacity-Constrained MRP Systems - Review, Formulation and Problem Reduction*. Management Science (1983) 10, S. 1126-1141.

- [BE+93] **Błazewicz, J.; Ecker, K.; Schmidt, G. und J. Węglarz** *Scheduling in Computer and Manufacturing Systems*. Heidelberg: Springer Verlag, 1993.
- [BG00] **Blömer, F. und H.O. Günther** *LP-based heuristics for scheduling chemical batch processes*. International Journal of Production Research (2000) 38, S. 1029-1051;
- [BH+98] **Burkhard, R.; Huijter, M.; Klinz, B. und M. Wennink** *A process scheduling problem arising from chemical production planning*. Optimization Methods & Software (1998) 10, S. 175-196
- [BH00] **Brucker, P. und J. Hurink** *Solving a chemical batch scheduling problem by local search*. Annals of Operations Research (2000) 96, S. 17-38
- [BHH03] **Brucker, P.; Heitmann, S. und J. Hurink**. *Flow-shop problems with intermediate buffers*. OR Spectrum (2003) 25, p. 549 - 574.
- [BKR98] **Burkhard, R.; Kocher, M. und R. Rudolf** *Rounding strategies for mixed integer programs arising from chemical production planning*. Yugoslav Journal of Operations Research (1998) 8, S. 9-23
- [B-H03] **Baecke-Heger, F.** *Das Ende der Dinosaurier-Systeme*. Markt und Mittelstand (2003) 1, S. 82-85
- [BJS95] **Bohachevsky, I.O.; Johnson, M.E. und M.L. Stein** *Simulated Annealing and generalizations*. in [Kv95], S. 3-24
- [Blö99] **Blömer, F.** *Produktionsplanung und -steuerung in der chemischen Industrie*. Wiesbaden: DUV, 1999.
- [BNS98] **Bartsch, R.; Noack K. und B. von Sta.** *R/3-Gesamtlösung für die Prozessindustrie*. in [JU98], S. 219-236
- [BPS99] **Brailsford, S.C.; Potts, C.N. und B.M. Smith** *Constraint satisfaction problems: Algorithms and applications*. European Journal of Operational Research (1999) 119, S. 557-581
- [Bro05] **Brockhaus** *Der Brockhaus - Computer und Informationstechnologie*. Mannheim: F.A. Brockhaus, 2005
- [Bru95] **Brucker, P.** *Scheduling Algorithms*. Heidelberg: Springer Verlag, 1995.
- [BD+99] **Brucker, P.; Drexel, A.; Möhring, R.; Neumann, K. und E. Pesch** *Resource-constrained project scheduling: notification classification, models and methods*. European Journal of Operational Research (1999) 111; S. 3-41
- [Brü02] **Brüssau, K.** *Evolutionäre Algorithmen zur simultanen Losgrößen- und Ablaufplanung*. Wiesbaden: DUV, 2002.

- [CC+95] **Chrétienne, P.; Coffmann, E.G.; Lenstra, J.K. und Z. Liu** *Scheduling Theory and its Applications*. Chichester: Jon Wiley & Sons, 1995
- [Cer85] **Černý, V.** *Thermodynamical Approach to the Travelling Salesman Problem: An Efficient Simulation Algorithm*.
Journal of Optimization Theory and Applications (1985) 45, S. 41-51
- [CHG97] **Cerda, J.; Henning, G. und I. Grossmann** *A Mixed-Integer Linear Programming Model for Short-Term Scheduling of Single-Stage Multiproduct Batch Plants with Parallel Lines*.
Industrial & Engineering Chemistry Research (1997) 36, S. 1695-1707
- [CM94] **Corsten, H. und C. May** *Besonderheiten der Produktion in der Verfahrensindustrie und ihre Auswirkungen auf PPS-Systeme*.
in [Cor94], S. 871-889
- [CMM67] **Conway, R.W.; Maxwell, W.L. und L.W. Miller** *Theory of Scheduling*. Addison-Wesley, 1967
- [Cor94] **Corsten, H. (Hrsg.)** *Handbuch Produktionsmanagement*. Wiesbaden: Gabler Verlag, 1994
- [Cor98] **Corsten, H.** *Produktionswirtschaft*.
München: Oldenbourg Wissenschaftsverlag, 1998, 7. Aufl.
- [Cra96] **Crawford, J.** *An approach to resource constrained project scheduling*. Proceedings 1996 AI and Manufacturing Research Planning Workshop. Albuquerque, NM, AAAI Press
- [Dash05] **dashoptimization** *Case Studies*. URL:
<http://www.dashoptimization.com/home/services/publications/casestudies.html>; abgerufen am: 20.10.2005
- [DD95] **Domschke, W. und A. Drexl** *Einführung in Operations Research*. Heidelberg: Springer Verlag, 1995, 3. Aufl.
- [Der95] **Derstroff, Matthias C.** *Mehrstufige Losgrößenplanung mit Kapazitätsbeschränkungen* Heidelberg: Physica-Verlag, 1995
- [DIN87] **Deutsches Institut für Normung e.V.** *DIN 8580 - Fertigungsverfahren*. Berlin: Beuth Verlag, 1987
- [DIN88] **Deutsches Institut für Normung e.V.** *DIN 28004 - Fließbilder verfahrenstechnischer Anlagen*. Berlin: Beuth Verlag, 1988
- [DIN01] **Deutsches Institut für Normung e.V.** *EN ISO 10628 - Fließschemata für verfahrenstechnische Anlagen*.
Berlin: Beuth Verlag, 2001
- [DJV01] **Dattal, B.; Jayaraman, V. K. und B. D. Kulkarni** *A Comparative Study of Annealing Methods for Batch Scheduling Problems* Chemical Engineering Research and Design (2001) 79, S. 673-683

- [DKV99] **Dessouky, M., Kijowski B. und S. Verma.** *Simultaneous batching and scheduling for chemical processing with earliness and tardiness penalties.* Production and Operations Management (1999) 8, S. 433 - 444.
- [DoS03] **Domschke, W. und A. Scholl** *Grundlagen der Betriebswirtschaftslehre.* Heidelberg: Springer Verlag, 2003, 2. Aufl.
- [Dor90] **Dorninger, C.** *Produktionsplanung und -steuerung: Konzepte, Methoden und Kritik.* Wien: Ueberreuter Verlag, 1990
- [DPR01] **Dangelmaier, W.; Pape, U. und M. Rüther (Hrsg.)** *Die Supply Chain in Zeitalter von E-Business und Global Sourcing* Paderbon: ALB-HNI-Verlagsschriftenreihe, 2001
- [DSV93] **Domschke, W.; Scholl, A. und S. Voß** *Produktionsplanung: Ablauforganisatorische Aspekte.* Heidelberg: Springer Verlag, 1993, 1. Aufl.
- [Due93] **Dueck, G.** *New Optimization Heuristics: The Great Deluge Algorithm and the Record-to-Record Travel.* Journal of Computational Physics (1993) 104:1, S. 86-92
- [DuS90] **Dueck, G. und T. Scheuer** *Threshold accepting: a general purpose optimization algorithm.* Journal of Computational Physics (1990) 90:1, S. 161-175
- [Dyc94] **Dyckhoff, H.** *Betriebliche Produktion - Theoretische Grundlagen einer umweltorientierten Produktionswirtschaft.* Heidelberg: Springer Verlag, 1994, 2. Aufl.
- [ER86] **Egli, U.M. und D.W.T. Rippin** *Short-term scheduling for multiproduct batch chemical plants.* Computers & Chemical Engineering (1986) 10, S. 303-325
- [FH97] **Fortemps, P. und M. Hapke** *On the disjunctive graph for projekt scheduling.* Foundations of Computing and Decision Sciences (1997) 22:3, S. 195-209
- [Fri03] **Friedrich, S.** *Stand der IT-Unterstützung für das Supply Chain Management in der Prozessindustrie.* in [SVK03], S. 145-160
- [FGK90] **Fourer, F.; Gay, D.M. und B.W. Kernighan** *A Modeling Language for Mathematical Programming.* Management Science (1990) 36, S. 519-554
- [FG02] **Fourer, F. und D.M. Gay** *Extending an Algebraic Modeling Language to Support Constraint Programming.* INFORMS Journal on Computing (2002) 14:4, S. 322-344
- [FV02] **Fink, A. und S. Voß** *HOTFRAME: A Heuristic Optimization Framework.* in [VW02], S. 81-154
- [Ge95] **Georgi, G.** *Job Shop Scheduling in der Produktion: einsatzorientierte Lösungen für ein Kernproblem der Produktionsplanung bei mittleren Auftragszahlen und variierenden Einsatzbedingungen.* Heidelberg: Physica-Verlag, 1995

- [GL+79] **Graham, R.L.; Lwaler, E.L.; Lenstra, J.K. und A.H.G. Rinnoy Kan** *Optimization and approximation in deterministic sequencing and scheduling: a survey.*
Annals of Discrete Mathematics (1979) 5, S. 287-326
- [GNZ86] **Grabowski, J.; Nowicki, E. und S. Zdrzalka** *A block approach for single machine scheduling with release dates und due dates.*
European Journal of Operational Research (1986) 26, S. 278-285
- [Gro04] **Gronau, N.** *Enterprise Resource Planning und Supply Chain Management - Architektur und Funktionen.*
München: Oldenbourg Verlag, 2004
- [GT04] **Günther, H.-O. und H. Tempelmeier** *Produktion und Logistik.* Heidelberg: Springer Verlag, 2004, 6. Aufl.
- [GTr01] **Günther, H.O. und N. Trautmann** *Benchmark: Westenberger & Kallrath (1995).*
URL: http://www.wior.uni-karlsruhe.de/LS_Neumann/Forschung/WK95_html; erstellt am: 16.01.2002; abgerufen am: 16.08.2002
- [Gut83] **Gutenberg, E.** *Grundlagen der Betriebswirtschaftslehre - Band 1: Die Produktion.* Heidelberg: Springer Verlag, 1983
- [Haa94] **Haase, K.** *Lotsizing and Scheduling for Production Planning.* Heidelberg: Springer Verlag, 1994
- [Häs79] **Hässig, L.** *Graphentheoretische Methoden des Operations Research.* Stuttgart: Teubner Studienbücher, 1979
- [HBV02] **Hartmann, E.; Bäumler, M. und S.U. Vallerien** *Die Reserven liegen in der Produktion.* Chemie Produktion (2002) 10, S. 14-15
- [HD95] **Herroelen, W.S. und E.L. Demeulemeester** *Recent Advances in Branch-and-Bound Procedures for Resource-Constrained Project Scheduling Problems.* in [CC+95], S. 259-276
- [Hei99] **Heipcke, S.** *Comparing constraint programming and mathematical programming approaches to discrete optimisation - the change problem.*
Journal of the Operational Research Society (1999) 50, S. 581-595
- [Hei02] **Heipcke, S.** *Applications of optimization with Xpress-MP.* - Revised translation from the french language edition of "Programmation Lineaire" by C. Guéret, C. Prins and M. Sevaux, 2000; Paris: Editions Eyrolles und Dash Optimization Ltd., 2002
- [Hem99] **Hemming, W.** *Verfahrenstechnik.*
Würzburg: Vogel Buchverlag, 1999, 8. Auflage
- [Hen91] **Hennicke, L.** *Wissenbasierte Erweiterung der Netzplantechnik.*
Heidelberg: Physica-Verlag, 1991
- [HS86] **Heinrich, C. und Ch. Schneeweiss** *Multi-Stage Lot-Sizing for General Production Systems.* in [ASS86]

- [Hol99] **Holzer, A.J.** *Prioritätsregelbasierte Ressourcenplanung für Projekte mit komplexer Ablaufstruktur*. Universität Passau: Dissertation, 1999
- [ILOG03] **ILOG** *ILOG OPL Studio 3.7 Language Manual*. September 2003; URL: http://ace.acadiau.ca/math/ACMMaC/private_docs/opl_studio/pdf/languageman.pdf; abgerufen am: 20.10.2005
- [ILOG05] **ILOG** *Customers*. URL: <http://www.ilog.com/success>; abgerufen am: 20.10.2005
- [Jän03] **Jänicke, W.** *Add-ons zu SAP APO PP/DS*. PPS Management (2003) 8, S. 26-28
- [JMM90] **Jehle, E.; Müller, K. und H. Michael** *Produktionswirtschaft*. Heidelberg; Verlag Recht und Wirtschaft, 1990
- [JR02] **Jänicke, W. und E. Roth** *Konkurrierende Fertigungsaufträge bei der betriebsübergreifenden Planung*. PPS Management (2002) 2, S. 55-57
- [JU98] **Jacob, O. und H.-J. Uhink**. *SAP R/3 im Mittelstand*. Braunschweig: vieweg, 1998
- [Joh54] **Johnson, S.M.** *Optimal Two- and Three-Stage Production Schedules with Setup Times Included*. Naval Research Logistics Quarterly (1954) 1, S. 61-68
- [KGV82] **Kirkpatrick, S.; Gelatt, C.D. und M.P. Vecchi** *Optimization by Simulated Annealing*. IBM Research Report RC 9355
- [KGV83] **Kirkpatrick, S.; Gelatt, C.D. und M.P. Vecchi** *Optimization by Simulated Annealing*. Science (1983) 220, S. 671-680
- [KH95] **Karimi, I.A. und S. Hasebe** *Chemical batch process scheduling*. in [Kv95], S. 181-203
- [Kie99] **Kießwetter, M.** *Ablaufplanung in der chemischen Industrie - Optimierung mit Evolutionären Algorithmen*. Wiesbaden: Gabler Verlag und DUV, 1999
- [Kle00] **Klein, R.** *Constraint Programming*. in [SK00], S. 351-359
- [KNS00] **Krauth, J.; Noche, B. und A. Schalla**. *Wenn die Leistung nicht reicht - Simulation verbessert Produktionsplanung und -steuerung*. Der Betriebsleiter (2000) 4, S. 50-51
- [KPS93] **Kondili, E.; Pantelides, C.C. und R.W.H. Sargent**. *A General Algorithm for Short-Term Scheduling of Batch Operations - 1. MILP Formulation*. Computers & Chemical Engineering (1993) 17, S. 211-227
- [Kr02] **Kallrath, J.** *Planning and scheduling in the process industry*. OR Spectrum (2002) 24, S. 219-250

- [KSW96] **Kern, W.; Schröder, H.-H. und J. Weber (Hrsg.)** *Handwörterbuch der Produktionswirtschaft - Enzyklopädie der Betriebswirtschaftslehre, Band VII.* Stuttgart: Schäfer-Poeschel Verlag, 1996, 2. Aufl.
- [KT+99] **Käschel, J.; Teich, T.; Köbernig, G. und B. Meier** *Algorithms for the Job Shop Scheduling Problem - a comparison of different methods* Vortrag: European Symposium on Intelligent Techniques, Kreta (Griechenland), 3. Juni 1999
URL: http://www.erudit.de/erudit/events/esit99/12553_P.pdf
abgerufen am: 20.10.2005
- [Kv95] **Kalivas, J.H.** *Adaption of simulated annealing to chemical optimization problems - Data Handling in Science and Technology, Vol. 15.* Amsterdam: Elsevier, 1995
- [KW03] **Kipp, R. und H.-H. Wiendahl** *MES Lösungen - Kriterien bei der Systemauswahl.* IT & Production (2003) 4, S. 14-16
- [LA92] **van Laarhoven, P.J.M. und E.H.L. Aarts** *Simulated Annealing - Theory and Applications.* Boston: Kluwer Academic Publishers, 1992, 3. Auflage
- [Lin05] **Lindemann, M.** *Manufacturing Execution Systems.* PPS Management (2005) 3, S. 12
- [Lö04] **Löhl, T.** *Modellbasierte Feinplanung von Chargenprozessen mit genetischen Algorithmen.* Aachen: Shaker Verlag, 2004
- [Loo99] **Loos, P.** *Konzeption und Umsetzung einer Systemarchitektur für die Produktionssteuerung in der Prozeßindustrie.* in [Wen99], S. 385-418
- [Loo01] **Loos, P.** *Gozintographs for By-Products and Cyclic Production: An Approach for ERP System Application.* in [SS01], S. 1111-1117
- [LP01] **Lustig, I.J. und J.-F. Puget** *Program \neq Program: Constraint Programming and its Relationship to Mathematical Programming.* Interfaces (2001) 31, S. 29-53
- [LS05] **Lindemann, M. und S. Schmid** *Marktüberblick: Manufacturing Execution Systems.* PPS Management (2005) 3, S. 55-67
- [LVP02] **Lee, D.S; Vassiliadis, V.S. und J.M. Park** *List-Based Threshold-Accepting Algorithm for Zero-Wait Scheduling of Multiproduct Batch Plants.* Industrial & Engineering Chemistry Research (2002) 41, S. 6579-6588
- [Ma96] **Mayr, M.** *Hierarchische Produktionsplanung mit zyklischen Auflagenmustern.* Regensburg: S. Roederer Verlag, 1996
- [MC+05] **Méndez, C.A.; Cerdá, J; Grossmann I.E.; Harjunkski, I. und M. Fahl** *State-of-the-art Review of Optimization Methods for short-term Scheduling of Batch Processes.* URL: <http://egon.cheme.cmu.edu/Papers/BatchReviewMendezGrossmann.pdf>; abgerufen am: 17.10.2005

- [Me99] **Meyr, H.** *Simultane Losgrößen- und Reihenfolgeplanung für kontinuierliche Produktionslinien.* Wiesbaden: DUV, 1999
- [MG05] **Mediavilla, V. und E. Gerdes** *Welchen Nutzen bieten ME-Systeme.* PPS Management (2005) 3; S. 17-20
- [MH+01] **Meylan, P.; Hildebrand, K.; Jänicke, W. und F. Staudtmeister** *Etablierung eines Supply Chain Coordinators bei der EMS-Dottikon AG in Dottikon.* CITplus, Chemie Ingenieur Technik (2001) 4, S. 48-51
- [MP02] **Mascis, A. und D. Pacciarelli** *Job-shop scheduling with blocking and no-wait restrictions* European Journal of Operational Research (2002) 143, S. 498 - 517
- [MRW00] **Meyr, H.; Rohde, J. und M. Wagner** *Architecture of Seletec APS.* in [SK00], S. 241-249
- [MR+53] **Metropolis, N.; Rosenbluth, A.; Rosenbluth, M.; Teller, A. und E. Teller** *Equation of State Calculations by Fast Computing Machines.* Journal of Chemical Physics (1953) 21, S. 1087-1159
- [NAM93] **NAMUR NE33** - *Anforderungen an Systeme zur Rezeptfahrweise.* Leverkusen: NAMUR Geschäftsstelle c/o Bayer AG, 1993
- [NAM94] **NAMUR NA46** - *Beispiel zur Rezeptfahrweise nach NAMUR-Empfehlung NA33* Leverkusen: NAMUR Geschäftsstelle c/o Bayer AG, 1993
- [NAM99] **NAMUR NA85** - *Feinplanung.* Leverkusen: NAMUR Geschäftsstelle c/o Bayer AG, 1999
- [Nis97] **Nissen, V.** *Einführung in Evolutionäre Algorithmen.* Wiesbaden: vieweg Verlag, 1997
- [NS05] **Nowicki, E. und C. Smutnicki** *An Advanced Tabu Search Algorithm for the Job Shop Problem.* Journal of Scheduling (2005) 8, S. 145-159
- [NW88] **Nemhauser, G.L. und L.A. Wolsey** *Integer and Combinatorial Optimization.* New York: Wiley Interscience Publication, 1988
- [OC94] **Osman, I.H. und N. Christofides** *Capacitated clustering problems by hybrid simulated annealing and tabu search.* International Transactions in Operational Research (1994) 1, S. 317-336
- [OK96a] **Osman, I.H. und J.P. Kelly (Hrsg.)** *Meta-Heuristics: Theory & Applications.* Boston: Kluwer Academic Publishers, 1996
- [OK96b] **Osman, I.H. und J.P. Kelly** *Meta-Heuristics: An Overview.* in [OK96a], S. 1-22
- [Pan94] **Pantelides, C.C.** *Unified Frameworks for optimal process planning and scheduling.* in [RH94], S. 253-274
- [Pin95] **Pinson, E.** *The Job Shop Scheduling Problem: A Concise Survey and Recent Developments.* in [CC+95], S. 277-293

- [PSP94] **Papageorgiou, L.G.; Shah, N. und C.C. Pantelides** *Optimal Scheduling of Heat-Integrated Multipurpose Plants*. Industrial & Engineering Chemistry Research (1994) 33, S. 3168
- [PP96] **Papageorgiou, L.G. und C.C. Pantelides** *Optimal Campaign Planning/Scheduling of Multipurpose Batch/Semicontinuous Plants - 1. Mathematical Formulation*. Industrial & Engineering Chemistry Research (1996) 35, S. 488-509
- [Pre96] **Preßmar, D.B.** *Stücklisten und Rezepturen*. in [KSW96], Sp. 1923-1930
- [Pu04] **Puchta, M.** *Optimierung von Problemstellungen aus der diskreten und der Prozess-Industrie unter Verwendung physikalischer Verfahren*. Regensburg: Dissertation, 2004;
URN: urn:nbn:de:bvb:355-opus-4382
- [RaH00] **Raaymakers, W.H.M. und J.A. Hoogeveen** *Scheduling multipurpose batch process industries with no-wait restrictions by simulated annealing* European Journal of Operational Research (2000) 126, S. 131 - 151
- [RH94] **Rippin, D.W.T. und J. Hale (Hrsg.)**. *Proceedings of the 2nd International Conference on Foundations of Computer-Aided Process Operations*. Austin (Texas), USA: Cache Publications, 1994
- [RS00] **Richter, M. und V. Stockrahm**. *Scheduling of Synthetic Granulate*. in [SK00], S. 251-265
- [Sch+00] **Schrumpf, G.; Schneider, J.; Stamm-Wilbrandt, H. und G. Dueck** *Record Breaking Optimization Results Using the Ruin and Recreate Principle*. Journal of Computational Physics (2000) 159:2, S. 139-171
- [Schn99] **Schneider, J.J.** *Effiziente parallelisierbare physikalische Optimierungsverfahren*. Regensburg: Dissertation, 1999;
URL: <http://homepages.uni-regensburg.de/~scj04369/diss.ps.gz>
abgerufen am: 25.10.2005
- [Schu03] **Schuster, J.C.** *No-wait Job-Shop-Scheduling - Komplexität und Local Search*. Duisburg: Dissertation, 2003;
URN: urn:nbn:de:hbz:464-duett-04242003-1316182
- [Schz02] **Schulz, C.** *Modellierung und Optimierung einer verfahrenstechnischen Mehrproduktanlage*. Aachen: Shaker Verlag, 2001
- [SFR02] **Schultmann, F.; Fröhling, M. und O. Rentz** *Simulationsgestützte Auftragsreihenfolgeplanung auf Basis von Prioritätsregeln*. PPS Management (2002) 4, S. 21-24
- [SHF94] **Schöneburg, E.; Heinzmann, F. und S. Feddersen** *Genetische Algorithmen und Evolutionsstrategien - Ein Einführung in Theorie und Praxis der simulierten Evolution*. Bonn: Addison-Wesley, 1994

- [SK00] **Stadtler, H. und Ch. Kilger (Hrsg.)** *Supply Chain Management and Advanced Planning - Concepts, Models, Software and Case Studies*. Heidelberg: Springer Verlag, 2000
- [Ski01] **Skiscim, C.** *Software Review OPL Studio 3.1* OR/MS Today (2001) April
- [SNT96] **Sadeh, N.M.; Nakakuki, Y. und S.R. Thangiah** *Learning to Recognize (Un)Promising Simulated Annealing Runs: Efficient Search Procedures for Job Shop Scheduling and Vehicle Routing*.
in [OK96a], S. 277-297
- [SS01] **Strong, D. und D. Straub (Hrsg.)** *Proceedings of the Seventh Americas Conference on Information Systems*.
AMCIS 2001, Boston, MA, August 3-5, 2001
- [Sta00a] **Stadtler, H.** *Production Planning and Scheduling*.
in [SK00], S. 147-165
- [Sta00b] **Stadtler, H.** *Supply Chain Management - An Overview*.
in [SK00], S. 7-28
- [Sta01] **Stadtler, H.** *Losgrößenplanung - Von der Grundlagenforschung zur Anwendung im Supply Chain Management*.
Vortrag: Tagung der AG Produktionsplanung der Gesellschaft für Operations Research (GOR) e.V., St. Leon-Rot, 11. Januar 2001
URL: http://130.149.112.124/PM/inhalt/de/forschung/scm-ag/2001_01_11/Stadtler.pdf; abgerufen am: : 24.10.2002
- [Sth05] **Stahl, M.** *Manufacturing Execution System: Korridor zwischen Fertigung und Verwaltung*. PPS Management (2005) 3, S. 29-31
- [Sto02] **Stockrahm, V.** *Korrespondenz zu [RS00]*. 06.11.2002
- [SVK03] **Spengler, Th.; Voß, S. und H. Kopfer** *Logistik Management - Prozesse, Systeme, Ausbildung*. Heidelberg: Physica Verlag, 2003
- [Tem01] **Tempelmeier, H.** *Supply Chain Planning with Advanced Planning Systems*.
Vortrag 3rd Aegean International Conference on „Design and Analysis of Manufacturing Systems“, Tinos Island (Griechenland), 19.-22. Mai 2001
URL: <http://www.pom-consult.de/docs/TempelmeierTinos2001.pdf>; abgerufen am: 20.10.2005
- [Tra01] **Trautmann, N.** *Anlagenbelegungsplanung in der Prozessindustrie* Wiesbaden: DUV, 2001
- [Tra05] **Trautmann, N.** *Operative Planung der Chargenproduktion* Wiesbaden: DUV, 2005
- [VCI98] **VCI** *Konzept für die Zusammenlagerung von Chemikalien - Leitfaden*. Frankfurt, 1998
- [VCI03] **VCI** *Chemische Industrie im Überblick*. Frankfurt, 2003

- [Vah00] **Vahrenkamp, R.** *Produktionsmanagement*. München: Oldenbourg Wissenschaftsverlag, 2000, 4. Auflage.
- [Vah03] **Vahrenkamp, R.** *Quantitative Logistik für das Supply Chain Management*. München: Oldenbourg Wissenschaftsverlag, 2003.
- [VAL96] **Vaessens, R.J.M. ; Aarts, E.H.L. und J.K. Lenstra** *Job shop scheduling by local search*. INFORMS Journal on Computing (1996) 8, S. 302-317
- [VM+99] **Van Hentenryck, P.; Michel, L.; Perron, L. und J.C. Régim** *Constraint Programming in OPL*. Vortrag auf der Konferenz *PLI 99 - Principles, Logics, and Implementations of high-level programming languages*, Paris, Frankreich, 27.09 - 01.10. 1999
URL: <http://www.cs.brown.edu/people/pvh/ppdp99.ps>;
abgerufen am: 20.10.2005
- [Vog98] **Vogt, M.** *Tourenplanung in Ballungsgebieten - Entwicklung eines PC-gestützten Verfahrens*. Wiesbaden: DUV, 1998.
- [VW02] **Voß, S. und D.L. Woodruff** *Optimization Software Class Libraries*. Boston: Kluwer Academic Publishers, 2002
- [Wen99] **Wenzel, P. (Hrsg.)** *Business Computing with SAP R/3*. Wiesbaden: Gabler Verlag, 1999
- [Wi01] **Witt, A.** *Maschinenbelegungsplanung im Rahmen des Supply Chain Management eines Unternehmens der Stahlindustrie - Problemdefinition und Lösungsansätze*. in [DPR01], S. 339-357
- [WK94] **Westenberger, H. und J. Kallrath** *Formulation of a Job Shop Problem in Process Industry*. Arbeitspapier der Bayer AG und der BASF AG, veröffentlicht in [Kr02], Appendix, S. 239-250
- [WMK05] **Wiendahl, H.-H.; Mussbach-Winter, U. und R. Kipp** *ERP und MES - Widerspruch oder sinnvolle Ergänzung*. PPS Management (2005) 3, S. 13-16
- [WN03] **Weidenhaun, J. und F. Niehsen** *ERP-Lösungen für die Nahrungsmittel und Prozessindustrie*. PPS Management (2003) 4, S. 36-39
- [Wö90] **Wöhe, G.** *Einführung in die Allgemeine Betriebswirtschaftslehre*. München: Verlag Vahlen, 1990, 17. Auflage
- [Zäp82] **Zäpfel, G.** *Produktionswirtschaft - Operatives Produktions-Management*. Berlin: de Gruyter, 1982.
- [ZE+98] **Zentner, M.G; Elkamel, A; Pekny, J.F und G.V. Reklaitis** *A language for describing process scheduling problems*. Computers & Chemical Engineering (1998) 22, S. 125-145.