

Kostenoptimale Kapazitätsabstimmung in einer getakteten Variantenfließlinie

- unter expliziter Berücksichtigung des Unterstützereinsatzes und
unterschiedlicher Planungszeiträume -

Dissertation

zur Erlangung der Würde eines
DOKTORS DER WIRTSCHAFTSWISSENSCHAFTEN
(Dr. rer. pol.)
der Universität Paderborn

vorgelegt von:

Dipl.-Wirt.-Inf Simon Altemeier
Heinz-Kemper-Weg 4
33104 Paderborn

Paderborn, im Mai 2009

Erstprüfer: Prof. Dr.-Ing. habil. Wilhelm Dangelmaier
Zweitprüferin: Prof. Dr. Leena Suhl

Vorwort

Die vorliegende Arbeit entstand im Rahmen einer Kooperation zwischen dem Heinz Nixdorf Institut der Universität Paderborn, dem Fraunhofer Anwendungszentrum für Logistikorientierte Betriebswirtschaft Paderborn, der International Graduate School Paderborn und dem Forschungszentrum der Daimler AG in Ulm.

Ich möchte mich bei meinem Doktorvater Prof. Dr.-Ing. habil. Wilhelm Dangelmaier für die wissenschaftliche Betreuung, die andauernde Bereitschaft zur Diskussion und den vielseitigen fachlichen Rat bedanken. Er hat es mir ermöglicht, meine Promotion in einem Industrieunternehmen durchzuführen, so dass ich nicht nur rein wissenschaftlich forschen, sondern die entwickelten Methoden auch in der Praxis zur Anwendung bringen konnte. Bei Frau Prof. Dr. Leena Suhl bedanke ich mich für die Möglichkeiten zur fachlichen Diskussion während der Promotionszeit und die Übernahme des Zweitgutachtens.

Der International Graduate School for Dynamic Intelligent Systems Paderborn und ihrem Leiter Herrn Prof. Dr. Eckhard Steffen danke ich für die Förderung durch ein Stipendium. Die Möglichkeit zum wissenschaftlichen und persönlichen Austausch mit anderen Stipendiaten in zahlreichen Veranstaltungen hat die Promotionszeit für mich sehr bereichert.

Ich bedanke mich bei dem Leiter des Forschungsbereiches Vehicle Manufacturing Herrn Prof. Dr. Heinrich Flegel sowie Herrn Dr. Stefan Kienzle, meinem Abteilungsleiter Herrn Dr. Gerhard Jünemann, meinen Teamleitern Thomas Sommer-Dittrich und Dr. Stephan Bürkner sowie meinem Teilprojektleiter Dr. Marcus Ziegler für die Möglichkeit, mein Wissen in unterschiedlichen praxisnahen Forschungsprojekten ausbauen zu können und daneben den nötigen Freiraum zur Anfertigung dieser Arbeit bekommen zu haben. Danken möchte ich auch den zahlreichen Ansprechpartnern und Planern in den Werken Sindelfingen, Rastatt und Bremen, die mir die Möglichkeit gegeben haben, direkte Praxiserfahrung zu sammeln und die entwickelten Methoden in der Praxis zu gewinnbringenden Lösungen voranzutreiben.

Bei den Kollegen im Team Produktionsmanagement am Daimler Forschungszentrum Ulm und der Arbeitsgruppe Wirtschaftsinformatik, insb. CIM, bedanke ich mich für die stets vertrauensvolle Atmosphäre und die interessanten Gespräche, die mir noch lange in Erinnerung bleiben werden. Besonderer Dank gilt meinen direkten Weggefährten in der Paderborner Doktorandengruppe Jens Peter Kempkes, Jan-Erik Gans und Thomas Sillekens die in der Zeit sowohl Kollegen waren als auch Freunde geworden sind. Wir haben eine tolle Zeit miteinander verbracht.

Dank sagen möchte ich auch den zahlreichen Studenten, mit deren Engagement eine Umsetzung der Ideen erst möglich geworden ist.

Bei meinen Eltern möchte ich mich dafür bedanken, dass sie meine wissenschaftliche Ausbildung jederzeit unterstützt und mir den notwendigen Rückhalt gegeben haben.

Mein besonderer Dank gilt meiner Freundin Katharina, die viel Verständnis für die entgangenen gemeinsamen Stunden aufgebracht und mir in schwierigen Phasen meiner Promotion mit bestärkenden Worten zur Seite gestanden hat. Ohne sie wäre all dies nicht möglich gewesen.

Simon Altemeier

für Katharina

Inhaltsverzeichnis

Abbildungsverzeichnis	vi
Tabellenverzeichnis	ix
Symbolverzeichnis	xi
1 Einleitung	1
2 Problemdefinition	5
2.1 Systembeschreibung	5
2.1.1 Aufbau einer getakteten Variantenfließlinie	5
2.1.2 Produktionsablauf in einer getakteten Variantenfließlinie	8
2.1.2.1 Belastungsschwankungen durch Variantenvielfalt	9
2.1.2.2 Kompensation von Belastungsschwankungen	11
2.1.3 Kostenstruktur in einer getakteten Variantenfließlinie	16
2.1.3.1 Bestimmung der Betriebskosten	17
2.1.3.2 Bestimmung der Rekonfigurationskosten	23
2.2 Systematisierung der Aufgaben zur Kapazitätsabstimmung in einer getakteten Variantenfließlinie	24
2.2.1 Rekonfiguration einer getakteten Variantenfließlinie	27
2.2.1.1 Ziel der Rekonfiguration	28
2.2.1.2 Restriktionen der Rekonfiguration	30
2.2.1.3 Voraussetzungen der Rekonfiguration	32
2.2.2 Vorranggraphenentwicklung für variantenreiche Produkte	33
2.2.2.1 Ziel der Vorranggraphenentwicklung	35
2.2.2.2 Restriktionen der Vorranggraphenentwicklung	36
2.2.2.3 Voraussetzungen der Vorranggraphenentwicklung	37
2.2.3 Reihenfolgeplanung für eine getaktete Variantenfließlinie	39
2.2.3.1 Ziel der Reihenfolgeplanung	39
2.2.3.2 Restriktionen der Reihenfolgeplanung	40
2.2.3.3 Voraussetzungen der Reihenfolgeplanung	41
2.3 Anforderungen an die Problemlösung	43
2.3.1 Anforderungen an ein Verfahren zur kostenoptimalen Rekonfiguration	43
2.3.2 Anforderungen an ein Verfahren zur Vorranggraphenerstellung für variantenreiche Produkte	44
2.3.3 Anforderungen an ein Verfahren zur Reihenfolgeplanung zur Minimierung des Unterstützereinsatzes	45

3	Stand der Forschung	47
3.1	Stand der Forschung zur Rekonfiguration von Fließlinien	47
3.1.1	Ansätze zur Rekonfiguration	47
3.1.2	Übertragbare Ansätze aus der Erstinstallation von Fließlinien	50
3.1.3	Fazit	54
3.2	Stand der Forschung zur Erstellung von Vorranggraphen	55
3.2.1	Manuelle Methoden zur Erstellung von Vorranggraphen	55
3.2.2	Automatisierte und hybride Methoden zur Erstellung von Vorranggraphen	56
3.2.2.1	Geometriedatenorientierte Ansätze	57
3.2.2.2	Stücklistenorientierte Ansätze	57
3.2.2.3	Ansätze unter Verwendung erweiterter Produktdokumentation	59
3.2.3	Fazit	59
3.3	Stand der Forschung zur Reihenfolgeplanung	61
3.3.1	Ansätze zum Mixed-Model Sequencing	62
3.3.1.1	Klassifikationskriterien	62
3.3.1.2	Einordnung existierender Ansätze	64
3.3.2	Fazit	66
4	Zu leistende Arbeiten	67
5	Kostenoptimale Kapazitätsabstimmung in einer getakteten Variantenfließlinie	69
5.1	Kostenoptimale Rekonfiguration	69
5.1.1	Formalisierung des Rekonfigurationsproblems	69
5.1.1.1	Entwicklung eines antizipativen Zielkriteriums	71
5.1.1.2	Formulierung des Modells	75
5.1.2	Vorgehensmodell zur Rekonfiguration	77
5.1.3	Entscheidungsunterstützungssystem zur kostenoptimalen Rekonfiguration	78
5.1.3.1	Analyse-Phase	79
5.1.3.2	Design-Phase	85
5.1.3.3	Auswahl-Entscheidungsphase	97
5.1.3.4	Implementationsphase	100
5.2	Teilautomatisierte Erstellung eines Variantenvorranggraphen	101
5.2.1	Vorgehensmodell zur Vorranggraphenerstellung	101
5.2.2	Datengrundlage	102
5.2.2.1	Produktdokumentation mit Hilfe der offenen Variantenstückliste	103
5.2.2.2	Aufbereitung der Coderegeln	105
5.2.2.3	Reihenfolge der Bearbeitung von Arbeitsvorgängen	107
5.2.3	Automatische Generierung von Freiheitsgraden	108
5.2.3.1	Strukturierung nach Werkstückansprache	108
5.2.3.2	Freiheitsgrade auf Basis der Coderegel	109
5.2.3.3	Integration unterschiedlicher gültiger Zuteilungen	113
5.2.4	Ergänzende manuelle Feinstrukturierung des Graphen	118

5.3	Belastungsorientierte Reihenfolgeplanung zur Reduzierung des Unterstützereinsatzes	120
5.3.1	Formalisierung der Reihenfolgeplanung	120
5.3.1.1	Reihenfolgeabhängige Positionsbestimmung der Werker	123
5.3.1.2	Modellierung von Abhängigkeiten zwischen Werkern	124
5.3.1.3	Abbildung der Unterstützereinsätze	127
5.3.2	Vorgehensmodell zur Reihenfolgeplanung	136
5.3.3	Konzept zur Fließliniensimulation	136
5.3.3.1	Ermittlung der Eingangsdaten für die Bewertung	137
5.3.3.2	Abbildung des rollierenden Bewertungshorizontes	139
5.3.3.3	Verfahren zur Ermittlung der Werkerpositionen	141
5.3.3.4	Auswertung der Ergebnisse	145
5.3.4	Zweistufiger Optimierungsansatz zur Reihenfolgeplanung	146
5.3.4.1	Initialheuristik	146
5.3.4.2	Iteratives Verbesserungsverfahren	151
6	Validierung und Ergebnisse	159
6.1	Validierung der Entscheidungsunterstützung zur Rekonfiguration	159
6.1.1	Testszenario	159
6.1.2	Ergebnisse	160
6.1.2.1	Kennzahlensystem zur Auswahl des Betrachtungsbereiches	160
6.1.2.2	Anwendungsbeispiele	161
6.2	Validierung der teilautomatisierten Vorranggraphenentwicklung	166
6.2.1	Testszenario	166
6.2.2	Ergebnisse	166
6.2.2.1	Bestimmung der erreichten Komplexitätsreduktion	166
6.2.2.2	Verwendung des Vorranggraphen innerhalb einer Rekonfigurationsmethode	168
6.3	Validierung der belastungsorientierten Reihenfolgeplanung zur Reduzierung des Unterstützereinsatzes	170
6.3.1	Testszenario	170
6.3.2	Ergebnisse	171
6.3.2.1	Einfluss der Abhängigkeit zwischen Werkern	171
6.3.2.2	Einfluss der Organisationsform von Unterstützereinsätzen	173
6.3.2.3	Parametrisierung der Clusteringmethode	173
6.3.2.4	Anwendungsbeispiele	178
7	Zusammenfassung und Ausblick	181
	Literaturverzeichnis	185
A	Anhang	199
A.1	Ergebnisse Rekonfiguration	199
A.2	Ergebnisse Vorranggraphenbildung	202
A.3	Ergebnisse Reihenfolgeoptimierung	205

Abbildungsverzeichnis

2.1	Beispiel für den Aufbau einer getakteten Variantenfließlinie in der automobilen Endmontage	7
2.2	Beispiel für die Ausrichtung an einer Durchschnittsvariante	10
2.3	Beispiel für den Effizienzverlust bei Ausrichtung an der Maximalvariante . . .	11
2.4	Variable Bearbeitungsbereiche durch die Einführung von Driftbereichen . . .	12
2.5	Beispiel für die reihenfolgeabhängige Position der Werker	13
2.6	Beispiel für den Verzug durch Abhängigkeiten zwischen Werkern	14
2.7	Beispiel für einen reaktiven Unterstützereinsatz	15
2.8	Beispiel für einen präventiven Unterstützereinsatz	15
2.9	Darstellung von gleichzeitigen Situationen im Fließsystem	19
2.10	Beispiel für die Belegung von Unterstützern über mehrere Takte hinweg . . .	20
2.11	Verschiebung des Kapazitätsbedarfs von Werkern auf Unterstützer	22
2.12	Schematisches Beispiel für die Verfügbarkeit von Auftragsdaten in der Automobilindustrie	26
2.13	Zielkonflikt der Rekonfiguration	28
2.14	Beispiel für die Zuordnung von Arbeitsvorgängen auf Stationen unter Beachtung der Vorrangbeziehungen	31
2.15	Beispiel für eine Rekonfiguration unter Beachtung der Vorrangbeziehungen . .	32
2.16	Verbotene Konstruktionen in Vorranggraphen	34
2.17	Rollierender Aspekt der Bewertung einer Reihenfolgeplanung	41
3.1	Vergleich der internen und der externen Auslastungsunterschiede bei einer Variantenproduktion	52
3.2	Vorgehen von Ammer zu Vorranggraphenbildung	58
5.1	Darstellung von gleichzeitigen Situationen im Fließsystem	73
5.2	Eingliederung der Verfahren in Phasen eines Entscheidungsunterstützungssystems	79
5.3	Beispiel für einen Komprimierungsvorgang im Rahmen der Rekonfiguration . .	82
5.4	Beispiel für die Visualisierung der Bearbeitungszeiten (exakt und geclustert) .	88
5.5	What-If Analyse mit Hilfe von Histogrammen	89
5.6	Beispiel für korrelierende Arbeitsvorgänge	91
5.7	Varianzreduktion durch Trennung korrelierender Arbeitsvorgänge	92
5.8	Identifikation einer Verschiebung mit Hilfe der AIA-Kennzahl	94
5.9	Beispiel für den Suchbaum der Heuristik nach einer Verschiebung	97
5.10	Kompletter Suchbaum der Heuristik	97
5.11	Versionierungssystematik im Alternativenbaum	98

5.12	Vergleich des Top-Down und Bottom-Up Ansatzes zur Vorranggraphenentwicklung	102
5.13	Vorgehen zu Vorranggraphenbildung	103
5.14	Beispielhafte Struktur einer offenen Variantenstückliste	104
5.15	Starrer Vorranggraph als Startpunkt	108
5.16	Beispiel für die Trennung in Teilgraphen mit Hilfe von unabhängigen Werkstückansprachen	109
5.17	Beispiel für den gegenseitigen Ausschluss zweier Coderegeln	110
5.18	Vorgehen zur Trennung nach sich ausschließenden XOR-Zweigen	111
5.19	Beispiel für die Trennung in sich ausschließende XOR-Zweige	113
5.20	Entstehung von Zyklen in vereinigten Graphen	115
5.21	Vorgehen zur Beseitigung von Zyklen	115
5.22	Beispiel für Zyklen mit Schnittmenge	116
5.23	Beispiel für aufgelöste Zyklen mit Schnittmenge	117
5.24	Beispiel für Werkerabhängigkeiten	125
5.25	Beispiel für die Beeinflussung durch einen Vorgängerarbeitsplatz mit voranliegender Ansprechzone in anderer Station	126
5.26	Beispiel für die Beeinflussung durch einen Vorgängerarbeitsplatz mit gleicher oder dahinter liegender Ansprechzone in anderer Station	126
5.27	Beispiel für die Beeinflussung durch einen Vorgängerarbeitsplatz in der gleichen Station	127
5.28	Beispiel für einen reaktiven Unterstützereinsatz	128
5.29	Beispiel für einen präventiven Unterstützereinsatz	131
5.30	Vorgehen der zweistufigen Reihenfolgeheuristik	137
5.31	Darstellung des zu bewertenden Zeitraumes	140
5.32	Vorgehen der dreiteiligen Initialheuristik	147
5.33	Vermeidung der Ausnutzung später Reihenfolgepositionen durch doppelte Bewertungsequenz	149
5.34	Identifikation von zu vermeidenden Unterstützereinsätzen	153
5.35	Identifikation potentieller Tauschpartner	155
6.1	Zwei ausgewählte Arbeitsplätze zur Rekonfiguration	161
6.2	Zwei Arbeitsplätze nach einer Verschiebung in der Rekonfiguration	163
6.3	Zwei Arbeitsplätze nach der zweiten Verschiebung in der Rekonfiguration	164
6.4	Ergebnisse der Lösungsheuristik mit und ohne Vorranggraphen im Vergleich	170
6.5	Vergleich Abhängigkeiten im Best-Case und im Worst-Case	172
6.6	Entwicklung der Homogenität (Quadratsumme) bei unterschiedlicher Clusteranzahl	174
6.7	Entwicklung der Laufzeit in ms bei unterschiedlicher Clusteranzahl	175
6.8	Entwicklung der Laufzeit für die optimale Zentroidensequenzierung bei unterschiedlicher Clusteranzahl in sek	177
6.9	Vergleich der durchschnittlichen Kennzahlen für die Reihenfolgeheuristik	180
A.1	Beispiel für einen vollständigen Vorranggraphen mit 472 Arbeitsvorgängen	202
A.2	Vergrößerter Ausschnitt aus dem Beispiel-Vorranggraphen	203
A.3	Vergrößerter Ausschnitt aus dem Beispiel-Vorranggraphen	204

Tabellenverzeichnis

2.1	Notation zur Formalisierung des Aufbaus einer Variantenfließlinie	6
2.2	Erweiterung der Notation zur Formalisierung des Produktionsablaufes in einer Variantenfließlinie	8
2.3	Erweiterung der Notation zur Formalisierung der Kostenstruktur in einer Variantenfließlinie	17
2.4	Erweiterung der Notation zur Formalisierung der Rekonfiguration	27
2.5	Freiheitsgrade im Vergleich (Rekonfiguration und Erstinstallation)	30
2.6	Notation zur Definition des Vorranggraphen	36
2.7	Erweiterung der Notation zur Formalisierung der Reihenfolgeplanung	39
3.1	Notation zur Formalisierung bestehender Konzepte aus dem Stand der Forschung	51
3.2	Klassifikationstabelle von Mixed-Model-Sequencing Ansätzen zur Reihenfolgeoptimierung	65
5.1	Notation zur Formalisierung des Rekonfigurationsproblems	71
5.2	Notation Herleitung der Wahrscheinlichkeit für gleichzeitige Unterstützereinsätze	73
5.3	Notationserweiterung für die Analyse-Phase des ESS	80
5.4	Kennzahlensystem zur Auswahl des Betrachtungsbereiches für eine Rekonfiguration	84
5.5	Notationserweiterung für die Design-Phase des ESS	86
5.6	Notation zur AIA-Kennzahl	93
5.7	Notationserweiterung für die Auswahl- und Entscheidungs-Phase des ESS	98
5.8	Eigenschaften zur Beschreibung eines Auftrages (Beispiel)	103
5.9	Vergleich der kurzen und der langen Coderegeln	105
5.10	Darstellung der langen Coderegeln als Venn-Diagramm	105
5.11	Beispiel für Positionsvarianten und ihre Code-Regeln	106
5.12	Beispiel für die Bildung der disjunktiven Normalform	106
5.13	Beispiel für die Bildung von Unterpositionsvarianten	106
5.14	Beispiel für die Bildung von sich gegenseitig ausschließenden Positionsvarianten	107
5.15	Beispiel für die Integration der Unterpositionsvarianten	107
5.16	Erweiterung der Notation zur Definition von Freiheitsgraden auf Basis der Coderegeln	109
5.17	Beispiele für Baubarkeitsregeln	110
5.18	Beispiel für eine Bedingungsmatrix	117
5.19	Beispiel für eine aufgelöste Bedingungsmatrix	117
5.20	Notation zur Formalisierung der Reihenfolgeplanung	122
5.21	Notationserweiterung für die Fließlinienbewertung	137
5.22	Notationserweiterung für die Reihenfolgeoptimierung	146

6.1	Ergebnis der Korrelationsanalyse für das Kennzahlensystem zur Auswahl des Betrachtungsbereiches	161
6.2	Kennzahlen für die ausgewählten Arbeitsplätze vor der Bearbeitung	162
6.3	Kennzahlen für die beiden ausgewählten Arbeitsplätze nach der ersten Verschiebung	162
6.4	Ergebnisse des ersten Anwendungsbeispiels für die Rekonfiguration	164
6.5	Ergebnisse der Simulation des ersten Anwendungsbeispiels für die Rekonfiguration	164
6.6	Ergebnisse des zweiten Anwendungsbeispiels für die Rekonfiguration	165
6.7	Ergebnisse der Simulation des zweiten Anwendungsbeispiels für die Rekonfiguration	165
6.8	Verbleibende Knoten im Vorranggraphen je Werkstückansprache	167
6.9	Ergebnisse Reduktion des Betrachtungshorizontes durch XOR-Aufsplittungen	168
6.10	Vergleich der Simulation ohne Werkerabhängigkeiten (Best-Case) und mit maximalen Werkerabhängigkeiten (Worst-Case) bei reaktivem Unterstützereinsatz	171
6.11	Vergleich der Simulation ohne Werkerabhängigkeiten (Best-Case) und mit maximalen Werkerabhängigkeiten (Worst-Case) bei präventivem Unterstützereinsatz	172
6.12	Vergleich der Auswirkungen von reaktivem (A) und präventivem (B) Unterstützereinsatz	173
6.13	Durchschnittliche Homogenität der Cluster je Auftrag in Abhängigkeit von der Auswahl der Startzentroiden und der Reihenfolge der Zuordnung über alle Szenarien	174
6.14	Entwicklung der Homogenität in Abhängigkeit von der Clusteranzahl	175
6.15	Entwicklung der Laufzeit in Abhängigkeit von der Clusteranzahl	176
6.16	Entwicklung der Laufzeit für die optimale Zentroidensequenzierung bei unterschiedlicher Clusteranzahl je getesteter Instanz	176
6.17	Durchschnittliche Unterstützerkennzahlen je Clusteranzahl in der Initialreihenfolge	177
6.18	Ergebnisse für reaktiven Unterstützereinsatz (A) im Best-Case (R = Zufallsreihenfolge, CS = Car Sequencing, IH = Initialheuristik, IVV = Iteratives Verbesserungsverfahren)	178
6.19	Ergebnisse für reaktiven Unterstützereinsatz (A) im Worst-Case (R = Zufallsreihenfolge, CS = Car Sequencing, IH = Initialheuristik, IVV = Iteratives Verbesserungsverfahren)	179
6.20	Ergebnisse für präventiven Unterstützereinsatz (B) im Best-Case (R = Zufallsreihenfolge, CS = Car Sequencing, IH = Initialheuristik, IVV = Iteratives Verbesserungsverfahren)	179
6.21	Ergebnisse für präventiven Unterstützereinsatz (B) im Worst-Case (R = Zufallsreihenfolge, CS = Car Sequencing, IH = Initialheuristik, IVV = Iteratives Verbesserungsverfahren)	179
6.22	Durchschnittliche Verbesserung der Heuristik (IH+IVV) im Vergleich zur zufälligen Reihenfolge (R) und zum Car Sequencing (CS)	180
A.1	Unterstützereinsätze vor und nach der Rekonfiguration für 100 zufällige Reihenfolgen	201

A.2 Durchschnittliche Homogenität der Cluster je Auftrag in Abhängigkeit von der Startzentroidenauswahl und der Reihenfolge der Zuordnung	205
---	-----

Symbolverzeichnis

$\mathcal{A}$	Menge von Aufträgen a
a	Auftrag $a \in \mathcal{A}$
$\mathcal{AA}$	Menge von Arbeitsaufgaben aa , die jeweils aus einer Menge von Arbeitsvorgängen $avo \in \mathcal{AVO}$ bestehen
aa	Arbeitsaufgabe $aa \in \mathcal{AA}$ (Gruppierung von mehreren Arbeitsvorgängen $avo \in \mathcal{AVO}$)
$AI A_{avo,ap}$	Verhältnis zwischen der durchschnittlichen Bearbeitungszeit aller Werkstücke $p \in \mathcal{P}_{avo}$, an denen Arbeitsvorgang $avo \in \mathcal{AVO}$ verrichtet werden muss, und der Kapazität am Arbeitsplatz $ap \in \mathcal{AP}$ (Auswirkung in Arbeitsplatz-Kennzahl)
$\mathcal{AP}$	Menge von Arbeitsplätzen ap
ap	Arbeitsplatz $ap \in \mathcal{AP}$
$\mathcal{AVO}$	Menge von Arbeitsvorgängen avo
$\mathcal{AVO}_{wa}$	Menge von Arbeitsvorgängen $avo \in \mathcal{AVO}$, die Werkstückansprache $wa \in \mathcal{WA}$ zugeordnet sind
avo	Arbeitsvorgang $avo \in \mathcal{AVO}$
β	Parameter für die lokale Suchheuristik in der Rekonfiguration. Definiert die Anzahl an Lösungen die zur weiteren Untersuchung generiert werden sollen.
b_{wa}	Kennzahl für die Menge der verbleibenden, zu betrachtenden Knoten an Werkstückansprache $wa \in \mathcal{WA}$ im Verhältnis zur Gesamtanzahl Knoten im Vorranggraphen
$\mathcal{BEL}_{pz}$	Matrix mit den Bearbeitungszeiten der Werkstücke $p \in \mathcal{BWP}_{pz}$, die in dem Produktionszeitraum $pz \in \mathcal{PZ}$ bewertet werden
$bp_{a,avo}$	Binär: 1, wenn der Auftrag $a \in \mathcal{A}$ den Arbeitsvorgang $avo \in \mathcal{AVO}$ benötigt; 0, sonst (Bauplan)
br_{avo}	Baurate des Arbeitsvorganges $avo \in \mathcal{AVO}$. Gibt die Häufigkeit des Auftretens eines Arbeitsvorganges im Verhältnis zur Gesamtanzahl Aufträge an.
$\mathcal{BWP}_{pz}$	Menge von Werkstücken $p \in \mathcal{P}$, die zur Bewertung des Produktionszeitraumes $pz \in \mathcal{PZ}$ betrachtet werden müssen
$\mathcal{BWR}\mathcal{F}_{pz}$	Menge von Reihenfolgepositionen $idx \in \mathcal{RF}$, die zur Bewertung des Produktionszeitraumes $pz \in \mathcal{PZ}$ betrachtet werden müssen
$\mathcal{BZ}$	Menge von Gewichten im Vorranggraphen. Jedes Gewicht entspricht der Bearbeitungszeit τ_{avo} eines Arbeitsvorganges.
$\mathcal{CL}$	Menge von Clustern CL_i
CL_i	Cluster $CL_i \in \mathcal{CL}$

$clgr$	Größe der Cluster $CL_i \in CL$
$\Delta\mu_{ap}$	Taktausgleich an Arbeitsplatz $ap \in \mathcal{AP}$
Δw_{ap}	Unterschied zwischen der ursprünglichen Anzahl Werker an einem Arbeitsplatz $ap \in \mathcal{AP}$ und der nach der Rekonfiguration zugewiesenen Anzahl. Zeigt den Handlungsbedarf für den Reparaturvorgang im Rekonfigurationsprozess an.
$\Delta\psi_{CL_i,ap}$	Differenz in der Auftrittshäufigkeit von Bearbeitungszeiten im Cluster CL_i nach der Verschiebung eines Arbeitsvorganges auf einen Arbeitsplatz $ap \in \mathcal{AP}$
$\Delta^{\mathcal{AP}}$	Summierte externe Auslastungsunterschiede in allen Arbeitsplätzen $ap \in \mathcal{AP}$ in Zeiteinheiten
Δ^p	Summierte interne Auslastungsunterschiede in Zeiteinheiten
$\Delta^{p,2}$	Summierte quadrierte interne Auslastungsunterschiede in Zeiteinheiten zum Quadrat
$\Delta^{p,3}$	Summierte Abweichung der Bearbeitungszeiten aller Varianten von der gesamten Durchschnittsauslastung μ in Zeiteinheiten
$\Delta^{p,max}$	Größter interner Auslastungsunterschied in Zeiteinheiten
Δ^{Lin}	Summierte lineare positive und negative Abweichung der Bearbeitungszeiten von der Taktzeit über alle Arbeitsplätze $ap \in \mathcal{AP}$
$\Delta^{LinTakt}$	Summierte positive lineare Abweichung der Bearbeitungszeiten von der Taktzeit über alle Arbeitsplätze $ap \in \mathcal{AP}$
Δ^{LinPos}	Summierte lineare positive Abweichung der Bearbeitungszeiten von der Kapazität ζ_{ap} über alle Arbeitsplätze $ap \in \mathcal{AP}$
$\Delta^{QuadPos}$	Summierte quadrierte positive Abweichung der Bearbeitungszeiten von der Kapazität ζ_{ap} über alle Arbeitsplätze $ap \in \mathcal{AP}$
Δ_{ap}^{LinPos}	Summe der linearen positiven Abweichungen der Bearbeitungszeiten auf Arbeitsplatz $ap \in \mathcal{AP}$ von der Kapazität in Zeiteinheiten
$\Delta_{ap}^{QuadPos}$	Summe der quadrierten positiven Abweichungen der Bearbeitungszeiten auf Arbeitsplatz $ap \in \mathcal{AP}$ von der Kapazität in Zeiteinheiten
Δ_{-avo}^{LinPos}	Veränderung in den linearen positiven Abweichungen der Bearbeitungszeiten Δ^{LinPos} nach Entfernung eines Arbeitsvorganges in einer Rekonfigurationslösung
Δ_{+avo}^{LinPos}	Veränderung in den linearen positiven Abweichungen der Bearbeitungszeiten Δ^{LinPos} nach Hinzufügen eines Arbeitsvorganges in einer Rekonfigurationslösung
dbl_{ap}	Größe des vorgelagerten ‘linken’ Driftbereiches des Arbeitsplatzes $ap \in \mathcal{AP}$ in Zeiteinheiten
dbr_{ap}	Größe des nachgelagerten ‘rechten’ Driftbereiches des Arbeitsplatzes $ap \in \mathcal{AP}$ in Zeiteinheiten
ϵ_{ap}	Arbeitsaufwand, der von Arbeitsplatz $ap \in \mathcal{AP}$ verschoben werden muss, um eine Komprimierung zur Reduzierung der Anzahl Werker auf dem Arbeitsplatz ap durchführen zu können. Zeigt den Handlungsbedarf zum Komprimieren im Rekonfigurationsprozess an.
gz	Anzahl gleichzeitiger Unterstützereinsätze

$\mathcal{G}$	Vorranggraph, der aus einer Menge von Arbeitsvorgängen $\mathcal{AVO}$, einer Menge von Vorrangbedingungen $\mathcal{V}$ und einer Menge von Gewichten $\mathcal{BZ}$ besteht
$h_{a,p}$	Binär: 1, wenn Auftrag $a \in \mathcal{A}$ zu Werkstück $p \in \mathcal{P}$ zugeordnet ist; 0, sonst
i	Indexvariable
$idx_{t,ap}$	Reihenfolgeposition des Werkstückes, welches sich im Takt $t \in \mathcal{T}$ an Arbeitsplatz $ap \in \mathcal{AP}$ befindet
idx	Reihenfolgeposition $idx \in \mathcal{RF}$
j	Indexvariable
κ	Parameter für die lokale Suchheuristik in der Rekonfiguration. Definiert die Anzahl an weiterverfolgten Lösungen je Verschiebungsstufe. Bestimmt die Genauigkeit des Algorithmus.
$\mathcal{K}^U$	Kostensatz für einen Unterstützer
$\mathcal{K}^{BGes}$	Betriebskosten
$\mathcal{K}^{Ges}$	Gesamtkosten
$\mathcal{K}^{LinPos}$	Kostensatz für eine Zeiteinheit der Kennzahl Δ^{LinPos}
$\mathcal{K}^{NGes}$	Gesamtkosten, welche durch zusätzliche Nacharbeitskosten auftreten, die auf Unterstützereinsätze zurückgeführt werden können
$\mathcal{K}^N$	Kostensatz für Nacharbeit
$\mathcal{K}^{QuadPos}$	Kostensatz für eine Zeiteinheit der Kennzahl $\Delta^{QuadPos}$
$\mathcal{K}^{RFGes}$	Gesamtkosten, die durch die Reihenfolgeplanung beeinflusst werden
$\mathcal{K}^{RGes}$	Gesamtkosten, die durch die Rekonfiguration und damit durch die Verschiebung von Arbeitsvorgängen verursacht werden
$\mathcal{K}_{avo,ap}^R$	Kostensatz für die Verschiebung eines Arbeitsvorganges $avo \in \mathcal{AVO}$ auf einen Arbeitsplatz $ap \in \mathcal{AP}$
$\mathcal{K}^{UGes}$	Gesamtkosten für die Unterstützer
$\mathcal{K}^{WGes}$	Gesamtkosten für die Werker
$\mathcal{K}^W$	Kostensatz für einen Werker
k	Parameter, der angibt, wie viele Cluster gebildet werden sollen
$K_a(b)$	Korrelation der Arbeitsvorgänge $a \in \mathcal{AVO}$ und $b \in \mathcal{AVO}$
l	Index einer Lösung für die Rekonfiguration
lz	Summierte Leerzeit der Werker
μ	Durchschnittliche Bearbeitungszeit für alle Werkstücke $p \in \mathcal{P}$ auf allen Arbeitsplätzen $ap \in \mathcal{AP}$ im Verhältnis zur Kapazität ζ_{ap}
μ_{ap}^{ABS}	Durchschnittliche Bearbeitungszeit für alle Werkstücke $p \in \mathcal{P}$ auf Arbeitsplatz $ap \in \mathcal{AP}$ in Zeiteinheiten
μ_{ap}^{MAX}	Obergrenze für die durchschnittliche Auslastung von Arbeitsplatz $ap \in \mathcal{AP}$
μ_{ap}^{MIN}	Untergrenze für die durchschnittliche Auslastung von Arbeitsplatz $ap \in \mathcal{AP}$
μ_{ap}	Durchschnittliche Bearbeitungszeit für alle Werkstücke $p \in \mathcal{P}$ auf Arbeitsplatz $ap \in \mathcal{AP}$ im Verhältnis zur Kapazität ζ_{ap}
μ_p	Durchschnittliche Bearbeitungszeit des Werkstückes $p \in \mathcal{P}$ auf allen Arbeitsplätzen $ap \in \mathcal{AP}$ in Zeiteinheiten

$M_{v,x,wa}$	Menge von Knoten $\mathcal{AVO}$, die einer ausgehenden Kante $v \in V$ eines XOR-Aufsplittungsknotens $x \in X_{wa}$ an Werkstückansprache $wa \in \mathcal{WA}$ zugeordnet sind
N_{wa}	Menge aller Knoten $avo \in \mathcal{AVO}$ aus Werkstückansprache $wa \in \mathcal{WA}$, welche nicht in XOR Aufsplittungen liegen
ω	Parameter für die lokale Suchheuristik in der Rekonfiguration. Definiert die Anzahl an erlaubten, aufeinander folgenden Verschiebungen.
p	Werkstück $p \in \mathcal{P}$
$\mathcal{P}$	Menge von Werkstücken p
$\mathcal{P}_{avo}$	Menge aller Werkstücke $p \in \mathcal{P}$, die Arbeitsvorgang $avo \in \mathcal{AVO}$ benötigen
$\mathcal{P}_{pz}$	Menge von Werkstücken, die im Produktionszeitraum $pz \in \mathcal{PZ}$ produziert werden
pb_{avo}	Notwendiger Lagerplatz für die Ausführung von Arbeitsvorgang $avo \in \mathcal{AVO}$
pl_{ap}	Zur Verfügung stehender Lagerplatz an Arbeitsplatz $ap \in \mathcal{AP}$
$pos_{idx,ap}$	Endposition des Werkers an Arbeitsplatz $ap \in \mathcal{AP}$ nach Bearbeitung des Werkstückes $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ relativ zur linken Stationsgrenze von Arbeitsplatz $ap \in \mathcal{AP}$ in Zeiteinheiten
$\mathcal{POS}_{pz}$	Matrix mit den Werkerendpositionen jeweils nach der Bearbeitung der Werkstücke $p \in \mathcal{BWP}$ an den Reihenfolgepositionen $idx \in \mathcal{BWRF}_{pz}$ je Arbeitsplatz $ap \in \mathcal{AP}$ im Produktionszeitraum $pz \in \mathcal{PZ}$.
$\psi_{CL_i,ap}$	Auftrittshäufigkeit von Bearbeitungszeiten im Cluster $CL_i \in \mathcal{CL}$ auf Arbeitsplatz $ap \in \mathcal{AP}$
$\psi_{i,ap}$	Auftrittshäufigkeit der Bearbeitungszeit i auf Arbeitsplatz $ap \in \mathcal{AP}$
$\mathcal{PZ}$	Menge von Produktionszeiträumen pz
pz	Produktionszeitraum $pz \in \mathcal{PZ}$
ρ_{ap}^{LinPos}	Verhältnis von Δ_{ap}^{LinPos} zum Taktausgleich μ_{ap} an Arbeitsplatz $ap \in \mathcal{AP}$
$\rho_{ap}^{QuadPos}$	Verhältnis von $\Delta_{ap}^{QuadPos}$ zum Taktausgleich μ_{ap} an Arbeitsplatz $ap \in \mathcal{AP}$
rb_{wa}	Kennzahl für die Reduzierung des Betrachtungshorizontes durch die Einführung von XOR-Aufsplittungsknoten im Vorranggraphen
$\mathcal{RF}$	Menge von Reihenfolgepositionen idx
$\mathcal{RF}_{pz}$	Menge von Reihenfolgepositionen $idx \in \mathcal{RF}$ im Produktionszeitraum $pz \in \mathcal{PZ}$
$\sigma_{p1,p2,ap}$	Abweichung der Bearbeitungszeiten der Werkstücke $p1 \in \mathcal{P}$ und $p2 \in \mathcal{P}$ an Arbeitsplatz $ap \in \mathcal{AP}$
s_{ap}	Seite $s \in \{l; r\}$ des Fließbandes, an der der Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist
sit	Situation im Fließsystem. Definiert durch ein Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ innerhalb einer bestimmten Station $st \in \mathcal{ST}$
$\mathcal{SP}$	Menge von Ansprechzonen sp
sp	Ansprechzone $sp \in \mathcal{SP}$
sp_{ap}	Ansprechzone, der Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist
$\mathcal{ST}$	Menge von Stationen st
st_{ap}	Station $st \in \mathcal{ST}$, welcher der Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist

st	Station $st \in \mathcal{ST}$
τ_{avo}	Bearbeitungszeit des Arbeitsvorganges $avo \in \mathcal{AVO}$ in Zeiteinheiten
t	Takt $t \in \mathcal{T}$
$t_{idx,ap}$	Taktindex des Taktes, in dem sich das Werkstück $p \in \mathcal{P}$ mit der Reihenfolgeposition $idx \in \mathcal{RF}$ in der Station $st_{ap} \in \mathcal{ST}$ von Arbeitsplatz $ap \in \mathcal{AP}$ befindet
T_{pz}	Menge von Takten $t \in \mathcal{T}$, die im Produktionszeitraum $pz \in \mathcal{PZ}$ stattfinden
$\tau_{p,ap}$	Bearbeitungszeit des Werkstückes $p \in \mathcal{P}$ an Arbeitsplatz $ap \in \mathcal{AP}$ in Zeiteinheiten
ts	Anzahl von gleichzeitigen Situationen sit innerhalb jedes Taktes $t \in \mathcal{T}$
tz	Taktzeit der Fließlinie in Zeiteinheiten
$u_{t,ap}^{AnzAP}$	Anzahl gleichzeitiger Unterstützereinsätze an Arbeitsplatz $ap \in \mathcal{AP}$ in Takt $t \in \mathcal{T}$
$u_{t,st}^{AnzST}$	Anzahl gleichzeitiger Unterstützereinsätze an Station $st \in \mathcal{ST}$ in Takt $t \in \mathcal{T}$
$u_{t,idx,ap}^{Belegt}$	Binär: 1, wenn ein Unterstützereinsatz, der an dem Werkstück $p \in \mathcal{P}$ mit der Reihenfolgeposition $idx \in \mathcal{RF}$ an Arbeitsplatz $ap \in \mathcal{AP}$ durchgeführt wird, in Takt $t \in \mathcal{T}$ noch nicht abgeschlossen ist; 0, sonst
$u_{idx,ap}^{Einsatz}$	Binär: 1, wenn ein Unterstützereinsatz an Reihenfolgeposition idx an Arbeitsplatz $ap \in \mathcal{AP}$ durchgeführt wird; 0, sonst
$u_{t,idx,ap}^{Ende}$	Binär: 1, wenn ein Unterstützereinsatz, der an dem Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ an Arbeitsplatz $ap \in \mathcal{AP}$ durchgeführt wird, in oder nach Takt $t \in \mathcal{T}$ abgeschlossen wird; 0, sonst
u^E	Gesamtanzahl Unterstützereinsätze
u^{MAX}	Maximale Anzahl gleichzeitiger Unterstützereinsätze $\hat{=}$ Notwendige Anzahl an Unterstützereinsatzkräften
$u_{t,idx,ap}^{Start}$	Binär: 1, wenn ein Unterstützereinsatz, der an dem Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ an Arbeitsplatz $ap \in \mathcal{AP}$ durchgeführt wird, in oder vor Takt $t \in \mathcal{T}$ angefangen hat; 0, sonst
U^{Zeit}	Summierte Unterstützereinsatzzeit
$u_{idx,ap}^{Zeit}$	Einsatzzeit für den Unterstützereinsatz, der an dem Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ an Arbeitsplatz $ap \in \mathcal{AP}$ durchgeführt wird
u_t	Anzahl gleichzeitiger Unterstützereinsätze in Takt $t \in \mathcal{T}$
$\mathcal{V}$	Menge von Vorrangbedingungen. Wenn $(i, j) \in \mathcal{V}$, dann muss Arbeitsvorgang $i \in \mathcal{AVO}$ abgeschlossen sein, bevor mit Arbeitsvorgang $j \in \mathcal{AVO}$ begonnen werden kann. Es gilt: $\mathcal{V} \subset \mathcal{AVO} \times \mathcal{AVO}$
$v_{avo,ap}^R$	Binär: 1, wenn Arbeitsvorgang $avo \in \mathcal{AVO}$ in der neuen Zuordnung zu einem neuen Arbeitsplatz $ap \in \mathcal{AP}$ verschoben wurde (Verschiebungsvariable)
$\mathcal{VAR}$	Menge aller bestellbaren Varianten
$\mathcal{VG}_{ap}$	Menge von Vorgängerarbeitsplätzen $vg \in \mathcal{AP}$, von denen Arbeitsplatz $ap \in \mathcal{AP}$ abhängt
vg	Vorgängerarbeitsplatz $vg \in \mathcal{AP}$

$vu_{idx,ap}$	Vorgängerüberhang, welcher die Startposition des Werkers an Arbeitsplatz $ap \in \mathcal{AP}$ bei der Bearbeitung des Werkstückes $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ beeinflusst
w_{ap}	Anzahl an Werkern, die Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen sind
$\mathcal{WA}$	Menge von Werkstückansprachen wa
wa	Werkstückansprache $wa \in \mathcal{WA}$
wa_{avo}	Werkstückansprache, an der Arbeitsvorgang $avo \in \mathcal{AVO}$ durchgeführt wird
$\mathcal{WP}$	Menge von Werkstückpositionen wp
wp	Werkstückposition $wp \in \mathcal{WP}$; Gibt die Lage eines Werkstückes an, z.B. horizontal oder vertikal
$\mathcal{WP}_{avo}$	Menge möglicher Werkstückpositionen $wp \in \mathcal{WP}$ für die Ausführung von Arbeitsvorgang $avo \in \mathcal{AVO}$. Es gilt: $\mathcal{WP}_{avo} \subseteq \mathcal{WP}$
wp_{st}	Werkstückposition $wp \in \mathcal{WP}$ an $st \in \mathcal{ST}$
$wv_{idx,ap}$	Werkerverzug des Werkers an Arbeitsplatz $ap \in \mathcal{AP}$ vor der Bearbeitung des Werkstückes $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$
$wz_{vg,ap,idx}$	Wartezeit durch Vorgängerarbeitsplatz $vg \in \mathcal{VG}_{ap}$, welche die Startposition des Werkers an Arbeitsplatz $ap \in \mathcal{AP}$ bei der Bearbeitung des Werkstückes $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ beeinflusst
χ	Parameter für die lokale Suchheuristik in der Rekonfiguration. Definiert die Zielfunktion der Suche.
$\mathcal{X}$	Menge von XOR-Aufsplittungsknoten x
$\mathcal{X}_{wa}$	Menge von XOR-Aufsplittungsknoten x , die Werkstückansprache $wa \in \mathcal{WA}$ zugeordnet sind. Es gilt: $\mathcal{X}_{wa} \subseteq \mathcal{X}$
x	XOR-Aufsplittungsknoten $x \in \mathcal{X}$
$x_{avo,ap}^{Soll}$	Binär: 1, wenn Arbeitsvorgang $avo \in \mathcal{AVO}$ Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist; 0, sonst (neue Zuordnungslösung)
$y_{p,idx}^{\mathcal{RF}}$	Binär: 1, wenn Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ produziert wird; 0, sonst (Reihenfolgelösung)
ζ_{ap}	Kapazität des Arbeitsplatzes $ap \in \mathcal{AP}$ in Zeiteinheiten
Z	Menge von Clusterzentrioden z_i
$z_{avo,ap}^{Ist}$	Binär: 1, wenn Arbeitsvorgang $avo \in \mathcal{AVO}$ dem Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist; 0, sonst (existierende Zuordnungslösung)
z_i	Zentroid $z_i \in Z$ des Clusters $CL_i \in CL$

1 Einleitung

Wer aufhört besser zu werden,
hat aufgehört gut zu sein.

(Philip Rosenthal)

Die Fließfertigung zeichnet sich durch eine Anordnung nach dem Objektprinzip aus:¹ Die Arbeitssysteme² sind entsprechend der Reihenfolge der durchzuführenden Arbeitsvorgänge angeordnet.³ Von getakteter Fließfertigung wird gesprochen, wenn die Werkstücke synchron automatisiert von Station zu Station weitergegeben werden. Getaktete Fließlinien haben ihre Vorzüge insbesondere bei der Produktion von Massenprodukten; ursprünglich wurden sie ausschließlich für die Produktion eines homogenen Erzeugnisses wie z.B. dem legendären Model T von Ford verwendet.⁴ Die Planung einer solchen Fließlinie beschränkt sich auf das Zuordnungsproblem *Simple-Assembly Line Balancing Problem* (SALBP); das erste Mal formuliert von Salveson⁵.

Im Zuge des Wechsels vom Anbieter- zum Käufermarkt wird eine enorme Individualisierbarkeit der Produkte gefordert.⁶ Daher werden heutzutage auf einer einzelnen Fließlinie viele verschiedene Varianten⁷ eines Produktes produziert.⁸ Mit jeder weiteren angebotenen Option steigt die Variantenvielfalt exponentiell⁹ an und mit ihr der Ressourceneinsatz, der je Werkstück an Personal, Betriebsmitteln und Material eingesetzt werden muss.¹⁰

¹Vgl. [GT04, S.13ff.]

²„Ein Arbeitssystem beinhaltet das Zusammenwirken von einer oder mehreren Personen mit den Arbeitsmitteln, um die Systemaufgabe im Arbeitsbereich, in der Arbeitsumgebung, unter den durch die Arbeitsaufgaben gesetzten Bedingungen zu erfüllen.“ [DIN EN614T1]

³[DW97, S.8]

⁴Vgl. [Sch99]

⁵Vgl. [Sal55]

⁶Vgl. [Ros74], [Sch89, S.60], [Lan90], [BS93] und [Mil93]

⁷„Varianten eines Erzeugnisses, einer Gruppe oder eines Teils sind die Veränderungen der Grundaussführung, die durch Weglassen oder Hinzufügen von Einzelteilen oder Gruppen hinsichtlich Gestalt, Beschaffenheit und Eigenschaften entstehen“ [Dan08]

⁸Insbesondere in der Automobilindustrie zeigt sich dieser Trend sehr deutlich. Das Mercedes C-Klasse Modell kann z.B. in 10²⁷ verschiedenen Varianten bestellt werden. [Röd06] BMW fertigt 10³² verschiedene Varianten in seinem Produktportfolio. [Mey04]

⁹Dies lässt sich einfach verdeutlichen, wenn man betrachtet, dass die Kombination von n möglichen Ausstattungsmerkmalen 2^n Varianten ergibt.

¹⁰Vgl. [Brä03]

Die Kapazität in einer Variantenfließlinie¹¹ wird durch das Personal und die Betriebsmittel definiert.¹² Zur Gewährleistung einer effizienten Produktion müssen diese kostenoptimal in der Fließlinie bereitgestellt und genutzt werden. Dieses sog. Kapazitätsabstimmungs-Problem¹³ wird bereits seit über 40 Jahren intensiv erforscht.¹⁴ Existierende Ansätze gehen davon aus, dass es genügt, die Anpassung der Kapazitäten einmalig für die Lebensdauer einer Modellreihe im Rahmen des *Mixed-Model Assembly Line Balancing Problem* (MALBP) durchzuführen und Schwankungseffekte in den Bearbeitungszeiten, die durch verschiedene Varianten auftreten, ausschließlich durch eine Reihenfolgeplanung abzufangen.¹⁵ Die Model-Mix-Verluste¹⁶ konnten so auf erträglichem Maß gehalten werden.

Dieser Ansatz reicht heute nicht mehr aus, da zunehmende Belastungsschwankungen zu steigenden Model-Mix-Verlusten und sinkenden Kapazitätsauslastungen in den Werken führen.¹⁷ Es wird daher ein Ansatz für eine Abstimmung von Kapazität und zu produzierenden Varianten benötigt. Dabei kann nicht mehr von einem ausgeglichenen Produktionsprogramm ausgegangen werden. Vielmehr muss es eine regelmäßige, kostenoptimale Kapazitätsabstimmung geben, die eine Anpassung auf die zukünftig zu produzierenden Varianten beinhaltet und die Umstellungskosten von einer Abstimmung auf die nächste betrachtet. Insbesondere müssen dabei Flexibilisierungsmaßnahmen, wie offene Stationsgrenzen und kurzfristige Kapazitätserweiterungen durch den Einsatz von Unterstützern, kostenoptimal genutzt werden.

Die Entscheidungen, die im Rahmen dieser Abstimmung zu treffen sind, können nicht simultan¹⁸ erfolgen. Die Zuordnung von Arbeitsvorgängen, Werkern und Betriebsmitteln zu Arbeitsplätzen sollte einen möglichst langfristigen, bekannten Planungszeitraum abdecken, damit sich die entstehenden Umstellungskosten amortisieren können. Die Reihenfolgeplanung und die Allokation von Unterstützereinsatzkräften müssen jedoch auf Grund kurzfristiger Änderungen¹⁹, der Abhängigkeit von der Produktionsprogrammplanung und der enormen Komplexität des Problems mit einem kleineren Planungshorizont durchgeführt werden.

¹¹Vgl. [Boy05b] sowie synonym „Model-Mix Montagelinie“ [Leo97], „Mixed-Model Assembly“ [Tho67]

¹²Vgl. [Got06b, S.28]

¹³„Aufgabe der Kapazitätsabstimmung ist es, Kapazitätsnachfrage und Kapazitätsangebot lang-, mittel- und kurzfristig abzustimmen.“ [Bul86, S.38]

¹⁴Vgl. [Tho67]

¹⁵Vgl. [Boy05b, S.137]

¹⁶Vgl. Definition von [Koe85, S.61]

¹⁷Vgl. für das Beispiel Automobilindustrie: [Bec07, S.83]

¹⁸Vgl. zur Definition der „Simultanplanung“ [Sch01, S.21]

¹⁹Vgl. [BRvB06, S.881] „In der Praxis wird es oft unmöglich einen früher errechneten Plan umzusetzen, da kurzfristige Änderungen die Daten, von denen ausgegangen wurde, verändern“

Abgeleitet aus den aktuellen Anforderungen, die sich in der Variantenfließfertigung ergeben, wird in dieser Arbeit eine Lösung für die folgende Aufgabenstellung erarbeitet:

„Kostenoptimale Kapazitätsabstimmung in einer getakteten Variantenfließlinie - unter expliziter Berücksichtigung des Unterstützereinsatzes und unterschiedlicher Planungszeiträume“

Diese Arbeit ist in sieben Kapitel gegliedert. In Kapitel 2 wird das System „getaktete Variantenfließlinie“ mit seinen Zusammenhängen formal beschrieben. Dabei werden die Kostentreiber im System identifiziert. Das Problem wird anschließend systematisiert und so in Teilprobleme zerlegt, dass die unterschiedlichen Fristigkeiten und Planungshorizonte der Entscheidungen beachtet werden. Für jedes der Teilprobleme werden das Ziel, die Aufgabe und die Anforderungen an eine Problemlösung formuliert. In Kapitel 3 werden existierende Lösungsansätze hinsichtlich ihrer Eignung zur Verwendung in einer Problemlösung je Teilproblem diskutiert. Darauf aufbauend werden in Kapitel 4 die Aufgaben zur notwendigen Erweiterung existierender Ansätze bzw. Entwicklung von neuen Konzepten systematisiert. In Kapitel 5 wird ein Ansatz zur kostenoptimalen Kapazitätsabstimmung in getakteten Variantenfließlinien vorgestellt. Dabei werden Konzepte für die einzelnen Teilprobleme erarbeitet. Insbesondere muss darauf geachtet werden, dass die getrennten Teilprobleme in einem hierarchischen System integriert funktionieren. Da die Ergebnisse dieser Arbeit nicht nur theoretischer Natur sind, sondern eine in der Praxis anwendbare Lösung darstellen sollen, werden die Methoden mit Hilfe einer prototypischen Umsetzung in Form einer Softwarelösung anhand von Praxisbeispielen in Kapitel 6 validiert. Zum Schluss folgt in Kapitel 7 eine kritische Auseinandersetzung mit dem Ergebnis dieser Arbeit sowie ein Ausblick auf weitere sich ergebende Forschungsfragestellungen.

2 Problemdefinition

Das Problem zu erkennen ist wichtiger als die Lösung zu erkennen,
denn die genaue Darstellung des Problems führt zur Lösung.

(Albert Einstein)

2.1 Systembeschreibung

Im Folgenden wird der Aufbau einer Variantenfließlinie auf Basis der Notation aus Tabelle 2.1 beschrieben. Darauf aufbauend werden eine Beschreibung des Produktionsablaufes und die Strukturierung der entstehenden Kosten in einer Variantenfließlinie vorgenommen.

2.1.1 Aufbau einer getakteten Variantenfließlinie

Eine Variantenfließlinie besteht aus einer festen Anzahl von Stationen $st \in \mathcal{ST}$. Zwischen den einzelnen Stationen existieren keine Puffer.²⁰ Die Werkstücke²¹ $p \in \mathcal{P}$ bewegen sich automatisiert durch jede einzelne Station (z.B. mit Hilfe eines Transportbandes oder einer Elektrohängebahn). Zu jedem Werkstück gehört genau ein Auftrag $a \in \mathcal{A}$, der spezifiziert, welche Arbeitsvorgänge²² $avo \in \mathcal{AVO}$ an dem Werkstück durchzuführen sind. Jedem Arbeitsvorgang $avo \in \mathcal{AVO}$ ist eine Werkstückansprache²³ $wa \in \mathcal{WA}$ durch wa_{avo} zugewiesen, die angibt, an welchem Bereich des Werkstückes der Arbeitsvorgang durchzuführen ist. Die Stationen sind in ihrer räumlichen Ausdehnung gleich groß. Jede Station besitzt eine „linke“ Stationsgrenze, an der das Werkstück in die Station einfährt, und eine „rechte“ Stationsgrenze, an der es die Station wieder verlässt. Je Station befindet sich das Werkstück in einer durch wp_{st} fest definierten Werkstückposition²⁴ $wp \in \mathcal{WP}$. Jedem Werkstück p ist eine Position idx

²⁰In der Realität wird die Fließlinie noch einmal in einzelne Linienabschnitte unterteilt. Die starre Verkettung lässt sich jedoch auch auf den Übergang zwischen zwei Linienabschnitten übertragen.

²¹Ein **Werkstück** bezeichnet das physikalische Objekt, an dem gearbeitet wird. Es stellt nach Ablauf des Produktionsauflaufes das Produkt dar. Daher wird im Folgenden synonym Produkt oder Werkstück verwendet.

²²Ein **Arbeitsvorgang** ist ein nicht mehr sinnvoll zu zerteilender Produktionsschritt [Boy05b, S.8]

²³Eine **Werkstückansprache** ist ein bestimmter Bereich an einem Werkstück, an dem gearbeitet wird.

²⁴Eine **Werkstückposition** beschreibt die Lage eines Werkstückes, wie z.B. horizontal oder vertikal.

Symbol	Definition
$\mathcal{A}$	Menge von Aufträgen a
$\mathcal{AP}$	Menge von Arbeitsplätzen ap
$\mathcal{AVO}$	Menge von Arbeitsvorgängen avo
$\mathcal{P}$	Menge von Werkstücken p
$\mathcal{RF}$	Menge von Reihenfolgepositionen idx
s_{ap}	Seite $s \in \{l; r\}$ des Fließbandes, an der der Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist
$\mathcal{SP}$	Menge von Ansprechzonen sp
sp_{ap}	Ansprechzone, der Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist
$\mathcal{ST}$	Menge von Stationen st
st_{ap}	Station $st \in \mathcal{ST}$, welcher der Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist
tz	Taktzeit der Fließlinie in Zeiteinheiten
w_{ap}	Anzahl an Werkern, die Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen sind
$\mathcal{WA}$	Menge von Werkstückansprachen wa
wa_{avo}	Werkstückansprache, an der Arbeitsvorgang $avo \in \mathcal{AVO}$ durchgeführt wird
$\mathcal{WP}$	Menge von Werkstückpositionen wp
wp	Werkstückposition $wp \in \mathcal{WP}$; Gibt die Lage eines Werkstückes an, z.B. horizontal oder vertikal
wp_{st}	Werkstückposition $wp \in \mathcal{WP}$ an $st \in \mathcal{ST}$
$y_{p,idx}^{\mathcal{RF}}$	Binär: 1, wenn Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ produziert wird; 0, sonst (Reihenfolgelösung)
$z_{avo,ap}^{Ist}$	Binär: 1, wenn Arbeitsvorgang $avo \in \mathcal{AVO}$ dem Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist; 0, sonst (existierende Zuordnungslösung)

Tabelle 2.1: Notation zur Formalisierung des Aufbaus einer Variantenfließlinie

in der Reihenfolge $\mathcal{RF}$ durch $y_{p,idx}^{\mathcal{RF}}$ zugewiesen.²⁵ Diese Position gibt an, in welchem Takt das Werkstück in die erste Station einfährt. Die Taktzeit²⁶ tz gibt vor, in welchem Rhythmus Werkstücke in die Fließlinie eingelastet werden. Diese Taktzeit wird durch das geplante Produktionsprogramm (gewünschte Stückzahl) und die innerhalb dieses Zeitraumes realisierbare Betriebszeit bestimmt (2.1).²⁷

$$tz = \frac{\text{Betriebszeit}}{\text{Stückzahl}} \quad (2.1)$$

Jeder Station sind ein oder mehrere Arbeitsplätze $ap \in \mathcal{AP}$ durch st_{ap} zugeordnet. Die Fläche einer Station ist aufgeteilt in durchnummerierte Ansprechzonen²⁸, denen jeweils maximal

²⁵Wird im Folgenden auf eine Reihenfolgeposition idx verwiesen, ist damit implizit das dahinter stehende Werkstück $p \in \mathcal{P}$ gemeint, für welches $y_{p,idx}^{\mathcal{RF}} = 1$ gilt.

²⁶Synonyme: Zykluszeit, Auflegeintervall

²⁷Vgl. [KSW96, S.28]

²⁸Eine **Ansprechzone** bezeichnet einen Bereich in der Station, in dem Arbeiten durchgeführt werden.

ein Arbeitsplatz durch sp_{ap} zugeordnet sein kann.²⁹ Jeder Arbeitsplatz wird durch s_{ap} einer Seite der Fließlinie zugeordnet. Jeder Arbeitsvorgang avo ist durch $z_{avo,ap}^{Ist}$ immer genau einem Arbeitsplatz ap zugewiesen (2.2), wobei jedem Arbeitsplatz mehrere Arbeitsvorgänge zugewiesen sein können.³⁰

$$\sum_{ap \in \mathcal{AP}} z_{avo,ap}^{Ist} = 1 \quad \forall \quad avo \in \mathcal{AVO} \quad (2.2)$$

Das Kapazitätsangebot eines Arbeitsplatzes innerhalb eines Taktes ist definiert durch die Anzahl Werker w_{ap} , die einem Arbeitsplatz ap zugewiesen sind.³¹ Falls mehrere Werker einem Arbeitsplatz zugeordnet sind ($w_{ap} > 1$), handelt es sich um einen sog. Mehrtakter und um horizontale Arbeitsteilung.³² In diesem Fall hat jeder Werker die w_{ap} -fache Zeit und damit mehrere Takte zur Verfügung, um ein Werkstück zu bearbeiten. Die Werker überholen sich gegenseitig und übernehmen abwechselnd die Werkstücke, so dass nur jedes w_{ap} -te Werkstück von demselben Werker bearbeitet wird.

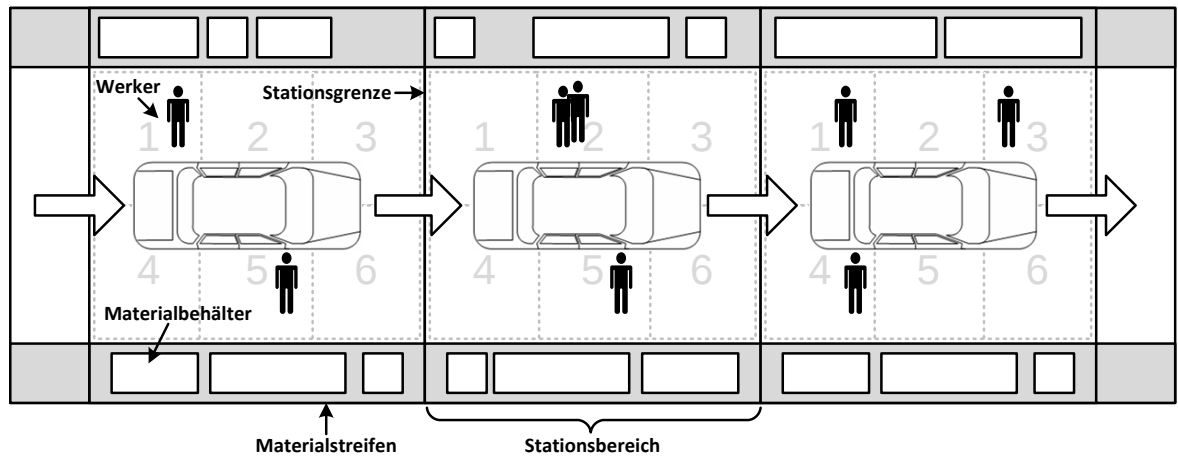


Abbildung 2.1: Beispiel für den Aufbau einer getakteten Variantenfließlinie in der automobilen Endmontage

²⁹Vgl. hierzu in Grafik 2.1 die Quadrate 1-6, welche die Ansprechzonen darstellen. Die sechs verfügbaren Ansprechzonen in dem Beispiel verdeutlichen, dass an bis zu sechs verschiedenen Arbeitsplätzen parallel an ein und derselben Station gearbeitet werden kann.

³⁰Diese Einschränkung hat zwei Gründe. Zum einen wird hierdurch dafür gesorgt, dass die notwendigen Teile für einen Arbeitsvorgang auch nur an einer Stelle des Bandes bereitgestellt werden müssen. Zum anderen wird der Arbeitsvorgang dadurch immer durch dieselben Werker durchgeführt, so dass eine höhere Montagequalität erwartet wird. Vgl. auch die Aussage in [Dan03, S.361]: “Es wird zusätzlich gefordert, dass diese Arbeitsvorgänge für alle Varianten an derselben Station ausgeführt werden sollen, um die Vorteile dieser Produktionsform überhaupt erst nutzen zu können.”

³¹Vgl. Definition von Kapazität durch Kern [Ker62, S.27]: “Kapazität kann als das Leistungsvermögen einer technischen oder wirtschaftlichen Einheit beliebiger Art, Größe oder Struktur in einem Zeitabschnitt verstanden werden.”

³²Vgl. [Got06b, S.36]

In Abbildung 2.1 wird der Aufbau einer Variantenfließlinie an einem Beispiel aus der Automobilindustrie verdeutlicht.³³

2.1.2 Produktionsablauf in einer getakteten Variantenfließlinie

Im Folgenden wird der Produktionsablauf innerhalb einer getakteten Variantenfließlinie beschrieben. Dabei wird immer von einem störungsfreien Ablauf der Produktion ausgegangen.³⁴ Zur Formalisierung des Produktionsablaufes wird die Notation um die Bezeichner in Tabelle 2.2 erweitert.

Symbol	Definition
$bp_{a,avo}$	Binär: 1, wenn der Auftrag $p \in \mathcal{P}$ den Arbeitsvorgang $avo \in \mathcal{AVO}$ benötigt; 0, sonst (Bauplan)
br_{avo}	Baurate des Arbeitsvorganges $avo \in \mathcal{AVO}$. Gibt die Häufigkeit des Auftretens eines Arbeitsvorganges im Verhältnis zur Gesamtanzahl Aufträge an.
dbl_{ap}	Größe des vorgelagerten „linken“ Driftbereiches des Arbeitsplatzes $ap \in \mathcal{AP}$ in Zeiteinheiten
dbr_{ap}	Größe des nachgelagerten „rechten“ Driftbereiches des Arbeitsplatzes $ap \in \mathcal{AP}$ in Zeiteinheiten
$h_{a,p}$	Binär: 1, wenn Auftrag $a \in \mathcal{A}$ zu Werkstück $p \in \mathcal{P}$ zugeordnet ist; 0, sonst
τ_{avo}	Bearbeitungszeit des Arbeitsvorganges $avo \in \mathcal{AVO}$ in Zeiteinheiten
$\tau_{p,ap}$	Bearbeitungszeit des Werkstückes $p \in \mathcal{P}$ an Arbeitsplatz $ap \in \mathcal{AP}$ in Zeiteinheiten
$\mathcal{PZ}$	Menge von Produktionszeiträumen pz
$\mathcal{VG}_{ap}$	Menge von Vorgängerarbeitsplätzen $vg \in \mathcal{AP}$, von denen Arbeitsplatz $ap \in \mathcal{AP}$ abhängt

Tabelle 2.2: Erweiterung der Notation zur Formalisierung des Produktionsablaufes in einer Variantenfließlinie

Zu Beginn eines Produktionszeitraumes³⁵ $pz \in \mathcal{PZ}$, der z.B. einer Schicht oder einem Tag entspricht, befinden sich unfertige Werkstücke in den einzelnen Stationen. Am Ende des Produktionszeitraumes pz wird die gleiche Menge unfertiger Werkstücke in den Stationen hinterlassen und erst im nächsten Produktionszeitraum $pz + 1$ weiter bearbeitet.

Die Bearbeitung eines Werkstückes stellt sich wie folgt dar:

³³Vgl. dazu auch die Darstellung von Dieckhoff in [Die95, S.66]

³⁴Störungen, die z.B. durch fehlerhafte Teile, technische Probleme oder menschliche Fehler während des Produktionsprozesses entstehen, können nicht vorhergesehen werden und werden im Rahmen dieser Arbeit nicht betrachtet.

³⁵Ein **Produktionszeitraum** beschreibt einen Zeitraum, in dem eine fest definierte Menge Werkstücke produziert wird.

Der Werker startet mit dem Werkstück an der linken Stationsgrenze und bewegt sich mit diesem in Fließrichtung des Transportsystems, während er die seinem Arbeitsplatz zugeordneten Arbeitsvorgänge abarbeitet. Nach Beendigung der Arbeitsvorgänge bewegt er sich entgegen der Fließrichtung zum nächsten Werkstück.³⁶

Die Bearbeitungszeit eines Arbeitsvorganges τ_{avo} wird als Vorgabezeit bezeichnet und wird nach REFA³⁷ ermittelt. Vorgabezeiten stellen den zu erwartenden Mittelwert der Arbeitsgeschwindigkeit dar und werden in dieser Arbeit als deterministisch vorgegeben betrachtet.

2.1.2.1 Belastungsschwankungen durch Variantenvielfalt

Die summierte Bearbeitungszeit $\tau_{p,ap}$ eines Werkstückes p auf einem Arbeitsplatz ap fällt je nach Spezifizierung des zugehörigen Auftrages a unterschiedlich aus. Diese Spezifizierung wird durch einen Bauplan $bp_{a,avo}$ vorgenommen, der definiert, ob ein Arbeitsvorgang zur Erfüllung eines Auftrages benötigt wird oder nicht. In Gleichung (2.3) wird die Bearbeitungszeit eines Werkstückes auf einem Arbeitsplatz durch die Summe aller Bearbeitungszeiten τ_{avo} der einzelnen Arbeitsvorgänge, die dem betrachteten Arbeitsplatz zugewiesen ($z_{avo,ap}^{Ist} = 1$) und für den zugehörigen Auftrag ($h_{a,p} = 1$) notwendig sind ($bp_{a,avo} = 1$), definiert. Hierdurch wird implizit die Beziehung jedes Auftrages zum zugeordneten Werkstück abgebildet.

$$\tau_{p,ap} = \sum_{avo \in AV\mathcal{O}} \tau_{avo} \cdot z_{avo,ap}^{Ist} \cdot bp_{a,avo} \quad \forall \quad p \in \mathcal{P} \wedge a \in \mathcal{A} : h_{a,p} = 1 \quad (2.3)$$

Auf Grund des großen Variantenreichtums, der auf einer Variantenfließlinie gefertigt wird, hat sich die Auslegung auf eine Durchschnittsvariante nach dem Ansatz von Thomopoulos³⁸ durchgesetzt.³⁹ Das Fließsystem wird dabei auf eine imaginäre Produktvariante ausgelegt, die sämtliche möglichen Arbeitsvorgänge benötigt. Die Bearbeitungszeit jedes Arbeitsvorganges wird mit der zugehörigen Baurate br_{avo} gewichtet, die angibt, wie häufig ein Arbeitsvorgang avo , gemessen an dem gesamten Auftragsvolumen, durchgeführt werden muss (2.4).

³⁶Entgegen einiger Ansätze aus der Literatur, z.B. [Mol96] und [Boc99], wird die Rückkehr zum nächsten Werkstück hier nicht als unendlich schnell, sondern in ihrer realen Geschwindigkeit beachtet, indem ein Arbeitsvorgang für den Rücklaufweg zum nächsten Werkstück existiert. Dies hat den Vorteil, dass unterschiedliche Geschwindigkeiten für den Rücklauf des Werkers dargestellt werden können. Je nach zu bewegendem Betriebsmittel oder der Anzahl an Werkstücken, die auf Grund eines Mehrtakters übersprungen werden müssen, ist der Rücklauf länger oder kürzer.

³⁷Vgl. [REF84] - Wonach mit Hilfe von vier verschiedenen Methoden die Vorgabezeiten ermittelt werden: Durch die Berechnung einflussgrößenbasierter Planzeiten, mit Hilfe von vorbestimmten Zeiten z.B. in MTM-Tabellen, durch Schätzung und Vergleich oder die einfache Berechnung im Fall von automatisierten Prozessschritten. Vgl. auch [Got06b, S.29]

³⁸Vgl. [Tho67]

³⁹Vgl. [Boy05b, S.137]

$$br_{avo} = \frac{\sum_{a \in \mathcal{A}} bp_{a,avo}}{|\mathcal{A}|} \quad (2.4)$$

Die Abstimmung des Fließsystems wird dann mit den gewichteten Bearbeitungszeiten der einzelnen Arbeitsvorgänge ($\tau_{avo} \cdot br_{avo}$) durchgeführt. Zielsetzung ist es dabei, eine möglichst hohe durchschnittliche Auslastung der Arbeitsplätze zu erreichen und wenig Effizienzverluste durch Leerzeiten zu erzeugen.⁴⁰

Die notwendige Werkeranzahl je Arbeitsplatz ergibt sich durch die berechnete Durchschnittsauslastung (2.5).

$$w_{ap} = \left\lceil \frac{\sum_{avo \in \mathcal{AVO}} \tau_{avo} \cdot br_{avo} \cdot z_{avo,ap}^{Ist}}{tz} \right\rceil \quad (2.5)$$

Wird ein Fließsystem auf die Durchschnittsvariante ausgelegt, wie in Abbildung 2.2 dargestellt, so führen die Unterschiede in den Bearbeitungszeiten, je nach Variante, die gerade produziert wird, zeitweise zu Überlastungen und zeitweise zu Leerzeiten.⁴¹

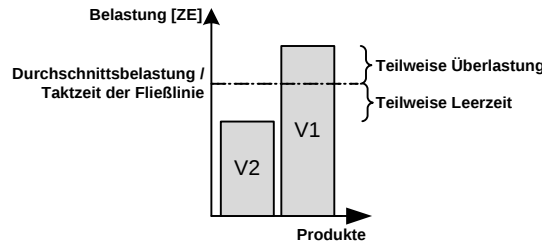


Abbildung 2.2: Beispiel für die Ausrichtung an einer Durchschnittsvariante

Im Vergleich dazu wird bei einer Auslegung auf eine Maximalvariante, wie sie z.B. von Deutsch⁴² vorgestellt wird, jeder Arbeitsplatz so geplant, dass die Variante mit der höchsten Bearbeitungszeit innerhalb der Taktzeit von einem Werker montiert werden kann. Betrachtet man dieselbe Belastungssituation wie in Abbildung 2.2, so müsste ein Fließsystem aggregiert betrachtet⁴³, für die größere der beiden Varianten ausgelegt werden. Bei gleicher Stückzahl der beiden Produkte ergibt sich in dem Beispiel in Abbildung 2.3 zwangsläufig eine Reduzierung

⁴⁰Vgl. [Dan03, S.361ff] und [DSV93, S.227]. Dies beinhaltet implizit auch die Minimierung der Stations- und Werkeranzahl

⁴¹Vgl. [BS03, S.16u]

⁴²Vgl. zu dem Ansatz [Deu71]

⁴³Eine genaue Betrachtung der einzelnen Arbeitsplätze sorgt unter Umständen durch die Kopplung des Systems für einen noch größeren Effizienzverlust, da die Maximalvariante nicht in Summe bestimmt werden kann, sondern das Maximum aus allen einzelnen Maximalvarianten ausschlaggebend ist.

der Effizienz um 25%.⁴⁴ Je größer die Streuung um die Durchschnittsauslastung eines Arbeitsplatzes ist, desto gravierender der Effekt. Diese Vorgehensweise wird aufgrund der zu hohen Effizienzverluste nur noch bei Produkten mit geringer Fertigungszeitspreizung⁴⁵ angewendet. Im weiteren Verlauf dieser Arbeit wird daher von einer Auslegung des Fließsystems auf eine Durchschnittsvariante nach Thomopoulos⁴⁶ ausgegangen und die angedeuteten Überlastungen und Leerzeiten müssen durch erhöhte Flexibilität im Produktionssystem abgefangen werden. Im Folgenden werden Möglichkeiten zur Kompensation dieser Belastungsschwankungen diskutiert.

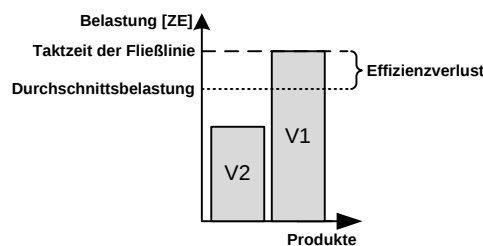


Abbildung 2.3: Beispiel für den Effizienzverlust bei Ausrichtung an der Maximalvariante

2.1.2.2 Kompensation von Belastungsschwankungen

2.1.2.2.1 Kapazitätsflexibilisierung durch Driftbereiche Wichtigstes Instrument zum Ausgleich von Belastungsschwankungen sind die Driftbereiche⁴⁷, die an der vorgelagerten „linken“ dbl_{ap} und der nachgelagerten „rechten“ Seite db_{ap} der Stationsgrenzen je Arbeitsplatz ap definiert werden.⁴⁸ Der Werker kann seinen rechten Driftbereich zusätzlich zum eigentlichen Stationsbereich verwenden, um seine eigene Kapazität kurzzeitig zu erhöhen und somit auch Werkstücke mit Bearbeitungszeiten, die größer als die Taktzeit sind, zu bearbeiten. Ein linker Driftbereich gibt dem Werker die Möglichkeit, seine Arbeit aufzunehmen, bevor sich das Werkstück innerhalb der eigentlichen Stationsgrenzen befindet. Hierdurch lässt sich die Kapazität für ein nachfolgendes Werkstück erhöhen. Eine Begrenzung der Größe von Driftbereichen ist in der Praxis insbesondere durch fest installierte Betriebsmittel als auch durch den Anfang und das Ende eines Linienabschnitts⁴⁹ gegeben. Arbeitsplätze, denen ein Driftbereich zugewiesen wird, werden *linksoffen* $dbl_{ap} > 0$ und *rechtsoffen* $db_{ap} > 0$ genannt. Dement-

⁴⁴Vgl. dazu die Äußerungen in [BS03, S.17]

⁴⁵Die **Fertigungszeitspreizung** bezeichnet die Differenz zwischen der Variante, die die geringste Bearbeitungszeit aufweist, und der „Maximalvariante“, welche die höchste Bearbeitungszeit benötigt.

⁴⁶Vgl. [Tho67]

⁴⁷Vgl. hierzu die Definitionen von [Dec93], [Mol96], [Boc99] und [GT04], in dem von einer „Abschwimm- und Aufholbewegung sowie einer Pufferzone“ gesprochen wird. Weiteres Synonym = Überlappungsbereiche

⁴⁸Vgl. hierzu Grafik 2.4.

⁴⁹Ein **Linienabschnitt** bezeichnet einen physikalischen Bereich der Fließlinie, der direkt zusammenhängt. Wie in Abschnitt 2.1.1 dargestellt sind alle Stationen des Fließsystems logisch und zeitlich gekoppelt, es kann jedoch im Layout physikalische Trennungen zur Optimierung des Platzbedarfs geben.

sprechend sind Arbeitsplätze, die keinen linken bzw. rechten Driftbereich besitzen, *links-* bzw. *rechtsgeschlossen*.

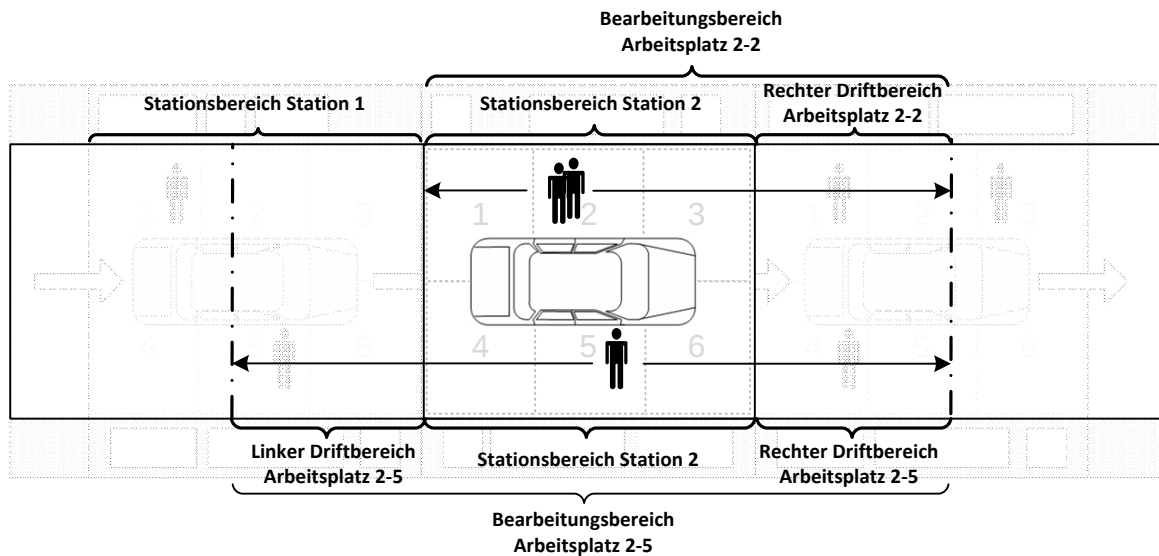


Abbildung 2.4: Variable Bearbeitungsbereiche durch die Einführung von Driftbereichen

Der **Bearbeitungsbereich** eines Arbeitsplatzes ist, wie in Abbildung 2.4 dargestellt, definiert durch den Bereich vom Beginn des linken Driftbereiches bis zum Ende des rechten Driftbereiches. Durch die Verwendung von Driftbereichen entstehen einige Besonderheiten, die im Folgenden diskutiert werden sollen.

Reihenfolgeabhängige Positionen der Werker Durch die Einführung von Driftbereichen kann nicht mehr davon ausgegangen werden, dass der Werker seine Arbeit bei jedem Werkstück wieder an der linken Stationsgrenze aufnimmt. Die Position der Werker im Zeitverlauf ist abhängig von den Bearbeitungszeiten der zu produzierenden Werkstücke und ihrer Reihenfolge. In Abbildung 2.5 wird dargestellt, wie ein Werker zeitweise aus seinem Stationsbereich herausdriftet.

Abhängigkeiten zwischen Werkern Wenn ein Werker seine Driftbereiche nutzt, werden zwangsläufig fremde Stationsbereiche verwendet. Hierdurch kann ein Verzug durch Abhängigkeiten zwischen den Arbeitsplätzen mit folgenden Gründen auftreten:⁵⁰

- Keine Überholmöglichkeit aufgrund des Platzbedarfs der Betriebsmittel (z.B. durch einen Werkstattwagen)

⁵⁰Vgl. "Interdependenzen zwischen Arbeitsgängen" HWProd [KSW96, S.188f.]

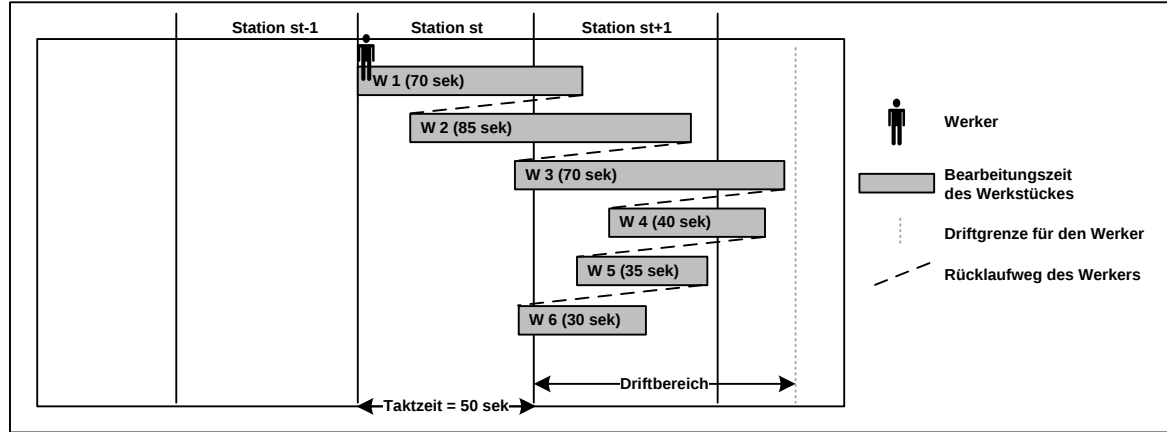


Abbildung 2.5: Beispiel für die reihenfolgeabhängige Position der Werker

- Prinzipielles Verbot des Überholens zur Vermeidung einer Motivationsminderung des Werkers am zu überholenden Arbeitsplatz
- Technische Vorrangbedingungen der Arbeitsvorgänge verhindern einen Überholvorgang
- Arbeit an der gleichen Ansprechzone⁵¹ macht eine Parallelarbeit unmöglich

Diese Abhängigkeiten sind je Arbeitsplatz ap durch eine Menge an Vorgängerarbeitsplätzen $vg \in \mathcal{VG}_{ap}$ gegeben, welche die Arbeit an Arbeitsplatz ap verzögern können. Dabei müssen sich die Vorgängerarbeitsplätze, zeitlich gesehen, vor dem abhängigen Arbeitsplatz (2.6),(2.7) befinden. Abhängigkeiten können nur zwischen Arbeitsplätzen auf derselben Seite des Fließbandes existieren (2.8).

$$st_{vg} \leq st_{ap} \quad \forall ap \in \mathcal{AP}, vg \in \mathcal{VG}_{ap} \quad (2.6)$$

$$sp_{vg} < sp_{ap} \quad \forall ap \in \mathcal{AP}, vg \in \mathcal{VG}_{ap} \mid st_{ap} = st_{vg} \quad (2.7)$$

$$\mathcal{VG}_{ap} \subset \mathcal{AP} \quad \forall ap \in \mathcal{AP}, vg \in \mathcal{VG}_{ap} \mid s_{vg} = s_{ap} \quad (2.8)$$

Des Weiteren darf der Driftbereich eines Vorgängerarbeitsplatzes nicht hinter dem Driftbereich eines davon abhängigen Arbeitsplatzes ap enden.⁵²

$$dbr_{vg} - (st_{ap} - st_{vg}) \cdot tz \leq dbr_{ap} \quad \forall ap \in \mathcal{AP}, vg \in \mathcal{VG}_{ap} \quad (2.9)$$

⁵¹Vgl. Abschnitt 2.1.1

⁵²Dies würde sonst dazu führen, dass die Arbeit des Werkers an Arbeitsplatz ap in bestimmten Situationen zwangsläufig nicht mehr durchgeführt werden kann.

Ein Beispiel für eine Verzögerung, die durch Abhängigkeiten erzeugt wird, findet sich in Abbildung 2.6, in der der Arbeitsplatz *AP2* seine Arbeit an dem Werkstück erst aufnehmen kann, nachdem der Vorgängerarbeitsplatz *AP1* aus der davor liegenden Station seine Arbeit beendet hat.

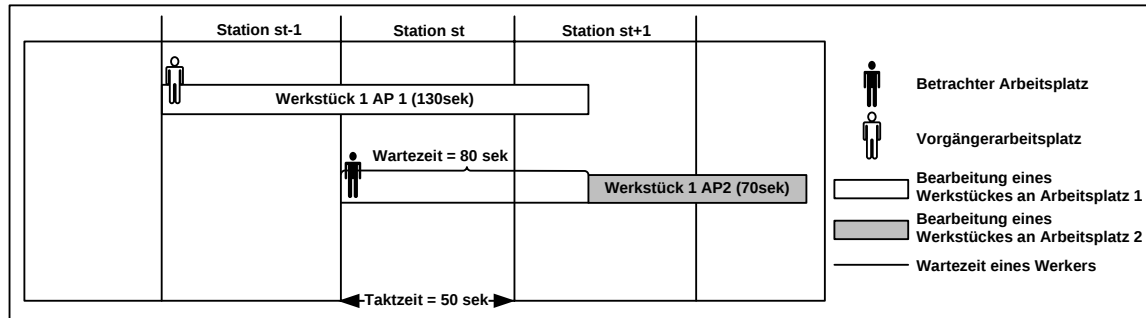


Abbildung 2.6: Beispiel für den Verzug durch Abhängigkeiten zwischen Werkern

2.1.2.2.2 Kurzfristige Kapazitätserweiterung durch den Einsatz von Unterstützern Wird ein Werker über den Bearbeitungsbereich eines Arbeitsplatzes hinaus belastet, so kommt es zu einer Überlastung⁵³. Eine Möglichkeit zur Vermeidung dieser Überlastungen stellt die kurzfristige Kapazitätserweiterung durch den Einsatz von sog. Unterstützern dar. Ein **Unterstützer**⁵⁴ ist eine zusätzliche, flexible Arbeitskraft, die innerhalb einer Fließlinie für die Abarbeitung von Überlastungen vorgehalten wird. Bei der Arbeit von Unterstützern wird davon ausgegangen, dass diese keine anderen Werker behindern.⁵⁵ Organisatorisch können diese kurzfristigen Kapazitätserweiterungen auf zwei Arten durchgeführt werden:

Reaktiver Unterstützereinsatz Beim reaktiven Unterstützereinsatz beginnt der Unterstützer mit seiner Arbeit, wenn ein Werkstück nicht innerhalb des vorgegebenen Bearbeitungsbereichs eines Arbeitsplatzes vollständig bearbeitet werden kann. In dieser Situation übernimmt der Unterstützer das Werkstück ab der Driftgrenze dbr_{ap} und führt alle restlichen Arbeitsvorgänge des Arbeitsplatzes ap aus.⁵⁶ Der Werker selbst kann zu diesem Zeitpunkt bereits zum nächsten Werkstück zurückgehen und parallel mit dessen Bearbeitung beginnen. In Abbildung 2.7 wird dies an einem Beispiel verdeutlicht.

⁵³Eine **Überlastung** tritt auf, wenn Arbeitsvorgänge außerhalb des erlaubten Bearbeitungsbereiches eines Arbeitsplatzes durchgeführt werden müssen.

⁵⁴Synonyme: Springer, Spitzenbrecher

⁵⁵Diese Annahme wird getroffen, da ein Unterstützer im Vergleich zum Werker z.B. keinen Werkstattwagen, der nicht überholt werden kann, mit sich führt, sondern die notwendigen Betriebsmittel nur kurzfristig bei sich trägt. Des Weiteren kann ein Unterstützer seine Arbeit kurz unterbrechen, wenn der nachfolgende Werker Arbeiten an derselben Ansprechzone des Werkstückes zu verrichten hat.

⁵⁶Vgl. zu dieser Organisationsform des Unterstützers existierende Definitionen in [Mol96],[Boy05b, S.218] und [Boc99, S.96]

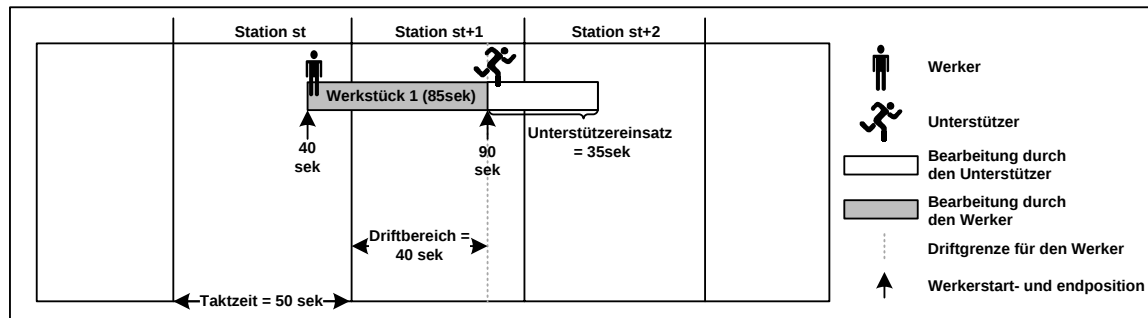


Abbildung 2.7: Beispiel für einen reaktiven Unterstützereinsatz

Präventiver Unterstützereinsatz Beim präventiven Unterstützereinsatz wird ein Unterstützer genau dann eingesetzt, wenn ein Werkstück den Werker über seinen Bearbeitungsbereich hinaus belasten würde. Es wird also präventiv vor der Überlastungssituation eingegriffen. Das Werkstück wird in diesem Fall vollständig vom Unterstützer übernommen. Im Beispiel in Abbildung 2.8 würde das Werkstück 2 den Werker über seine Driftgrenze hinaus belasten, daher übernimmt ein Unterstützer die Bearbeitung des gesamten Werkstückes in dem Arbeitsplatz und der Werker kann direkt zu Werkstück 3 zurücklaufen und dieses bearbeiten. Beim präventiven Unterstützereinsatz wird davon ausgegangen, dass der Unterstützer seine Arbeit am Werkstück bereits an der linken Stationsgrenze beginnen kann.

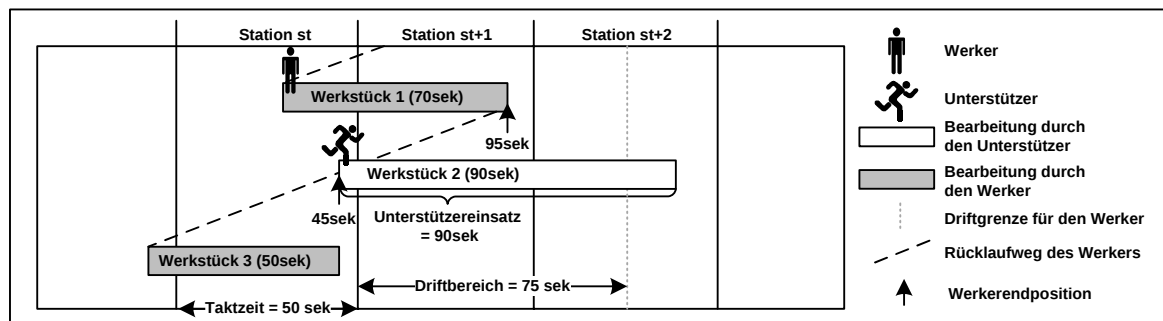


Abbildung 2.8: Beispiel für einen präventiven Unterstützereinsatz

Vor- und Nachteile der Organisationsformen von Unterstützern Beim „reaktiven Unterstützereinsatz“ wird nur die Überlastung durch den Unterstützer abgebaut, daher führt ein weiteres Werkstück mit hoher Bearbeitungszeit direkt zu einer erneuten Überlastung. Arbeitsvorbereitende Maßnahmen, wie z.B. das Einsteigen in ein Fahrzeug, müssen dadurch sehr häufig für wenig Arbeitsvorgänge durchgeführt werden. Der Einsatz der Unterstützer kann durch den Werker selbst gesteuert werden, indem er ein Signal absetzt, wenn er sich seiner Driftgrenze nähert.

Im Vergleich dazu wird bei der Anwendung des „präventiven Unterstützereinsatzes“ die Position des Werkers durch einen Unterstützereinsatz nachhaltiger beeinflusst, da die Bearbeitung eines ganzen Werkstückes abgenommen wird. Hierdurch kann sich jedoch eine zwangsläufige Leerzeit für den betroffenen Werker ergeben, da er das darauf folgende Werkstück erst übernehmen kann, wenn es sich in seinem linken Driftbereich befindet. Diese Gefahr besteht insbesondere, wenn die Driftbereiche sehr klein sind. Der Vorteil dieser Vorgehensweise ist, dass arbeitsvorbereitende Maßnahmen seltener durchgeführt werden müssen. Zur Realisierung des präventiven Unterstützereinsatzes muss eine Art Frühwarnsystem o.ä. existieren, welches die Bearbeitungszeiten der zukünftigen Werkstücke analysiert und die Unterstützer frühzeitig platziert.

Die Auswahl der in einem Praxisfall anzuwendenden Organisationsform hängt stark von den Gegebenheiten, wie z.B. der Menge an arbeitsvorbereitenden Maßnahmen und der Größe der Driftbereiche, ab. In dieser Arbeit sollen daher beide Alternativen betrachtet werden.

2.1.2.2.3 Kurzfristige Kapazitätserweiterung durch kurzzeitigen Bandstopp Eine weitere Möglichkeit, Überlastungen zu vermeiden, ist der Bandstopp. Die Fließlinie wird bei der Überlastung eines Werkers so lange angehalten, bis die Belastungsspitze abgearbeitet ist. Bei einem starr verketteten Fließsystem würde dies Leerzeit für alle nicht überlasteten Werker bedeuten. Die Kosten eines solchen Bandstopps je Sekunde lassen sich abschätzen, indem der durchschnittliche Produktwert durch die Taktzeit geteilt wird. Ein Einsatz dieses Mittels wäre nur sinnvoll, wenn die Kosten für Unterstützereinsätze höher wären als die Folgekosten der notwendigen Bandstopps, wie z.B. in extrem kleinen Montagen. Dieses Vorgehen wird daher in dieser Arbeit nicht weiter betrachtet.⁵⁷

2.1.3 Kostenstruktur in einer getakteten Variantenfließlinie

Im Folgenden werden alle Kosten diskutiert, die im Rahmen einer Kapazitätsabstimmung relevant und beeinflussbar sind.⁵⁸ Sie lassen sich aufteilen in Betriebs- und Rekonfigurationskosten. Zur Formalisierung der Kostenstruktur wird die Notation um die Bezeichner in Tabelle 2.3 erweitert.

⁵⁷Dieses Vorgehen sollte nicht verwechselt werden mit dem sog. „Qualitätsstopp“, der insbesondere in der Automobilindustrie Einzug gehalten hat. Hier wird ein Bandstopp eingeleitet, wenn es zu einem Fehler in der Montage kommt. Insbesondere die Toyota Werke haben dieses Vorgehen eingeführt und konnten die Montagequalität deutlich erhöhen [Mon98].

⁵⁸Kosten, auf die die Kapazitätsabstimmung keinen Einfluss hat, sind z.B. die Investitionskosten zur Erstinstallation der Fließlinie oder Kosten der werksexternen Logistik.

Symbol	Definition
$\Delta\mu_{ap}$	Taktausgleich an Arbeitsplatz $ap \in \mathcal{AP}$
$\mathcal{K}^{NGes}$	Gesamtkosten, welche durch zusätzliche Nacharbeitskosten auftreten, die auf Unterstützereinsätze zurückgeführt werden können
$\mathcal{K}^N$	Kostensatz für Nacharbeit
$\mathcal{K}^W$	Kostensatz für einen Werker
$\mathcal{K}^{WGes}$	Gesamtkosten für die Werker
$\mathcal{K}^U$	Kostensatz für einen Unterstützer
$\mathcal{K}^{UGes}$	Gesamtkosten für die Unterstützer
$\mathcal{K}^{RGes}$	Gesamtkosten, die durch die Rekonfiguration und damit durch die Verschiebung von Arbeitsvorgängen verursacht werden
$\mathcal{K}_{avo,ap}^R$	Kostensatz für die Verschiebung eines Arbeitsvorganges $avo \in \mathcal{AVO}$ auf einen Arbeitsplatz $ap \in \mathcal{AP}$
l_z	Summierte Leerzeit der Werker
$v_{avo,ap}^R$	Binär: 1, wenn Arbeitsvorgang $avo \in \mathcal{AVO}$ in der neuen Zuordnung zu einem neuen Arbeitsplatz $ap \in \mathcal{AP}$ verschoben wurde (Verschiebungsvariable)
sit	Situation im Fließsystem. Definiert durch ein Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ innerhalb einer bestimmten Station $st \in \mathcal{ST}$
t	Takt $t \in \mathcal{T}$
$t_{idx,ap}$	Taktindex des Taktes, in dem sich das Werkstück $p \in \mathcal{P}$ mit der Reihenfolgeposition $idx \in \mathcal{RF}$ in der Station $st_{ap} \in \mathcal{ST}$ von Arbeitsplatz $ap \in \mathcal{AP}$ befindet
ts	Anzahl von gleichzeitigen Situationen sit innerhalb jedes Taktes $t \in \mathcal{T}$
u^{MAX}	Maximale Anzahl gleichzeitiger Unterstützereinsätze $\hat{=}$ Notwendige Anzahl an Unterstützereinsatzkräften
U^{Zeit}	Summierte Unterstützereinsatzzeit
U^E	Gesamtanzahl Unterstützereinsätze
ζ_{ap}	Kapazität des Arbeitsplatzes $ap \in \mathcal{AP}$ in Zeiteinheiten

Tabelle 2.3: Erweiterung der Notation zur Formalisierung der Kostenstruktur in einer Variantenfließlinie

2.1.3.1 Bestimmung der Betriebskosten

Die Betriebskosten einer getakteten Variantenfließlinie beinhalten Kosten für die menschliche Arbeitskraft und die Verwendung von Betriebsmitteln.⁵⁹ Eine Kapazitätsabstimmung kann keinen Einfluss auf die Länge der Verwendung dieser Ressourcen nehmen, da die summierte Fertigungszeit vorher festgelegt ist. Allerdings können die Arbeitsvorgänge durch eine unterschiedliche Anzahl an Workern oder durch teurere Unterstützerarbeitskräfte durchgeführt werden. Des Weiteren kann ein zusätzliches Pensum an Nacharbeit an den Produkten notwendig werden, wenn ungünstige Belastungssituationen zu einer verminderten Montagequalität führen. Diese Kosten werden im Folgenden genauer systematisiert.

⁵⁹Vgl. [Got06b, S.28]

2.1.3.1.1 Kosten durch Werker Werker werden einem Arbeitsplatz fest zugewiesen und führen die dem Arbeitsplatz zugewiesenen Arbeitsvorgänge aus. Die Kapazität eines Arbeitsplatzes ζ_{ap} ist definiert durch Gleichung 2.10.

$$\zeta_{ap} = w_{ap} \cdot tz \quad (2.10)$$

Werker werden pauschal für den gesamten Produktionszeitraum entlohnt.⁶⁰ Unterschiedliche Kosten für verschiedene Tätigkeiten sind daher nicht notwendig. Zur Formalisierung der Kosten wird der Kostensatz $\mathcal{K}^W$ eingeführt, welcher die Kosten für die Bezahlung eines Werkers für einen Produktionszeitraum wiedergibt. Gleichung 2.11 bestimmt die Gesamtkosten für die Werker $\mathcal{K}^{WGes}$.

$$\mathcal{K}^{WGes} = \sum_{ap \in \mathcal{AP}} w_{ap} \cdot \mathcal{K}^W \quad (2.11)$$

2.1.3.1.2 Kosten durch Unterstützer Im Vergleich zu den Werkern sind Unterstützer für einen relativ großen Bereich einer Fließlinie verantwortlich und müssen sämtliche Arbeitsvorgänge dieses Bereiches beherrschen. Hierdurch ergibt sich die Notwendigkeit einer umfangreichen Ausbildung und somit auch einer höheren Entlohnung.⁶¹ Die Kosten für den Einsatz von Unterstützern können unterschiedlich bestimmt werden. Eine Möglichkeit ist es, die Einsatzzeit der Unterstützer zu ermitteln und diese in der Kostenbetrachtung zu berücksichtigen.⁶² Um die notwendigen Unterstützereinsätze innerhalb eines Produktionszeitraumes durchführen zu können, müssen über den gesamten Produktionszeitraum hinweg Unterstützerarbeitskräfte vorgehalten werden. Somit muss, selbst wenn nur ein einzelner Einsatz in der Planungsperiode notwendig ist, über die gesamte Zeit ein Unterstützer bezahlt werden. Die Anzahl an vorgehaltenen Unterstützern in einer Planungsperiode muss mindestens so groß sein wie die maximale Anzahl gleichzeitiger Unterstützereinsätze. Selbst wenn nur zwei Einsätze ungünstigerweise gleichzeitig stattfinden, müssen bereits zwei Unterstützer vorgehalten werden. Des Weiteren lässt sich keine unterschiedliche Entlohnung von Leerzeiten und Einsatzzeiten der Unterstützer definieren, da die exakte Einsatzzeit nicht feststeht, bevor das Gehalt eines Unterstützers fixiert wird.⁶³

⁶⁰Vgl. im Gegensatz dazu die Berechnung durch [Boc99, S.96], welcher je nach Verrichtungsart von Arbeitsvorgängen unterschiedliche Kostensätze verwendet und den Lohnkostensatz innerhalb einer Station standardisiert.

⁶¹Vgl. z.B. [Boc99, S.96]

⁶²Vgl. [Mol96]

⁶³Vgl. im Gegensatz dazu z.B. [Boc99, S.96], wo unterschiedliche Kostensätze definiert werden.

Zur korrekten Bestimmung der notwendigen Anzahl an Unterstützern in einem Produktionszeitraum wird im ersten Schritt der zeitliche Zusammenhang in einem Fließsystem verdeutlicht, um im zweiten Schritt die Gleichzeitigkeit zu definieren.

Zeitlicher Zusammenhang im Fließsystem Befindet sich das erste Werkstück in Station 2, so befindet sich das zweite Werkstück in Station 1. Allgemein beschrieben befindet sich das Werkstück mit der Reihenfolgeposition idx in Station st , wenn sich das Werkstück mit der Reihenfolgeposition $idx + n$ in der Arbeitsstation $st - n$ befindet. Dies beschreibt zwei Situationen, die im selben Takt t stattfinden.

Eine **Situation** ist definiert durch ein bestimmtes Werkstück an Reihenfolgeposition idx , das sich in Station st befindet. Der Takt t einer Situation kann mit Hilfe der Reihenfolgeposition eines Werkstückes und der Arbeitsplatznummer ermittelt werden (2.12). In Abbildung 2.9 entspricht ein Feld in der Matrix einer Situation.

$$t_{idx,ap} = idx + st_{ap} - 1 \quad (2.12)$$

Im Beispiel in Abbildung 2.9 sind gleichzeitige Situationen dadurch gekennzeichnet, dass sie auf einer Diagonalen liegen.⁶⁴ Dies entspricht einer Diskretisierung der Zeit in Taktlängen.

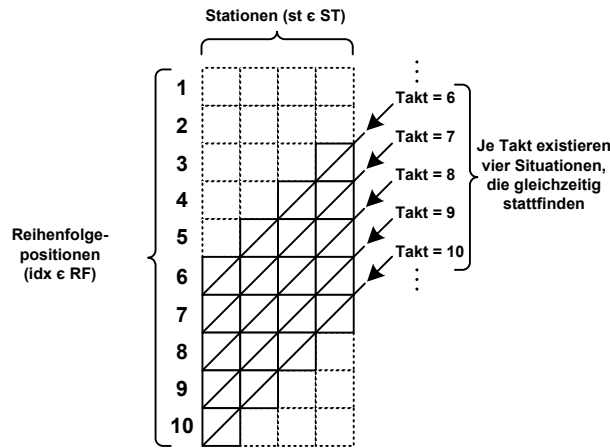


Abbildung 2.9: Darstellung von gleichzeitigen Situationen im Fließsystem

Gleichzeitigkeit von Unterstützereinsätzen Zwei Unterstützereinsätze finden **gleichzeitig** statt, wenn sie in gleichzeitigen Situationen und damit im selben Takt durchgeführt werden.

⁶⁴Für diese Definition ist es zwingend erforderlich, dass die Stationen gleich lang und ohne Unterbrechungen starr verkettet sind. Vgl. dazu die Definition in Abschnitt 2.1.1

Dabei wird davon ausgegangen, dass ein Unterstützer, der innerhalb eines Taktes t die Aufgaben eines Arbeitsplatzes ap übernimmt, an keinem anderen Arbeitsplatz aushelfen kann, unabhängig davon, wie lange er innerhalb eines Taktes t beschäftigt ist.

In Gleichung 2.13 wird die Obergrenze für die Anzahl an gleichzeitigen Unterstützereinsätzen definiert; diese entspricht der Anzahl möglicher gleichzeitiger Situationen ts innerhalb eines Taktes und damit der Länge der Diagonalen in Abbildung 2.9. Die Gesamtanzahl an Situationen im Fließsystem sit in einem Planungszeitraum ist gegeben durch die Anzahl an Feldern in der Matrix in Gleichung 2.14.

$$ts = \min[|\mathcal{RF}|; |ST|] \quad (2.13)$$

$$sit = |\mathcal{RF}| \cdot |ST| \quad (2.14)$$

Zu beachten ist, dass die Unterstützereinsätze sich nicht auf einen Takt beschränken müssen, wie im Beispiel in Abbildung 2.10 gezeigt.⁶⁵

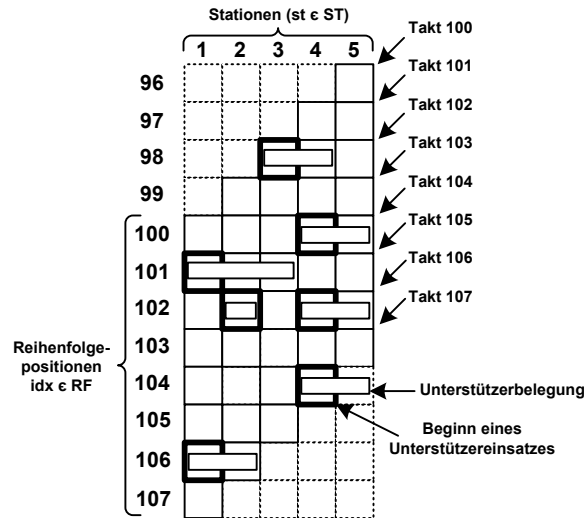


Abbildung 2.10: Beispiel für die Belegung von Unterstützern über mehrere Takte hinweg

Die beiden alternativen Organisationsformen von Unterstützern haben unterschiedlichen Einfluss auf die Notwendigkeit und die Gleichzeitigkeit von Unterstützereinsätzen.⁶⁶

⁶⁵Vgl. im Gegensatz dazu die Definition von Bock in [Boc99], der davon ausgeht, dass die Arbeit an einem Werkstück grundsätzlich innerhalb der Stationsgrenzen beendet werden kann, wenn ein Unterstützer zu Hilfe kommt.

⁶⁶Vgl. Abschnitt 2.1.2.2.2: Bei präventiver Einsatzform kann der Unterstützer an der linken Stationsgrenze den Auftrag übernehmen, während er bei reaktivem Einsatz erst am Ende der Driftgrenze seine Arbeit beginnen kann.

Ableitung der Kosten Entscheidend für die notwendige Anzahl an vorzuhaltenden Unterstützern ist die maximale Anzahl an gleichzeitigen Unterstützereinsätzen u^{MAX} . Diese wird mit dem Kostensatz für Unterstützer $\mathcal{K}^U$ multipliziert, um die Gesamtkosten für Unterstützer zu ermitteln (2.15).

$$\mathcal{K}^{UGes} = \mathcal{K}^U \cdot u^{MAX} \quad (2.15)$$

2.1.3.1.3 Kosten durch Leerzeit Entstehen Zustände, in denen der Werker keine Arbeitsvorgänge bearbeitet, so bedeutet dies, dass bezahltes Produktionspotential ungenutzt bleibt. Zur Herleitung dieser Werkerleerzeiten im Produktionsprozess wird im ersten Schritt der Taktausgleich definiert.

Der **Taktausgleich** beschreibt die Differenz zwischen der Kapazität eines Arbeitsplatzes und der durchschnittlichen Auslastung eines Arbeitssystems und wird im Verhältnis zur Kapazität angegeben (2.16).

$$\Delta\mu_{ap} = 1 - \frac{\sum_{p \in \mathcal{P}} \tau_{p,ap}}{|\mathcal{P}| \zeta_{ap}} \quad \forall \quad ap \in \mathcal{AP} \quad (2.16)$$

Wenn ein Produktionssystem im Durchschnitt zu 95 Prozent ausgelastet ist, existieren in Summe 5 Prozent Taktausgleich. Existieren keine Unterstützereinsätze, so entspricht die summierte Leerzeit der Werker dem Taktausgleich. Zusätzlich kann durch Belastungsschwankungen und ungünstige Produktionsreihenfolgen ein Anteil an vermeidbarer Leerzeit entstehen, der sich an anderer Stelle in der Produktion durch Unterstützereinsätze äußert. Der Zusammenhang wird in Grafik 2.11⁶⁷ verdeutlicht. Dabei tritt Leerzeit entweder auf, wenn ein Werkstück noch durch einen anderen Werker blockiert ist oder es sich noch nicht im Bearbeitungsbereich des Werkers befindet.

Die Unterstützereinsatzzeit entspricht der zusätzlich zum Taktausgleich existierenden Werkerleerzeit im System. Wird eine Minimierung der Unterstützereinsätze durchgeführt, bedingt dies eine Erhöhung der durchschnittlichen Werkerauslastung und damit eine Minimierung der summierten Leerzeit in Gleichung 2.17.

$$lz = \sum_{ap \in \mathcal{AP}} (\overbrace{\Delta\mu_{ap} \cdot tz}^{\text{Taktausgleich}}) + \overbrace{U^{Zeit}}^{\text{Unterstützereinsatzzeit}} \quad (2.17)$$

⁶⁷Erweitert in Anlehnung an [Mol96]

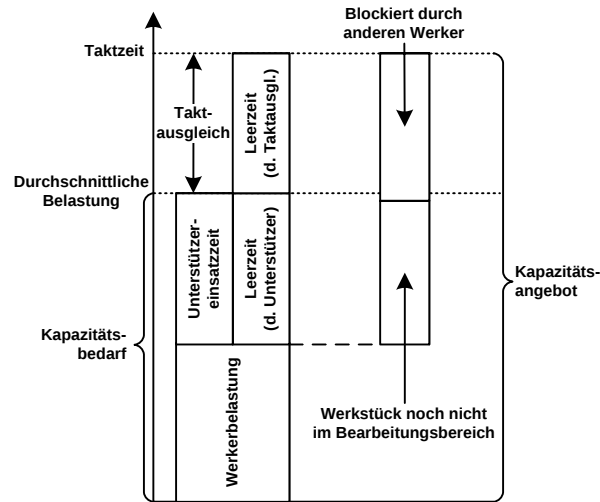


Abbildung 2.11: Verschiebung des Kapazitätsbedarfs von Werkern auf Unterstützer

Durch Leerzeit entstehen keine weiteren Kosten im Fließsystem. Die Kosten für Werker und Unterstützer beinhalten implizit die Auswirkungen von Leerzeit. Eine explizite Betrachtung der Leerzeit in den Kosten ist somit nicht notwendig.

2.1.3.1.4 Kosten durch Nacharbeit Eine erhöhte Fehlerrate kann unter anderem auf folgende Gründe zurückgeführt werden:⁶⁸

- Stresssituationen durch Überlastung der Werker und Ausgleich durch schnelleres Bearbeiten der Arbeitsvorgänge
- Aufträge, die durch Unterstützer bearbeitet werden, denen die Routine bei gewissen Arbeitsvorgängen fehlt
- Erhöhte Leerzeit der Werker, die zu abnehmender Konzentration führt

Eine genaue Bestimmung der Abhängigkeit von Nacharbeit und Güte der Kapazitätsabstimmung ist schwer durchzuführen. Jedoch zeigt sich eine Korrelation zwischen einer erhöhten Fehlerrate in der Montage und den aufgezählten Gründen.⁶⁹ Sie entstehen allesamt durch Überlastungen der Werker und die daraus resultierenden Unterstützereinsätze.⁷⁰ Zur Abschätzung der zu erwartenden erhöhten Nacharbeitskosten wird ein linearer Kostenzusammenhang unterstellt und die Anzahl an notwendigen Unterstützereinsätzen u^E mit dem Kostenfaktor $\mathcal{K}^N$ bewertet (2.18).

⁶⁸Vgl. zum Zusammenhang von Leerzeit und Überlastung mit schlechter Montagequalität [Sys07]

⁶⁹Vgl. [Sys07]

⁷⁰Dies trifft auch auf die vermeidbare Leerzeit zu, die sich in gleichem Maße in Unterstützereinsätzen darstellt. Vgl. Abschnitt 2.1.3.1.3

$$\mathcal{K}^{NGes} = \mathcal{K}^N \cdot u^E \quad (2.18)$$

2.1.3.2 Bestimmung der Rekonfigurationskosten

Im Rahmen der Kapazitätsabstimmung von Variantenfließlinien sind keine direkten Investitionskosten wie bei der Erstinstallation einer Fließlinie zu beachten. Zum Betrachtungszeitpunkt existiert bereits eine Fließlinie mit den Betriebsmitteln, Stationen und Arbeitsplätzen.⁷¹ Im Rahmen der Kapazitätsabstimmung soll eine regelmäßige Anpassung der Zuweisung von Arbeitsvorgängen zu Arbeitsplätzen durchgeführt werden.⁷² Durch die Verschiebung von Arbeitsvorgängen entstehen folgende Kosten:

- Kosten für Qualifizierungsmaßnahmen durch neue Kombinationen von Arbeitsvorgängen auf einem Arbeitsplatz, die durch den Werker beherrscht werden müssen.
- Kosten für Umbaumaßnahmen von Betriebsmitteln, die durch das Verschieben von Arbeitsvorgängen nötig werden.
- Anschaffungskosten für zusätzliche Betriebsmittel, die an weiteren Arbeitsplätzen benötigt werden, wenn zwei Arbeitsvorgänge, die dasselbe Betriebsmittel benötigen, zwei unterschiedlichen Arbeitsplätzen zugeordnet werden.
- Erweiterung von Lagerfläche am Montageband, die durch die neue Zuordnung eines Arbeitsvorganges notwendig wird.
- Erhöhte Fehlerrate und verringertes Kapazitätsangebot durch Verlust von Lerneffekten.⁷³

Je nach verschobenem Arbeitsvorgang sind ein oder mehrere der angesprochenen Punkte relevant. Ein Kostensatz $\mathcal{K}_{avo,ap}^R$ beschreibt die Kosten, die anfallen, wenn der Arbeitsvorgang *avo* an einen Arbeitsplatz *ap* verschoben wird. Darin sind alle oben beschriebenen Kosten enthalten. Handelt es sich um Kosten, die sich über eine längere Zeit auswirken, wie z.B. die Lerneffekteinbußen, so sind diese als Fixkosten mit im Kostensatz enthalten. Die binäre Variable $v_{avo,ap}^R$ beschreibt eine durchgeführte Verschiebung. Die Gesamtkosten, die durch eine Rekonfiguration entstehen, sind in Gleichung 2.19 definiert.

$$\mathcal{K}^{RGes} = \sum_{avo \in \mathcal{AVO}} \sum_{ap \in \mathcal{AP}} \mathcal{K}_{avo,ap}^R \cdot v_{avo,ap}^R \quad (2.19)$$

⁷¹Die Weiterentwicklung der Produkte und damit u.U. Erweiterung der angebotenen Varianten kann eine neue Investition notwendig machen. Diese muss jedoch unabhängig von der Kapazitätsabstimmung getätigt werden und wird daher hier nicht quantifiziert.

⁷²Vgl. zum genauen Vorgehen Abschnitt 2.2.1

⁷³Vgl. [Got06a, S.30]

2.2 Systematisierung der Aufgaben zur Kapazitätsabstimmung in einer getakteten Variantenfließlinie

Die Kapazitätsabstimmung beinhaltet nach Bullinger⁷⁴ allgemein eine Kapazitätsanpassung und einen Kapazitätsabgleich. Dabei werden die Kapazitätsnachfrage⁷⁵ und das Kapazitätsangebot⁷⁶ aufeinander abgestimmt. Bei der Kapazitätsanpassung geht man von einer gegebenen Belastung aus und ändert die Kapazität, wohingegen beim Kapazitätsabgleich davon ausgegangen wird, dass feste Kapazitäten existieren und die Belastung abgeglichen wird.⁷⁷ Wird diese Definition auf die getaktete Variantenfließlinie übertragen, so wird die Belastung durch die zu produzierenden Werkstücke und die verfügbare Kapazität durch das Personal und die Betriebsmittel definiert.

Zeitlich wird die Kapazitätsabstimmung der Produktionssteuerung zugesprochen und findet nach der Produktionsprogrammplanung⁷⁸ statt, welche die Auftragsermittlung sowie deren Zuordnung zu Produktionszeiträumen beinhaltet.⁷⁹ Die Zusammensetzung der Aufträge und somit die einzelnen notwendigen Arbeitsvorgänge je Werkstück sind durch die Produktionsprogrammplanung bereits fest definiert und lassen sich im Rahmen der Kapazitätsabstimmung nicht mehr ändern. Ebenfalls sind das Layout der Fließlinie sowie die Anzahl an physikalischen Stationen bereits bei der Erstinstallation der Fließlinie mit Hilfe von Methoden zur Fließbandabstimmung fixiert worden.⁸⁰

Die Entscheidungsvariablen einer Kapazitätsabstimmung beinhalten die Anzahl der Werker und Unterstützer sowie die Art und Menge der Betriebsmittel. Zusätzlich kann eine Kapazitätsstrukturierung durchgeführt werden. Dabei wird die Zuordnung von Arbeitsvorgängen, Werkern und Betriebsmitteln zu Arbeitsplätzen vorgenommen sowie eine Reihenfolgeplanung

⁷⁴Vgl. [Bul86, S.38]

⁷⁵Die **Kapazitätsnachfrage** in einer getakteten Variantenfließlinie ist definiert durch die Aufträge, die in einem Produktionszeitraum gefertigt werden sollen, sowie durch die Strukturierung dieser Nachfrage, gegeben durch die Zuordnung von Arbeitsvorgängen auf Arbeitsplätze und die Zuordnung von Aufträgen auf Reihenfolgepositionen.

⁷⁶Das **Kapazitätsangebot** in einer getakteten Variantenfließlinie ist definiert durch die Anzahl an Werkern und Betriebsmitteln an den Arbeitsplätzen sowie durch die Anzahl an Unterstützearbeitskräften, die für eventuelle Überlastungen vorgehalten werden.

⁷⁷Vgl. [Bul86, S.38]

⁷⁸Aufgabe der **Produktionsprogrammplanung** ist es, in einem aggregierten Raster festzulegen, welches Produkt zu welchem Zeitpunkt hergestellt werden soll. [KMSO06] Dieser Planungsschritt lässt sich nach [Sch06] der operativen Planung zuordnen. Lautenschläger definiert den Planungsschritt in [Lau99, S.8] auf Basis von [Fle88, S.348] so: „In der Phase der Produktionsprogrammplanung, die auch als *Master Production Schedule* (MPS) bezeichnet wird, erfolgt die Erstellung eines Produktionsprogramms für Endprodukte oder Produktgruppen. Dazu wird von Absatzprognosen, eventuell bereits vorliegenden Kundenaufträgen, einem aggregierten längerfristigen Produktionsprogramm sowie von den Lagerbeständen der Produkte ausgegangen. Die Abstimmung des Kapazitätsangebots mit der Kapazitätsnachfrage wird nur mit stark aggregierten Kapazitätsprofilen für einen bestimmten Produktmix durchgeführt.“

⁷⁹Vgl. Abbildung in [Hac89]

⁸⁰Vgl. zur Übersicht [BFS08]

durchgeführt, in der die Werkstücke einzelnen Reihenfolgepositionen zugewiesen werden. Folgende Entscheidungsvariablen müssen in einer Kapazitätsabstimmung in Variantenfließlinien gesetzt werden:

1. Bestimmung der Anzahl der Werker und Betriebsmittel
2. Zuordnung von Werkern und Betriebsmitteln zu Arbeitsplätzen
3. Zuordnung von Arbeitsvorgängen zu Arbeitsplätzen
4. Bestimmung der Anzahl an vorzuhaltenden Unterstützerarbeitskräften
5. Zuordnung von Werkstücken zu Reihenfolgepositionen

Die zu treffenden Entscheidungen können hinsichtlich ihrer Fristigkeiten unterschiedlich zu Teilaufgaben der Kapazitätsabstimmung zusammengefasst werden. Die Entscheidungsvariablen eins bis drei sollten bei der Rekonfiguration für einen möglichst langfristigen Planungszeitraum fixiert werden, damit sich die zuvor diskutierten Rekonfigurationskosten amortisieren.⁸¹ Auf der anderen Seite sollte dieser Planungsschritt auf einer möglichst sicheren Datenbasis durchgeführt werden. Bestenfalls sollte hierfür der Zeitpunkt⁸² gewählt werden, an dem nicht mehr mit prognostizierten Aufträgen gearbeitet wird, sondern bereits echte Aufträge festliegen.⁸³ Die Entscheidungsvariablen vier und fünf müssen im Rahmen einer Reihenfolgeplanung auf Grund der späten Festlegung der Aufträge für einen Produktionszeitraum durch die Produktionsprogrammplanung und der Möglichkeit für späte Änderungen⁸⁴ relativ kurzfristig festgelegt werden. Für die Abstimmung mit Zulieferern und anderen vorgelagerten Produktionsstufen kann jedoch eine möglichst große Vorlaufzeit notwendig sein. Daher sollten die Entscheidungsvariablen vier und fünf gesetzt werden, sobald das Auftragsprogramm für den zu sequenzierenden Produktionszeitraum durch die Programmplanung festliegt.

In Abbildung 2.12 wird ein Beispiel für die Verfügbarkeit von Auftragsinformationen gegeben. In diesem Fall würde die Rekonfiguration nach der Festlegung des Auftragsbestandes für den nächsten Monat durchgeführt und die Reihenfolgeplanung jeweils nach Festlegung des Auftragsbestandes für den zehnt-nächsten Tag.

Insbesondere die unterschiedlichen Planungszeiträume der beiden Planungsvorgänge machen eine simultane⁸⁵ Lösung dieses Problems unmöglich.⁸⁶ Aus diesem Grund müssen die beiden

⁸¹Vgl. Abschnitt 2.1.3.2

⁸²Vgl. z.B. [Mey04, S.451], der diese Zeiträume für die Automobilindustrie definiert und [KMSO06, S.150ff], welche den Übergang von prognostizierten zu echten Aufträgen beschreiben.

⁸³Vgl. [Sch95, S.108] zum Re-Balancing

⁸⁴Vgl. [BRvB06, S.881] „In der Praxis wird es oft unmöglich einen früher errechneten Plan umzusetzen, da kurzfristige Änderungen die Daten, von denen ausgegangen wurde, verändern.“

⁸⁵Vgl. zur Definition der „Simultanplanung“ [Sch01, S.21]

⁸⁶Vgl. zu dieser Tatsache die Aussagen in [BFS07b], [Boc99, S.18ff.] und [Sch95, S.108ff.]

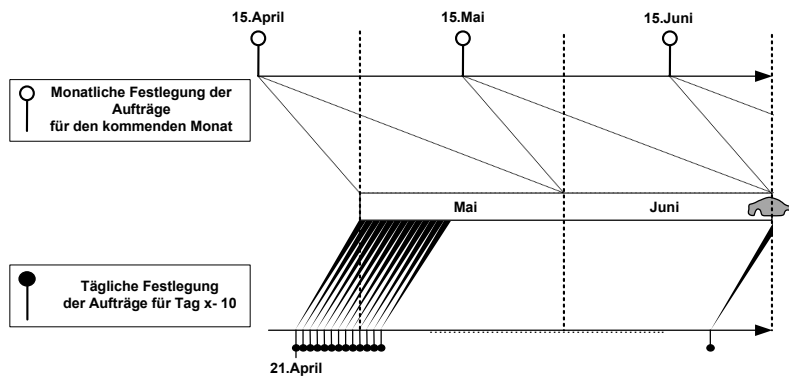


Abbildung 2.12: Schematisches Beispiel für die Verfügbarkeit von Auftragsdaten in der Automobilindustrie

Planungsschritte in einem hierarchischen Ansatz⁸⁷ integriert werden, in dem die Rekonfiguration die Auswirkungen auf der Reihenfolgeebene antizipiert⁸⁸ und die Reihenfolgeplanung die Entscheidungen aus der Rekonfiguration für die Planung verwendet.⁸⁹

Bei vielen in der Montageplanung existierenden Fragestellungen wird ein sog. Vorranggraph⁹⁰ verwendet, welcher die technischen Vorrangbeziehungen der einzelnen Arbeitsvorgänge wiedergibt.⁹¹ Insbesondere Methoden zur Erstinstallation von Fließlinien definieren diesen als Eingangsdatum.⁹² Die in dieser Arbeit zu entwickelnde Methode zur Rekonfiguration des Fließsystems kann mit Hilfe eines Vorranggraphen automatisiert neue Zuordnungslösungen finden. Wenn es um die Produktion von komplexen und variantenreichen Erzeugnissen geht, stellt die Forderung nach einem Vorranggraphen jedoch eine große Hürde dar.⁹³

„Die Abneigung potentieller Anwender gegen die Einführung von EDV-gestützten Verfahren zur Leistungsabstimmung resultiert aus der Tatsache, dass der zeitliche Gewinn, den diese Verfahren mit sich bringen, bei der Erstellung von Vorranggraphen verloren geht.“⁹⁴

⁸⁷Vgl. zur Definition der „Hierarchischen Planung“ [Sch01, S.35]

⁸⁸Vgl. Definition in [Pic75, S.53] nach welcher antizipativ als vorgehend definiert wird.

⁸⁹Vgl. hierarchische Modelle von [Sch95, S.108ff] und [BFS07b, S.4]

⁹⁰„Ein **Vorranggraph** $VRG = (KN, KA, t)$ ist ein gerichteter zyklensfreier Graph mit einer Menge $KN = \{1, \dots, n\}$ von Knoten, einer Kantenmenge und einem Vektor von Bewertungen. Die Knoten des Vorranggraphen stellen dabei die Arbeitsvorgänge dar, die an dem betrachteten Produkt zu leisten sind, während jede gerichtete Kante eine Vorrangfolge zwischen den Arbeitsvorgängen i und j in der Weise definiert, dass der Arbeitsvorgang i vor seinem Nachfolger j ausgeführt werden muss. Zudem legt die Bewertung t_i die Bearbeitungszeit für jeden Arbeitsvorgang i fest.“ in [Dan03, S.359]

⁹¹Bullinger et al. beschreiben den Nutzen eines Vorranggraphen in [BGLS95, S.94f.] folgendermaßen: „Der Montagevorranggraph ermöglicht es, die Montageablaufstruktur hinsichtlich der logisch-zeitlichen Abhängigkeiten abzubilden.“

⁹²Vgl. [BFS07a]

⁹³Vgl. Hardeck in [Har74], der sagt, dass die Hauptschwierigkeit bei der Fließbandabstimmung darin liegt, den Vorranggraphen zu bestimmen. Vgl. auch die Aussagen von [Amm85] und [Fal05]

⁹⁴Vgl. [Amm85]

Existierende Methoden zur manuellen Definition von Vorranggraphen werden bereits bei einer sehr geringen Anzahl an Produktvarianten zu komplex und fehleranfällig.⁹⁵ Daher wird in dieser Arbeit ein Verfahren benötigt, welches die Erstellung eines Vorranggraphen für variantenreiche Produkte ermöglicht.

Die zu lösenden Probleme zur Kapazitätsabstimmung von getakteten Variantenfließlinien lassen sich somit in drei Teilprobleme strukturieren, welche in den folgenden Abschnitten detailliert definiert werden:

- Rekonfiguration einer getakteten Variantenfließlinie
- Vorranggraphenentwicklung für variantenreiche Produkte
- Reihenfolgeplanung für eine getaktete Variantenfließlinie

2.2.1 Rekonfiguration einer getakteten Variantenfließlinie

Im Folgenden soll die Aufgabe der Rekonfiguration einer Fließlinie genauer definiert werden. Hierfür werden in drei Schritten das Ziel, die Restriktionen und die Voraussetzungen einer Rekonfiguration definiert. Zur Formalisierung wird die Notation um die Bezeichner in Tabelle 2.4 erweitert.

Symbol	Definition
$\mathcal{K}^{Ges}$	Gesamtkosten
$\mathcal{P}_{pz}$	Menge von Werkstücken, die im Produktionszeitraum $pz \in \mathcal{PZ}$ produziert werden
pl_{ap}	Zur Verfügung stehender Lagerplatz an Arbeitsplatz $ap \in \mathcal{AP}$
pb_{avo}	Notwendiger Lagerplatz für die Ausführung von Arbeitsvorgang $avo \in \mathcal{AVO}$
$\mathcal{WP}_{avo}$	Menge möglicher Werkstückpositionen $wp \in \mathcal{WP}$ für die Ausführung von Arbeitsvorgang $avo \in \mathcal{AVO}$. Es gilt: $\mathcal{WP}_{avo} \subseteq \mathcal{WP}$
$\mathcal{V}$	Menge von Vorrangbedingungen. Wenn $(i, j) \in \mathcal{V}$, dann muss Arbeitsvorgang $i \in \mathcal{AVO}$ abgeschlossen sein, bevor mit Arbeitsvorgang $j \in \mathcal{AVO}$ begonnen werden kann. Es gilt: $\mathcal{V} \subset \mathcal{AVO} \times \mathcal{AVO}$
$x_{avo,ap}^{Soll}$	Binär: 1, wenn Arbeitsvorgang $avo \in \mathcal{AVO}$ Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist; 0, sonst (neue Zuordnungslösung)

Tabelle 2.4: Erweiterung der Notation zur Formalisierung der Rekonfiguration

⁹⁵Vgl. [Wil95, S.309], welcher seine Methoden an Produkten mit 10-20 Vorgängen und einer Produktvariante testet. Vgl.[DFW87] zur Fehleranfälligkeit der manuellen Methoden

2.2.1.1 Ziel der Rekonfiguration

Die Rekonfiguration einer getakteten Variantenfließlinie muss folgende Entscheidungsvariablen definieren:⁹⁶

1. Bestimmung der Anzahl der Werker und Betriebsmittel
2. Zuordnung von Werkern und Betriebsmitteln zu Arbeitsplätzen
3. Zuordnung von Arbeitsvorgängen zu Arbeitsplätzen

Gegenstand der Rekonfiguration ist die Verschiebung von Arbeitsvorgängen von ihrem ursprünglich zugeordneten Arbeitsplatz $z_{avo,ap}^{Ist}$ auf einen anderen Arbeitsplatz $x_{avo,ap}^{Soll}$ zur Effizienzsteigerung in der Fließlinie. Referenz ist dabei immer die letzte gültige Zuordnung. Das Ergebnis einer Rekonfiguration ist eine Verschiebematrix, in der definiert ist, welche Arbeitsvorgänge wohin verschoben werden (2.20).

$$v_{avo,ap}^R \geq x_{avo,ap}^{Soll} - z_{avo,ap}^{Ist} \quad \forall \quad avo \in \mathcal{AVO}, ap \in \mathcal{AP} \quad (2.20)$$

Diese neue Zuordnung der Arbeitsvorgänge beinhaltet implizit auch die neue Zuordnung von Betriebsmitteln und Werkern zu Arbeitsplätzen.⁹⁷ Im Vergleich zur Erstinstallation einer Variantenfließlinie fallen zusätzlich zu den Betriebskosten auch Rekonfigurationskosten⁹⁸ an.⁹⁹ Prinzipiell kann festgestellt werden, dass mit jedem weiteren Arbeitsvorgang, der verschoben wird, weitere Umstellungskosten auftreten, jedoch auch mehr Potential besteht, die Betriebskosten zu reduzieren. Es entsteht somit ein Zielkonflikt zwischen der Minimierung von Umstellungskosten und der Minimierung von Betriebskosten. In Abbildung 2.13 wird dieser verdeutlicht.

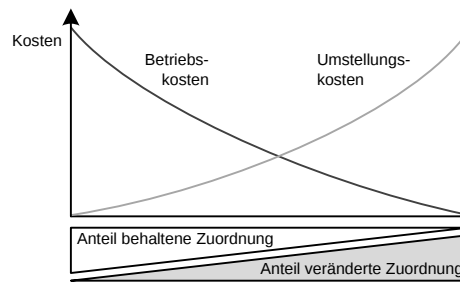


Abbildung 2.13: Zielkonflikt der Rekonfiguration

⁹⁶Vgl. Abschnitt 2.2

⁹⁷Vgl. Gleichung 2.5 in Abschnitt 2.1.2.1

⁹⁸Im Folgenden wird für Rekonfigurationskosten synonym „Umstellungskosten“ verwendet

⁹⁹Vgl. Abschnitt 2.1.3.2

2.2.1.1.1 Minimierung der Betriebskosten Die Betriebskosten, die im Rahmen einer Rekonfiguration beeinflusst werden können, beschränken sich auf die Anzahl der Werker im Fließsystem, welche sich direkt durch die durchschnittliche Auslastung der Arbeitsplätze ergeben¹⁰⁰, sowie die Anzahl an vorzuhaltenden Unterstützern und der zusätzliche Teil an Nacharbeit, der aus der Anzahl an Unterstützereinsätzen abgeleitet wird.¹⁰¹ In Gleichung 2.21 wird die Summe der Kosten ermittelt.

$$\mathcal{K}^U \cdot u^{MAX} + \mathcal{K}^N \cdot u^E + \sum_{ap \in \mathcal{AP}} \mathcal{K}^W \cdot w_{ap} \quad (2.21)$$

Eine exakte Ermittlung der notwendigen Unterstützereinsätze ist erst möglich, wenn die endgültige Produktionsreihenfolge bekannt ist. Die Rekonfiguration muss jedoch zu einem Zeitpunkt stattfinden, zu dem noch keine Produktionsreihenfolge festliegt.¹⁰² Daher muss zur Minimierung der notwendigen vorzuhaltenden Unterstützer sowie der Anzahl an notwendigen Unterstützereinsätzen in dieser Arbeit ein antizipatives Zielkriterium entwickelt werden, welches ohne Kenntnis der Produktionsreihenfolge ermittelt werden kann und zu einer implizierten Minimierung der genannten Ziele führt.

2.2.1.1.2 Minimierung der Rekonfigurationskosten Durch Umsortierung von Arbeitsvorgängen entstehen Rekonfigurationskosten¹⁰³, welche formal durch Gleichung 2.22 ermittelt werden.

$$\sum_{avo \in \mathcal{AVO}} \sum_{ap \in \mathcal{AP}} \mathcal{K}_{avo,ap}^R \cdot v_{avo,ap}^R \quad (2.22)$$

Die möglichen Verschiebevorgänge für einen Arbeitsvorgang entsprechen der Anzahl alternativer Arbeitsplätze $|\mathcal{AP}| - 1$. Um die Kosten für die Umsortierung aller Arbeitsvorgänge zu definieren, muss eine Verschiebekostenmatrix der Dimension $(|\mathcal{AP}| - 1) \cdot |\mathcal{AVO}|$ gefüllt werden. Nach einer durchgeführten Verschiebung müssen u.U. sämtliche Kostensätze für Verschiebungen aller Arbeitsvorgänge von und zu den beiden betroffenen Arbeitsplätzen neu bewertet werden.¹⁰⁴ Ein erfahrener Planer kann die Kosten der Verschiebung eines einzelnen Arbeitsvorganges abschätzen, eine genaue Definition aller Verschiebungskosten im Vorfeld ist jedoch auf Grund der Komplexität unmöglich. Innerhalb dieser Arbeit muss ein Konzept

¹⁰⁰Vgl. Gleichung (2.5) in Abschnitt 2.1.2.1

¹⁰¹Vgl. Abschnitt 2.1.3.1

¹⁰²Vgl. Diskussion in Abschnitt 2.2

¹⁰³Vgl. zur Definition der Rekonfigurationskosten Abschnitt 2.1.3.2

¹⁰⁴Dieser Schritt ist notwendig, da durch den Wegfall eines Prozesses auf einem Arbeitsplatz eine darauf folgende Verschiebung eines Arbeitsvorganges zu diesem Arbeitsplatz hin unter Umständen günstiger wird. Durch eine Verschiebung kann sich z.B. neuer Raum für die logistische Andienung ergeben, so dass eine Verschiebung hin zu diesem Arbeitsplatz günstiger wird. Das Gegenteil kann ebenso der Fall sein.

erarbeitet werden, welches es ermöglicht, das Wissen des Planers einzubeziehen, ohne die Rekonfigurationskosten vorab vollständig zu definieren.

Die ermittelte Rekonfigurationslösung bildet die Startlösung für die nächste Rekonfiguration. Daher sei an dieser Stelle angemerkt, dass Rekonfigurationskosten, die in Kauf genommen werden, um eine Verringerung der Betriebskosten zu erreichen, unter Umständen auch in der nächsten Planungsperiode für eine Verringerung der Betriebskosten sorgen. Dieser Effekt ist zu erwarten, da sich der Trend hinsichtlich bestimmter Ausstattungsoptionen von Produkten nicht vollständig in jedem Planungszeitraum ändern wird. Da eine genaue Abschätzung dieses Effektes nicht durchgeführt werden kann, wird davon ausgegangen, dass sich die investierten Rekonfigurationskosten innerhalb des zu planenden Zeitraumes amortisieren müssen. Der weitere positive Effekt in den folgenden Planungsperioden ergibt sich dann durch die wegfallende Notwendigkeit, bereits verschobene Arbeitsinhalte erneut zu bewegen.

Zusammenfassend wird die Zielfunktion der Rekonfiguration aus der Summe der Rekonfigurationskosten und Betriebskosten in Gleichung 2.23 definiert.

$$\begin{aligned} \text{Min } \mathcal{K}^{Ges} = & \mathcal{K}^U \cdot u^{MAX} + \mathcal{K}^N \cdot u^E + \sum_{ap \in \mathcal{AP}} \mathcal{K}^W \cdot w_{ap} \\ & + \sum_{ap \in \mathcal{AP}} \sum_{avo \in \mathcal{AVO}} \mathcal{K}_{avo,ap}^R \cdot v_{avo,ap}^R \end{aligned} \quad (2.23)$$

2.2.1.2 Restriktionen der Rekonfiguration

Im Vergleich zur Erstinstallation einer Fließlinie ist der Entscheidungsraum bei einer Rekonfiguration stärker eingeschränkt. In Tabelle 2.5 werden die Unterschiede aufgezeigt.

	Erstinstallation	Rekonfiguration
Taktzeit	variabel	fix
Layout der Linie	variabel	fix
Länge der Linie	variabel	fix
Installation von Betriebsmitteln	variabel	meist fix
Zuordnung von Arbeitsvorgängen	variabel (Zuordnungen kostenfrei)	variabel (Verschiebung erzeugt Kosten)
Auswahl von Prozessalternativen	variabel	fix
Definition von Lagerfläche	variabel	fix

Tabelle 2.5: Freiheitsgrade im Vergleich (Rekonfiguration und Erstinstallation)

Alle Restriktionen, die bei der Erstinstallation zu beachten sind, behalten auch innerhalb der Rekonfiguration ihre Gültigkeit. Nach der Rekonfiguration darf jeder Arbeitsvorgang nur einem Arbeitsplatz zugeordnet sein.¹⁰⁵ Des Weiteren muss die technische Machbarkeit von Arbeitsvorgangszuordnungen zu Arbeitsplätzen gewährleistet sein. Diese Einschränkungen der Zuordnung werden abgebildet in Form von einzuhaltenden Vorrangbeziehungen in einem Vorranggraphen^{106,107} Die Menge $\mathcal{V}$ beinhaltet die Vorrangbeziehungen (i, j) , die bei der Umsortierung von Arbeitsvorgängen eingehalten werden müssen (2.24).¹⁰⁸

$$\sum_{ap \in \mathcal{AP}} st_{ap} \cdot x_{i,ap}^{Soll} \leq \sum_{ap \in \mathcal{AP}} st_{ap} \cdot x_{j,ap}^{Soll} \quad \forall (i, j) \in \mathcal{V} \quad (2.24)$$

Folgendes Beispiel 2.14 soll die Verwendung eines Vorranggraphen innerhalb einer Rekonfiguration verdeutlichen.

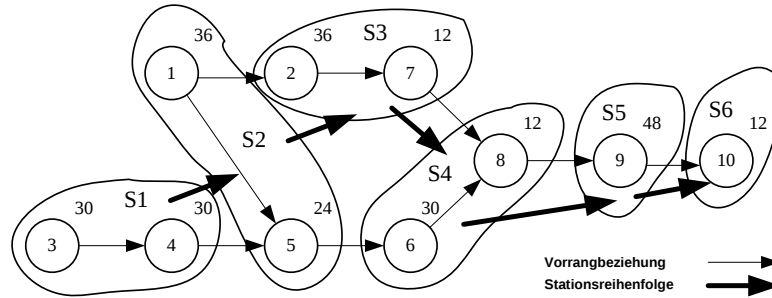


Abbildung 2.14: Beispiel für die Zuordnung von Arbeitsvorgängen auf Stationen unter Beachtung der Vorrangbeziehungen

Die einzelnen Arbeitsvorgänge werden vorrangbeziehungskonform abgearbeitet. Die Stationen werden dabei in der Reihenfolge S1, S2, S3, S4, S5 und S6 durchlaufen. Eine Rekonfiguration bzw. Verschiebung von Arbeitsvorgängen ist nur unter Einhaltung der Vorrangbeziehungen möglich. Abbildung 2.15 zeigt zwei mögliche Verschiebungen.

Arbeitsvorgang 10 kann nicht ohne weitere Verschiebung von anderen Arbeitsvorgängen auf Station S3 verschoben werden, da die Arbeitsvorgänge 8 und 9 vor diesem abgearbeitet werden müssen. Eine Verschiebung auf Station 5 hingegen ist möglich.

Zur Sicherstellung ergonomischer Bedingungen an den Arbeitsplätzen muss sich das zu bearbeitende Werkstück bei der Durchführung bestimmter Arbeitsvorgänge in einer passenden

¹⁰⁵Vgl. Ausführungen in Abschnitt 2.1.2 und die zugehörige Gleichung 2.2

¹⁰⁶Vgl. Definition in Abschnitt 2.2.1.1

¹⁰⁷Vgl. Definition in Abschnitt 2.2.2

¹⁰⁸Es wird dabei angenommen, dass die interne Reihenfolge der Durchführung von Arbeitsvorgängen auf den Arbeitsplätzen innerhalb einer Station durch einen weiteren Planungsprozess bestimmt wird. Daher reicht es hier, die Vorrangbeziehungen zwischen den Stationen sicherzustellen.

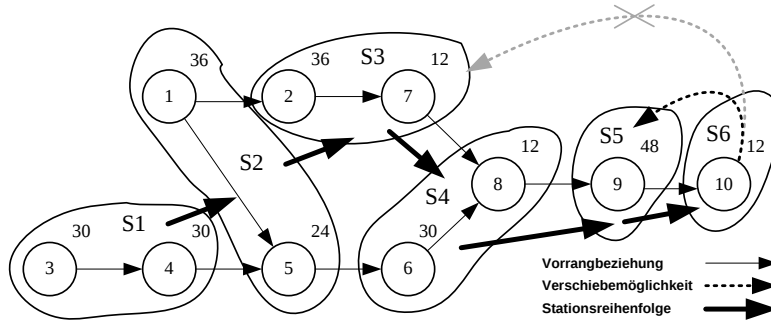


Abbildung 2.15: Beispiel für eine Rekonfiguration unter Beachtung der Vorrangbeziehungen

Werkstückposition¹⁰⁹ befinden. Das Werkstück kann sich innerhalb eines Stationsbereiches immer nur in einer Position befinden.¹¹⁰ Im Rahmen der Rekonfiguration wird diese nicht angepasst; daher können die Arbeitsvorgänge immer nur Arbeitsplätzen zugeordnet werden, welche sich an Stationen befinden, in denen die korrekte Werkstückposition gegeben ist (2.25).

$$x_{avo,ap}^{Soll} = 0 \quad \forall \quad ap \in \mathcal{AP}, avo \in \mathcal{AVO} \quad | \quad wp_{st_{ap}} \notin \mathcal{WP}_{avo} \quad (2.25)$$

Ein Großteil der Arbeitsvorgänge beinhaltet den Verbau von Teilen am Werkstück. Diese Teile werden an der entsprechenden Station am Fließband angeliefert. Jeder Stationsbereich hat nur eine begrenzte Lagerfläche. Die notwendige Lagerfläche für die Zuweisung von Arbeitsvorgängen darf in Summe die existierende Lagerfläche an einer Station nicht überschreiten (2.26).

$$\sum_{avo \in \mathcal{AVO}} \sum_{ap \in \mathcal{AP}: st_{ap}=st} x_{avo,ap}^{Soll} \cdot pl_{avo} \leq pl_{st_{ap}} \quad \forall \quad st \in \mathcal{ST} \quad (2.26)$$

2.2.1.3 Voraussetzungen der Rekonfiguration

Im Folgenden sollen die Voraussetzungen und getroffenen Annahmen, die im Rahmen der Rekonfiguration gelten, zusammengefasst werden.

- Es existiert genau eine Fließlinie.
- Es existieren Stationen $st \in \mathcal{ST}$.

¹⁰⁹Eine **Werkstückposition** ist eine bestimmte Lage, in der sich ein Werkstück befindet. Dies kann in der Automobilindustrie z.B. eine „horizontale“ oder „vertikale“ Lage bedeuten.

¹¹⁰Vgl. Abschnitt 2.1.1

- Es existieren Arbeitsplätze $ap \in \mathcal{AP}$ mit beschränktem Platz pl_{ap} und einer gegebenen Werkstückposition wp_{ap} , die jeweils einer Station zugewiesen sind st_{ap} .
- Jeder Arbeitsvorgang avo ist bei der Erstinstallation oder einer vorherigen Rekonfiguration bereits genau einem Arbeitsplatz ap durch $z_{avo,ap}^{Ist}$ zugeordnet und benötigt den Lagerplatz pl_{avo} .
- Jeder Arbeitsvorgang hat eine Menge möglicher Werkstückpositionen $\mathcal{WP}_{avo}$, bei denen er durchführbar ist.
- Das Werkstück befindet sich in jeder Station in einer fest definierten Werkstückposition wp_{st} .
- Die im Produktionszeitraum pz zu produzierenden Werkstücke $p \in \mathcal{P}_{pz}$ stehen fest.
- Alle Bearbeitungszeiten τ_{avo} sind fix vorgegeben.
- Die linken dbl_{ap} und rechten Driftbereiche dbr_{ap} liegen für jeden Arbeitsplatz ap fest.

Die Rekonfiguration findet dabei jeweils an dem Zeitpunkt statt, an dem die Aufträge und damit die Werkstückzusammensetzung für den folgenden Planungszeitraum feststehen.¹¹¹

2.2.2 Vorranggraphenentwicklung für variantenreiche Produkte

Zur Beschreibung der Vorranggraphenentwicklung für variantenreiche Produkte soll im ersten Schritt die Definition des Vorranggraphen geklärt werden. Im Anschluss werden das Ziel, die Restriktionen sowie die Voraussetzungen für die Vorranggraphenentwicklung diskutiert.

Vorranggraphen werden insbesondere im Rahmen der Fließbandabstimmung verwendet. So schreibt Bock¹¹²: „Bei der Abtaktung von Fließlinien ist eine Menge von Arbeitsvorgängen, zwischen denen Abhängigkeiten bestehen und die im Vorranggraph angegeben sind, einer Menge von aufeinanderfolgenden Stationen zuzuordnen.“¹¹³ Bullinger et. al.¹¹⁴ beschreiben den Nutzen eines Vorranggraphen folgendermaßen: „Der Montagevorranggraph ermöglicht es, die Montageablaufstruktur hinsichtlich der logisch-zeitlichen Abhängigkeiten abzubilden.“ Darauf aufbauend wurden sowohl Erweiterungen in der Anwendung des Graphen, wie auch in seiner Definition entwickelt. Die Definition eines Vorranggraphen kann grundsätzlich wie folgt vorgenommen werden:

¹¹¹Vgl. Diskussion zu Planungszeiträumen in Abschnitt 2.2

¹¹²Vgl. [Boc99]

¹¹³Vgl. auch [Dan03]

¹¹⁴Vgl. [BGLS95, S.94f.]

Ein **Vorranggraph** $VRG = (KN, KA, t)$ ist ein gerichteter zyklensfreier Graph mit einer Menge $KN = \{1, \dots, n\}$ von Knoten, einer Kantenmenge KA und einem Vektor t von Bewertungen. Die Knoten des Vorranggraphen stellen dabei die Arbeitsvorgänge dar, die an dem betrachteten Produkt zu leisten sind, während jede gerichtete Kante eine Vorrangfolge zwischen den Arbeitsvorgängen i und j in der Weise definiert, dass der Arbeitsvorgang i vor seinem Nachfolger j ausgeführt werden muss. Zudem legt die Bewertung t_i die Bearbeitungszeit für jeden Arbeitsvorgang i fest.¹¹⁵

Weiter definiert Braun¹¹⁶, dass zusätzlich zu den Zyklen keine Schleifen und Transitionen¹¹⁷ im Vorranggraphen erlaubt sind.¹¹⁸

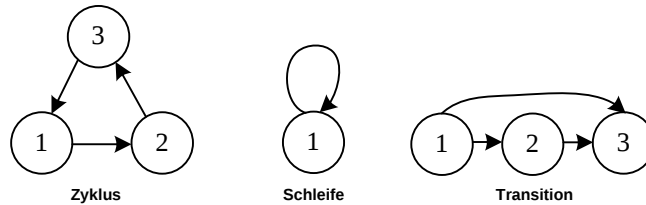


Abbildung 2.16: Verbotene Konstruktionen in Vorranggraphen

Bereits Thomopoulos¹¹⁹ erweitert den normalen Vorranggraphen zu einem sog. Mischgraphen¹²⁰, in dem er ein ganzes Variantenprogramm abbildet. Dafür werden die Knoten und Kanten aus den einzelnen Vorranggraphen jeder Variante vereinigt und die Bearbeitungszeiten der Arbeitsvorgänge mit der Baurate multipliziert.¹²¹ Dieser Algorithmus wird detailliert beschrieben von Braun¹²². Durch die Vereinigung können Zyklen auftreten, die beseitigt werden müssen.¹²³ Diese vereinfachte Erstellung eines Vorranggraphen ermöglicht eine Fließbandabstimmung für variantenreiche Produkte mit Methoden, die für die Einproduktabstimmung entwickelt wurden.¹²⁴ Braun definiert weiter, dass im Gegensatz zum Standardvorranggra-

¹¹⁵Zitat: [Dan03, S.359]; Vgl. [DSV93, S.227] sowie ähnliche Definitionen in [Bul86], [DSV93, S.171] und [NM93, S.177f.]

¹¹⁶Vgl. [Bra95, S.52]

¹¹⁷Transitiv ist in diesem Zusammenhang die Vorrangbeziehung zwischen Arbeitsvorgang i und k , wenn nur die Vorrangbeziehungen zwischen i und j sowie j und k abgebildet sind.

¹¹⁸Vgl. zur Erklärung Grafik 2.16 aus [Bra95, S.52]

¹¹⁹Vgl. [Tho67]

¹²⁰Synonyme: Komplexgraph, aggregierter Vorranggraph, Variantenvorranggraphen z.B. in [Mül02, S.60]

¹²¹Vgl. zur Erstellungssystematik von Mischgraphen [Tho67, S.60ff.] und [Gör78, S.47ff.] sowie [Bra95, S.53ff.] und [DSV93, S.227]

¹²²Vgl. [Bra95, S.59 ff.]

¹²³Vgl. zu Algorithmen zur Zyklenbeseitigung [AH94]

¹²⁴Vgl. [Dan03] und Abschnitt 2.1.2.1

phen im Mischgraphen Transitionen zulässig sind.¹²⁵ Über dies hinaus werden je nach Anwendungsfall weitere Erweiterungen in der Literatur genannt.¹²⁶

In dieser Arbeit soll der Vorranggraph insbesondere im Rahmen der zuvor definierten Rekonfiguration von getakteten Variantenfließlinien¹²⁷ verwendet werden. Die hierfür relevante Definition wird im folgenden Abschnitt 2.2.2.1 vorgenommen.

2.2.2.1 Ziel der Vorranggraphenentwicklung

Ziel der Entwicklung eines Vorranggraphen ist es, die technischen Restriktionen zwischen den einzelnen Arbeitsvorgängen aufzunehmen und als Restriktionen für Planungsprobleme zur Verfügung zu stellen.

Im Rahmen einer Rekonfiguration werden die Informationen aus dem Vorranggraphen dazu verwendet, das Planungsproblem hinsichtlich der technischen Machbarkeit von Verschiebungen durch die Gleichung 2.27 einzugrenzen.¹²⁸

$$\sum_{ap \in \mathcal{AP}} st_{ap} \cdot x_{i,ap}^{Soll} \leq \sum_{ap \in \mathcal{AP}} st_{ap} \cdot x_{j,ap}^{Soll} \quad \forall (i, j) \in \mathcal{V} \quad (2.27)$$

Werden alle Vorrangbeziehungen im Graphen eingehalten, so kann zumindest im Bezug auf technische Restriktionen davon ausgegangen werden, dass die Abfolge der Arbeitsvorgänge gültig ist. Im Vergleich zur Erstinstallation von Fließlinien existieren zum Zeitpunkt einer Rekonfiguration bereits ausspezifizierte Varianten.¹²⁹ Die Anpassung auf den zukünftigen Bedarf steht bei der Rekonfiguration im Vordergrund; daher ist eine explizite Verwendung der Bearbeitungszeiten für die einzelnen zu fertigenden Produkte notwendig.¹³⁰ Dadurch ergibt sich die Notwendigkeit, die ungewichteten Bearbeitungszeiten der einzelnen Arbeitsvorgänge beizubehalten. Zur einheitlichen Datenhaltung über mehrere Rekonfigurationen hinweg sollte

¹²⁵Vgl. [Bra95, S.58]

¹²⁶Weitere Erweiterungen des Vorranggraphen umfassen z.B. Beschreibungsmerkmale für Montagevorgänge, eingeführt in [Sei98, S.69 f.] und Erweiterungen in [Mül02, S.62 f.], in denen z.B. auch die Lage des Werkstücks und die Werkstückansprache definiert sind. Haller et al. ergänzen in [HSMZ00, S.6f.] die Möglichkeit, an den Kanten im Vorranggraphen zusätzliche Informationen zu hinterlegen, indem der Vorranggraph z.B. für Vorgangsfolgezwänge oder auch für simultan auszuführende Montagevorgänge um spezielle Montageablaufrestriktionen erweitert wird. Müller definiert in [Mül02, S.63ff.] einen mehrschichtigen Vorranggraphen, indem mehrere Prozesszeiten je Arbeitsvorgang gespeichert werden. Ein Arbeitsvorgang kann in diesem Konzept mehrere Bearbeitungszeiten haben: die betriebsmittelorientierte, die mitarbeiterorientierte und die arbeitsgegenstandsorientierte Arbeitszeit. Die Differenz zwischen den Zeiten kann z.B. aus vorbereitenden Tätigkeiten eines Mitarbeiters resultieren, für die das Werkstück selbst nicht erforderlich ist. Mit Hilfe dieses mehrschichtigen Vorranggraphen ist es möglich, die unterschiedlichen Sichten der Prozessbeteiligten mit ihren verschiedenen Zeitarten gleichrangig in einem gemeinsamen Vorranggraphen abzubilden.

¹²⁷Vgl. Abschnitt 2.2.1

¹²⁸Vgl. Abschnitt 2.2.1.2

¹²⁹Vgl. Abschnitt 2.2.1.3

¹³⁰Vgl. Abschnitt 2.2.1.1.1

der Vorranggraph auch Arbeitsvorgänge von Varianten beinhalten, welche in dem Planungszeitraum unnötig sind, da auch diese Arbeitsvorgänge und die zugehörigen Betriebsmittel an dem zugeordneten Arbeitsplatz verweilen.

In einem komplexen praktischen Szenario, wie z.B. in der Automobilindustrie, kann die übliche Vorgehensweise, für jede Variante einen Graphen zu bilden und nachher den Mischgraphen zu erzeugen, nicht gewählt werden.¹³¹ Dennoch müssen alle möglichen Arbeitsvorgänge im Vorranggraphen enthalten sein, um jede mögliche Variante bei Bedarf bauen zu können. Daher wird in Gleichung 2.28 eine variantenunabhängige Definition eines vollständigen Mischgraphen mit Hilfe der Notation in Tabelle 2.6 vorgenommen.

$$\mathcal{G} = (\mathcal{AVO}, \mathcal{V}, \mathcal{BZ}) \quad (2.28)$$

Symbol	Definition
$\mathcal{AVO}$	Menge von Arbeitsvorgängen <i>avo.</i> (Knoten)
$\mathcal{G}$	Vorranggraph, der aus einer Menge von Arbeitsvorgängen $\mathcal{AVO}$, einer Menge von Vorrangbedingungen $\mathcal{V}$ und einer Menge von Gewichten $\mathcal{BZ}$ besteht.
$\mathcal{BZ}$	Menge von Gewichten im Vorranggraphen. Jedes Gewicht entspricht der Bearbeitungszeit τ_{avo} eines Arbeitsvorganges. (Gewichtsvektor)
$\mathcal{V}$	Menge von Vorrangbedingungen. Wenn $(i, j) \in \mathcal{V}$, dann muss Arbeitsvorgang $i \in \mathcal{AVO}$ abgeschlossen sein, bevor mit Arbeitsvorgang $j \in \mathcal{AVO}$ begonnen werden kann. Es gilt: $\mathcal{V} \subset \mathcal{AVO} \times \mathcal{AVO}$ (Kanten)

Tabelle 2.6: Notation zur Definition des Vorranggraphen

2.2.2.2 Restriktionen der Vorranggraphenentwicklung

Ein Problem bei der Vorranggraphenentwicklung ist die Tatsache, dass alle Vorrangbeziehungen aufgenommen sein müssen, bevor der Graph z.B. im Rahmen einer Rekonfiguration oder Erstinstallation einer Fließlinie verwendet werden kann. Fehlen Beziehungen im Graphen, können Planungsmethoden, die auf dem Vorranggraphen aufbauen, keine gültigen Lösungen generieren. Somit bleibt an dieser Stelle festzuhalten, dass sämtliche mit dem Graphen erzeugte Zuordnungen von Arbeitsvorgängen gültig sein müssen. Daher sollte eher auf Freiheitsgrade¹³² im Graphen verzichtet werden, sofern nicht sichergestellt werden kann, dass sämtliche aus dem Graphen abgeleiteten Planungslösungen auch nach der Einführung noch valide sind.

¹³¹Vgl. die von der Produktvariante abhängige Definition eines Mischgraphen in [BFS07a]

¹³²Ein **Freiheitsgrad** im Vorranggraphen ist als Möglichkeit zu verstehen, im Rahmen einer Rekonfiguration einen Arbeitsvorgang zu verschieben.

2.2.2.3 Voraussetzungen der Vorranggraphenentwicklung

Die Hauptproblematik bei der Erstellung von Vorranggraphen ist die Komplexität, die durch die Menge an zu definierenden Vorrangbeziehungen zwischen den Arbeitsvorgängen erzeugt wird. Bei einer Anzahl von $|\mathcal{AVO}|$ Arbeitsvorgängen müssten zur vollständigen Betrachtung $0,5 \cdot |\mathcal{AVO}| \cdot (|\mathcal{AVO}| - 1)$ Beziehungen geklärt werden.¹³³ In einem praktischen Umfeld wie in der Automobilindustrie müssen in der Endmontage zur Erstellung eines Vorranggraphen bei mehr als 5000 Arbeitsvorgängen ca. 12,5 Mio Beziehungen geklärt werden. Diese Komplexität muss im Rahmen einer Vorranggraphenentwicklung für variantenreiche Produkte berücksichtigt werden und kann z.B. mittels Automatisierung beherrscht werden. Um den Aufwand möglichst gering zu halten, sollte diese Automatisierung auf existierenden Daten aufbauen. Die Produktdokumentation in Form einer Stückliste hat sich branchenübergreifend durchgesetzt. Daher werden im Folgenden verschiedene Stücklistenarten systematisiert.

Eine **Stückliste** ist ein „für den jeweiligen Zweck vollständiges, formal aufgebautes Verzeichnis für einen Gegenstand, das alle zugehörigen Gegenstände, unter Angabe von Bezeichnung (Benennung, Sachnummer), Menge und Einheit, enthält. Als Stücklisten werden nur solche Verzeichnisse bezeichnet, die sich auf die Menge 1 eines Gegenstandes beziehen.“ Im Folgenden werden die unterschiedlichen Arten von Stücklisten angelehnt an die Definition in DIN 199 2002 Teil 3 vorgestellt.¹³⁴

In der **Mengenübersichts-Stückliste** werden die Bestandteile eines Erzeugnisses unstrukturiert aufgezählt. Dabei werden die Bestandteile eines Erzeugnisses, dazu gehören sowohl Baugruppen als auch Einzelteile, mit ihren Mengenangaben geführt. Bei dieser einfachen Form der Stückliste wird kein Hinweis auf die Erzeugnisstruktur des Produktes gegeben. In der Mengenübersichts-Stückliste können „bei Bedarf auch nicht aufgelöste Gruppen aufgeführt werden.“¹³⁵

Eine weitere Stücklistenform ist die **Strukturstückliste**, welche die hierarchische Stellung des jeweiligen Elementes eines Erzeugnisses in der Erzeugnisstruktur darstellt. Bei der Strukturstückliste, die auch als ein nach Fertigungsebene sortierter Baum angesehen werden kann, werden allerdings Baugruppen, die mehrfach verwendet werden, mit ihren Ebenen wiederholt erfasst. Hierbei wird jede Gruppe bis zu ihrer niedrigsten Stufe aufgegliedert.

Die **Baukasten-Stückliste** ist eine „Stücklistenform, bei der alle Teile und Gruppen aus der nächsttieferen Stufe aufgeführt sind.“¹³⁶ Bei der Baukastenstückliste beziehen sich Mengenangaben auf die Baugruppe und nicht auf das Enderzeugnis. Daher werden je Baugruppe

¹³³Vgl. [Bra95, S.19]

¹³⁴Vgl. dazu auch [DW97, S.225ff]

¹³⁵Vgl. [DIN 199 2002 Teil 3, S.13]

¹³⁶Vgl. [DIN 199 2002 Teil 3, S.9]

nur die Einzelteile aufgeführt, die unmittelbar in sie eingehen. Bei komplexen Produkten entsteht so allerdings eine große Anzahl an Stücklisten, da das Endprodukt in mehrere Gruppen aufgeteilt ist.

Um die Speicherung redundanter Informationen zu vermeiden, eignen sich **Variantenstücklisten**. Eine Variantenstückliste ist eine „Zusammenfassung mehrerer Stücklisten auf einem Vordruck, um verschiedene Gegenstände mit einem in der Regel hohen Anteil identischer Bestandteile gemeinsam aufführen zu können.“¹³⁷ Bei der **Variantenstückliste mit Gleichteilsatz** wird jede Variante durch eine Gleichteilegruppe und zusätzliche variantenabhängige Baugruppen oder Einzelteile beschrieben. Bei der **Plus-Minus-Stückliste** werden Varianten durch die Angabe von Zusatz- und Entfallteilen beschrieben.

Geht es um die Abbildung besonders variantenreicher Produkte, wird die **offene Variantenstückliste** verwendet¹³⁸, bei der die Merkmale mit entsprechenden Ausprägungen so verwaltet werden, dass für einen Erzeugnistyp nur noch eine Stückliste vorhanden ist. Durch diese Stückliste wird kein konkretes Erzeugnis, sondern eine komplette Erzeugnisfamilie beschrieben und damit das Abspeichern und Pflegen der darin enthaltenen Daten erleichtert. Für einen konkreten Auftrag wird, in Abhängigkeit von den durch den Kunden festgelegten Eigenschaften, jeweils eine Strukturstückliste erzeugt, welche den Auftrag vom Eingang bis zur Fertigstellung begleitet. An die Bauteile dieser Stückliste sind die notwendigen Arbeitsvorgänge fest gekoppelt.

Die im Folgenden entwickelten Methoden bauen zum Teil auf der Struktur der offenen Variantenstückliste auf. Eine genauere Beschreibung der Struktur mit der notwendigen Aufbereitung wird im Konzeptteil der Arbeit in den Abschnitten 5.2.2.1 und 5.2.2.2 vorgenommen.

Im Folgenden sollen die Voraussetzungen und getroffenen Annahmen, die im Rahmen der zu entwickelnden Vorranggraphenerstellung gelten, zusammengefasst definiert werden.

- Es existiert genau eine Fließlinie.
- Es existieren Stationen $st \in \mathcal{ST}$.
- Es existieren Arbeitsplätze $ap \in \mathcal{AP}$, die jeweils einer Station zugewiesen sind st_{ap} .
- Es existieren eine oder mehrere gültige Zuordnungen $z_{avo,ap}^{Ist}$ von Arbeitsvorgängen $avo \in \mathcal{AVO}$ auf Arbeitsplätze $ap \in \mathcal{AP}$.
- Es existieren eine oder mehrere gültige Reihenfolgen der Abarbeitung von Arbeitsvorgängen $avo \in \mathcal{AVO}$.
- Es existiert eine Zuordnung von Arbeitsvorgängen zu Werkstückansprachen durch wa_{avo} , an welchen unabhängig voneinander gearbeitet werden kann.

¹³⁷Vgl. [DIN 199 2002 Teil 3, S.15]

¹³⁸Vgl. [Dan03, S.266] und [Dan09, S.241]

- Es wird eine offene Variantenstückliste mit Bezug zu den Arbeitsvorgängen zur Dokumentation verwendet.

2.2.3 Reihenfolgeplanung für eine getaktete Variantenfließlinie

Im Folgenden soll die Aufgabe der Reihenfolgeplanung für eine getaktete Variantenfließlinie genauer definiert werden. Hierfür werden in drei Schritten das Ziel, die Restriktionen und die Voraussetzungen einer Reihenfolgeplanung definiert. Die Notation wird hierfür um die Bezeichner in Tabelle 2.7 erweitert.

Symbol	Definition
$\mathcal{RF}_{pz}$	Menge von Reihenfolgepositionen $idx \in \mathcal{RF}$ im Produktionszeitraum $pz \in \mathcal{PZ}$
$\mathcal{K}^{RFGE}$	Gesamtkosten, die durch die Reihenfolgeplanung beeinflusst werden.

Tabelle 2.7: Erweiterung der Notation zur Formalisierung der Reihenfolgeplanung

2.2.3.1 Ziel der Reihenfolgeplanung

Die Reihenfolgeplanung muss folgende Entscheidungsvariablen bestimmen:¹³⁹

1. Bestimmung der Anzahl an vorzuhaltenden Unterstützerarbeitskräften
2. Zuordnung von Aufträgen zu Reihenfolgepositionen

Gegenstand der Reihenfolgeplanung ist die Zuordnung von Werkstücken $p \in \mathcal{P}_{pz}$ zu Reihenfolgepositionen $idx \in \mathcal{RF}_{pz}$ eines Produktionszeitraumes durch $y_{p,idx}^{\mathcal{RF}}$. Dabei muss jeder Reihenfolgeposition genau ein Werkstück (2.29) und jedem Werkstück genau eine Reihenfolgeposition zugewiesen sein (2.30).

$$\sum_{p \in \mathcal{P}} y_{p,idx}^{\mathcal{RF}} = 1 \quad \forall \quad idx \in \{1, \dots, |\mathcal{RF}|\} \quad (2.29)$$

$$\sum_{idx \in \mathcal{RF}} y_{p,idx}^{\mathcal{RF}} = 1 \quad \forall \quad p \in \{1, \dots, |\mathcal{P}|\} \quad (2.30)$$

Im Sinne der Effizienzsteigerung einer Fließlinie gilt es durch die Reihenfolgeplanung ein Optimum der Betriebskosten für den zu planenden Produktionszeitraum zu erreichen. Das Kapazitätsangebot wird bereits im Rahmen der Rekonfiguration für den Planungszeitraum definiert.¹⁴⁰ Die Kosten, die durch Unterstützereinsätze entstehen, teilen sich auf in die Anzahl

¹³⁹Vgl. Abschnitt 2.2

¹⁴⁰Vgl. Abschnitt 2.2.1

an vorzuhaltenden Unterstützern und die Anzahl an Unterstützereinsätzen, die zu erhöhten Nacharbeitskosten führen (2.31).¹⁴¹

$$\text{Min } \mathcal{K}^{GesRF} = \mathcal{K}^U \cdot u^{MAX} + \mathcal{K}^N \cdot u^E \quad (2.31)$$

Zur Bestimmung der notwendigen Unterstützereinsätze muss der Produktionsablauf mit den Positionen der Werker vollständig simuliert werden. Dabei ist insbesondere darauf zu achten, dass mehrere Planungszeiträume denselben Bewertungszeitraum beeinflussen können. Die einzelnen Reihenfolgepositionen sind durch $\mathcal{RF} = [1, \dots, |\mathcal{RF}|]$ definiert und über alle Planungszeiträume hinweg eindeutig (2.32).

$$\mathcal{RF}_{pz} \subset \mathcal{RF} \quad | \quad idx1 < idx2 \quad \forall \quad idx1 \in \mathcal{RF}_{pz} \wedge idx2 \in \mathcal{RF}_{pz+1} \quad (2.32)$$

Die Reihenfolgeplanung definiert, in welcher Reihenfolge die Werkstücke $\mathcal{P}_{pz}$ in einem Planungszeitraum pz eingesteuert werden. Wenn der erste Auftrag eines Produktionszeitraumes pz in das System eingelastet wird, befinden sich bereits $|\mathcal{ST}| - 1$ Werkstücke im Fließsystem, die in der Vorperiode in eine Reihenfolge gebracht wurden. Bei der Bewertung einer aufzulegenden Produktionsreihenfolge für den Produktionszeitraum pz muss die Fertigstellung der Werkstücke des Produktionszeitraumes $pz - 1$ mit betrachtet werden da durch die Fertigstellung der $|\mathcal{ST}| - 1$ Werkstücke, die sich noch im System befinden, ebenfalls Unterstützereinsätze auftreten können.¹⁴² Die innerhalb des Produktionszeitraumes pz zu bewertenden Situationen sind in Abbildung 2.17 durch die durchgezogenen Diagonalen gekennzeichnet.¹⁴³ Folglich hat die Bildung einer Produktionsreihenfolge für den Produktionszeitraum pz einen Einfluss auf die Ereignisse im folgenden Produktionszeitraum $pz + 1$. In Abbildung 2.17 werden diese Takte durch die gestrichelten Diagonalen dargestellt. Um eine korrekte Bewertung der Auswirkungen einer Reihenfolgeplanung durchzuführen, müssen somit mehrere Produktionszeiträume bewertet werden. Des Weiteren muss beachtet werden, dass ein eventueller Werkerdrift sich nicht über einen Bewertungszeitraum hinweg in den nächsten verschieben kann, da die Fließlinie zwischen zwei Produktionszeiträumen angehalten und die Arbeiten an jedem Arbeitsplatz zu Ende geführt werden. Die Werker beginnen in einem neuen Produktionszeitraum immer an der linken Stationsgrenze.

2.2.3.2 Restriktionen der Reihenfolgeplanung

Die Zuweisung der Aufträge zu den Produktionszeiträumen ist durch die Programmplanung, hinsichtlich logistischer Restriktionen, bereits abgeschlossen. Wichtig ist somit, dass die Werk-

¹⁴¹Vgl. Abschnitt 2.1.3.1

¹⁴²Vgl. Abschnitt 2.1.3.1.2

¹⁴³Vgl. Definition in Abschnitt 2.1.3.1

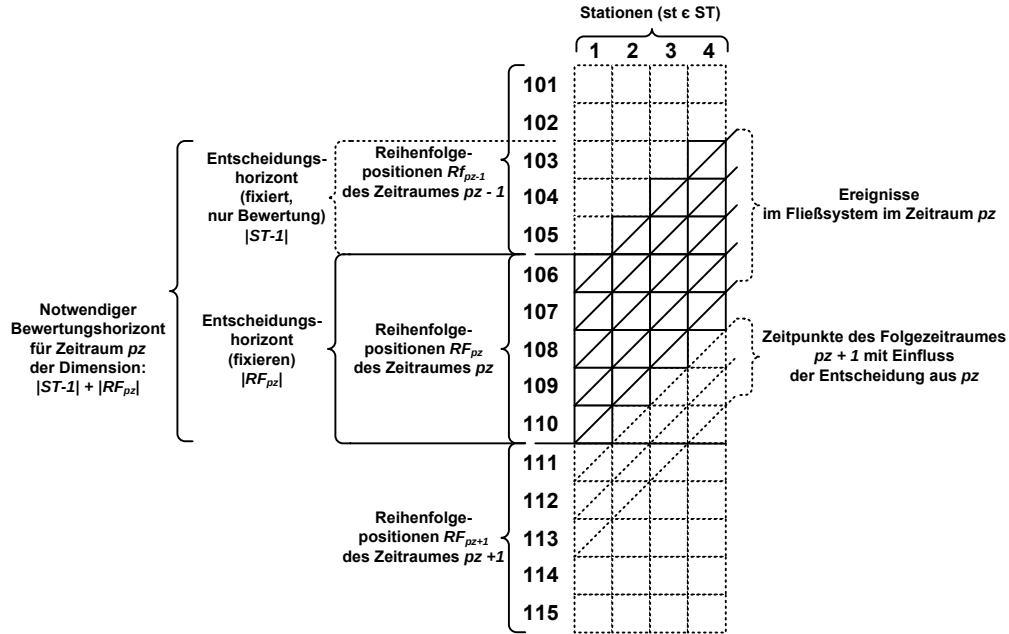


Abbildung 2.17: Rollierender Aspekt der Bewertung einer Reihenfolgeplanung

stücke innerhalb des zugewiesenen Zeitraumes produziert werden, da sonst die durch die Programmplanung betrachteten logistischen Restriktionen unter Umständen nicht eingehalten werden. Innerhalb des zu planenden Produktionszeitraumes pz können die Werkstücke $p \in \mathcal{P}_{pz}$ hinsichtlich der logistischen Restriktionen beliebig angeordnet werden.

2.2.3.3 Voraussetzungen der Reihenfolgeplanung

Im Folgenden sollen die Voraussetzungen und getroffenen Annahmen, die im Rahmen der Reihenfolgeplanung gelten, zusammengefasst werden.

- Es existiert genau eine Fließlinie.
- Es existieren Stationen $st \in \mathcal{ST}$.
- Es existieren Arbeitsplätze $ap \in \mathcal{AP}$, die jeweils einer Station st_{ap} und einer Ansprechzone sp_{ap} zugewiesen sind.
- Jeder Arbeitsvorgang avo ist durch die Rekonfiguration genau einem Arbeitsplatz ap durch $z_{avo,ap}^{Ist}$ zugeordnet.
- Alle Bearbeitungszeiten τ_{avo} sind deterministisch und fix.
- Die Aufträge und damit die entsprechenden Werkstücke $p \in \mathcal{P}_{pz} \subset \mathcal{P}$ sind durch eine Programmplanung bereits einem Produktionszeitraum $pz \in \mathcal{PZ}$ zugeordnet.

- Der organisatorische Umgang mit Überlastungssituationen (reaktiver oder präventiver Unterstützereinsatz) liegt fest.¹⁴⁴
- Die Abhängigkeiten der Arbeitsplätze untereinander sind durch Vorgängerarbeitsplätze $vg \in \mathcal{VG}_{ap}$ je Arbeitsplatz definiert.¹⁴⁵
- Die linken dbl_{ap} und rechten Driftbereiche dbr_{ap} liegen für jeden Arbeitsplatz ap fest.

¹⁴⁴Vgl. Abschnitt 2.1.2.2.2

¹⁴⁵Vgl. Abschnitt 2.1.2.2.1

2.3 Anforderungen an die Problemlösung

Basierend auf den zuvor definierten Teilaufgaben zur Kapazitätsabstimmung von getakteten Variantenfließlinien werden im Folgenden die Anforderungen an die einzelnen Verfahren zur Lösung der Teilprobleme abgeleitet.

2.3.1 Anforderungen an ein Verfahren zur kostenoptimalen Rekonfiguration

Folgende Anforderungen an ein Verfahren zur kostenoptimalen Rekonfiguration können genannt werden:

- **Betrachtung der Betriebskosten**

Als wichtigster Kostenfaktor bei der Montage in getakteten Variantenfließlinien wurde die menschliche Arbeit identifiziert. Das Verfahren sollte daher den Einsatz des Personals bewerten und die entstehenden Kosten bei der Entscheidungsfindung berücksichtigen.

- **Unabhängigkeit von der Reihenfolgeplanung**

Auf Grund der unterschiedlichen Planungszeiträume sollte das Verfahren in der Lage sein, die Betriebskosten ohne Kenntnis der Reihenfolge abschätzen zu können. Hierfür muss ein antizipatives Zielkriterium gefunden werden, das die Auswirkungen der Planungsalternative korrekt abschätzt und in die Entscheidung einfließen lässt.

- **Betrachtung des zukünftigen Auftragsprogramms**

Das Verfahren sollte das zukünftige Auftragsprogramm, bestehend aus voll ausspezifizierten Varianten, für die Planung verwenden können.

- **Betrachtung der Rekonfigurationskosten**

Die entstehenden Rekonfigurationskosten, die sich durch die Planung ergeben, müssen mit in der Entscheidung berücksichtigt werden. Das Verfahren muss einsatzfähig sein, ohne dass die Kosten vorher vollständig bestimmt werden.

- **Beachtung von Vorrangrestriktionen**

Das Verfahren sollte die Möglichkeit beinhalten, sowohl mit als auch ohne explizite Verfügbarkeit eines Vorranggraphen in Datenform Lösungen zu erzeugen.

- **Einflussmöglichkeit durch den Planer**

Um über die definierten Restriktionen hinausgehende praktische Begebenheiten betrachten zu können, sollte der Planer in den Entscheidungsprozess integriert werden. Das Verfahren sollte für den Planer verständlich sein und ihm die Möglichkeit geben, in den Entscheidungsprozess einzugreifen.

- **Transparenz in der Entscheidungsfindung**

Auswirkungen von gefundenen Planungsalternativen sollten für den Planer transparent in einem IT-System dargestellt werden. So können Einzelentscheidungen jederzeit revidiert werden und ein Überblick über die gefundenen Planungsalternativen bleibt erhalten.

- **Schnelle Rückmeldung**

Das Verfahren sollte fähig sein, die hohe Komplexität einer praktischen Problemgröße zu beherrschen. Dabei sollte eine softwaretechnische Umsetzung in seiner Laufzeit performant genug sein, um in einer praktischen Anwendung auf handelsüblichen Rechnern mit Antwortzeiten im Sekundenbereich zu reagieren.

2.3.2 Anforderungen an ein Verfahren zur Vorranggraphenerstellung für variantenreiche Produkte

Folgende Anforderungen an ein Verfahren zur Vorranggraphenentwicklung für variantenreiche Produkte können genannt werden:

- **Aufbau auf existierender Produktdokumentation**

Eine große Problematik bei der Erstellung von Vorranggraphen stellt die Datenverfügbarkeit dar. Daher soll das Verfahren auf einer existierenden Produktdokumentation aufbauen können. Diese sollte bestmöglich analysiert und interpretiert werden, um automatisiert Informationen für eine Graphenstrukturierung ableiten zu können.

- **Beherrschung der Komplexität variantenreicher Produkte**

Die Erstellung eines Vorranggraphen für variantenreiche Produkte ist mit erheblicher Komplexität verbunden. Das Verfahren sollte diese Komplexität beherrschen und Lösungen für praktische Problemgrößen erzeugen können.

- **Möglichkeit der Teilung des Aufwandes zur Vorranggraphengenerierung**

Eine vollständige Vorranggraphenerstellung ist mit erheblichem Aufwand verbunden. Das Verfahren sollte daher die Möglichkeit beinhalten, den Aufwand zur Erstellung des Vorranggraphen in verschiedene Entwicklungsstufen oder in verschiedene thematische Bereiche zu teilen, um eine parallele Bearbeitung möglich zu machen.

- **Pflege des Vorranggraphen über die Zeit**

Ein Vorranggraph ist kein statisches Konstrukt, welches einmal erzeugt wird und ab diesem Zeitpunkt auf Dauer verwendet werden kann. Es ergeben sich ständig neue Anforderungen an den Produktionsprozess, so dass eine Pflege des Graphen über die Zeit hinweg möglich sein muss.

- **Sicherstellung der Gültigkeit jeglicher abgeleiteter Lösungen zu jeder Zeit**
Der Vorranggraph soll insbesondere zur Beschränkung der Lösungen in einer Rekonfiguration eingesetzt werden. Daher sollte zu jeder Zeit, auch während der Entwicklung des Graphen, eine Gültigkeit sämtlicher generierter Lösungen auf Basis des erzeugten Vorranggraphen gewährleistet sein.
- **Schnelle Rückmeldung**
Die softwaretechnische Umsetzung des Verfahrens sollte auf handelsüblichen Rechnern Lösungen im Minutenbereich generieren.

2.3.3 Anforderungen an ein Verfahren zur Reihenfolgeplanung zur Minimierung des Unterstützereinsatzes

Folgende Anforderungen an eine Reihenfolgeplanung zur Minimierung des Unterstützereinsatzes können genannt werden:

- **Genaue Bewertung der resultierenden Kosten**
Zum Zeitpunkt der Reihenfolgeplanung sind alle notwendigen Entscheidungsvariablen fixiert. Das Bewertungsverfahren sollte daher eine vollständige Bewertung der Produktion in Form einer Fließliniensimulation unter Berücksichtigung der definierten Einflussfaktoren vornehmen.
- **Betrachtung der Betriebskosten unter expliziter Berücksichtigung des Unterstützereinsatzes**
Der Unterstützereinsatz ist der entscheidende Kostenfaktor bei der Reihenfolgeplanung. Die Bewertung der Lösungen sollte daher beide vorgestellten Organisationsformen des Unterstützereinsatzes beherrschen. Dabei sollte insbesondere auch der gleichzeitige Unterstützereinsatz bewertet werden.
- **Verwendung der Rekonfigurationslösung**
Auf Grund der unterschiedlichen Planungszeiträume der beiden Planungsschritte müssen die in der Rekonfiguration fixierten Entscheidungsvariablen übernommen werden. Das Verfahren zur Reihenfolgeplanung muss auf der Planungslösung aufbauen.
- **Rollierenden Bewertungshorizont beachten**
Die Bewertung des Produktionsprozesses muss passend zum Produktionszeitraum durchgeführt werden. Dabei müssen die Auswirkungen der Reihenfolgeplanung auf eine Folgeperiode sowie die Auswirkungen der Reihenfolge der Vorperiode korrekt bewertet werden.

- **Verwendung als Frühwarnsystem**

Das Verfahren zur Bewertung des Produktionsprozesses sollte als Frühwarnsystem eingesetzt werden können, um die notwendigen Einsatzpunkte für Unterstützer sowie die Anzahl an notwendigen Unterstützern abzuleiten.

- **Beherrschung der Komplexität realer Problemgrößen**

Die Reihenfolgeplanung ist mit einer erheblichen Komplexität verbunden. Das Verfahren sollte in einer softwaretechnischen Umsetzung dazu fähig sein, reale Problemgrößen im Minutenbereich zu lösen.

3 Stand der Forschung

Forschung ist immer das Weiterforschen,
wo andere aufgehört haben, das Weiterbauen
auf Grundsteinen und Gerüsten, die andere
vorbereitet haben, und damit allerdings
leider zugleich auch mitunter das Weitergehen
auf Irrwegen, die andere eingeschlagen haben.

(Hubert S. Markl)

Im Folgenden wird der Stand der Forschung aufgeteilt in die drei Teilprobleme Rekonfiguration, Vorranggraphenerstellung und Reihenfolgeplanung analysiert.

3.1 Stand der Forschung zur Rekonfiguration von Fließlinien

Zur Beschreibung der Literatur zur Rekonfiguration von Fließlinien werden zuerst Ansätze dargestellt, die explizit die Rekonfiguration betreffen. Danach werden Ansätze für die Ersteinstallation von Fließlinien auf ihre Übertragbarkeit auf die Rekonfiguration hin geprüft.

3.1.1 Ansätze zur Rekonfiguration

Das Problem der Rekonfiguration unter Betrachtung einer existierenden Fließlinie wird das erste Mal in „Line Balancing in the Real World“¹⁴⁶ von Falkenauer explizit erwähnt. Er beschreibt die große Diskrepanz zwischen Theorie und Praxis im Bereich der Fließbandabstimmung. Die Gründe hierfür liegen in der Komplexität des Problems und in der unterschiedlichen Ausgangsproblematik in Theorie und Praxis. Er forciert den notwendigen Forschungsbedarf im Bereich der Rekonfiguration, da selbst bei Einführung eines neuen Produktes meist auf

¹⁴⁶Vgl. hierzu [Fal05]

einem existierenden Fließsystem aufgebaut werden muss. Als wichtigstes Zielkriterium bezeichnet er die gleichmäßige Auslastung der Arbeitsplätze, vernachlässigt jedoch die Schwankungen der Bearbeitungszeiten. Lediglich die Minimierung von maximalen Belastungen wird kurz diskutiert, aber kein Lösungsansatz in der Arbeit vorgestellt.¹⁴⁷

Boysen et al.¹⁴⁸ beschreiben die Rekonfiguration als ein Verfahren zur Auswahl zwischen Fließbandabstimmungslösungen im Zusammenhang mit einer durchzuführenden Programmplanung. Sie platzieren den Schritt der Rekonfiguration zwischen der Programmplanung und der Reihenfolgeplanung in einem hierarchischen Modell, stellen jedoch einen simultanen Ansatz zur Programmplanung und Rekonfiguration vor, in dem aggregierte Kosten für den Wechsel von einer Abstimmungsalternative zur anderen existieren. Es wird kein Anhaltspunkt gegeben, wie die verschiedenen Alternativen an Abstimmungen zu generieren sind. In der Arbeit wird noch einmal explizit auf den Mangel an Lösungen des Rekonfigurationsproblems hingewiesen.

„Während für die Erstinstallation bereits leistungsfähige Verfahren für die genannten Besonderheiten existieren, sollte zukünftig vor allem die Rekonfiguration von Fließsystemen verstärkt in der Forschung berücksichtigt werden.“¹⁴⁹

Scholl¹⁵⁰ definiert das sog. *Re-Engineering* und *Re-Balancing* in seiner Arbeit zum „Balancing and sequencing of assembly lines“. Beide Planungsschritte platziert er vor dem sog. *Master-Sequencing*, in dem erstmals echte Aufträge zur Verfügung stehen. Er beschreibt, dass die Planungsschritte eine erneute Durchführung des *MALBP-2 Problems* unter Verwendung besserer Prognosedaten bedeuten. Des Weiteren erwähnt er, dass beim *Re-Balancing* zusätzliche Ziele verfolgt werden sollten, wie z.B. die Antizipation der Reihenfolgeplanung. Im weiteren Verlauf konzentriert er sich auf die Standard Problemklassen des MALBP und stellt keine Lösung für die Antizipation vor.¹⁵¹

Domschke et al.¹⁵² beschreiben einen Ansatz zur antizipativen mittelfristigen Leistungsabstimmung. Zusätzlich zu den traditionellen Zielen, wie der Minimierung der Stationsanzahl und der Minimierung der Taktzeit, wird auch eine simultane Minimierung der Schwankungen im Fließsystem untersucht. Die nachgelagerten Zielsetzungen zur Vermeidung von Überlastungen werden mit Hilfe von einigen Beispielen verifiziert und verglichen. Auch dieser Ansatz vernachlässigt die Betrachtung einer existierenden Fließbandabstimmung und die Rekonfigurationskosten.

¹⁴⁷Vgl. [Fal05, S.8]

¹⁴⁸Vgl. [BFS07b]

¹⁴⁹Zitat aus [BFS07b, S.9]

¹⁵⁰Vgl. [Sch95, S.106ff.]

¹⁵¹Vgl. dazu die Ausführungen in Abschnitt 3.1.2

¹⁵²Vgl. [DKS96]

Bock¹⁵³ erstellt in seiner Arbeit ein umfassendes und genaues Kostenmodell für den Betrieb einer Variantenfließlinie. Dabei geht er zu Beginn auf die Problematik der MALB-Probleme ein, in denen lediglich eine Durchschnittsvariante verwendet wird, und zeigt die Notwendigkeit auf, echte Varianten zu betrachten. In der Arbeit konzentriert er sich dann hauptsächlich auf die Ein-Produkt-Modelle (SALBP). Die vorgestellte Lösungsmethode führt eine Ersteinstallation einer Fließlinie durch und betrachtet keine Rekonfigurationskosten.¹⁵⁴

Bock et al.¹⁵⁵ stellen eine simultane Lösung der Umsortierung von Arbeitsvorgängen und der Anpassung der Produktionsreihenfolge im Fall von Störungen während des laufenden Betriebs vor. Dabei werden einige Arbeitsvorgänge fixiert, die nicht neu verteilt werden können. Alle anderen Arbeitsvorgänge lassen sich kostenneutral neu zuordnen.

Gamberini et al. beschäftigen sich in „A new multi-objective heuristic algorithm for solving the stochastic assembly line re-balancing problem“¹⁵⁶ mit der Rekonfiguration eines Ein-Produkt-Problems bei stochastischen Montagezeiten. Hierbei wird die Abstimmung angepasst, wenn sich die zu erwartenden Montagezeiten durch Lerneffekte etc. geändert haben. Da jedoch von einem Ein-Produkt-Problem ausgegangen wird, werden keine Zielkriterien einer Variantenfertigung betrachtet.

Mollemeier¹⁵⁷ stellt in „Integrierte Steuerung getakteter Variantenfließlinien“ einen Ansatz vor, in dem innerhalb einer operativen Feinabstimmung der Fließlinie bestimmte Arbeitsvorgänge je Auftrag unterschiedlichen Arbeitsstationen zugewiesen werden. Die Fließbandabstimmung kann somit während der Produktion für jeden Auftrag flexibel vorgenommen werden. Des Weiteren untersucht er, welchen Einfluss die Größe der Überlappungsbereiche auf das Problem hat, und stellt Algorithmen vor, mit denen die optimalen Größen bestimmt werden können.

Bartholdi et al.¹⁵⁸ stellen einen Ansatz mit erhöhtem Flexibilitätsmaß vor, indem die Werker die Stationen entlang laufen und einfachen Regeln folgen, um herauszufinden, welche Arbeitsschritte sie als nächstes durchführen sollen. Hierfür müssen die Werker sämtliche Aufgaben innerhalb einer Fließlinie beherrschen, wodurch sich die Methode eher für komplexe Produkte, wie in der Möbel- oder Bekleidungsindustrie, eignet.

In weiteren Arbeiten wird das Problem der Rekonfiguration nur am Rande erwähnt.¹⁵⁹

Eine explizite Betrachtung des Rekonfigurationsproblems, wie es in Abschnitt 2.2.1 definiert wurde, existiert somit noch nicht; allerdings wird seit langer Zeit Forschung im Bereich der

¹⁵³Vgl. [Boc99, S.81ff.]

¹⁵⁴Vgl. [Boc99, S.122]

¹⁵⁵Vgl. [BRvB06]

¹⁵⁶Vgl. [GGR06]

¹⁵⁷Vgl. [Mol96]

¹⁵⁸Vgl. [BE96]

¹⁵⁹Vgl. [BFS06],[BFS07b] und [BFS07c]

Erstinstallation einer Fließlinie betrieben. Das *Assembly Line Balancing Problem* (ALBP) hat sowohl im Bereich des Operations Research als auch im industriellen Umfeld Aufmerksamkeit auf sich gezogen. Die erste mathematische Formulierung des Problems wurde von Salveson im Jahre 1955 erstellt.¹⁶⁰ Aktuelle Übersichten über den mächtigen Bestand an Literatur zu diesem Problem werden unter anderem von Becker und Scholl¹⁶¹ und Boysen et al.¹⁶² gegeben. Der größte Teil der Literatur beschäftigt sich mit vereinfachten Versionen des Problems, die sich auf die Zuordnung von Arbeitsvorgängen auf Stationen unter Betrachtung von Vorrangrestriktionen konzentrieren. Geht es um Ein-Produkt-Probleme, werden sie *Simple Assembly Line Balancing Problems* (SALBP) genannt; geht es um Mehr-Produkt-Probleme, werden sie *Mixed Model Assembly Line Balancing Problem* (MALBP) genannt.¹⁶³ Erweiterungen um die Betrachtung von parallelen Stationen, Mehrfachbemannung oder U-Linien werden allgemein unter dem Begriff *General Assembly Line Balancing Problem* (GALBP) zusammengefasst. Die Zielsetzungen sind dabei typischerweise die Minimierung der Stationsanzahl und der Taktzeit. Aktuellere Modelle versuchen darüber hinaus, die Produktionskosten zu betrachten und zu minimieren.¹⁶⁴ Eine Betrachtung einer existierenden Fließlinie wird dabei jedoch nicht vorgenommen.

3.1.2 Übertragbare Ansätze aus der Erstinstallation von Fließlinien

Im Folgenden sollen Ansätze zur Erstinstallation von Fließlinien im Hinblick auf ihre Eignung für das Rekonfigurationsproblem diskutiert werden. Die Notation wird hierfür um die Bezeichner in Tabelle 3.1 erweitert.

Im Rahmen der Erstinstallation einer Fließlinie werden hauptsächlich zwei Ziele und deren Kombination verfolgt:

- *MALBP* – 1 Minimierung der notwendigen Stationsanzahl $|ST|$ ¹⁶⁵
- *MALBP* – 2 Minimierung der Taktzeit tz ¹⁶⁶
- *MALBP* – *E* Minimierung von Stationsanzahl $|ST|$ und Taktzeit tz (Maximierung des Bandwirkungsgrads)¹⁶⁷

¹⁶⁰Vgl. [Sal55]

¹⁶¹Vgl. [BS06]

¹⁶²Vgl. [BFS06] und [BFS08]

¹⁶³Vgl. erstmalige Systematisierung der Problemklassen von [Bay86].

¹⁶⁴Vgl. dazu auch [Dan03, S.359ff.]

¹⁶⁵Vgl. zu Ansätzen zur Minimierung der Stationsanzahl [MNP99], [SM98], [Tho70], [VS02], [VS06],[EG99], [RV70]

¹⁶⁶Vgl. zu Ansätzen zur Minimierung der Taktzeit [Saw02]

¹⁶⁷Vgl. [Arc66], [Mac72] sowie [PW97], die eine Modellfamilie vorstellen, welche sowohl Ein- als auch Mehrvariantenprobleme realitätsnah abbildet. Dabei konzentrieren sie sich auf verschiedene Stationskonfigurationen wie z.B. die Parallelisierung.

Symbol	Definition
$\Delta^{\mathcal{AP}}$	Summierte externe Auslastungsunterschiede in allen Arbeitsplätzen $ap \in \mathcal{AP}$ in Zeiteinheiten
$\Delta^{LinTakt}$	Summierte positive lineare Abweichung der Bearbeitungszeiten von der Taktzeit über alle Arbeitsplätze $ap \in \mathcal{AP}$
Δ^{Lin}	Summierte lineare positive und negative Abweichung der Bearbeitungszeiten von der Taktzeit über alle Arbeitsplätze $ap \in \mathcal{AP}$
Δ^p	Summierte interne Auslastungsunterschiede in Zeiteinheiten
$\Delta^{p,2}$	Summierte quadrierte interne Auslastungsunterschiede in Zeiteinheiten zum Quadrat
$\Delta^{p,3}$	Summierte Abweichung der Bearbeitungszeiten aller Varianten von der gesamten Durchschnittsauslastung μ in Zeiteinheiten
$\Delta^{p,max}$	Größter interner Auslastungsunterschied in Zeiteinheiten
μ	Durchschnittliche Bearbeitungszeit für alle Werkstücke $p \in \mathcal{P}$ auf allen Arbeitsplätzen $ap \in \mathcal{AP}$ im Verhältnis zur Kapazität ζ_{ap}
μ_{ap}	Durchschnittliche Bearbeitungszeit für alle Werkstücke $p \in \mathcal{P}$ auf Arbeitsplatz $ap \in \mathcal{AP}$ im Verhältnis zur Kapazität ζ_{ap}
μ_p	Durchschnittliche Bearbeitungszeit des Werkstückes $p \in \mathcal{P}$ auf allen Arbeitsplätzen $ap \in \mathcal{AP}$ in Zeiteinheiten

Tabelle 3.1: Notation zur Formalisierung bestehender Konzepte aus dem Stand der Forschung

Im Rahmen einer Rekonfiguration sind die genannten Zielgrößen irrelevant, da sie zu diesem Zeitpunkt bereits fixierte Parameter darstellen.¹⁶⁸ Darüber hinaus gibt es einige kostenorientierte Ansätze, die jedoch ebenso eine Minimierung der Stationsanzahl und der Taktzeit beabsichtigen und daher nicht übertragbar sind.¹⁶⁹

In der jüngeren Literatur wurden weitere nachgelagerte Ziele zur „gleichmäßigen Auslastung“ einer Fließlinie betrachtet. Dabei wird zwischen der Minimierung von internen und externen Auslastungsunterschieden unterschieden, wie in Abbildung 3.1 verdeutlicht. Die folgenden Ansätze beziehen sich immer auf Arbeitsstationen, in denen nicht zwischen Stationen und Arbeitsplätzen unterschieden wird. Um im Kontext des zuvor beschriebenen Problems zu bleiben, wird der Bezug jeweils auf die Arbeitsplätze abgeändert.

Bei der Minimierung **externer** Auslastungsunterschiede¹⁷⁰ werden die Arbeitsplätze im Durchschnitt möglichst gleich ausgelastet.¹⁷¹ Zur Berechnung werden die durchschnittliche Auslastung je Arbeitsplatz μ_{ap} und die Durchschnittsauslastung der gesamten Fließlinie μ in Gleichung 3.1 definiert.

¹⁶⁸Vgl. dazu Formalisierung in 2.2.1.2

¹⁶⁹Vgl. [Ste77], [Ame96], [Ame00], [Ame98], [Ame01], [Ame06], [BR06], [MF98], [MT03]

¹⁷⁰Häufig verwendetes Synonym ist das „Vertical Balancing“.

¹⁷¹Vgl. Ansätze und leichte Abwandlungen in [MY65], [Tho70, S.595], [RT91], [DKS96], [MNP99, S.2840ff.], [MY01], [Mil02], [VS02], [VS06], [KKK06], [KKK00a]

$$\begin{aligned}\mu_{ap} &= \frac{\sum_{p \in \mathcal{P}} \frac{\tau_{p,ap}}{\zeta_{ap}}}{|\mathcal{P}|} \\ \mu &= \frac{\sum_{ap \in \mathcal{AP}} \mu_{ap}}{|\mathcal{AP}|}\end{aligned}\tag{3.1}$$

Die externen Auslastungsunterschiede sind durch die Abweichungen der einzelnen durchschnittlichen Auslastungen μ_{ap} von der gesamten Durchschnittsauslastung μ in Gleichung 3.2 definiert.

$$\text{Min } \Delta^{\mathcal{AP}} = \sum_{ap \in \mathcal{AP}} |(\mu_{ap} - \mu)|\tag{3.2}$$

In weiteren Ansätzen wird der Fokus auf die **internen** Auslastungsunterschiede¹⁷² gelegt, die jedoch unterschiedlich definiert werden. Übereinstimmende Grundidee ist es, die Bearbeitungszeiten innerhalb der einzelnen Arbeitsplätze möglichst gleichmäßig zu halten.

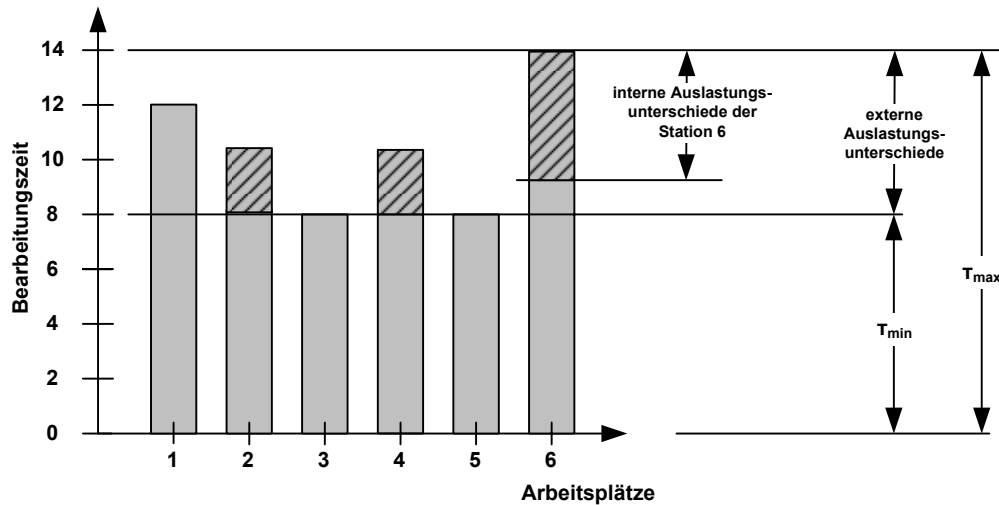


Abbildung 3.1: Vergleich der internen und der externen Auslastungsunterschiede bei einer Variantenproduktion

Minimierung der Abweichung der Bearbeitungszeit einer Variante im Verhältnis zur durchschnittlichen Bearbeitungszeit der Variante

Thomopoulos schlägt vor, eine Abstimmung zu präferieren, in der die einzelnen Varianten auf allen Arbeitsplätzen eine möglichst gleich hohe Auslastung erzeugen (3.3).¹⁷³

¹⁷²Häufig verwendetes Synonym ist das „Horizontal Balancing“.

¹⁷³Vgl. [Tho70] und Erweiterungen um einen Gewichtungsfaktor je Produkt p in [DSV97]

$$\text{Min } \Delta^p = \sum_{ap \in \mathcal{AP}} \sum_{p \in \mathcal{P}} |(\tau_{p,ap} - \mu_p)| \quad (3.3)$$

Domschke et al. untersuchen dieses Zielkriterium mit der Erweiterung, die Abweichungen quadratisch zu bestrafen (3.4).¹⁷⁴

$$\text{Min } \Delta^{p,2} = \sqrt{\sum_{ap \in \mathcal{AP}} \sum_{p \in \mathcal{P}} (|\tau_{p,ap} - \mu_p|)^2} \quad (3.4)$$

Decker betrachtet die Minimierung der stärksten Abweichung (3.5)¹⁷⁵ und erwähnt, dass das Problem der Reihenfolgeplanung durch die Anwendung dieser Zielsetzung vereinfacht wird, da hierdurch Strukturbrüche in den Bearbeitungszeiten der Varianten minimiert werden.¹⁷⁶ Strukturbrüche zeichnen sich dadurch aus, dass für eine Variante unterschiedliche Bearbeitungszeiten in unterschiedlichen Teilen der Fließlinie notwendig sind (heterogene Produkte). Durch das hier diskutierte Zielkriterium wird eine Homogenisierung von heterogenen Produkten verfolgt; jedoch ergeben sich diese Strukturbrüche in der Praxis zwangsläufig. So existieren häufig Teilbereiche der Produktion, in denen die Bearbeitungszeiten nur sehr geringen Schwankungen unterzogen sind, und andere Bereiche, in denen hohe Belastungsschwankungen verarbeitet werden müssen.¹⁷⁷

$$\text{Min } \Delta^{p,max} = \max [|(\tau_{p,ap} - \mu_p)|] \quad (3.5)$$

Minimierung der Abweichungen bezogen auf die gesamte Durchschnittsauslastung

Matanachai und Yano untersuchen die Abweichungen der Bearbeitungszeiten von der gesamten Durchschnittsauslastung μ (3.6).¹⁷⁸

$$\text{Min } \Delta^{p,3} = \sum_{ap \in \mathcal{AP}} \sum_{p \in \mathcal{P}} |(\tau_{p,ap} - \mu)| \quad (3.6)$$

Minimierung der Abweichungen bezogen auf die Taktzeit

Domschke et al.¹⁷⁹ untersuchen einige dieser Ansätze hinsichtlich ihres Potenzials, Überlas-

¹⁷⁴Vgl. [DKS96, S.1484]

¹⁷⁵Vgl. [Dec93]

¹⁷⁶Vgl. [Dec93, S.78]

¹⁷⁷Aufgrund der technischen Vorrangrestriktionen lässt sich keine vollständige Verteilung der varianzbehafteten Inhalte über alle Linienabschnitte erreichen.

¹⁷⁸Vgl. [MY01]

¹⁷⁹Vgl. [DKS96] - Untersuchungen von nachgelagerten Zielsetzungen für die Erstinstallation einer Fließlinie mit dem Ziel der Betrachtung operationaler Aspekte.

tungen durch die spätere Reihenfolgeplanung zu vermeiden, und stellen ein weiteres Kriterium vor, welches nur die positiven linearen Abweichungen von der Taktzeit betrachtet (3.7).

$$\text{Min } \Delta^{LinTakt} = \sum_{ap \in \mathcal{AP}} \sum_{p \in \mathcal{P}} \max[0; |(\tau_{p,ap} - tz)|] \quad (3.7)$$

Des Weiteren wird eine Zielsetzung mit der gleichzeitigen Reduzierung von positiven und negativen linearen Abweichungen formuliert (3.8).

$$\text{Min } \Delta^{Lin} = \sum_{ap \in \mathcal{AP}} \sum_{p \in \mathcal{P}} |(\tau_{p,ap} - tz)| \quad (3.8)$$

Bei den in der Arbeit von Domschke et al. durchgeführten Studien schneidet das Kriterium $\Delta^{LinTakt}$ zur Minimierung von Überlastungen am besten ab.¹⁸⁰

Bukchin¹⁸¹ diskutiert in seiner Arbeit unterschiedliche antizipative Zielkriterien, geht dabei aber von einer variablen Taktzeit aus und untersucht die Kriterien hinsichtlich des Potentials, den Ausstoß eines Fließsystems zu erhöhen.

Boysen¹⁸² systematisiert die existierenden antizipativen Zielkriterien zur impliziten Vermeidung von Überlastungen im Fließsystem, verwendet diese jedoch nicht in seinen Ansätzen und konzentriert sich auf die Erstinstallation von Fließlinien ohne nachgelagerte Zielkriterien.

Für die nachgelagerten, antizipativen Zielkriterien existiert eine Vielzahl an leistungsfähigen und ausgereiften Lösungsmethoden, die jedoch allesamt nicht von einem existierenden Fließsystem ausgehen.¹⁸³

3.1.3 Fazit

Obwohl der Bedarf eines Konzeptes zur Rekonfiguration in einigen Arbeiten explizit konstatiert wird, existiert noch keine Methode, die die Anpassung einer bestehenden Fließlinie vornimmt und dabei den Zielkonflikt zwischen der Minimierung von Rekonfigurationskosten und Betriebskosten betrachtet. Im Kontext der Erstinstallation von Fließlinien wurden jedoch einige Anstrengungen unternommen, nachgelagerte Zielkriterien zu entwickeln, die die Belastungsschwankungen im Fließsystem minimieren. Eine Übertragung dieser Zielkriterien auf das in dieser Arbeit zu lösende Rekonfigurationsproblem erscheint sinnvoll und soll im Konzeptteil der Arbeit diskutiert werden.

¹⁸⁰Vgl. [DKS96, S.1483]

¹⁸¹Vgl. [Buk98]

¹⁸²Vgl. [Boy05b, S.147]

¹⁸³Eine gute Übersicht zu den existierenden Methoden geben Boysen et al. in [BFS08] und Becker und Scholl in [BS03].

3.2 Stand der Forschung zur Erstellung von Vorranggraphen

Der Stand der Forschung zur Erstellung von Vorranggraphen kann in manuelle und automatische bzw. hybride Methoden geteilt werden und wird in dieser Strukturierung im Folgenden vorgestellt.

3.2.1 Manuelle Methoden zur Erstellung von Vorranggraphen

Die älteste und einfachste Methode, einen Vorranggraphen zu erstellen, ist die manuelle Aufnahme durch Fragen an einen Montageplaner:

Die Vorrangrestriktionen zwischen Teilen werden normalerweise festgestellt durch Antworten eines Montageplaners auf die Frage, ob zwei Teile eine Vorrangbeziehung beinhalten.¹⁸⁴

Darauf aufbauend haben sich zwei unterschiedliche Vorgehensweisen herauskristallisiert, um die Anzahl an Fragen zu reduzieren.

Im ersten Ansatz, eingeführt von Bourjault et al.¹⁸⁵ und De Fazio et al.¹⁸⁶, konzentriert man sich auf eine Verbindung von zwei Teilen aus einem zu montierenden Produkt. Für jede dieser Verbindungen wird gefragt, welche anderen Verbindungen vorher existieren müssen und welche erst danach hergestellt werden können. Die Antworten werden überführt in Vorrangbedingungen zwischen logischen Kombinationen von Verbindungen. Zur vollständigen Erstellung eines Vorranggraphen nach dieser Methode müssen nur $2 \cdot V$ Fragen für ein Produkt mit V Verbindungen beantwortet werden. Diese relativ geringe Anzahl notwendiger Fragen geht jedoch einher mit einer komplizierten Fragestellung an den Planer. Zur Aufnahme einer Vorrangbedingung müssen alle möglichen Rahmenbedingungen beachtet werden, die die betroffenen Montageverbindungen erlauben oder verbieten. Selbst bei Produkten mit geringer Komplexität ist diese Methode sehr fehlerträchtig.¹⁸⁷

In einem alternativen Ansatz wird die Eigenschaft genutzt, dass eine Reihenfolge zur Demontage eines Produktes mit der Montagereihenfolge umgekehrt übereinstimmt. Das vollständige, fertig montierte Produkt wird Stück für Stück zerlegt. In jedem Zustand wird zu jedem Arbeitsvorgang gefragt, ob er in dieser Situation durchführbar wäre. Diese Frage ist recht einfach gestaltet und es reicht eine Ja/Nein-Antwort. Um die immense Anzahl an notwendigen Fragen bei diesem Ansatz zu reduzieren, kann die sog. Unter-/Obermengen-Regel von Bourjault¹⁸⁸

¹⁸⁴Vgl. [Che89, S.243] und [JK84] sowie [DFW87]

¹⁸⁵Vgl. [Bou84]

¹⁸⁶Vgl. [DFW87]

¹⁸⁷Vgl. Aussage von [DFW87] zu ihrem eigenen Ansatz.

¹⁸⁸Vgl. [Bou84]

angewendet werden. Dabei werden bereits existierende Antworten auf neue Fragen übertragen, so dass diese nicht noch einmal gestellt werden müssen. Eine genaue Beschreibung dieses Vorgehens wurde von Wilson¹⁸⁹ vorgenommen. Trotz der Anwendung dieses Konzeptes müssen für die vollständige Ermittlung sämtlicher Vorrangbedingungen bei Produkten mit 10 bis 20 Teilen bereits einige hundert Fragen gestellt werden.¹⁹⁰ Baldwin et al.¹⁹¹ und Henrioud und Bourjault¹⁹² führen zur weiteren Reduzierung der Anzahl Fragen die sog. “Strategische Regeln” ein, die für einige Arbeitsvorgänge bereits eine feste Reihenfolge vorgeben.

Hardeck¹⁹³ stellt einen Ansatz vor, in welchem der Planer zwar immer noch jede Frage einzeln beantworten muss, jedoch mit Hilfe eines graphischen Modellierungstools unterstützt wird. Eine Maßnahme zur Reduktion der Komplexität wird nicht vorgestellt. Müller¹⁹⁴ zeigt ebenso einen Ansatz auf, der eine Unterstützung bei der Erstellung eines Vorranggraphen durch eine geeignete grafische Darstellung bietet. Er ergänzt den Vorranggraphen um zusätzliche Schichten, welche einem Arbeitsvorgang unterschiedliche Bearbeitungszeiten zuordnen können: eine betriebsmittellorientierte, eine mitarbeiterorientierte und eine arbeitsgegenstandsorientierte Bearbeitungszeit. Eine neue Methode zur Erstellung der Vorrangbedingungen wird jedoch nicht vorgestellt.

3.2.2 Automatisierte und hybride Methoden zur Erstellung von Vorranggraphen

Bereits früh wurde erkannt, dass die steigende Komplexität von Produkten und damit die Größe und Komplexität des Vorranggraphen eine manuelle Erstellung sehr fehleranfällig und aufwendig macht.¹⁹⁵ Daher wurden unterschiedliche Ansätze verfolgt, den Vorranggraphen automatisiert oder teilautomatisiert zu erstellen. Diese lassen sich unterteilen in geometriedatenorientierte Ansätze, die aus Geometrieinformationen Vorrangbedingungen ableiten, Stücklistenorientierte Ansätze, die ihre Informationen aus der Produktdokumentation extrahieren, sowie Ansätze, die auf Basis von neu definierten Datenkonstrukten Vorranggraphen erstellen. Die meisten Systeme sind dabei als hybrid zu klassifizieren, da sie eine manuelle Nachbehandlung oder die Anwendung zweier Systematiken beinhalten.

¹⁸⁹Vgl. [Wil95, S.310]

¹⁹⁰Vgl. [Wil95, S.309]

¹⁹¹Vgl. [BAL⁺91]

¹⁹²Vgl. [HB91]

¹⁹³Vgl. [Har74]

¹⁹⁴Vgl. [Mül02]

¹⁹⁵Vgl. dazu die Diskussion in [Bra95, S.20]

3.2.2.1 Geometriedatenorientierte Ansätze

Geometriedatenorientierte Ansätze verwenden CAD-Informationen über jedes Einzelteil in einem Werkstück und führen eine Kollisionsanalyse durch, um mögliche Demontagereihenfolgen abzuleiten.

Kaufman et al.¹⁹⁶ stellen das *Archimedes Mechanical Assembly Planning System* vor. In dem System wird eine geometrische Analyse durchgeführt, um mögliche Bearbeitungsreihenfolgen für Arbeitsvorgänge zu definieren.

Sanderson et al.¹⁹⁷ stellen mit dem *Pleiades-System* ein hybrides Konzept vor, mit welchem sie aus vollständigen geometrischen Daten, einem Beziehungsmodell der Teile sowie einer Verbindungs- und Fügungsmatrix mögliche Bearbeitungsreihenfolgen von Arbeitsvorgängen ableiten. Sie verwenden einen AND/OR Graphen zur Visualisierung der Bearbeitungsreihenfolgen. Aus diesem wird dann ein Vorranggraph abgeleitet.

Weitere Konzepte zur Ableitung von Bearbeitungsreihenfolgen aus CAD-Informationen wurden z.B. von Bourjault et al.¹⁹⁸, De Fazio et al.¹⁹⁹ und Santochi und Dini²⁰⁰ sowie Lin et al.²⁰¹ vorgestellt.

Wilson²⁰² erstellt einen hybriden Ansatz, in welchem nach einer CAD-Analyse weitere Vorrangbedingungen manuell abgefragt werden. Der Fokus der Arbeit liegt darauf, die Anzahl notwendiger Fragen an den Montageplaner zu reduzieren.

Jones et al.²⁰³ bauen ebenfalls auf einer CAD-basierten Suche nach möglichen Bearbeitungsreihenfolgen auf.²⁰⁴ Auf dieser Basis stellen sie ein Konzept vor, mit dem ein Planer nachträglich weitere Informationen zu den erkannten Vorrangbedingungen ergänzen kann.

3.2.2.2 Stücklistenorientierte Ansätze

Ammer stellt einen Ansatz zur rechnerunterstützten Planung von Montageablaufstrukturen vor. Dafür bildet er einen Vorranggraphen als Zwischenergebnis. Die Vorgehensweise ist in Grafik 3.2²⁰⁵ dargestellt und soll im Folgenden genauer beschrieben werden.

¹⁹⁶Vgl. [KWJC96]

¹⁹⁷Vgl. [SHdMZ90]

¹⁹⁸Vgl. [Bou84]

¹⁹⁹Vgl. [DFW87]

²⁰⁰Vgl. [SD92]

²⁰¹Vgl. [LC93]

²⁰²Vgl. [Wil95]

²⁰³Vgl. [JWC97]

²⁰⁴Sie verwenden das *Archimedes Mechanical Assembly Planning System* aus [KWJC96].

²⁰⁵Vgl. [Amm85, S.35]

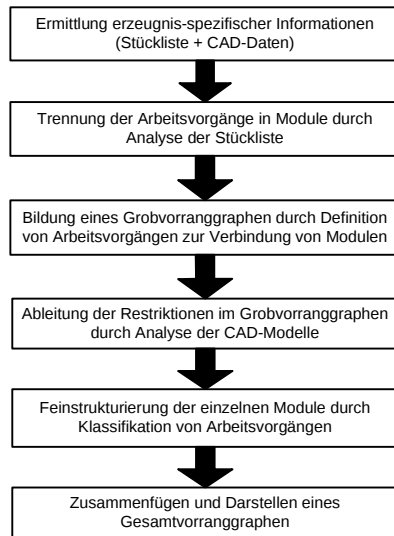


Abbildung 3.2: Vorgehen von Ammer zu Vorranggraphenbildung

Im ersten Schritt wird die Stückliste analysiert und die Erzeugnisstruktur in einem Strukturbaum aufbereitet, in dem hierarchisch dargestellt ist, welche Bauteile in welche Baugruppe eingehen und welche Baugruppen zusammen wieder in eine höhere Baugruppe eingehen. Um von einer Ebene dieses Strukturbaumes in die nächste zu gelangen, ist laut Ammer mindestens ein Fügearbeitsvorgang notwendig. Dieser wird automatisch erstellt und vom Planer benannt. Die möglichen Abarbeitungsreihenfolgen dieser Vorgänge werden nun mit Hilfe einer CAD-Analyse erstellt. Ergebnis ist ein Grobvorranggraph, der die Vorrangbedingungen zwischen den Baugruppen/Modulen beinhaltet. Im nächsten Schritt folgt eine Feinstrukturierung der einzelnen Arbeitsvorgänge in der untersten Ebene des Strukturbaumes, indem je Einzelteil Arbeitsvorgänge manuell als vorbereitend, fügend, nachbereitend, prüfend oder justierend klassifiziert werden. Anschließend werden diese Arbeitsvorgänge automatisch nach einer Standardabfolge der Klassen in eine Reihenfolge gebracht.²⁰⁶ Die resultierenden Einzelvorranggraphen werden in den Grobvorranggraphen integriert. Im letzten Schritt werden die je Variante erstellten Vorranggraphen in einem Mischgraphen zusammengeführt.²⁰⁷ Im Anschluss werden die Arbeitsvorgänge für eine graphische Darstellung sortiert, so dass eine geeignete Darstellung erreicht wird.²⁰⁸

Hellberg²⁰⁹ beschreibt mögliche Wege, eine gültige Reihenfolge der Arbeitsvorgänge abzuleiten, wenn die Vorrangbedingungen zwischen den einzelnen Arbeitsvorgängen vorher bekannt

²⁰⁶Vgl. [Amm85, S.57-73]

²⁰⁷Vgl. [Tho67, S.60ff.] und [Gör78, S.47ff.] sowie [Bra95, S.53ff.] und [DSV93, S.227]

²⁰⁸Vgl. [Amm85, S.74-77]

²⁰⁹Vgl. [Hel93]

sind.²¹⁰ Dabei baut er auf Stücklisten auf, zersplittet das Produkt immer weiter in Zwischenprodukte und baut mit heuristischen Lösungsmethoden einen Suchbaum auf.

3.2.2.3 Ansätze unter Verwendung erweiterter Produktdokumentation

Neben den oben dargestellten Methoden existieren einige Ansätze zur automatischen Ableitung von Bearbeitungsreihenfolgen, die auf eigens dafür entwickelten Datenstrukturen aufbauen. Dazu gehören das Liaison-Diagramm von Bourjault et al.²¹¹, das Beziehungsmodell von Homem et al.²¹², der Montagerestriktionsgraph von Wolter²¹³, der funktionsbezogene Verbindungs- und Bearbeitungsgraph von Huang und Kusiak²¹⁴, die hierarchische Objektrepräsentation von Yokota und Rough²¹⁵, der Kontakt-Graph und die Vorgangsmatrix von Santochi und Dini²¹⁶, der Teilverknüpfungsgraph in Verbindung mit dem Zusammenhangsgraph und dem Spaltmaßrestriktionsgraph von Lin und Chang²¹⁷ und das mathematische Modell von Dini und Santochi²¹⁸ sowie der Teileverknüpfungslevelgraph von Cho et al.²¹⁹. Alle genannten Datenstrukturen sind lediglich Zwischenschritte und müssen existieren, bevor ein Vorranggraph abgeleitet werden kann.

3.2.3 Fazit

Zusammenfassend kann gesagt werden, dass dem Thema der Vorranggraphenentwicklung bereits große Aufmerksamkeit in der Literatur gewidmet wurde. Jedoch treten je nach Ansatz unterschiedliche Probleme hinsichtlich der Übertragbarkeit auf das diskutierte Problem auf.

Die Geometriedatenorientierten Ansätze haben den entscheidenden Nachteil, dass sie mit weichen Bauteilen nicht richtig umgehen können.²²⁰ Insbesondere elastische Materialien, wie Klipse, Dichtungsringe, Gummimanschetten und andere Kunststoffteile können von den Algorithmen nicht korrekt demontiert werden, da hierfür eine Verformung der Materialien notwendig ist und die Systeme keine Information über die Materialeigenschaft haben. Des Weiteren vernachlässigen diese Ansätze die praktische Machbarkeit der Montagereihenfolgen, die z.B. durch den Platzbedarf für das Ansetzen eines Werkzeuges weiter eingeschränkt ist. Darüber

²¹⁰Vgl. [Hel93, S.34]

²¹¹Vgl. [Bou84]

²¹²Vgl. [HdMS91]

²¹³Vgl. [Wol89]

²¹⁴Vgl. [HL89]

²¹⁵Vgl. [YR92]

²¹⁶Vgl. [SD92]

²¹⁷Vgl. [LC93]

²¹⁸Vgl. [DS92]

²¹⁹Vgl. [CSC93]

²²⁰Vgl. [KWJC96, S.3368]

hinaus lassen sich aus Geometriedaten lediglich Fügearbeitsvorgänge ableiten; Laufwege, Vorarbeiten oder Prüfvorgänge werden vernachlässigt. Eine Variantenbetrachtung ist in keinem der Ansätze vorgesehen. Eine Übertragung der Methoden auf das in dieser Arbeit vorliegende Problem ist somit nicht möglich.

Der Stücklistenorientierte Ansatz von Ammer beinhaltet einige interessante Aspekte, um sich dem Problem unter Verwendung von Stücklisten zu nähern. Allerdings werden auch hier CAD-basierte Daten verwendet, um den Grobvorranggraphen zu erstellen. In dem Ansatz wird davon ausgegangen, dass sich ein Arbeitsvorgang immer auf ein einzelnes Teil bezieht. Gerade bei sehr komplexen Produkten ist eine Verbindung zwischen zwei Bauteilen häufig nicht durch nur einen Arbeitsvorgang möglich. Dieses Problem tritt z.B. auf, wenn in der Automobilindustrie der Motor in eine Karosserie eingebaut wird und Hunderte von Arbeitsvorgängen benötigt werden, um die notwendigen Verbindungen zwischen dem Motor und der Karosserie herzustellen. Diese Vorgänge beziehen sich nicht nur auf eine höher- oder tieferliegende Ebene im Strukturbaum, wie von Ammer angenommen, sondern auch auf Bauteile, die sich in derselben Ebene des Strukturbaumes befinden. Daher lässt sich diese Lösungsmethode nicht auf das vorgestellte Problem übertragen.

Die Verfahren zur Erstellung von Vorranggraphen unter Verwendung von zusätzlichen Datenstrukturen verlagern lediglich das Problem und klären nicht, wie diese zusätzlichen Datenstrukturen gefüllt werden.

Die manuelle Erstellung von Vorranggraphen kann in jedem Fall unabhängig von den Voraussetzungen angewendet werden. Bei komplexen, variantenreichen Produkten scheitern diese Ansätze jedoch aufgrund des hohen Aufwandes sowie der Fehleranfälligkeit. Dennoch scheint die manuelle Aufnahme eines Vorranggraphen die einzige Möglichkeit zu sein, auch die speziellen Anforderungen einer manuellen Montage zu betrachten. Eine Übertragung der Ansätze soll daher im Konzeptteil der Arbeit geprüft werden.

Bei allen vorgestellten Ansätzen können Vorrangbedingungen nicht erkannt oder vergessen werden, wodurch Freiheitsgrade entstehen, die im Rahmen einer Rekonfiguration zu ungültigen Lösungen führen. Dies führt zu einer mangelnden Akzeptanz der Systeme in der Praxis.

3.3 Stand der Forschung zur Reihenfolgeplanung

In der Literatur zur Reihenfolgeplanung wurden zwei grundsätzlich verschiedene Zielsetzungen verfolgt:²²¹

Bedarfsglättung für Zwischenprodukte: Diese Ansätze konzentrieren sich auf eine Planung der Reihenfolge, in der eine gleichmäßige Nachfrage nach Zwischenprodukten und Zulieferteilen entsteht. Dabei können das **Level-Scheduling**²²² und das **Min-Inventory-Scheduling**²²³ unterschieden werden. Letztes betrachtet zusätzlich zum gleichmäßigen Bedarf, dass die Anlieferung von Bauteilen in unterschiedlich großen Mengen vorgenommen wird, und beabsichtigt die Minimierung des notwendigen Lagerbedarfs an der Fließlinie.²²⁴ Beide Konzepte vernachlässigen die Bearbeitungszeitschwankungen in der Fließlinie und werden daher im Folgenden nicht weiter betrachtet. Die notwendige Betrachtung logistischer Restriktionen ist in der zugrunde liegenden Problemstellung dieser Arbeit bereits durch die mittelfristige Programmplanung eingehalten.²²⁵ Eine Zusammenstellung der Level-Scheduling Ansätze wird von Boysen et al.²²⁶ gegeben.

Vermeidung von Überlastungen: Bei diesen Ansätzen werden die Werkstücke in einer Reihenfolge so verteilt, dass sich Varianten mit großen Bearbeitungszeiten und Varianten mit geringen Bearbeitungszeiten abwechseln. Hierdurch wird versucht, eine möglichst gleichmäßige Auslastung und damit effiziente Nutzung der eingesetzten Arbeitssysteme zu erreichen. Zwei grundsätzliche Methoden zur Vermeidung von Überlastungen können unterschieden werden. Das **Car-Sequencing**²²⁷ nimmt die Reihenfolgeplanung anhand von abstrahierten Regeln vor; sog. $H_0 : N_0$ Dichtekriterien, welche definieren, wie häufig eine Ausstattungsoption innerhalb einer bestimmten Anzahl Produkte in der Reihenfolge auftreten darf. Eine typische Regel wäre z.B.: *1 aus 5 (Option A)*. Dies bedeutet, dass in fünf aufeinander folgenden Aufträgen nur ein Auftrag die Option A beinhalten darf. Großer Nachteil an den Car-Sequencing Ansätzen ist, dass es bis dato keine geeignete Methode gibt, die Regeln analytisch herzuleiten.²²⁸ Eine Definition der Regeln rein durch die Intuition von Planern zeigt in der Praxis keine zufriedenstellenden Ergebnisse. Ebenfalls stellt sich die Pflege dieser Regeln als sehr schwierig dar, weil nicht klar ist, wann und wie sie angepasst werden müssen. Werden zu viele Regeln definiert, so wird das Problem durch die Kombination der Regeln unlösbar und es können nur noch Ansätze verfolgt werden, die die Verletzungen dieser Regeln minimieren. Eine

²²¹Vgl. [BSJ94]

²²²Die Idee zum Level-Scheduling entstammt dem berühmten Toyota-Production-System. Vgl. [Mon98]

²²³Vgl. [BFS07e]

²²⁴Vgl. [FF89]

²²⁵Vgl. dazu Abschnitt 2.2.3.1 und 2.2.3.3

²²⁶Vgl. [BFS07d, S.16-19]

²²⁷Eingeführt von [Par88] in der Automobilindustrie und in der Theorie erstmals vorgestellt in [PKW86]

²²⁸Vgl. [BFS07d, S.21 und S.23u], nur Bolat und Yano konzentrieren sich in [BY92b] auf diese Fragestellung, erzeugen jedoch nur Regeln für einfache Ja/Nein-Optionen.

Klassifikation und Sammlung verschiedener Ansätze zum Car-Sequencing wird von Boysen et al.²²⁹ vorgestellt.²³⁰ Das **Mixed-Model-Sequencing**²³¹ betrachtet explizit die Bearbeitungszeiten der einzelnen Varianten sowie deren Auswirkungen und soll daher im Folgenden genauer untersucht werden.

3.3.1 Ansätze zum Mixed-Model Sequencing

Um die Fülle an existierenden Ansätzen zu beherrschen, wird eine Klassifikation vorgenommen. Dabei werden ausschließlich die Ansätze eingeordnet, welche mit deterministischen Bearbeitungszeiten und einer seriellen Linienanordnung arbeiten, um den Rahmen nicht zu groß werden zu lassen.

3.3.1.1 Klassifikationskriterien

Bandeigenschaften

Offene Stationsgrenzen

- Es gibt keinen Driftbereich.
- Die Werker können über ihre offenen Stationsgrenzen hinaus in Driftbereichen arbeiten.

Parallelarbeit in Station

- Keine Parallelarbeit innerhalb der Stationen.
- Parallelarbeit an unterschiedlichen Arbeitsplätzen je Station.

Abhängigkeiten zwischen Werkern

- Es werden keine Abhängigkeiten zwischen den Werkern betrachtet. Ein Überholen der Werker untereinander wird zugelassen.
- Es kann individuell je Arbeitsplatz definiert werden, welche Abhängigkeiten existieren.
- Die Werker sind voneinander abhängig, ein Überholen ist nicht möglich.

Parallele Stationen

- Es werden keine parallelen, gleich ausgestatteten Stationen betrachtet.

²²⁹Vgl. [BFS07d]

²³⁰Aktuelle Lösungsansätze sowie theoretische Betrachtungen zum Car-Sequencing finden sich in [Ben08], [BNM08], [CLP08], [DH08], [Gav08], [PR08], [RAN⁺08a], [RAN⁺08b] und [SCNA08], [Sol08]

²³¹Eingeführt von [WK64]

- Parallele Stationen mit gleicher Ausstattung sind möglich und werden betrachtet.

Flexible Anzahl Stationen

- Die Anzahl Stationen ist fix (meist 1 zur theoretischen Betrachtung).
- Die Anzahl Stationen ist in der Betrachtung flexibel wählbar.

Rüstkosten

- Es werden keine Rüstkosten und Rüstzeiten betrachtet.
- Rüstkosten werden betrachtet.

Verschiedene Stationstypen

- Alle Stationen sind in ihrer Charakteristik homogen.
- Es existieren unterschiedliche Stationen (z.B. offene und geschlossene) im Fließsystem.

Steuerung

Reaktion auf Überlast

- u* Die Überlastungen werden durch Unterstützer abgearbeitet.
- st* Bei Überlastung wird ein Bandstopp eingeleitet.
- c* Überlastungen werden anderweitig verarbeitet, z.B. durch Ausschleusen eines Werkstückes.

Werkerrücklauf vernachlässigt

- Die Zeit für den Rückweg ist vernachlässigbar, beziehungsweise indirekt als Arbeitsvorgang modelliert.
- Die Zeit für den Rückweg ist variabel und wird in die Betrachtung direkt einbezogen.

Auflegeintervall fixiert

- Das Auflegeintervall, in dem die Werkstücke in das Montagesystem eingelastet werden, ist variabel.
- Das Auflegeintervall, in dem die Werkstücke in das Montagesystem eingelastet werden, ist fix.

Zielsetzung

Bei den Zielsetzungen wird jeweils durch einen vollen Kreis ● signalisiert, dass dieses Zielkriterium betrachtet wird, sonst ist ein leerer Kreis ○ dargestellt.

- Min Linienlänge: Die Linienlänge und damit die Anzahl Stationen wird minimiert.
- Min Bandstillstände: Die Anzahl an notwendigen Bandstillständen wird minimiert.
- Min maximale Überlast: Die maximale Überlast der Werker wird minimiert.
- Min Leerzeit: Die summierte Leerzeit der Werker wird minimiert.
- Min Überlast: Die summierte Überlastungszeit der Werker wird minimiert.
- Min Unterstützereinsätze: Die Anzahl an notwendigen Unterstützereinsätzen zur Kompensation der Überlastungen wird minimiert.
- Min notwendige Unterstützer: Die Anzahl an notwendigen Unterstützereinsatzkräften wird minimiert.
- Kostenbetrachtung: Die Zielkriterien sind mit einem Kostenfaktor versehen und die Gesamtkosten werden minimiert.

Methode

Die Ansätze stellen jeweils unterschiedliche Methoden mit Bezug auf das klassifizierte Problem vor:

M - Mathematisches Modell

E - Exakte Lösungsmethode

H - Heuristische Lösungsmethode

T - Theoretische Untersuchungen des Problems (Lösungsraumgröße etc.)

3.3.1.2 Einordnung existierender Ansätze

In Tabelle 3.2 werden existierende Lösungsansätze zum Mixed-Model-Sequencing in das vorgestellte Klassifikationsschema eingeordnet. Außerdem wird das in dieser Arbeit betrachtete Reihenfolgeplanungsproblem, wie in Abschnitt 2.2.3 definiert, als Referenzproblem klassifiziert und in die Tabelle eingeordnet.

Zusätzlich zu den klassifizierten Arbeiten existieren einige Methoden, die das Reihenfolgeplanungsproblem und das Fließbandabstimmungsproblem simultan lösen.²³² Leopold²³³ stellt z.B. einen Ansatz zur simultanen Personal- und Reihenfolgeplanung vor, klammert aber die Möglichkeit der Verwendung von Unterstützern in seinem Ansatz aus.²³⁴ Sofern mit dem

²³²Vgl. Ansätze von [Boc99], [Dec93], [KOP07], [KKK00b], [KKK00a], [MNP99], [Mil02], [Mol96], [RDDB02], [Sch95] und [SBF06]

²³³Vgl. [Leo97]

²³⁴Vgl. [Leo97, S.35]

Ansatz	Bandeigenschaften							Steuerung		Zielsetzung										Methode
	Offene Stationsgrenzen	Parallelarbeit in Station	Abhängigkeiten zw. Werkern	Parallele Stationen	Flexible Anzahl Stationen	Rüstkosten	Verschiedene Stationstypen	Reaktion auf Überlast	Werkerrücklauf vernachl.	Auflegeintervall fixiert	Min Linienlänge	Min Durchlaufzeit	Min Bandstillstände	Min maximale Überlast	Min Leerzeit	Min Überlast	Min Unterstützereinsätze	Min notwendige Unterstützter	Kostenbetrachtung	
Referenzproblem	●	●	○	○	●	○	●	u	●	●	○	○	○	○	○	○	●	●	●	M
Bard et al. [BDES92]	●	○	●	○	●	○	○	c	●	○	●	●	○	○	○	○	○	○	○	M
Bolat [Bol97]	○	○	●	○	○	○	○	c	○	●	○	○	○	○	○	●	○	○	○	MH
Bolat und Yano [BY92a]	○	○	●	○	○	○	○	c	●	●	○	○	○	○	○	●	○	○	○	MH
Bolat und Yano [BY92b]	○	○	●	○	○	○	○	c	●	●	○	○	○	○	○	●	○	○	○	T
Bolat et al. [BSAF94]	○	○	○	○	○	●	○	c	○	●	○	○	○	○	○	●	○	○	●	MEHT
Boysen [Boy05a]	○	○	●	●	●	○	○	c	●	●	○	○	○	○	○	●	○	○	○	MHT
Chutima et al. [CNY03]	○	○	●	○	●	○	○	c	●	●	○	●	○	○	○	○	○	○	○	H
Dar-El [DE78]	○	○	●	○	●	○	●	c	○	○	●	●	○	○	○	○	○	○	○	H
Dar-El und Cother [DEC75]	○	○	●	○	●	○	○	c	●	●	●	○	○	○	○	○	○	○	○	HT
Dar-El und Cucuy [DEC77]	●	○	●	○	○	○	○	c	●	●	●	○	○	○	○	○	○	○	○	H
Dar-El und Nadivi [DEN81]	●	○	●	○	●	○	○	c	●	●	●	○	○	○	○	○	○	○	○	H
Decker [Dec92]	●	○	●	○	●	○	○	c	●	●	○	○	○	○	○	●	○	○	○	H
Decker [Dec93]	●	○	●	○	●	○	○	c/u	●	●	○	○	○	○	○	●	○	○	○	H
Domschke [DSV97]	○	○	●	○	●	○	○	c	●	●	○	○	○	○	○	○	○	○	●	M
Felbecker [Fel80]	○	○	○	○	●	○	○	c	●	●	○	○	○	○	●	○	○	○	○	MH
Goldschmidt et al. [GBT97]	●	○	●	○	○	○	○	c	●	●	○	○	○	○	○	○	○	○	○	T
Kim et al. [KHK96]	○	○	●	○	●	○	○	c	●	●	○	○	○	○	○	○	○	○	○	H
Koether [Koe85]	●	○	●	○	●	○	●	c/u	●	●	○	○	○	○	○	●	○	○	●	H
Macaskill [Mac72]	●	○	○	○	●	○	○	c	●	●	○	○	○	○	●	●	○	○	○	HT
Mollemeier [Mol96]	●	○	●	○	●	○	○	u	○	○	○	○	○	○	○	●	●	○	○	MHT
Okamura und Yamashina [OY79]	●	○	○	○	●	○	●	c	○	○	○	○	○	○	○	○	○	○	○	H
Sarker und Pan [SH98],[SP01]	●	○	●	○	●	○	○	c	○	○	○	○	○	○	●	●	○	○	●	M
Schneeweiss und Söhner [SS91]	●	○	●	○	●	○	○	c	●	●	○	○	○	○	○	●	○	○	○	H
Scholl [Sch95]	○	○	●	○	●	○	○	c	●	●	○	○	○	○	○	●	○	○	○	MEHT
Scholl et al. [SKD98]	○	○	●	○	●	○	○	c	●	●	○	○	○	○	○	●	○	○	○	MH
Sumichrast et al. [SOC00]	○	○	●	○	●	○	○	c	●	●	○	○	○	○	○	●	○	○	○	MH
Thomopoulos [Tho67]	●	○	●	○	●	○	○	c	●	●	○	○	○	○	●	●	○	○	●	H
Tsai [Tsa95]	○	○	●	○	○	○	○	c	○	○	○	○	○	○	○	●	○	○	○	ET
Wester und Kilbridge [WK64]	○	○	●	○	○	○	○	c	●	●	○	○	○	○	●	●	○	○	○	H
Xiaobo und Ohno [XO94]	○	○	●	○	●	○	○	st	●	●	○	○	●	○	○	○	○	○	○	MT
Xiaobo und Ohno [XO97]	○	○	●	○	●	○	○	st	○	○	○	○	○	○	○	○	○	○	○	EHT
Xiaobo und Ohno [XO00]	○	○	●	○	○	○	○	st	○	○	○	○	●	○	○	○	○	○	○	HT
Yano und Bolat [YB89]	○	○	●	○	●	○	○	c	●	●	○	○	○	○	○	●	○	○	○	M
Yano und Rachamadugu [YR91]	○	○	●	○	○	○	○	c	●	●	○	○	○	○	○	●	○	○	○	MEH
Yoo et al. [YSH05]	○	○	●	○	●	○	○	c/u	○	○	○	○	○	○	○	○	○	○	○	MH

Tabelle 3.2: Klassifikationstabelle von Mixed-Model-Sequencing Ansätzen zur Reihenfolgeoptimierung

vorgestellten heuristischen Verfahren zur Reihenfolgeplanung keine gültige Reihenfolge ohne Überlastung gefunden werden kann, wird die Personalkapazität an den Stationen erhöht, an denen der Drift am größten wird. Dieser Prozess wiederholt sich so lange, bis eine gültige Reihenfolge gefunden werden kann. Eine gleichzeitige Lösung der beiden Probleme ist auf

Grund der unterschiedlichen Planungszeiträume nicht möglich.²³⁵ Daher werden simultane Lösungsansätze nicht weiter betrachtet.

3.3.2 Fazit

Vergleicht man die Annahmen des Referenzproblems mit denen der existierenden Mixed-Model-Sequencing Ansätze, so existieren einige Parallelen. Die Mehrheit der Ansätze beherrscht eine flexible Anzahl an Stationen und betrachtet ein fixiertes Auflegeintervall sowie einen vernachlässigbaren, bzw. indirekt modellierten Werkerrücklauf. Etwa die Hälfte der Ansätze beachtet bereits offene Stationsgrenzen und wenige beherrschen verschiedene Stationstypen. Kein Ansatz unterscheidet jedoch bisher zwischen Arbeitsplätzen und Stationen, wodurch auch eine Parallelarbeit von mehreren Arbeitsplätzen innerhalb einer Station vernachlässigt wird.

Bei den Zielsetzungen fällt auf, dass die meisten Autoren die Überlast minimieren. Eine Minimierung der Summe dieser Überlast bedingt jedoch keine Minimierung der Anzahl notwendiger Unterstützereinsätze. Es lässt sich leicht ein Beispiel konstruieren, in dem zwar wenig Überlast auftritt, jedoch viele kurze Unterstützereinsätze notwendig sind. Andersherum kann eine große Überlast mit wenigen langen Unterstützereinsätzen einhergehen. Lediglich Mollemeier²³⁶ und Yoo²³⁷ und Köther²³⁸ betrachten einen reaktiven Unterstützereinsatz.²³⁹ Decker²⁴⁰ versteht unter Unterstützern eine gegenseitige lokale Aushilfe der Werker und modelliert keine Unterstützer in der in dieser Arbeit vorliegenden Definition. Ausschließlich Mollemeier misst die Anzahl Unterstützereinsätze und betrachtet diese Größe innerhalb der Zielfunktion. Jedoch vernachlässigt auch er, dass neben der Anzahl an Unterstützereinsätzen insbesondere die Anzahl gleichzeitiger Unterstützereinsätze kostenrelevant ist.

Keiner der Lösungsansätze kann daher das in dieser Arbeit beschriebene Problem lösen.

²³⁵Vgl. Abschnitt 2.2

²³⁶Vgl. [Mol96]

²³⁷Vgl. [YSH05]

²³⁸Vgl. [Koe85]

²³⁹Vgl. Abschnitt 2.1.2.2.2

²⁴⁰Vgl. [Dec93]

4 Zu leistende Arbeiten

Leben ist arbeiten, und alles,
was man tut, bringt Erfahrung.

(Henry Ford)

Für jedes der drei beschriebenen Teilprobleme in Kapitel 2 soll eine Lösungsmethode entwickelt werden.

Im ersten Schritt soll ein Verfahren für eine kostenoptimale Rekonfiguration entwickelt werden, welches die Zuordnung von Arbeitsvorgängen, Arbeitskräften und Betriebsmitteln zu Arbeitsplätzen auf eine bekannte Menge von Aufträgen anpasst und dabei die Rekonfigurationskosten betrachtet. Hierfür soll das Rekonfigurationsproblem zuerst als mathematisches Modell formuliert werden. Da eine Bestimmung der Rekonfigurationskosten a priori nicht möglich ist, muss das zu entwickelnde Lösungsverfahren in ein Entscheidungsunterstützungssystem eingebettet werden, welches dem Planer die Möglichkeit gibt, die Kosten direkt im Verfahren zu ergänzen. Die Bewertung der Planungslösungen muss ohne Kenntnis der Produktionsreihenfolge, die erst in einem späteren Schritt fixiert wird, durchgeführt werden können. Für diese Antizipation der Auswirkungen einer Rekonfigurationslösung auf den Produktionsprozess wurden in Kapitel 3.1.2 nachgelagerte Zielsetzungen aus der Erstinstallation von Fließlinien diskutiert. Zur Übertragung dieser Zielsetzungen soll gezeigt werden, dass sie sich sowohl für die Minimierung der Anzahl an Unterstützern als auch für die Minimierung von gleichzeitigen Unterstützereinsätzen eignen.

Im zweiten Schritt soll ein Verfahren entwickelt werden, das eine Vorranggraphenentwicklung für variantenreiche Produkte ermöglicht. Bei der Literaturrecherche hat sich gezeigt, dass auf die Integration von geometrischen Daten verzichtet werden sollte und die Verwendung einer Produktdokumentation in Form einer Stückliste die verlässlichste Quelle für ein automatisiertes Vorgehen darstellt. Das Konzept sollte daher auf einer Stücklistenstruktur aufbauen, welche in der Praxis für variantenreiche Produkte etabliert ist. Existierende Verfahren leiden insbesondere unter einer hohen Fehleranfälligkeit und der mangelnden Fähigkeit, Varianten abzubilden. Daher muss ein alternatives Vorgehen entwickelt werden, welches gewährleistet,

dass jederzeit eine gültige Zuordnungslösung für die Arbeitsvorgänge aus dem Vorranggraphen abgeleitet werden kann. Kein voll automatisiertes Verfahren kann die manuelle Machbarkeit der Bearbeitungsreihenfolgen gewährleisten. Daher muss eine teilautomatisierte Methode entwickelt werden, die die Komplexität weit genug reduziert, um im Nachgang eine manuelle Feinstrukturierung des Graphen zu ermöglichen.

Im dritten Schritt soll ein Konzept für eine belastungsorientierte Reihenfolgeplanung entwickelt werden, das explizit den Unterstützereinsatz minimiert. Zur Formalisierung des Problems muss eine mathematische Problemmodellierung erarbeitet werden, die insbesondere auch die parallelen Arbeiten innerhalb einer Station und die unterschiedlichen Abhängigkeiten der Werker berücksichtigt. Zur genauen Bewertung der erzeugten Lösungen ist zuerst ein Verfahren zur Fließliniensimulation zu entwickeln, das die Anzahl der Unterstützereinsätze und auch die Anzahl notwendiger Unterstützer korrekt ermittelt. Die Methode sollte dabei die fixierte Zuordnungslösung, die mit Hilfe der Rekonfigurationsmethode erzeugt wurde, verwenden. Die Komplexität des Reihenfolgeplanungsproblems wurde in den existierenden Arbeiten als größte Herausforderung identifiziert und soll vom zu entwickelnden Optimierungsansatz beherrscht werden.

Die Verfahren sollen anschließend im Rahmen praktischer Einsätze einzeln verifiziert und der erzielte Nutzen anhand dieser Erprobungen aufgezeigt werden.

5 Kostenoptimale Kapazitätsabstimmung in einer getakteten Variantenfließlinie

Probleme kann man niemals mit derselben Denkweise lösen, durch die sie entstanden sind.

(Albert Einstein)

5.1 Kostenoptimale Rekonfiguration

5.1.1 Formalisierung des Rekonfigurationsproblems

Im Folgenden wird das Problem der Rekonfiguration auf Basis der Notation in Tabelle 5.1 als Optimierungsmodell mathematisch formuliert.

Symbol	Definition
Mengen	
$\mathcal{A}$	Menge von Aufträgen a
$\mathcal{AP}$	Menge von Arbeitsplätzen ap
$\mathcal{AVO}$	Menge von Arbeitsvorgängen avo
$\mathcal{P}$	Menge von Werkstücken p
$\mathcal{ST}$	Menge von Stationen st
$\mathcal{V}$	Menge von Vorrangbedingungen. Wenn $(i, j) \in \mathcal{V}$, dann muss Arbeitsvorgang $i \in \mathcal{AVO}$ abgeschlossen sein, bevor mit Arbeitsvorgang $j \in \mathcal{AVO}$ begonnen werden kann. Es gilt: $\mathcal{V} \subset \mathcal{AVO} \times \mathcal{AVO}$
$\mathcal{WP}_{avo}$	Menge möglicher Werkstückpositionen $wp \in \mathcal{WP}$ für die Ausführung von Arbeitsvorgang $avo \in \mathcal{AVO}$. Es gilt: $\mathcal{WP}_{avo} \subseteq \mathcal{WP}$
Parameter	
$bp_{a,avo}$	Binär: 1, wenn der Auftrag $p \in \mathcal{P}$ den Arbeitsvorgang $avo \in \mathcal{AVO}$ benötigt; 0, sonst (Bauplan)

Symbol	Definition
br_{avo}	Baurate des Arbeitsvorganges $avo \in \mathcal{AVO}$. Gibt die Häufigkeit des Auftretens eines Arbeitsvorganges im Verhältnis zur Gesamtanzahl Aufträge an.
dbl_{ap}	Größe des vorgelagerten „linken“ Driftbereiches des Arbeitsplatzes $ap \in \mathcal{AP}$ in Zeiteinheiten
dbr_{ap}	Größe des nachgelagerten „rechten“ Driftbereiches des Arbeitsplatzes $ap \in \mathcal{AP}$ in Zeiteinheiten
$h_{a,p}$	Binär: 1, wenn Auftrag $a \in \mathcal{A}$ zu Werkstück $p \in \mathcal{P}$ zugeordnet ist; 0, sonst
$\mathcal{K}^W$	Kostensatz für einen Werker
$\mathcal{K}^U$	Kostensatz für einen Unterstützer
$\mathcal{K}^N$	Kostensatz für Nacharbeit
$\mathcal{K}_{avo,ap}^R$	Kostensatz für die Verschiebung eines Arbeitsvorganges $avo \in \mathcal{AVO}$ auf einen Arbeitsplatz $ap \in \mathcal{AP}$
$\mathcal{K}^{LinPos}$	Kostensatz für eine Zeiteinheit der Kennzahl Δ^{LinPos}
$\mathcal{K}^{QuadPos}$	Kostensatz für eine Zeiteinheit der Kennzahl $\Delta^{QuadPos}$
pl_{ap}	Zur Verfügung stehender Lagerplatz an Arbeitsplatz $ap \in \mathcal{AP}$
pb_{avo}	Notwendiger Lagerplatz für die Ausführung von Arbeitsvorgang $avo \in \mathcal{AVO}$
st_{ap}	Station $st \in \mathcal{ST}$, welcher der Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist
$\mathcal{SP}$	Menge von Ansprechzonen sp
sp_{ap}	Ansprechzone, der Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist
tz	Taktzeit
τ_{avo}	Bearbeitungszeit des Arbeitsvorganges $avo \in \mathcal{AVO}$ in Zeiteinheiten
wp_{ap}	Werkstückposition $wp \in \mathcal{WP}$ an $st \in \mathcal{ST}$
$z_{avo,ap}^{Ist}$	Binär: 1, wenn Arbeitsvorgang $avo \in \mathcal{AVO}$ dem Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist; 0, sonst (existierende Zuordnungslösung)

Variablen

Δ^{LinPos}	Summierte lineare positive Abweichung der Bearbeitungszeiten von der Kapazität ζ_{ap} über alle Arbeitsplätze $ap \in \mathcal{AP}$
$\Delta^{QuadPos}$	Summierte quadrierte positive Abweichung der Bearbeitungszeiten von der Kapazität ζ_{ap} über alle Arbeitsplätze $ap \in \mathcal{AP}$
$\mathcal{K}^{Ges}$	Gesamtkosten
l	Index einer Lösung für die Rekonfiguration
$\tau_{p,ap}$	Bearbeitungszeit des Werkstückes $p \in \mathcal{P}$ an Arbeitsplatz $ap \in \mathcal{AP}$ in Zeiteinheiten
u^E	Gesamtanzahl Unterstützereinsätze
u^{MAX}	Maximale Anzahl gleichzeitiger Unterstützereinsätze $\hat{=}$ Notwendige Anzahl an Unterstützereinsatzkräften

Symbol	Definition
$v_{avo,ap}^R$	Binär: 1, wenn Arbeitsvorgang $avo \in \mathcal{AVO}$ in der neuen Zuordnung zu einem neuen Arbeitsplatz $ap \in \mathcal{AP}$ verschoben wurde (Verschiebungsvariable)
w_{st}	Anzahl an Werkern, die Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen sind
$x_{avo,ap}^{Soll}$	Binär: 1, wenn Arbeitsvorgang $avo \in \mathcal{AVO}$ Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist; 0, sonst (neue Zuordnungslösung)

Tabelle 5.1: Notation zur Formalisierung des Rekonfigurationsproblems

5.1.1.1 Entwicklung eines antizipativen Zielkriteriums

Ziel der Rekonfiguration ist die Minimierung der Betriebs- und Rekonfigurationskosten für den zukünftigen Planungszeitraum.²⁴¹ Da die Reihenfolge, in der die Werkstücke das Fließsystem durchlaufen, zum Zeitpunkt der Rekonfiguration noch nicht feststeht, muss für reihenfolgeabhängige Teile der Zielfunktion, wie in Gleichung (5.1) dargestellt, ein alternatives Zielkriterium gewählt werden.²⁴²

$$\begin{aligned}
 \text{Min } \mathcal{K}^{Ges} = & \overbrace{\mathcal{K}^U \cdot u^{MAX} + \mathcal{K}^N \cdot u^E}^{\text{Durch antizipatives Zielkriterium zu ersetzen}} \\
 & + \sum_{ap \in \mathcal{AP}} \mathcal{K}^W \cdot w_{ap} + \sum_{ap \in \mathcal{AP}} \sum_{avo \in \mathcal{AVO}} \mathcal{K}_{avo,ap}^R \cdot v_{avo,ap}^R
 \end{aligned} \tag{5.1}$$

In der Literatur wurden, wie in Abschnitt 3.1.3 diskutiert, einige Anstrengungen unternommen, bereits während der Erstinstallation einer Fließlinie die Auswirkungen auf die Reihenfolgeplanung zu antizipieren. Die Übertragung eines dieser Ziele auf das Rekonfigurationsproblem liegt nahe und soll nachfolgend diskutiert werden.

Minimierung externer Auslastungsunterschiede Die Minimierung externer Auslastungsunterschiede sorgt für eine gleichmäßige Durchschnittsbelastung auf den Arbeitsplätzen (3.2). Diese Maßzahl ist aus zwei Gründen nicht für die Minimierung von Unterstützereinsätzen im Rahmen einer Rekonfiguration geeignet. Zum einen werden die Schwankungen der Bearbeitungszeiten nicht erfasst, zum anderen ist eine gleichmäßige Auslastung der Arbeitsplätze

²⁴¹Vgl. dazu 2.2.1.1

²⁴²Vgl. dazu die Diskussion in Abschnitt 2.2.1.1

unter Umständen nicht gewollt. Es kann sinnvoll sein, Arbeitsplätze, die keinerlei Varianz in den Bearbeitungszeiten aufweisen, mit einer hohen Durchschnittsauslastung zu betreiben, während Arbeitsplätze, die hohe Schwankungen aufweisen, einen größeren Taktausgleich²⁴³ benötigen, um hohe Bearbeitungszeiten durch niedrige zu kompensieren.

Minimierung interner Auslastungsunterschiede Die meisten Ansätze zur Minimierung interner Auslastungsunterschiede konzentrieren sich auf die *Minimierung der Abweichung der Bearbeitungszeit einer Variante im Verhältnis zur durchschnittlichen Bearbeitungszeit der Variante* (3.3),(3.4),(3.5). Eine gezielte Minimierung der Unterstützereinsätze wird mit der Zielsetzung nicht vorgenommen, da Überlastungen nicht erfasst werden und der Arbeitsplatzbezug fehlt. Eine Verbesserung dieses Zielfunktionswertes würde auch dann erreicht, wenn der Arbeitsumfang einer Produktvariante, die auf jedem Arbeitsplatz eine niedrigere Bearbeitungszeit als die Taktzeit hat, gleichmäßiger auf die Arbeitsplätze verteilt wird.

Die *Minimierung der Abweichungen bezogen auf die gesamte Durchschnittsauslastung* (3.6) beinhaltet ebenfalls keinen Arbeitsplatzbezug und erfasst keine Überlastungen. Demzufolge ist auch von dieser Zielsetzung im Rahmen der Rekonfiguration abzusehen.

Die *Minimierung der Abweichungen von der Taktzeit* soll laut Domschke et. al.²⁴⁴ bei gleichzeitiger Betrachtung der Unter- und Überlastungen sowohl Unterstützereinsätze als auch Leerzeiten minimieren. Wie in Abschnitt 2.1.3.1.3 beschrieben hat die Minimierung von Leerzeit keine direkte Auswirkungen auf die Kosten.²⁴⁵ Die Beschränkung auf positive Abweichungen von der Taktzeit, eingeführt von Domschke et al. in Gleichung (3.4), erscheint sinnvoll, da hierdurch eine Fokussierung auf die Überlastungen vorgenommen wird. Dieses Zielkriterium wird im Folgenden als “Minimierung der linearen Übertaktzeit” bezeichnet. Domschke et al. zeigen an einem Beispiel, dass eine Verfolgung dieses Zieles eine Minimierung der Überlastungssituationen und damit auch der Anzahl Unterstützereinsätze impliziert.²⁴⁶ Da die Anzahl gleichzeitiger Unterstützereinsätze definiert, wie viele Unterstützereinsatzkräfte vorgehalten werden müssen²⁴⁷, muss bei Übertragung dieser Zielsetzung gezeigt werden, dass die Verringerung der Unterstützereinsatzanzahl auch die Wahrscheinlichkeit für gleichzeitige Einsätze verringert. Im Folgenden wird dieser Zusammenhang unter Verwendung der Notation in Tabelle 5.2 formal hergeleitet.

Aufbauend auf der Definition der Gleichzeitigkeit im Fließsystem in Abschnitt 2.1.3.1.2 kann das Problem als Urnenmodell beschrieben werden. Jede mögliche Situation stellt eine Kugel

²⁴³Vgl. Abschnitt 2.1.3.1 und Definition in Gleichung (2.16)

²⁴⁴Vgl. [DKS96, S.1483]

²⁴⁵Es sei denn, es wird davon ausgegangen, dass die Minimierung der Leerzeiten eine Minimierung der Unterstützereinsätze nach sich zieht. In diesem Fall ist es dennoch sinnvoll, sich direkt auf die Minimierung der Unterstützereinsätze zu konzentrieren, da das Ergebnis dasselbe ist.

²⁴⁶Vgl. [DKS96, S.1483]

²⁴⁷Vgl. Abschnitt 2.1.3.1.2

Symbol	Definition
gz	Anzahl gleichzeitiger Unterstützereinsätze
idx	Reihenfolgeposition $idx \in \mathcal{RF}$
sit	Situation im Fließsystem. Definiert durch ein Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ innerhalb einer bestimmten Station $st \in \mathcal{ST}$
t	Takt $t \in \mathcal{T}$
ts	Anzahl von gleichzeitigen Situationen sit innerhalb jedes Taktes $t \in \mathcal{T}$
u^E	Gesamtanzahl Unterstützereinsätze

Tabelle 5.2: Notation Herleitung der Wahrscheinlichkeit für gleichzeitige Unterstützereinsätze

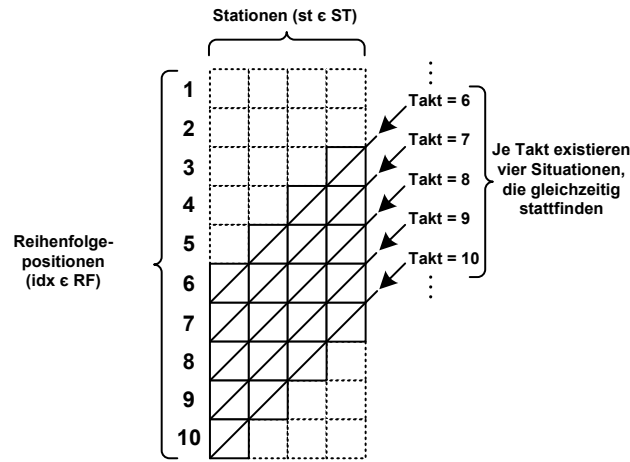


Abbildung 5.1: Darstellung von gleichzeitigen Situationen im Fließsystem

in der Urne dar. Die Kugeln haben unterschiedliche Farben und jede Farbe repräsentiert jeweils einen bestimmten Takt t in Abbildung 5.1. Es gibt somit genauso viele Farben wie Takte existieren. Je Farbe gibt es ts Kugeln; also genau so viele, wie es gleichzeitige Situationen in einem Takt gibt. In einem Experiment werden nun u^E -Anzahl Kugeln aus der Urne gezogen. Dabei wird von einer Gleichverteilung der Wahrscheinlichkeiten für einen Unterstützereinsatz in einer bestimmten Situation ausgegangen.²⁴⁸ Die Wahrscheinlichkeit dafür, dass gz gleichfarbige Kugeln gezogen werden, und somit, dass die Unterstützereinsätze innerhalb eines Taktes liegen, wird über die Einzelwahrscheinlichkeiten für jeden Zug hergeleitet (5.2). Dabei ist die Wahrscheinlichkeit, beim ersten Zug eine Kugel einer bestimmten Farbe zu ziehen $\frac{ts}{sit}$, beim zweiten, die gleiche Farbe zu ziehen $\frac{ts-1}{sit-1}$, usw. Da bestimmt werden soll, wie hoch die Wahrscheinlichkeit ist, eine bestimmte Anzahl an gleichzeitigen Unterstützereinsätzen gz und damit gleichfarbigen Kugeln zu ziehen, muss noch gewährleistet sein, dass die anderen $u^E - gz$ gezogenen Kugeln andere Farben haben. Auch hierfür werden die Einzelwahrscheinlichkeiten

²⁴⁸ Dies ist eine Vereinfachung, da die Länge eines Unterstützereinsatzes länger als eine Taktzeit sein kann und die Wahrscheinlichkeit für Einsätze je Arbeitsplatz in der Realität unterschiedlich hoch ist. Dennoch kann dies an dieser Stelle vernachlässigt werden, da nur die Abhängigkeit der Wahrscheinlichkeit gleichzeitiger Unterstützereinsätze von der Anzahl an Unterstützereinsätzen gezeigt werden soll.

aufgezählt. Die Wahrscheinlichkeit dafür, dass eine Kugel beim $gz + 1$ -ten Zug einer anderen Farbe entspricht, ist $\frac{sit-ts}{sit-gz}$, beim nächsten Zug $\frac{sit-ts-1}{sit-(gz-1)}$, usw. Der Binomialkoeffizient $\binom{u^E}{gz}$ gibt an, wie viele Möglichkeiten es gibt, u^E Anzahl Objekte aus der Urne mit gz Anzahl Kugeln zu ziehen. Daher wird mit diesem Faktor noch multipliziert.

$$\begin{aligned}
 P(X = gz) &= \binom{u^E}{gz} \cdot \frac{ts}{sit} \cdot \frac{ts-1}{sit-1} \cdot \dots \cdot \frac{ts-(gz-1)}{sit-(gz-1)} \cdot \\
 &\quad \frac{sit-ts}{sit-gz} \cdot \frac{sit-ts-1}{sit-(gz-1)} \cdot \dots \cdot \frac{sit-ts-(u-(gz-1))}{sit-gz-(u^E-(gz-1))} \\
 &= \frac{u!}{gz! \cdot (u^E - gz)!} \cdot \frac{ts!}{(ts - gz)!} \cdot \frac{(sit-ts)!}{(sit-ts-u^E-gz)!} \\
 &\quad \frac{sit!}{(sit-u)!}
 \end{aligned} \tag{5.2}$$

Dieser Term entspricht der Beschreibung der hypergeometrischen Verteilung (5.3). Somit wird die Wahrscheinlichkeit für gleichzeitige Unterstützereinsätze in Abhängigkeit von der Anzahl Einsätze durch diese Verteilung beschrieben.

$$\begin{aligned}
 P(gz, u^E, t) &= \frac{ts!}{gz! \cdot (ts - gz)!} \cdot \frac{(sit - ts)!}{(u^E - gz)! \cdot (sit - ts - u^E - gz)!} \cdot \\
 &\quad \frac{u^E! \cdot (sit - u^E)!}{sit!} \\
 &= \frac{\binom{ts}{gz} \cdot \binom{sit-ts}{u^E-gz}}{\binom{sit}{u^E}}
 \end{aligned} \tag{5.3}$$

Der Erwartungswert dieser Verteilung kann auf das Problem ebenfalls übertragen werden und ist definiert durch (5.4).²⁴⁹

$$E(X = gz) = u^E \cdot \frac{ts}{sit} \tag{5.4}$$

Bei einer steigenden Anzahl Unterstützereinsätze u^E steigt der Erwartungswert für gleichzeitige Unterstützereinsätze. Die genaue Anzahl vorzuhaltender Unterstützer, die sich durch die Anzahl gleichzeitiger Unterstützereinsätze u^{MAX} ergibt, lässt sich dadurch nicht genau

²⁴⁹Vgl. Diskussion der hypergeometrischen Verteilung in [Eck03, S.228ff.]

vorhersagen, jedoch wurde gezeigt, dass die Wahrscheinlichkeit für gleichzeitige Unterstützereinsätze mit der Minimierung der Anzahl Unterstützereinsätze sinkt.

Das Zielkriterium „Minimierung der linearen Übertaktzeit“ wird somit auf das Rekonfigurationsproblem übertragen. Wie in Abschnitt 2.1 beschrieben, existiert im betrachteten Fließsystem eine unterschiedliche Kapazität je Arbeitsplatz. Daher müssen zur Übertragung des Zielkriteriums die positiven Abweichungen der Bearbeitungszeiten auf die durch ζ_{ap} definierte Produktionskapazität an dem entsprechenden Arbeitsplatz bezogen werden (5.5).

$$\text{Min } \Delta^{LinPos} = \sum_{ap \in \mathcal{AP}} \sum_{p \in \mathcal{P}} \max[0; (\tau_{p,ap} - \zeta_{ap})] \quad (5.5)$$

Alternativ soll zur erhöhten Bestrafung großer Abweichungen deren Quadrat betrachtet werden (5.6).

$$\text{Min } \Delta^{QuadPos} = \sum_{ap \in \mathcal{AP}} \sum_{p \in \mathcal{P}} \begin{cases} (\tau_{p,ap} - \zeta_{ap})^2 & \text{wenn } \tau_{p,ap} > \zeta_{ap} \\ 0 & \text{sonst} \end{cases} \quad (5.6)$$

Da in der Zielfunktion Kosten in ein Verhältnis gesetzt werden, muss die Zielgröße noch mit einem Kostenfaktor $\mathcal{K}^{LinPos}$ bzw. $\mathcal{K}^{QuadPos}$ multipliziert werden. Der Faktor muss so gewählt werden, dass das Produkt aus einer Zeiteinheit linearer bzw. quadrierter Übertaktzeit und dem Kostenfaktor die zu erwartenden Kosten durch Unterstützereinsätze und vorzuhaltende Unterstützer wiedergibt. Die neue zu minimierende Zielfunktion wird in (5.7) beispielhaft mit den linearen Abweichungen formuliert.

$$\text{Min } \mathcal{K}^{Ges} = \Delta^{LinPos} \cdot \mathcal{K}^{LinPos} + \sum_{ap \in \mathcal{AP}} \mathcal{K}^W \cdot w_{ap} + \sum_{ap \in \mathcal{AP}} \sum_{avo \in \mathcal{AVO}} \mathcal{K}_{avo,ap}^R \cdot v_{avo,ap}^R \quad (5.7)$$

5.1.1.2 Formulierung des Modells

Unter Verwendung der Notation in Tabelle 5.1 wird das Problem als Optimierungsmodell formuliert. Dabei wird die Zielfunktion aus Abschnitt 5.1.1.1 übernommen.

In der Zielfunktion (5.8) wurde der reihenfolgeabhängige Teil der Gleichung durch die zuvor diskutierte Zielfunktion ersetzt. Durch Gleichung (5.9) wird sichergestellt, dass jeder Arbeitsvorgang nur einem Arbeitsplatz zugewiesen wird und damit die Anlieferung von bestimmten Einzelteilen immer nur an einem Arbeitsplatz in der Montagelinie notwendig ist. Gleichung (5.10) definiert den Unterschied zwischen der alten Zuordnung von Arbeitsvorgängen und der

neuen. Dabei wird die neue Zuordnung mit der alten verglichen. Bei allen Arbeitsvorgangs-/Arbeitsplatzkombinationen, bei denen die neue Zuordnung von der alten abweicht, hat eine Verschiebung stattgefunden. Die resultierenden Verschiebungen machen eine Kalkulation der Verschiebekosten in der Zielfunktion möglich. Die Beachtung der gegebenen Menge an Vorrangbeziehungen wird in Gleichung (5.11) sichergestellt. Es wird dabei angenommen, dass die interne Reihenfolge der Durchführung von Arbeitsvorgängen auf den Arbeitsplätzen innerhalb einer Station durch einen weiteren Planungsprozess bestimmt wird. Daher reicht es hier, die Vorrangbeziehungen zwischen den Stationen sicherzustellen. Wichtig ist dabei, dass die Stationsnummern von vorne bis hinten im Fließsystem durchnummeriert sind. Die Anzahl Werker, die einem Arbeitsplatz zugewiesen sind, wird von der durchschnittlichen Belastung eines Arbeitsplatzes in Gleichung (5.12) abgeleitet. Die Kapazität des Arbeitsplatzes muss immer mindestens die durchschnittliche Bearbeitungszeit auf dem Arbeitsplatz abdecken. Dabei wird zum einen sichergestellt, dass nur ein Minimum an Werkern zugewiesen wird, und zum anderen keine dauerhafte Überlastung entsteht. Die Bearbeitungszeit eines Werkstückes auf einem Arbeitsplatz wird in Gleichung (5.13) durch die Summe der Ausführungszeiten aller notwendigen Arbeitsvorgänge für den zugehörigen Auftrag berechnet. Jeder Arbeitsvorgang hat einen bestimmten Platzbedarf für die Anlieferung von Einzelteilen sowie die Betriebsmittel, die an dem Arbeitsplatz installiert werden müssen. Hierfür stellt Gleichung (5.14) sicher, dass das verfügbare Platzangebot am Arbeitsplatz durch die Zuordnung von zu vielen Arbeitsvorgängen nicht überschritten wird. Das Werkstück kann je Station in eine andere Werkstückposition gebracht werden.²⁵⁰ Einige Arbeitsvorgänge können nur durchgeführt werden, wenn sich das Werkstück in einer bestimmten Werkstückposition befindet. In Gleichung (5.15) wird daher sichergestellt, dass diese Anforderung erfüllt wird. Gleichungen (5.16), (5.17) und (5.18) definieren die Gültigkeitsbereiche der Variablen.

²⁵⁰Vgl. Abschnitt 2.1.1

$$\begin{aligned}
Min \quad \mathcal{K}^{Ges} = & \mathcal{K}^{LinPos} \cdot \sum_{ap \in \mathcal{AP}} \sum_{p \in \mathcal{P}} \max\{0, \tau_{p,ap} - tz \cdot w_{ap}\} \\
& + \sum_{ap \in \mathcal{AP}} \mathcal{K}^W \cdot w_{ap} + \sum_{ap \in \mathcal{AP}} \sum_{avo \in \mathcal{AVO}} \mathcal{K}_{avo,ap}^R \cdot v_{avo,ap}^R
\end{aligned} \tag{5.8}$$

u.B.d.R.

$$\sum_{ap \in \mathcal{AP}} x_{avo,ap}^{Soll} = 1 \quad \forall avo \in \mathcal{AVO} \tag{5.9}$$

$$v_{avo,ap}^R \geq x_{avo,ap}^{Soll} - z_{avo,ap}^{Ist} \quad \forall ap \in \mathcal{AP}, avo \in \mathcal{AVO} \tag{5.10}$$

$$\sum_{ap \in \mathcal{AP}} st_{ap} \cdot x_{i,ap}^{Soll} \leq \sum_{ap \in \mathcal{AP}} st_{ap} \cdot x_{j,ap}^{Soll} \quad \forall (i, j) \in \mathcal{G} \tag{5.11}$$

$$\sum_{avo \in \mathcal{AVO}} br_{avo} \cdot \tau_{avo} \cdot x_{avo,ap}^{Soll} \leq tz \cdot w_{ap} \quad \forall ap \in \mathcal{AP} \tag{5.12}$$

$$\tau_{p,ap} = \sum_{avo \in \mathcal{AVO}} \tau_{avo} \cdot bp_{a,avo} \cdot x_{avo,ap}^{Soll} \quad \forall p \in \mathcal{P}, ap \in \mathcal{AP}, a \in \mathcal{A} : h_{a,p} = 1 \tag{5.13}$$

$$\sum_{avo \in \mathcal{AVO}} pb_{avo} \cdot x_{avo,ap}^{Soll} \leq pl_{ap} \quad \forall ap \in \mathcal{AP} \tag{5.14}$$

$$x_{avo,ap}^{Soll} = 0 \quad \forall ap \in \mathcal{AP}, avo \in \mathcal{AVO} : wp_{ap} \notin \mathcal{WP}_{avo} \tag{5.15}$$

$$x_{avo,ap}^{Soll} \in \{0, 1\} \quad \forall ap \in \mathcal{AP}, avo \in \mathcal{AVO} \tag{5.16}$$

$$v_{avo,ap}^R \in \{0, 1\} \quad \forall ap \in \mathcal{AP}, avo \in \mathcal{AVO} \tag{5.17}$$

$$w_{ap} \geq 0 \quad \wedge \quad w_{ap} \in \{0, 1\} \quad \forall ap \in \mathcal{AP} \tag{5.18}$$

5.1.2 Vorgehensmodell zur Rekonfiguration

Für die Lösung des Modells können exakte oder heuristische Lösungsmethoden entwickelt werden. Eine Anwendung der Lösungsmethoden in einem praktischen Umfeld ist jedoch aus folgenden Gründen nicht machbar:²⁵¹

- Die Rekonfigurationskosten, die $|\mathcal{AVO}| \cdot (|\mathcal{AP}| - 1)$ verschiedene Kostensätze beinhalten, müssen zur Anwendung von exakten Lösungsmethoden vorweg definiert werden. Die Kosten hängen von der Kombination von Arbeitsvorgängen auf einem Arbeitsplatz ab, so dass nach jeder Verschiebung eines Arbeitsvorganges alle Kosten von und zu den

²⁵¹Vgl. auch die Anforderungen an die Problemlösung in Abschnitt 2.3.1

beiden betroffenen Arbeitsplätzen neu definiert werden müssen. Die Aufnahme dieser Kosten für alle Kombinationsmöglichkeiten von Arbeitsvorgängen auf allen Arbeitsplätzen im Vorfeld ist aufgrund der enormen Komplexität nicht machbar.

- Eine exakte Lösungsfindung ist nur möglich, wenn ein vollständiger und aktuell gepflegter Vorranggraph vorliegt. Die Definition eines korrekten Vorranggraphen ist jedoch mit extrem großem Aufwand verbunden.²⁵²
- Das inhaltliche Verständnis für die Lösungsfindung ist für die Akzeptanz eines Planungsverfahrens sowie die Erkennung von Fehlern in der Planung sehr wichtig.

Daher wird im Folgenden ein Entscheidungsunterstützungssystem (ESS)²⁵³ entwickelt, welches auf dem vorgestellten Modell basiert und den Planer mit in den Entscheidungsprozess einbezieht.²⁵⁴ Der Aufwand für die Definition von Rekonfigurationskosten und Vorrangbeziehungen beschränkt sich dabei auf Verschiebungen von Arbeitsvorgängen, die eine bessere Lösung erwarten lassen.

5.1.3 Entscheidungsunterstützungssystem zur kostenoptimalen Rekonfiguration

Entscheidungsunterstützungssysteme werden seit einigen Jahrzehnten für verschiedenste Anwendungsfälle eingesetzt. Trotz der unterschiedlichen Problemspezifika hat sich eine grundsätzliche Strukturierung herauskristallisiert. Nach Turban²⁵⁵ sowie Hotman²⁵⁶ wird der Entscheidungsprozess unterteilt in die Schritte Analysephase²⁵⁷, Design-Phase²⁵⁸, Auswahl und Entscheidungsphase²⁵⁹ und die Implementationsphase²⁶⁰.

Andere Konzepte verfolgen ähnliche Strukturierungen.²⁶¹ Hieran angelehnt sollen die Bausteine des Konzeptes, wie in Abbildung 5.2 aufgezeigt, für das zu entwickelnde ESS zur Rekonfiguration beschrieben werden.

²⁵²Vgl. [Amm85]

²⁵³Entscheidungsunterstützungssysteme (ESS) werden von [SC82] definiert als: „ESS sind interaktive computerbasierte Systeme, die Entscheidern helfen schlecht strukturierte Probleme mit Daten- sowie Analysemodellen zu lösen“. In [Fin89, S.35ff] werden alternative Definitionen verglichen. Die Grunddefinition ist jedoch dieselbe.

²⁵⁴Vgl. Charakteristik eines Entscheidungsunterstützungssystems [SM06]

²⁵⁵Vgl. [TA98]

²⁵⁶Vgl. [Hot06, S.31]

²⁵⁷In der **Analysephase** werden Daten aufbereitet und Kennzahlen erzeugt, die das Problem beschreiben. Hierbei steht die Identifikation des Problems und des Handlungsbedarfs im Vordergrund.

²⁵⁸Die **Design-Phase** beinhaltet die Erzeugung des Problemverständnisses beim Benutzer. Außerdem werden in diesem Schritt Möglichkeiten zur Selektion von Entscheidungsalternativen vorgestellt.

²⁵⁹In der **Auswahl-Entscheidungsphase** werden die unterschiedlichen Alternativen bewertet und die Beste ausgewählt.

²⁶⁰Die **Implementation-Phase** beinhaltet alle Prozesse, die notwendig sind, um die Entscheidung umzusetzen.

²⁶¹Vgl. [Smi98, S.6]

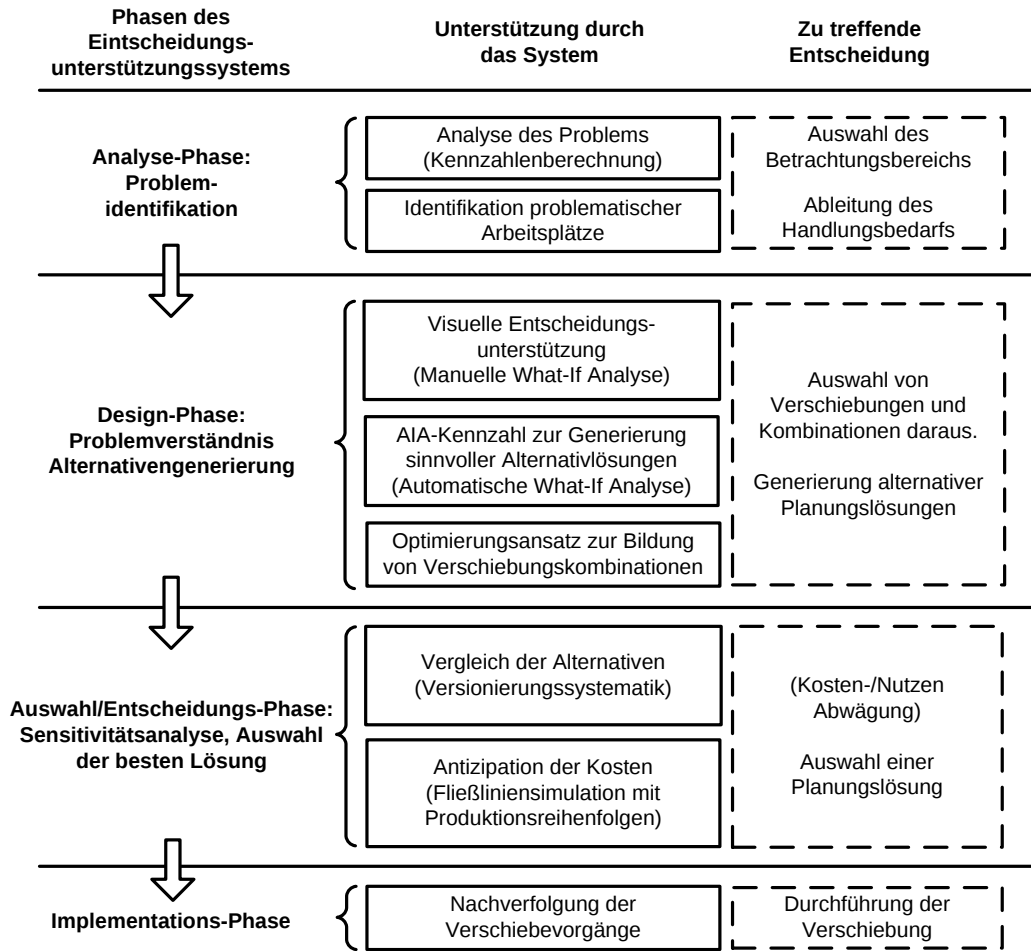


Abbildung 5.2: Eingliederung der Verfahren in Phasen eines Entscheidungsunterstützungssystems

5.1.3.1 Analyse-Phase

Zur Beschreibung der Analyse-Phase wird die Notation um die Bezeichner in Tabelle 5.3 erweitert.

5.1.3.1.1 Ableitung des Handlungsbedarfs Die Rekonfiguration wird angestoßen, wenn das Auftragsprogramm für den neuen Planungszeitraum festliegt. Die bisherige Zuordnung der Arbeitsvorgänge, gegeben durch $z_{avo,ap}^{Ist}$, bietet die Grundlage zur Ableitung von Verschiebungen. Daher wird als Startlösung für die Rekonfiguration im ersten Schritt die alte Arbeitszuordnung gewählt (5.19).

$$x_{avo,ap}^{Soll} \Leftarrow z_{avo,ap}^{Ist} \quad \forall \quad avo \in \mathcal{AVO}, ap \in \mathcal{AP} \quad (5.19)$$

Symbol	Definition
$\Delta\mu_{ap}$	Taktausgleich an Arbeitsplatz $ap \in \mathcal{AP}$
Δ_{ap}^{LinPos}	Summe der linearen positiven Abweichungen der Bearbeitungszeiten auf Arbeitsplatz $ap \in \mathcal{AP}$ von der Kapazität in Zeiteinheiten
$\Delta_{ap}^{QuadPos}$	Summe der quadrierten positiven Abweichungen der Bearbeitungszeiten auf Arbeitsplatz $ap \in \mathcal{AP}$ von der Kapazität in Zeiteinheiten
Δw_{ap}	Unterschied zwischen der ursprünglichen Anzahl Werker an einem Arbeitsplatz $ap \in \mathcal{AP}$ und der nach der Rekonfiguration zugewiesenen Anzahl. Zeigt den Handlungsbedarf für den Reparaturvorgang im Rekonfigurationsprozess an.
ϵ_{ap}	Arbeitsaufwand, der von Arbeitsplatz $ap \in \mathcal{AP}$ verschoben werden muss, um eine Komprimierung zur Reduzierung der Anzahl Werker auf dem Arbeitsplatz ap durchführen zu können. Zeigt den Handlungsbedarf zum Komprimieren im Rekonfigurationsprozess an.
μ_{ap}	Durchschnittliche Bearbeitungszeit für alle Werkstücke $p \in \mathcal{P}$ auf Arbeitsplatz $ap \in \mathcal{AP}$ im Verhältnis zur Kapazität ζ_{ap}
μ_{ap}^{ABS}	Durchschnittliche Bearbeitungszeit für alle Werkstücke $p \in \mathcal{P}$ auf Arbeitsplatz $ap \in \mathcal{AP}$ in Zeiteinheiten
ρ_{ap}^{LinPos}	Verhältnis von Δ_{ap}^{LinPos} zum Taktausgleich μ_{ap} an Arbeitsplatz $ap \in \mathcal{AP}$
$\rho_{ap}^{QuadPos}$	Verhältnis von $\Delta_{ap}^{QuadPos}$ zum Taktausgleich μ_{ap} an Arbeitsplatz $ap \in \mathcal{AP}$

Tabelle 5.3: Notationserweiterung für die Analyse-Phase des ESS

Durch das neue Auftragsprogramm entsteht eine veränderte Auslastungssituation, aus der ein Handlungsbedarf abgeleitet werden kann.

Aufbauend auf der Formalisierung aus 5.1.1 wird das vorliegende Problem untersucht. Hierfür muss im ersten Schritt ermittelt werden, welche Arbeitsvorgänge durch welchen Auftrag benötigt werden. Diese Zuordnung muss durch eine Auflösung der Arbeitsvorgänge aus einer Stückliste ermittelt werden. Für das eigentliche Problem der Rekonfiguration ist diese Zuordnung exogen vorgegeben durch $bp_{a,avo}$. Alle weiteren Kennzahlen im Modell hängen direkt oder indirekt von dieser Auftragsdefinition ab, insbesondere die Bearbeitungszeit durch ein Werkstück, definiert in (5.20).²⁶²

$$\tau_{p,ap} = \sum_{avo \in \mathcal{AVO}} \tau_{avo} \cdot x_{avo,ap}^{Soll} \cdot bp_{a,avo} \quad \forall \quad a \in \mathcal{A}, p \in \mathcal{P} : h_{a,p} = 1 \quad (5.20)$$

Die Veränderungen in den Häufigkeiten des Auftretens eines Arbeitsvorganges, definiert durch die Baurate br_{avo} , beeinflussen direkt die Durchschnittsbelastung (5.21)

$$\mu_{ap}^{ABS} = \sum_{avo \in \mathcal{AVO}} \tau_{avo} \cdot br_{avo} \cdot x_{avo,ap}^{Soll} \quad \forall \quad ap \in \mathcal{AP} \quad (5.21)$$

²⁶²Vgl. dazu Erläuterung in Abschnitt 2.1.2.1, in dem die Übersetzung von Aufträgen zu Werkstücken erläutert wird.

Der entstehende Handlungsbedarf nach der Analyse des neuen Auftragsprogramms wird in drei Schritten systematisiert:

1. **(Reparieren)**
Behebung von Auslastungsüberschreitungen.
2. **(Komprimieren)**
Minimierung des Werkereinsatzes durch Freiräumen von Arbeitsplätzen.
3. **(Antizipieren)**
Minimierung von Unterstützereinsätzen durch Verringerung von Übertaktsituationen.

Zu jedem dieser Schritte sollen Kennzahlen vorgestellt werden, welche den Planer dabei unterstützen, den Handlungsbedarf und damit die zu bearbeitenden Arbeitsplätze zu identifizieren.²⁶³

Reparieren Durch eine leichte Erhöhung von Bauraten der Arbeitsvorgänge, die einem Arbeitsplatz zugewiesen sind, kann die Durchschnittsbelastung eines Arbeitsplatzes größer als die verfügbare Kapazität werden. Da die Anzahl Werker eines Arbeitsplatzes w_{ap} direkt durch die durchschnittliche Belastung eines Arbeitsplatzes ermittelt wird (2.5)²⁶⁴, wird dem entsprechenden Arbeitsplatz ein weiterer Worker zugewiesen. Im Sinne der Kostenminimierung ist dies zu vermeiden, da der zusätzliche Worker sehr gering ausgelastet sein wird. Um dem Planer die Möglichkeit zu geben, diese Situation schnell zu erkennen, wird der Handlungsbedarf zum „Reparieren“ eines Arbeitsplatzes aus der positiven Differenz der alten Werkeranzahl zur neuen Werkeranzahl abgeleitet (5.22). Dabei wird als Referenz die alte Baurate des Arbeitsvorganges br_{avo}^{alt} verwendet.

$$\Delta w_{ap} = \max\left[0; \left\lceil \frac{\sum_{avo \in \mathcal{AVO}} \tau_{avo} \cdot br_{avo} \cdot x_{avo,ap}^{Soll}}{tz} \right\rceil - \left\lceil \frac{\sum_{avo \in \mathcal{AVO}} \tau_{avo} \cdot br_{avo}^{alt} \cdot z_{avo,ap}^{Ist}}{tz} \right\rceil \right] \quad \forall \quad ap \in \mathcal{AP} \quad (5.22)$$

Der entgegengesetzte Fall, in dem sich die Werkeranzahl aufgrund eines veränderten Auftragsprogramms verringert, ist sehr unwahrscheinlich, da hierfür eine deutliche Änderung des Auftragsprogramms notwendig wäre. Ein direkter Handlungsbedarf für das „Reparieren“ entsteht jedoch auch in diesem Fall nicht.

²⁶³Dies entspricht dem „Issue Detection“ Prozess, definiert von Smith in [Smi98, S.44] bzw. dem Analyseprozess aus [TA98]

²⁶⁴Vgl. Abschnitt 2.1.2.1

Komprimieren Durch eine Verringerung der Bauraten an bestimmten Arbeitsplätzen kann Freiraum für das Hinzufügen von weiteren Arbeitsvorgängen entstehen. Lohnenswert ist dieser Schritt jedoch erst, wenn die Chance besteht, einen kompletten Werker einzusparen, und sich die Umstellungskosten hierfür rechnen. Da sich die Werkeranzahl wie oben beschrieben direkt aus der durchschnittlichen Belastung eines Arbeitsplatzes ableitet, müssen hierfür so lange Arbeitsvorgänge von dem Arbeitsplatz herunter genommen werden, bis dort ein Werker weniger benötigt wird. Hierfür muss es möglich sein, diese Arbeitsvorgänge auf andere Arbeitsplätze zu verteilen.²⁶⁵

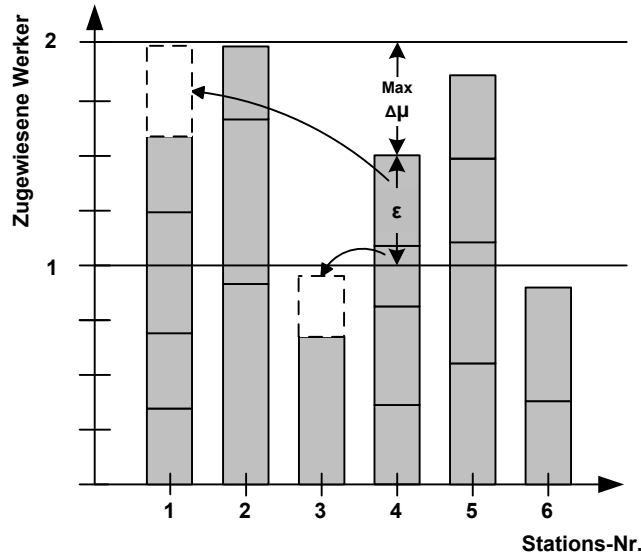


Abbildung 5.3: Beispiel für einen Komprimierungsvorgang im Rahmen der Rekonfiguration

Im ersten Schritt wird der Arbeitsplatz mit dem größten prozentualen Taktausgleich, definiert in Gleichung (5.23), ausgewählt. Dies ist der Arbeitsplatz mit der geringsten durchschnittlichen Auslastung μ_{ap} , gemessen an seiner Kapazität ζ_{ap} .

$$\Delta\mu_{ap} = 1 - \frac{\mu_{ap}}{\zeta_{ap}} \quad \forall \quad ap \in \mathcal{AP} \quad (5.23)$$

Zusätzlich muss gewährleistet sein, dass sich die Arbeitsvorgänge des ausgewählten Arbeitsplatzes auf andere Arbeitsplätze verteilen lassen und dies auch kostenmäßig sinnvoll ist. Dies ist wiederum nur durch eine Einbeziehung des Planers bewertbar. Die Variable ϵ_{ap} , definiert in Gleichung (5.24), gibt den mindestens zu verschiebenden Arbeitsaufwand zur Komprimierung und damit zur Reduzierung um einen Werker auf einem Arbeitsplatz ap an.

²⁶⁵Vgl. dazu Grafik 5.3

$$\epsilon_{ap} = \mu_{ap} - (tz \cdot (w_{ap} - 1)) \quad \forall \quad ap \in \mathcal{AP} \quad (5.24)$$

Es muss die Bedingung in Gleichung (5.25) gelten, damit eine Komprimierung möglich ist.²⁶⁶

$$\epsilon_{ap1} \leq \sum_{ap2 \in \mathcal{AP} \mid ap1 \neq ap2} \Delta\mu_{ap2} \quad (5.25)$$

Antizipieren Nach der Minimierung der eingesetzten Werker folgt die Minimierung der Übertaktzeiten zur Minimierung der daraus folgenden Unterstützereinsätze. Im Folgenden werden verschiedene Kennzahlen vorgestellt, um zu bearbeitende Arbeitsplätze zu identifizieren. Mit Bezug zu den beiden alternativen Zielfunktionen kann der Arbeitsplatz ap mit der größten linearen Übertaktzeit Δ_{ap}^{LinPos} (5.26) oder der Arbeitsplatz ap mit der größten quadrierten Übertaktzeit $\Delta_{ap}^{QuadPos}$ (5.27) gewählt werden.

$$\Delta_{ap}^{LinPos} = \sum_{p \in \mathcal{P}} \max[0; (\tau_{p,ap} - tz \cdot w_{ap})] \quad \forall \quad ap \in \mathcal{AP} \quad (5.26)$$

$$\Delta_{ap}^{QuadPos} = \sum_{p \in \mathcal{P}} \begin{cases} (\tau_{p,ap} - \zeta_{ap})^2 & \text{wenn } \tau_{p,ap} \geq \zeta_{ap} \\ 0 & \text{sonst} \end{cases} \quad \forall \quad ap \in \mathcal{AP} \quad (5.27)$$

Obwohl eine Nutzung dieser Kennzahlen logisch erscheint, kann ein relativ hoher Taktausgleich $\Delta\mu_{ap}$ das Potential für die Erzeugung von Unterstützereinsätzen auf einem Arbeitsplatz senken. Eine Maßzahl, die die Übertaktzeiten ins Verhältnis zum Taktausgleich setzt, wird in (5.28) definiert.

$$\rho_{ap}^{LinPos} = \frac{\Delta_{ap}^{LinPos}}{\Delta\mu_{ap}} \quad \forall \quad ap \in \mathcal{AP} \quad (5.28)$$

Analog dazu kann auch wieder eine quadrierte Version verwendet werden, um besonders große Ausreißer stärker zu bestrafen (5.29).

$$\rho_{ap}^{QuadPos} = \sum_{p \in \mathcal{P}} \begin{cases} \frac{(\tau_{p,ap} - \zeta_{ap})^2}{\Delta\mu_{ap}}, & \text{wenn } \tau_{p,ap} \geq \zeta_{ap} \\ 0, & \text{sonst} \end{cases} \quad \forall \quad ap \in \mathcal{AP} \quad (5.29)$$

²⁶⁶Die ist keine hinreichende, jedoch eine notwendige Bedingung. Der Vorranggraph sowie das Verhältnis von Umstellungskosten zur Verringerung der Werkerkosten und weitere Beschränkungen, wie z.B. Lagerplatz etc. schränken die Bedingung weiter ein.

5.1.3.1.2 Auswahl des Betrachtungsbereiches Wichtig für die Reduzierung der Komplexität im Sinne einer Entscheidungsunterstützung ist es, dass der Betrachtungsbereich eines Planers sinnvoll reduziert wird. Dies hat den Vorteil, dass der Planer nicht mit irrelevanten Informationen überflutet wird und die Komplexität für Algorithmen zur Ermittlung möglicher Lösungen deutlich reduziert werden kann. Durch die vorgestellten Kennzahlen, die in Tabelle 5.4 noch einmal zusammengefasst sind, können für die drei Schritte Reparieren, Komprimieren und Antizipieren die zu betrachtenden Arbeitsplätze ausgewählt werden.

Symbol	Definition
Δw_{ap}	Gibt die Differenz der Anzahl zugewiesener Werker an Arbeitsplatz ap im zukünftigen Planungszeitraum im Vergleich zur momentanen Abstimmung wieder. Bei positivem Wert zeigt diese Kenngröße, dass es notwendig ist, die Belastung zu “reparieren”.
$\Delta \mu_{ap}$	Beschreibt den Taktausgleich auf Arbeitsplatz ap . Für ein “Komprimieren” hat der Arbeitsplatz mit dem größten Taktausgleich das höchste Potential. Je höher der Taktausgleich, desto mehr lineare Übertaktzeit kann kompensiert werden.
ϵ_{ap}	Beschreibt die Höhe der Bearbeitungszeit, die von einem Arbeitsplatz ap verteilt werden muss, damit ein Werker weniger zugewiesen wird.
Δ_{ap}^{LinPos}	Geht direkt in die Zielfunktion ein und beschreibt die lineare Übertaktzeit auf Arbeitsplatz ap . Je größer diese Kennzahl ist, umso höher die Wahrscheinlichkeit, dass Überlastungssituationen auf dem Arbeitsplatz auftreten und somit Unterstützereinsätze hervorgerufen werden.
$\Delta_{ap}^{QuadPos}$	Geht direkt in die alternative Zielfunktion ein und beschreibt die quadrierte Übertaktzeit auf Arbeitsplatz ap . Je größer diese Kennzahl ist, umso höher die Wahrscheinlichkeit, dass Überlastungssituationen auf dem Arbeitsplatz auftreten und somit Unterstützereinsätze hervorgerufen werden. Höhere Ausreißer werden durch die Quadrierung stärker bestraft.
ρ_{ap}^{LinPos}	Ist definiert durch das Verhältnis aus linearer Übertaktzeit und Taktausgleich auf einem Arbeitsplatz ap . Diese Kennzahl erweitert die lineare Übertaktzeit um den existierenden Taktausgleich, der, bei guter Wahl der Reihenfolge, Überlastungen kompensieren kann. Sie wird verwendet zur Identifikation von Arbeitsplätzen, die im Antizipationsvorgang bearbeitet werden sollten.
$\rho_{ap}^{QuadPos}$	Ist definiert durch das Verhältnis aus quadrierter Übertaktzeit und Taktausgleich auf einem Arbeitsplatz ap . Diese Kennzahl erweitert die quadrierte Übertaktzeit um den existierenden Taktausgleich, der, bei guter Wahl der Reihenfolge, Überlastungen kompensieren kann. Sie wird verwendet zur Identifikation von Arbeitsplätzen, die im Antizipationsvorgang bearbeitet werden sollten. Höhere Ausreißer werden durch die Quadrierung stärker bestraft.

Tabelle 5.4: Kennzahlensystem zur Auswahl des Betrachtungsbereiches für eine Rekonfiguration

Nachdem ein zu bearbeitender Arbeitsplatz identifiziert wurde, müssen alternative Arbeitsplätze für die zu verteilenden Arbeitsvorgänge ausgewählt werden. Die Auswahl der alternativen Arbeitsplätze sollte nach verschiedenen Kriterien erfolgen. Insbesondere hinsichtlich der Werkstückposition $wp_{st_{ap}}$, in der sich das Werkstück an Arbeitsplatz ap befindet, sollten die alternativen Arbeitsplätze übereinstimmen. So ist gewährleistet, dass sich Arbeitsvorgänge zwischen den Arbeitsplätzen gut verschieben lassen. Des Weiteren sollte genügend Taktausgleich $\Delta\mu_{ap}$ und Lagerplatz pl_{ap} auf den alternativen Arbeitsplätzen existieren. Je näher die zum zu bearbeitenden Arbeitsplatz ap ausgewählten Alternativarbeitsplätze liegen, desto höher ist die Wahrscheinlichkeit, dass Verschiebungen gefunden werden, die den technischen Restriktionen aus dem Vorranggraphen genügen. Verschiebt man Arbeitsvorgänge nach hinten im Produktionsprozess, so müssen alle Arbeitsvorgänge zwischen dem alten und dem neuen Arbeitsplatz ohne Vorrangbeziehung zum zu verschiebenden Arbeitsvorgang sein. Analog dazu muss dies für eine Verschiebung nach vorne gelten. Es sei denn, der Vorgang, der eine Vorrangbeziehung zum Arbeitsvorgang besitzt, lässt sich ebenfalls weiter nach hinten oder vorne verschieben. Daher sollten möglichst nah liegende alternative Arbeitsplätze bevorzugt werden. Im Sinne der Ergonomie und Bearbeitungszeitreduzierung ist es sinnvoll, Arbeitsplätze zu betrachten, die ähnlichen Ansprechzonen sp_{ap} zugeordnet sind.²⁶⁷ Werden Arbeitsvorgänge auf einen Arbeitsplatz verschoben, welcher einer anderen Ansprechzone zugeordnet ist, so kann hierdurch ein zusätzlicher Laufweg und u.U. eine Bearbeitungszeiterhöhung entstehen. Eine Einschränkung auf diese alternativen Arbeitsplätze wäre jedoch falsch, da hierdurch Flexibilität in der Planungslösung verloren geht.

Die endgültige Entscheidung über die Auswahl der zu betrachtenden Arbeitsplätze kann nur vom Planer selbst getroffen werden, da diese zusätzlich auch von potentiellen Verschiebungskosten und Umschulungskosten der Werker abhängt. Durch die Kennzahlen und Informationen, die über die Arbeitsplätze zur Verfügung gestellt werden, wird jedoch erreicht, dass die Auswahl erfolgsversprechender Betrachtungsbereiche durch den Planer begünstigt wird.

5.1.3.2 Design-Phase

Nach der Ableitung des Handlungsbedarfs und Auswahl des Betrachtungsbereiches sollen im Folgenden Konzepte zur Erhöhung des Problemverständnisses beim Planer sowie Methoden zum Auffinden von sinnvollen Verschiebungen entwickelt werden. Hierfür wird die Notation um die Symbole in Tabelle 5.5 erweitert.

5.1.3.2.1 Visuelle Entscheidungsunterstützung Im Folgenden soll eine Visualisierung entwickelt werden, welche zusätzlich zur Durchschnittsbelastung auch die tatsächlichen Bearbeitungszeiten der nächsten Planungsperiode für einen Arbeitsplatz visualisiert. Dadurch soll das

²⁶⁷Vgl. Abschnitt 2.1.1

Symbol	Definition
$\mathcal{AA}$	Menge von Arbeitsaufgaben aa , die jeweils aus einer Menge von Arbeitsvorgängen $avo \in \mathcal{AVO}$ bestehen
$AlA_{avo,ap}$	Verhältnis zwischen der durchschnittlichen Bearbeitungszeit aller Werkstücke $p \in \mathcal{P}_{avo}$, an denen Arbeitsvorgang $avo \in \mathcal{AVO}$ verrichtet werden muss, und der Kapazität am Arbeitsplatz $ap \in \mathcal{AP}$ (Auswirkung in Arbeitsplatz-Kennzahl)
β	Parameter für die lokale Suchheuristik in der Rekonfiguration. Definiert die Anzahl an Lösungen, die zur weiteren Untersuchung generiert werden sollen
CL	Menge von Clustern CL_i
$clgr$	Größe der Cluster $CL_i \in CL$
$\Delta\psi_{CL_i,ap}$	Differenz in der Auftrittshäufigkeit von Bearbeitungszeiten im Cluster CL_i nach der Verschiebung eines Arbeitsvorganges auf einen Arbeitsplatz $ap \in \mathcal{AP}$
Δ_{-avo}^{LinPos}	Veränderung in den linearen positiven Abweichungen der Bearbeitungszeiten Δ^{LinPos} nach Entfernung eines Arbeitsvorganges in einer Rekonfigurationslösung
Δ_{+avo}^{LinPos}	Veränderung in den linearen positiven Abweichungen der Bearbeitungszeiten Δ^{LinPos} nach Hinzufügen eines Arbeitsvorganges in einer Rekonfigurationslösung
κ	Parameter für die lokale Suchheuristik in der Rekonfiguration. Definiert die Anzahl an weiterverfolgten Lösungen je Verschiebungsstufe. Bestimmt die Genauigkeit des Algorithmus.
$K_a(b)$	Korrelation der Arbeitsvorgänge $a \in \mathcal{AVO}$ und $b \in \mathcal{AVO}$
μ_{ap}^{MIN}	Untergrenze für die durchschnittliche Auslastung von Arbeitsplatz $ap \in \mathcal{AP}$
μ_{ap}^{MAX}	Obergrenze für die durchschnittliche Auslastung von Arbeitsplatz $ap \in \mathcal{AP}$
$\mathcal{P}_{avo}$	Menge aller Werkstücke $p \in \mathcal{P}$, die Arbeitsvorgang $avo \in \mathcal{AVO}$ benötigen
$\psi_{i,ap}$	Auftrittshäufigkeit der Bearbeitungszeit i auf Arbeitsplatz $ap \in \mathcal{AP}$
$\psi_{CL_i,ap}$	Auftrittshäufigkeit von Bearbeitungszeiten im Cluster $CL_i \in CL$ auf Arbeitsplatz $ap \in \mathcal{AP}$
ω	Parameter für die lokale Suchheuristik in der Rekonfiguration. Definiert die Anzahl an erlaubten, aufeinander folgenden Verschiebungen.
χ	Parameter für die lokale Suchheuristik in der Rekonfiguration. Definiert die Zielfunktion der Suche

Tabelle 5.5: Notationserweiterung für die Design-Phase des ESS

Problemverständnis für die Belastungsschwankungen beim Planer erhöht und eine What-If Analyse ermöglicht werden.

Visualisierung der Bearbeitungszeiten Die Bearbeitungszeiten jedes Werkstückes p auf einem Arbeitsplatz ap können auf Basis der Gleichung (5.20) in Abschnitt 5.1.3.1.1 grafisch dargestellt werden. Dies geschieht mit Hilfe eines modifizierten Histogramms, in dem die

Y-Achse die unterschiedlichen Bearbeitungszeiten je Werkstück darstellt. Mit waagerechten Balken wird angezeigt, wie häufig eine bestimmte Bearbeitungszeit auftritt (5.30).

$$\psi_{i,ap} = \sum_{p \in \mathcal{P} | \tau_{p,ap} = i} 1 \quad (5.30)$$

Ein Beispiel für ein solches Histogramm ist in Abbildung 5.4 auf der linken Seite abgebildet. Die maximal abgebildete Häufigkeit beträgt nur zwei Werkstücke. Durch die Kombination von einer Vielzahl an unterschiedlichen Arbeitsvorgängen kann, wie im Beispiel, eine große Anzahl unterschiedlicher Bearbeitungszeiten entstehen. Dadurch wird das Histogramm unübersichtlich und erschwert das Problemverständnis. Den schlimmsten Fall stellt ein Produktionsprogramm dar, bei dem alle Werkstücke auf einem Arbeitsplatz unterschiedliche Bearbeitungszeiten benötigen.

Clustern der Häufigkeiten Um die Übersichtlichkeit der Darstellung zu erhöhen, werden die Häufigkeiten geclustert. Hierfür wird der Wertebereich in eine vorgegebene Anzahl von gleich großen Clustern CL_i zerlegt.²⁶⁸ Die Clustergröße $clgr$ ergibt sich aus der Größe des Wertebereiches und der gewünschten Anzahl Cluster $|CL|$. Die Häufigkeit für eine Bearbeitungszeit in einem Cluster wird durch Gleichung (5.31) ermittelt.

$$\psi_{CL_i,ap} = \sum_{p \in \mathcal{P} | (i-1) \cdot clgr \leq \tau_{p,ap} \leq i \cdot clgr} 1 \quad (5.31)$$

In dem geclusterten Histogramm werden nicht mehr die Häufigkeiten für einzelne Bearbeitungszeiten $\psi_{i,ap}$ angezeigt, sondern die Häufigkeiten für Bearbeitungszeiten innerhalb eines Clusters $\psi_{CL_i,ap}$. Der Balken wird in der Mitte des zugehörigen Clusters $(i \cdot clgr) - \frac{clgr}{2}$ angezeigt. Nach der Clusterung im Beispiel in Grafik 5.4 auf der rechten Seite lässt sich nun deutlich erkennen, welche Belastungen tendenziell häufiger oder weniger häufig auftreten.

Manuelle What-If Analyse Eine What-If Analyse bietet Messmethoden, um den Einfluss unabhängiger Variablen auf die abhängigen zu bewerten.²⁶⁹ Im Fall der Rekonfiguration ist die Zuteilung der Arbeitsvorgänge auf die Arbeitsplätze $x_{avo,ap}^{Soll}$ die unabhängige Variable. Alle Kennzahlen, die die Lösungen charakterisieren, stellen die abhängigen Variablen dar.

Das folgende Konzept hilft dem Planer, die Auswirkungen jeder Entscheidung und somit der Verschiebung eines Arbeitsvorganges auf einen neuen Arbeitsplatz direkt zu bewerten. Da

²⁶⁸Vgl. [BEPW06]

²⁶⁹Vgl. zur Definition von "What-If Analysen" [Phi88]

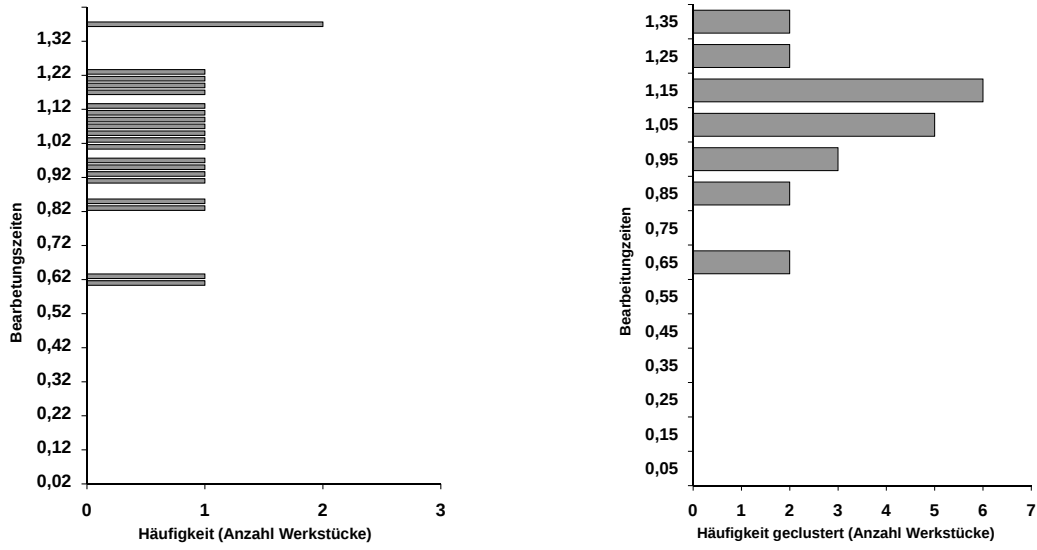


Abbildung 5.4: Beispiel für die Visualisierung der Bearbeitungszeiten (exakt und geclustert)

sich der Planer hierfür die alte Häufigkeitsverteilung der Bearbeitungszeiten auf den betroffenen Arbeitsplätzen merken müsste, wird eine Möglichkeit geschaffen, den Arbeitsvorgang im ersten Schritt nur vorläufig zu verschieben. Die mit dieser Verschiebung zusammenhängenden Auswirkungen in den Kennzahlen werden im Vergleich zur alten Lösung dargestellt um z.B. direkt überprüfen zu können, ob eine Verschiebung zu einer Überlastung eines Arbeitsplatzes führt. Die Veränderungen in den Histogrammen werden mit Hilfe von unterschiedlichen Mustern visualisiert. Eine Verringerung der Häufigkeit von Bearbeitungszeiten in einem Cluster wird durch eine Schraffierung des Balkens visualisiert. Eine erhöhte Häufigkeit ist durch einen gepunkteten Balken zu erkennen. Errechnet werden diese durch ein virtuelles Hinzufügen oder Wegnehmen des Arbeitsvorganges bei der Berechnung der Bearbeitungszeiten, wie in (5.32) und (5.33) definiert.

$$\tau_{p,ap} = \sum_{avo \in \mathcal{AVO}} \tau_{avo} \cdot x_{avo,ap}^{Soll} \cdot bp_{a,avo} \quad \forall ap \in \mathcal{AP}, p \in \mathcal{P}, a \in \mathcal{A} : h_{a,p} = 1 \quad (5.32)$$

$$\tau_{p,ap}^{NEU} = \sum_{avo \in \mathcal{AVO}} \tau_{avo} \cdot x_{avo,ap}^{Soll} \cdot bp_{a,avo} \pm \tau_{avoNEU} \cdot bp_{a,avo} \quad \forall ap \in \mathcal{AP}, p \in \mathcal{P}, a \in \mathcal{A} : h_{a,p} = 1 \quad (5.33)$$

Für den Anteil der veränderten Häufigkeit im Cluster wird dann die Differenz in den Clusterhäufigkeiten genommen (5.34), die jeweils mit den neuen bzw. alten Bearbeitungszeiten berechnet wurde. Ist sie positiv, handelt es sich um eine erhöhte Häufigkeit, sonst um eine niedrigere. Unveränderte Häufigkeiten werden in der ursprünglichen Farbe und im ursprünglichen Muster angezeigt.

$$\Delta\psi_{CL_i,ap} = \psi_{CL_i,ap}^{NEU} - \psi_{CL_i,ap}^{ALT} \quad (5.34)$$

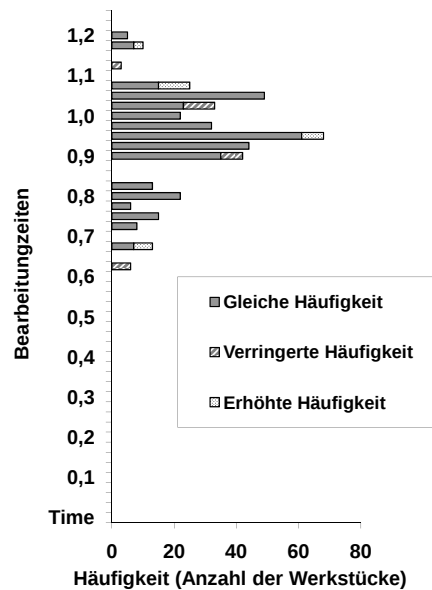


Abbildung 5.5: What-If Analyse mit Hilfe von Histogrammen

Abbildung 5.5 zeigt die Veränderungen in einem Histogramm durch die Verschiebung eines Arbeitsvorganges. Ebenfalls können Kombinationen von Verschiebungen mit diesem Verfahren visualisiert werden. Die unterschiedlichen Kennzeichnungen der Balken zeigen dann an, wie die gesamte Veränderung zu der Startlösung aussieht.

Erweiterte What-If Analyse In der erweiterten What-If Analyse wird die Möglichkeit gegeben, einen Arbeitsvorgang auf einem problematischen Arbeitsplatz auszuwählen und diesen virtuell gleichzeitig auf alle möglichen alternativen Arbeitsplätze zu verschieben. Hierdurch lässt sich über die Visualisierung der Auswirkungen gut erkennen, an welchem Arbeitsplatz der Arbeitsvorgang am besten mit den anderen Arbeitsvorgängen harmonisiert und wenig Überlastungen hervorruft.

5.1.3.2.2 AIA-Kennzahl zur Auswahl von Verschiebungen Im Rahmen der vorgestellten manuellen What-If Analyse müsste der Planer sämtliche Arbeitsvorgänge durchprobieren, um sicher zu gehen, dass er eine gute Lösung findet. Da dies in einem praktischen Umfeld einen erheblichen Aufwand bedeuten würde, wird weiterführend ein Konzept vorgestellt, welches sinnvolle Verschiebungen identifiziert.

Grundidee der Korrelationsanalyse Um Schwankungen in Bearbeitungszeiten zu vermeiden, ist es sinnvoll, Arbeitsvorgänge, die häufig zusammen auftreten, zu trennen, da sonst immer ein Werkstück mit beiden oder mit keinem der beiden Arbeitsvorgänge bearbeitet werden muss. Umgekehrt sollten Arbeitsvorgänge, die sich gegenseitig ausschließen, zusammengelegt werden, da entweder der eine oder der andere Arbeitsvorgang durchgeführt werden muss. Dies sorgt für eine gleichmäßigere Auslastung. Um die gleichzeitige Auftretshäufigkeit zweier Arbeitsvorgänge zu messen, wird im Folgenden eine veränderte Definition der Korrelation eingeführt.²⁷⁰

Definition der Korrelation von zwei Arbeitsvorgängen: Die Korrelation von zwei Arbeitsvorgängen wird auf einer Skala von -1 bis 1 beschrieben. Eine Korrelation von 1 bedeutet, dass die zwei betrachteten Arbeitsvorgänge immer gleichzeitig auftreten. Ergibt der Korrelationswert zweier Arbeitsvorgänge einen Wert von -1 , so wird von einer vollständig negativen Korrelation gesprochen und beide Arbeitsvorgänge treten immer getrennt auf. Die Werte dazwischen beschreiben eine Mischung der beiden Extreme. Korreliert a negativ mit b , so kann geschlussfolgert werden, dass a öfters ohne als mit b vorkommt. Ist diese Korrelation positiv, so gilt der umgekehrte Fall. Des Weiteren ist die Korrelation zwischen zwei Arbeitsvorgängen nicht symmetrisch. Angenommen, a korreliert mit b vollständig positiv. Arbeitsvorgang b kommt aber viel häufiger vor als a . Somit muss die Korrelation von b mit a einen geringeren Wert annehmen, da b oft ohne a auftritt. Durch diese Art der Betrachtung ist die Baurate implizit mit im Korrelationswert enthalten. Gleichung (5.35) beschreibt die Berechnung der Korrelation.

$$K_a(b) = \frac{H(a \cap b) - H(a \setminus b)}{H(a)} \quad (5.35)$$

Dabei steht $K_a(b)$ für die Korrelation von Arbeitsvorgang a mit Arbeitsvorgang b . $H()$ steht für die Häufigkeit, mit der ein bestimmtes Ereignis auftritt. Dabei ist $a \cap b$ das Auftreten von a und b an einem Werkstück, während $a \setminus b$ ein Werkstück beschreibt, das a , aber nicht Arbeitsvorgang b benötigt. $H(a)$ ist die Anzahl der Werkstücke mit Arbeitsvorgang a , unabhängig davon, welche anderen Arbeitsvorgänge an dem Werkstück durchgeführt werden müssen. Die

²⁷⁰Vgl. zur Definition in der Statistik [Har99]

Korrelationswerte beschreiben somit das Kundenverhalten, dass sich in der Werkstückzusammensetzung widerspiegelt. Zusätzlich zur Baurate, die die eindimensionale Häufigkeit eines Arbeitsvorganges wiedergibt, ist es wichtig zu betrachten, welche Ausstattungsmerkmale häufig zusammen auftreten.

Im Folgenden wird ein Beispiel beschrieben, welches demonstriert, wie das Identifizieren stark negativ korrelierender Arbeitsvorgänge bei der Rekonfiguration hilfreich sein kann. Dafür müssen Arbeitsvorgänge gefunden werden, die sich auf unterschiedlichen Arbeitsplätzen befinden und möglichst vollständig negativ miteinander korrelieren. Im Beispiel 5.6 gibt es zwei Arbeitsplätze mit einigen Standard-Arbeitsvorgängen und jeweils zwei variantenabhängigen Arbeitsvorgängen.

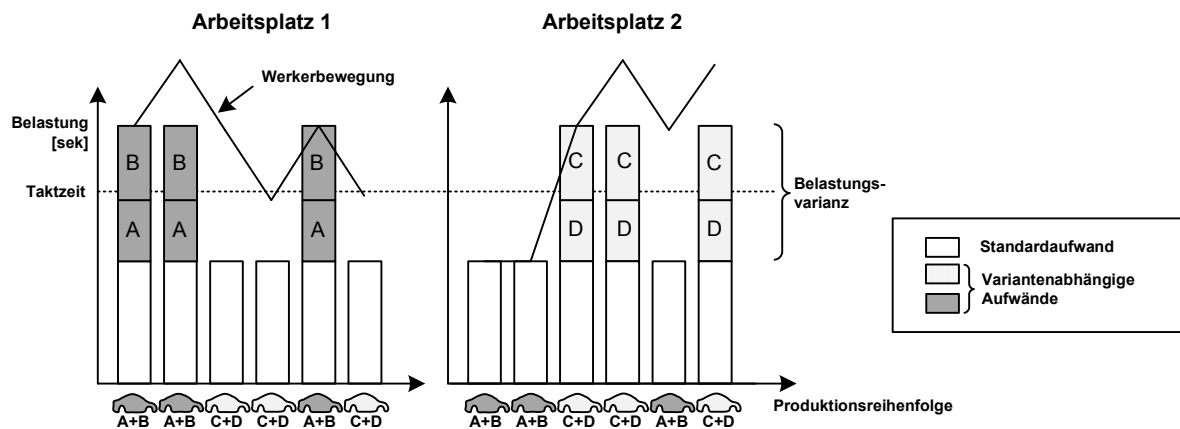


Abbildung 5.6: Beispiel für korrelierende Arbeitsvorgänge

Sowohl auf Arbeitsplatz 1 als auch auf Arbeitsplatz 2 sind die Bearbeitungszeiten stark schwankend. Bei einer genauen Betrachtung lässt sich erkennen, dass Arbeitsvorgang a und b genauso wie c und d immer gleichzeitig auftreten; $K_a(b) = 1$ und $K_c(d) = 1$ und im Gegensatz dazu a und c genauso wie b und d niemals gemeinsam auftreten; $K_a(c) = -1$ und $K_b(d) = -1$. Wird nun Arbeitsvorgang a auf Arbeitsplatz 2 und Arbeitsvorgang c auf Arbeitsplatz 1 verschoben (5.7), so sind die beiden variantenabhängigen Arbeitsvorgänge auf jedem Arbeitsplatz vollständig negativ korrelierend. Dadurch werden Spitzenbelastungen und Wartezeiten reduziert und somit die Schwankungen auf beiden Arbeitsplätzen minimiert.

Die Identifikation stark negativ korrelierender Arbeitsvorgänge auf unterschiedlichen Arbeitsplätzen kann somit vom Planer genutzt werden, um Arbeitsvorgänge auszuwählen, die verschoben werden sollen. Dieses Verfahren führt jedoch in einigen Fällen zu Problemen. Ist z.B. die Ausführungszeit eines Arbeitsvorganges wesentlich größer als die des negativ korrelierenden, so hat die Verschiebung nur einen geringen Effekt auf die Schwankung der Bearbeitungszeiten. Noch problematischer ist, dass nur die Korrelation von jeweils zwei Arbeits-

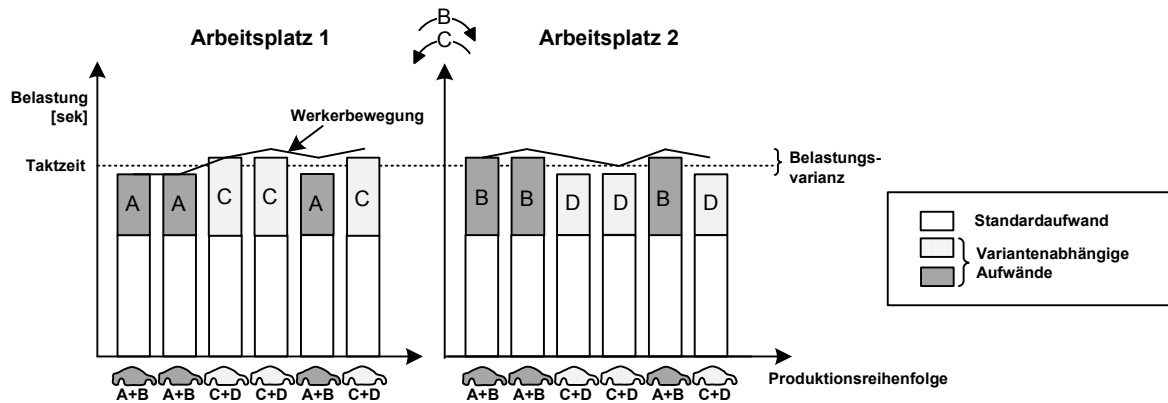


Abbildung 5.7: Varianzreduktion durch Trennung korrelierender Arbeitsvorgänge

vorgängen miteinander verglichen wird. In einer komplexen Produktionsumgebung mit vielen variantenabhängigen Arbeitsvorgängen werden die Bearbeitungszeiten auf den Arbeitsplätzen durch sehr viele Kombinationen von Arbeitsvorgängen generiert. Würde in so einem Szenario z.B. ein Arbeitsvorgang auf Grund einer vollständig negativen Korrelation mit einem beliebigen anderen Arbeitsvorgang verschoben, so kann dies sogar zu einer Zunahme der Belastungsschwankungen führen, falls der Arbeitsvorgang mit allen anderen Arbeitsvorgängen auf dem Ziellarbeitsplatz stark positiv korreliert. Darüber hinaus wird durch dieses Vorgehen nur die Schwankung von Bearbeitungszeiten betrachtet und kein Fokus auf Überlastungen gelegt.²⁷¹

Dies macht deutlich, dass eine zweidimensionale Korrelationsanalyse der Arbeitsvorgänge bei der Auswahl von Verschiebungen nicht weiterhilft. Es wird eine Maßzahl benötigt, die die Korrelation eines Arbeitsvorganges mit allen anderen Arbeitsvorgängen auf dem Arbeitsplatz bewertet und die Belastungsauswirkungen dieses Zusammenspiels bewertet. Hierfür wird die (Auswirkung in Arbeitsplatz) AIA-Kennzahl eingeführt.

Definition der AIA-Kennzahl Eine hilfreiche Verschiebung verringert hohe Auslastungen auf dem Quellarbeitsplatz²⁷², insbesondere über der Kapazität, und führt zu einer Erhöhung niedriger Auslastungen auf dem Ziellarbeitsplatz²⁷³. Die AIA-Kennzahl eines Arbeitsvorganges auf einem Arbeitsplatz $AIA_{avo,ap}$ gibt das Verhältnis zwischen der durchschnittlichen Bearbeitungszeit aller Werkstücke, an denen der Arbeitsvorgang verrichtet werden muss, und der Kapazität am Arbeitsplatz an (5.36).

²⁷¹Vgl. Diskussion in Abschnitt 5.1.1.1

²⁷²Der **Quellarbeitsplatz** ist der Arbeitsplatz, von dem ein Arbeitsvorgang entfernt wird.

²⁷³Der **Ziellarbeitsplatz** ist der Arbeitsplatz, dem der Arbeitsvorgang neu zugeordnet wird.

$$AIA_{avo,ap} = \frac{\frac{1}{|\mathcal{P}_{avo}|} \cdot \sum_{p \in \mathcal{P}_{avo}} \tau_{p,ap}}{\zeta_{ap}} \quad (5.36)$$

Symbol	Definition
$\mathcal{P}_{avo}$	Menge aller Werkstücke $p \in \mathcal{P}$ die Arbeitsvorgang $avo \in \mathcal{AVO}$ benötigen
$\tau_{p,ap}$	Bearbeitungszeit des Werkstückes $p \in \mathcal{P}$ an Arbeitsplatz $ap \in \mathcal{AP}$ in Zeiteinheiten
ζ_{ap}	Kapazität des Arbeitsplatzes $ap \in \mathcal{AP}$ in Zeiteinheiten

Tabelle 5.6: Notation zur AIA-Kennzahl

Soll die AIA-Kennzahl für einen alternativen Arbeitsplatz für einen Arbeitsvorgang avo berechnet werden, so werden die Bearbeitungszeiten der Werkstücke, die den Arbeitsvorgang benötigen, um den hinzukommenden Arbeitsvorgang korrigiert. Dafür müssen alle Bearbeitungszeiten für Werkstücke aus der Menge $\mathcal{P}_{avo}$ auf dem Zielarbeitsplatz durch eine Addition der Bearbeitungszeit τ_{avo} korrigiert werden.

Auswahlprozess zur Bestimmung der Verschiebungen durch die AIA-Kennzahl Ist der Wert einer AIA unter 1, so bedeutet dies, dass die Bearbeitungszeiten der Werkstücke, die den betrachteten Arbeitsvorgang enthalten, auf dem Arbeitsplatz durchschnittlich unter der Kapazitätsgrenze des Arbeitsplatzes liegen. Bei einer AIA über 1 liegen sie durchschnittlich darüber. Ziel ist es, Verschiebungen von Arbeitsvorgängen zu identifizieren, die auf dem momentan zugeordneten Arbeitsplatz eine hohe AIA (>1) besitzen, auf einem anderen Arbeitsplatz jedoch zu einer niedrigen AIA (<1) führen.²⁷⁴ Durch eine Verschiebung dieser Arbeitsvorgänge werden auf den ursprünglichen Arbeitsplätzen hohe Bearbeitungszeiten reduziert, während auf den neuen Arbeitsplätzen hauptsächlich niedrige Bearbeitungszeiten aufgebaut werden.

²⁷⁴Vgl. dazu Grafik (5.8)

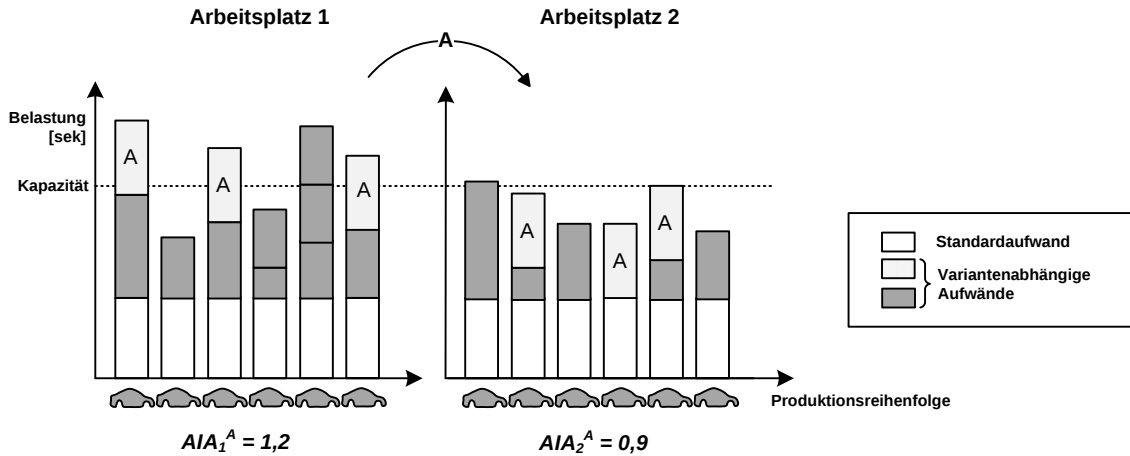


Abbildung 5.8: Identifikation einer Verschiebung mit Hilfe der AIA-Kennzahl

Zur weiteren Unterstützung des Planers wird für jeden Arbeitsvorgang im Betrachtungsbereich die AIA auf dem momentanen Arbeitsplatz sowie allen alternativen Arbeitsplätzen berechnet. Die so entstehende Liste von möglichen Verschiebungen wird nach dem Effekt an AIA-Verringerung sortiert. Bevor einer vorgeschlagenen Verschiebung zugestimmt wird, sollte der Planer immer auch die Visualisierung der Bearbeitungszeiten nutzen, da die AIA-Kennzahl nur einen Durchschnittswert berechnet und unter Umständen wenige aber sehr hohe Spitzen durch eine Verschiebung aufgebaut werden können.

5.1.3.2.3 Optimierungsansatz zur Bildung von Verschiebungskombinationen Die bisherigen Ansätze des ESS unterstützen nur jeweils die Evaluation einer Verschiebung eines Arbeitsvorganges hintereinander. Dies hat den Vorteil, dass der Planer jede einzelne Verschiebung validiert, bevor sie durchgeführt wird, und es relativ leicht fällt, dem Prozess zu folgen und den Überblick zu behalten. Im Folgenden wird das ESS durch einen Optimierungsansatz ergänzt, der auch mehrere aufeinander folgende Verschiebungskombinationen bewerten kann und die beste Kombination ermittelt. Wenn gewünscht, lässt sich auch die Bestimmung einer rechnerisch optimalen Lösung unter den gegebenen Bedingungen durchführen.²⁷⁵ Basierend auf einer existierenden Lösung werden mit Hilfe eines an eine lokale Suche angelehnten Verfahrens mehrere Lösungsalternativen generiert. Der Benutzer hat dann die Möglichkeit, diese Lösungen zu kontrollieren und die Kosten und den Nutzen jeder einzelnen Verschiebung abzuwägen. Der Algorithmus durchsucht dabei einen durch den Benutzer begrenzten Bereich und präsentiert die besten gefundenen Lösungen. Folgende Parameter dienen der Beeinflussung des Suchprozesses:

- χ - Definition der Zielgröße Δ^{LinPos} oder $\Delta^{QuadPos}$

²⁷⁵Vgl. hierzu „From what if to what’s best“ in [Asi87]

- β - Anzahl an Lösungen, die generiert werden sollen
- ω - Anzahl an erlaubten, aufeinander folgenden Verschiebungen
- κ - Anzahl an weiterverfolgten Lösungen je Verschiebungsstufe. Bestimmt die Genauigkeit des Algorithmus. Ist $\kappa > |\mathcal{AVO}|$, wird der Algorithmus die rechnerisch optimale Lösung bestimmen, dies geht jedoch deutlich zu Lasten der Laufzeit.
- μ_{ap}^{MAX} - Obergrenze an durchschnittlicher Auslastung je Arbeitsplatz ap
- μ_{ap}^{MIN} - Untergrenze an durchschnittlicher Auslastung je Arbeitsplatz ap
- Fixierung von Arbeitsvorgängen auf bestimmten Arbeitsplätzen

Nach einer Bewertung der Startlösung evaluiert der Algorithmus die vollständige Nachbarschaft, die mit der Verschiebung eines Arbeitsvorganges erreicht werden kann. Einige Verschiebungen sind dabei ungültig aufgrund von folgenden betrachteten Restriktionen:

- Vorrangrestriktionen: Jeder Arbeitsvorgang kann einen oder mehrere Vorgänger haben, die vor diesem Arbeitsvorgang durchgeführt werden müssen.²⁷⁶
- Grenzen für die Belastung von Arbeitsplätzen: Es können Ober- und Untergrenzen für die durchschnittliche Auslastung von Arbeitsplätzen definiert werden, um eine zu niedrige oder zu hohe Belastung des Arbeitsplatzes zu vermeiden.

$$\mu_{ap}^{MIN} \leq \mu_{ap} \leq \mu_{ap}^{MAX} \quad (5.37)$$

- Werkstückpositionen: Jeder Prozess hat eine Liste mit erlaubten Werkstückpositionen. Alle anderen Verschiebungen sind ungültig.²⁷⁷
- Fixierte Arbeitsvorgänge: Einige Arbeitsvorgänge müssen an bestimmten Arbeitsplätzen verbleiben, da an diesen spezielle Betriebsmittel vorhanden sind oder speziell qualifiziertes Personal vorhanden ist, und können daher fixiert werden. Mit Hilfe dieser Beschränkung kann der Planer den Algorithmus gezielt einschränken.
- Gruppierung von Arbeitsvorgängen: Der Benutzer kann Gruppen von Arbeitsvorgängen generieren, die immer gemeinsam verschoben werden müssen. Diese Gruppen, definiert in (5.38), werden vom Algorithmus wie ein summierter gemeinsamer Arbeitsvorgang behandelt.²⁷⁸

²⁷⁶Vgl. Restriktion (5.11) im Modell in Abschnitt 5.1.1.2

²⁷⁷Vgl. Restriktion (5.15) im Modell in Abschnitt 5.1.1.2

²⁷⁸In praktischen Szenarien existieren immer wieder einige Arbeitsvorgänge, bei denen es sinnvoll ist, diese zusammen von einem Werker durchführen zu lassen. Diese Situation kann z.B. durch einen Stempelvorgang gegeben sein, den nur ein Werker machen darf, wenn er auch den zugehörigen Arbeitsvorgang durchgeführt hat.

$$\sum_{avo \in aa} x_{avo,ap}^{Soll} = |AA| \cdot \max_{avo \in aa} \{x_{avo,ap}^{Soll}\} \quad \forall \quad aa \in AA, ap \in AP \quad (5.38)$$

Mit Rücksicht auf die Laufzeit des Algorithmus wird bei der Bewertung einer Verschiebung nicht alles neu berechnet.²⁷⁹ Es ist ausreichend, den Unterschied in der summierten Zielgröße, der von einer durchgeführten Verschiebung ausgeht, zu bestimmen. Dafür wird zuerst die Verringerung der Zielgröße auf dem Quellarbeitsplatz abgeleitet. Hierfür müssen nur die Werkstücke, die den Arbeitsvorgang benötigen, betrachtet werden. Für jedes dieser Werkstücke wird die Übertaktzeit ohne und mit dem Arbeitsvorgang bestimmt. Die Differenz entspricht der reduzierten Übertaktzeit in Gleichung (5.39).

$$\Delta_{-avo}^{LinPos} = \frac{1}{\overline{\mathcal{P}}_{avo}} \cdot \sum_{p \in \mathcal{P}_{avo}} \left(\begin{cases} (\tau_{p,ap} - \tau_{avo}) - tz \cdot w_{ap}, & \text{wenn } (\tau_{p,ap} - \tau_{avo}) \geq tz \cdot w_{ap} \\ 0, & \text{sonst} \end{cases} \right. \quad (5.39)$$

$$\left. - \begin{cases} \tau_{p,ap} - tz \cdot w_{ap}, & \text{if } \tau_{p,ap} \geq tz \cdot w_{ap} \\ 0, & \text{sonst} \end{cases} \right) \quad ap \in AP \mid x_{avo,ap}^{Soll} = 1$$

Im zweiten Schritt wird die Erhöhung der Zielgröße analog dazu auf dem Ziellarbeitsplatz bestimmt. In diesem Fall wird die Differenz zwischen der ursprünglichen Zielgröße und der Zielgröße unter Hinzunahme des Arbeitsvorganges in Gleichung (5.40) bestimmt.

$$\Delta_{+avo}^{LinPos} = \frac{1}{\overline{\mathcal{P}}_{avo}} \cdot \sum_{p \in \mathcal{P}_{avo}} \left(\begin{cases} (\tau_{p,ap} + \tau_{avo}) - tz \cdot w_{ap}, & \text{wenn } (\tau_{p,ap} + \tau_{avo}) \geq tz \cdot w_{ap} \\ 0, & \text{sonst} \end{cases} \right. \quad (5.40)$$

$$\left. - \begin{cases} \tau_{p,ap} - tz \cdot w_{ap}, & \text{if } \tau_{p,ap} \geq tz \cdot w_{ap} \\ 0, & \text{sonst} \end{cases} \right) \quad ap \in AP \mid x_{avo,ap}^{Soll} = 1$$

Zuletzt werden die beiden Größen Δ_{-avo}^{LinPos} und Δ_{+avo}^{LinPos} summiert. Je kleiner der Wert ist, umso höher ist die resultierende Reduktion der Zielgröße und umso besser ist somit die Lösung. Analog dazu funktioniert die Vorgehensweise bei der quadrierten Version der Zielfunktion. Die besten κ Lösungen aus der Nachbarschaft werden gespeichert. Jede dieser Lösungen unterscheidet sich von der Ausgangslösung nur um eine Verschiebung.²⁸⁰

²⁷⁹Speziell in einem praktischen Umfeld ist es wichtig, dass ein ESS akzeptable Antwortzeiten aufweist, da sonst keine Akzeptanz der Benutzer gewährleistet werden kann. Vgl. auch die Anforderungen an die Problemlösung in Abschnitt 2.3.1.

²⁸⁰Vgl. hierzu Grafik 5.9

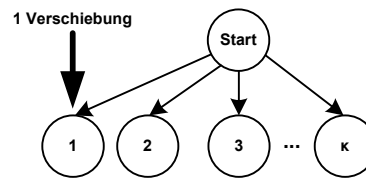


Abbildung 5.9: Beispiel für den Suchbaum der Heuristik nach einer Verschiebung

Jede gespeicherte Lösung ist Ausgangspunkt für eine erneute Nachbarschaftsanalyse auf der nächsten Stufe im Suchbaum. Auch hiernach werden wieder die besten κ Lösungen gespeichert und als Startlösung für die nächsten zu untersuchenden Verschiebungen verwendet. Diese Vorgehensweise wird wiederholt bis, wie in Grafik 5.10 gezeigt, Level ω im Suchbaum erreicht wird.

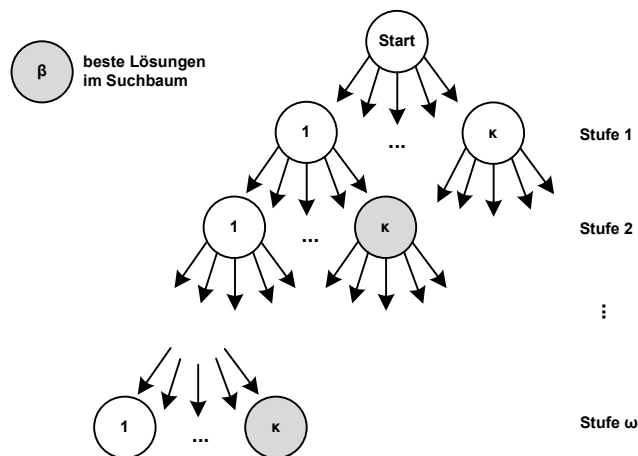


Abbildung 5.10: Kompletter Suchbaum der Heuristik

Zum Schluss werden die besten β Lösungen im Suchbaum selektiert. Identische Lösungen, die sich nur durch die Reihenfolge der Verschiebungen unterscheiden, werden dabei ignoriert. Basierend auf diesen Lösungen können sowohl mit der visuellen Entscheidungsunterstützung, der AIA-Kennzahl, als auch mit dem Optimierungsansatz weitere Lösungen generiert werden.²⁸¹

5.1.3.3 Auswahl-Entscheidungsphase

Mit Hilfe der Methoden in der Design-Phase wurden unterschiedliche Lösungsalternativen generiert. Es muss jedoch noch die Entscheidung für eine der generierten Zuordnungslösungen getroffen werden. Um diesen nicht immer stringenten Prozess der Entscheidungsfin-

²⁸¹Vgl. hierzu die im Folgenden vorgestellte Versionierungssystematik in Abschnitt 5.1.3.3.1

dung besser zu unterstützen, wird eine Versionierungssystematik entworfen. Für die Auswahl-Entscheidungsphase wird die Notation um die Bezeichner in Tabelle 5.7 erweitert.

Symbol	Definition
$\mathcal{K}^{BGes}$	Betriebskosten
$\mathcal{K}^{RGes}$	Gesamtkosten, die durch die Rekonfiguration und damit durch die Verschiebung von Arbeitsvorgängen verursacht werden.

Tabelle 5.7: Notationserweiterung für die Auswahl- und Entscheidungs-Phase des ESS

5.1.3.3.1 Versionierungssystematik Für die zuvor beschriebenen Methoden aus der Design-Phase sowie zur Entscheidung für eine der Lösungen ist es besonders wichtig, zu wissen, welche Lösungsalternative gerade betrachtet oder verändert wird und wie sich die verschiedenen Lösungen zueinander verhalten. Die Startlösung bildet dabei für alle Veränderungen die Referenz. Daher wird diese als bestätigte Zuordnung besonders gekennzeichnet - z.B. durch die Farbe blau. Alle alternativen Lösungen, die auf Basis dieser Zuordnung generiert werden, stellen Lösungsalternativen dar, die noch nicht bestätigt sind. Daher werden sie z.B. rot angezeigt. Da es sich bei diesen Alternativen um Veränderungen der blauen Startlösung handelt, werden sie innerhalb einer hierarchischen Baumstruktur als Unterknoten der Startlösung dargestellt, wie in Grafik 5.11 gezeigt.

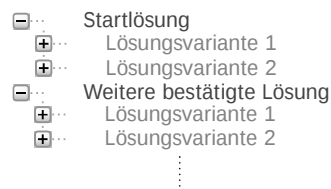


Abbildung 5.11: Versionierungssystematik im Alternativenbaum

Von der Startlösung aus können beliebig viele Lösungsalternativen gebildet werden, welche jeweils einen roten Unterknoten unter der Startlösung darstellen. Der Planer kann so sehr einfach die verschiedenen Alternativen auswählen und vergleichen. Die Veränderungen in den Kennzahlen werden in Form der in Abschnitt 5.1.3.2.1 vorgestellten What-If Analyse dargestellt und beziehen sich immer auf eine blaue bestätigte Lösung, die eine Ebene höher liegt. Aus einer roten Lösungsalternative kann jederzeit eine blaue, bestätigte Lösung generiert werden, welche dann wieder als Referenz für neue rote Lösungen genutzt werden kann. Mit Hilfe dieser Versionierungssystematik ist es so möglich, auf Basis jeder Lösung jeweils die verschiedenen Methoden aus der Design-Phase anzuwenden sowie bei bereits erstellten Alternativen Rücksprünge zur Ausgangslösung zuzulassen.

5.1.3.3.2 Kosten/Nutzen-Abwägung Nachdem unterschiedliche Lösungsalternativen generiert wurden, ist es Aufgabe des Planers, die beste Lösung auszuwählen. Hierfür werden alle erläuterten Kennzahlen je Lösung dargestellt. Die Rekonfigurationskosten $\mathcal{K}^{RGes}$ der einzelnen Lösungen l müssen durch den Planer abgeschätzt werden. Dafür werden die Kosten für jede einzelne Verschiebung abgefragt und in Gleichung (5.41) summiert.

$$\mathcal{K}^{RGes} = \sum_{ap \in \mathcal{AP}} \sum_{avo \in \mathcal{AVO}} \mathcal{K}_{avo,ap}^R \cdot v_{avo,ap}^R \quad (5.41)$$

Mit Hilfe der Kostensätze für die notwendigen Werker und dem Kostensatz für die lineare Übertaktzeit werden die erwarteten Betriebskosten bei Umsetzung der einzelnen Alternativlösungen (5.42) ermittelt.

$$\mathcal{K}^{BGes} = \Delta^{LinPos} \cdot \mathcal{K}^{LinPos} + \sum_{ap \in \mathcal{AP}} \mathcal{K}^W \cdot w_{ap} \quad (5.42)$$

Die Verbesserungen in den Betriebskosten der Alternativlösungen müssen jeweils vom Planer um die abgeschätzten Rekonfigurationskosten bereinigt werden. Auf dieser Basis kann dann die beste Lösung ermittelt werden.

Zusätzlich können die erstellten Planungslösungen mit verschiedenen Reihenfolgen der Werkstücke simuliert werden. Dabei sollte von einer Abschätzung einer Reihenfolge, indem die Reihenfolgeplanung aus Abschnitt 5.3 verwendet wird, an dieser Stelle aus folgenden Gründen abgesehen werden:

1. Alle Aufträge des Rekonfigurationszeitraumes müssten sequenziert werden (Komplexitätsproblem).
2. Eine Zuordnung auf Produktionszeiträume durch die Programmplanung ist noch nicht durchgeführt worden (Falsche Datengrundlage).
3. Die fiktiv gebildete Reihenfolge könnte zufällig überdurchschnittlich gut mit der Abstimmungslösung harmonieren und somit eine Lösung bevorzugen, die lediglich mit genau dieser Reihenfolge gute Ergebnisse erzielt.

Um dennoch die Fließliniensimulation aus Abschnitt 5.3.3 zur genaueren Abschätzung der Betriebskosten nutzen zu können, werden unabhängig von der Reihenfolgeplanung ausreichend zufällige Reihenfolgen gebildet. Diese werden jeweils mit jeder Abstimmungslösung simuliert und die durchschnittliche Anzahl Unterstützereinsätze sowie die Anzahl an notwendigen Unterstützerarbeitskräften ermittelt. Wenn die Anzahl getesteter Produktionsreihenfolgen groß genug gewählt wird, kann so eine von der Reihenfolge unabhängige Bewertung

der Abstimmung durchgeführt werden. Auf diesem Wert aufbauend wird dem Planer eine genauere Abschätzung der Betriebskosten möglich gemacht.

5.1.3.4 Implementationsphase

In der Implementationsphase wird die Lösung, die in der Entscheidungsphase ausgewählt wurde, umgesetzt. Dafür werden die notwendigen Verschiebungen von Arbeitsvorgängen in Gleichung (5.43) abgeleitet.

$$v_{avo,ap}^R = x_{avo,ap}^{Soll} - z_{avo,ap}^{Ist} \quad \forall \quad avo \in \mathcal{AVO}, ap \in \mathcal{AP} \mid x_{avo,ap}^{Soll} = 1 \quad (5.43)$$

Während der praktischen Implementation einer Lösung entstehen Probleme und Beschränkungen, die vorher nicht absehbar waren. Hierfür ist in dem Konzept immer ein Rück- sowie Vorwärtssprung zwischen den einzelnen Phasen des ESS sowie verschiedenen Lösungsalternativen möglich.

Nach erfolgreicher Implementation ist die Zuordnung $z_{avo,ap}^{Ist}$ auf die in der Rekonfiguration geplante Zuordnung $x_{avo,ap}^{Soll}$ angepasst worden.

5.2 Teilautomatisierte Erstellung eines Variantenvorranggraphen

Ziel der teilautomatisierten Erstellung eines Variantenvorranggraphen ist die Erzeugung eines Vorranggraphen, der alle Varianten eines Produktes beschreibt. Durch das Konzept soll der Graph möglichst weit automatisiert erstellt und die Komplexität so weit reduziert werden, dass eine anschließende manuelle Feinstrukturierung durch einen Montageplaner möglich ist.

5.2.1 Vorgehensmodell zur Vorranggraphenerstellung

Existierende Methoden zur Vorranggraphenerstellung beginnen mit einem leeren Vorranggraphen.²⁸² Aufgabe des Planers ist es, Vorrangbedingungen zu definieren und in den Graphen einzubauen.²⁸³ Bei diesem Bottom-Up Verfahren ist die Gefahr, Restriktionen zu vergessen und damit ungültige Freiheitsgrade zu erzeugen, sehr hoch.²⁸⁴

Im Folgenden wird ein alternativer Ansatz vorgestellt, der ausgehend von existierenden Zuordnungen und Stücklisteninformationen eine automatische Generierung des Vorranggraphen ermöglicht. Bei diesem Vorgehen wird mit einer existierenden Bearbeitungsreihenfolge der Arbeitsvorgänge begonnen. Nun werden sukzessive Freiheitsgrade im Graphen hinzugefügt, indem Vorrangbedingungen, die nicht gültig sind, entfernt werden. Durch dieses Top-Down Verfahren ist sichergestellt, dass keine ungültigen Freiheitsgrade identifiziert werden und somit bei einer Verwendung des Graphen im Rahmen einer Rekonfiguration alle Verschiebungen, die durch den Ansatz zur Optimierung in Abschnitt 5.1.3.2.3 generiert werden, valide sind. Im Vergleich zum vorher vorgestellten Bottom-Up kann im Gegenzug jedoch nicht garantiert werden, dass sämtliche Freiheitsgrade erkannt werden. Dieses Verfahren entspricht somit dem umgekehrten Weg, um zum selben Ziel zu kommen. In Grafik 5.12 wird der Unterschied verdeutlicht.

Über die erkannten Freiheitsgrade hinaus kann in einem nachgelagerten Schritt eine manuelle Feinstrukturierung des Graphen vorgenommen werden. Dabei ist der Graph bis zu diesem Schritt so weit strukturiert, dass die reduzierte Komplexität eine manuelle Bearbeitung möglich macht. Zusätzlich zur Erkennung der Freiheitsgrade werden weitere Informationen, die aus der automatischen Analyse der Produktdokumentation hervorgehen, im Graphen zur Verfügung gestellt.

²⁸²Leer bedeutet in diesem Zusammenhang, dass zwar alle Knoten vorhanden sind, jedoch keine Vorrangbedingungen existieren.

²⁸³Vgl. Abschnitt 3.2.1 und [Che89, S.243],[JK84] sowie [DFW87]

²⁸⁴Es existieren Forschungsansätze, die dafür sorgen, dass jede Beziehung überprüft wird und keine Inkonsistenzen zwischen den transitiven und direkten Vorrangbeziehungen auftreten. Allerdings sind diese Ansätze bisher den komplexen Anforderungen einer Variantenabbildung nicht gewachsen. Vgl. dazu z.B. [Mei07].

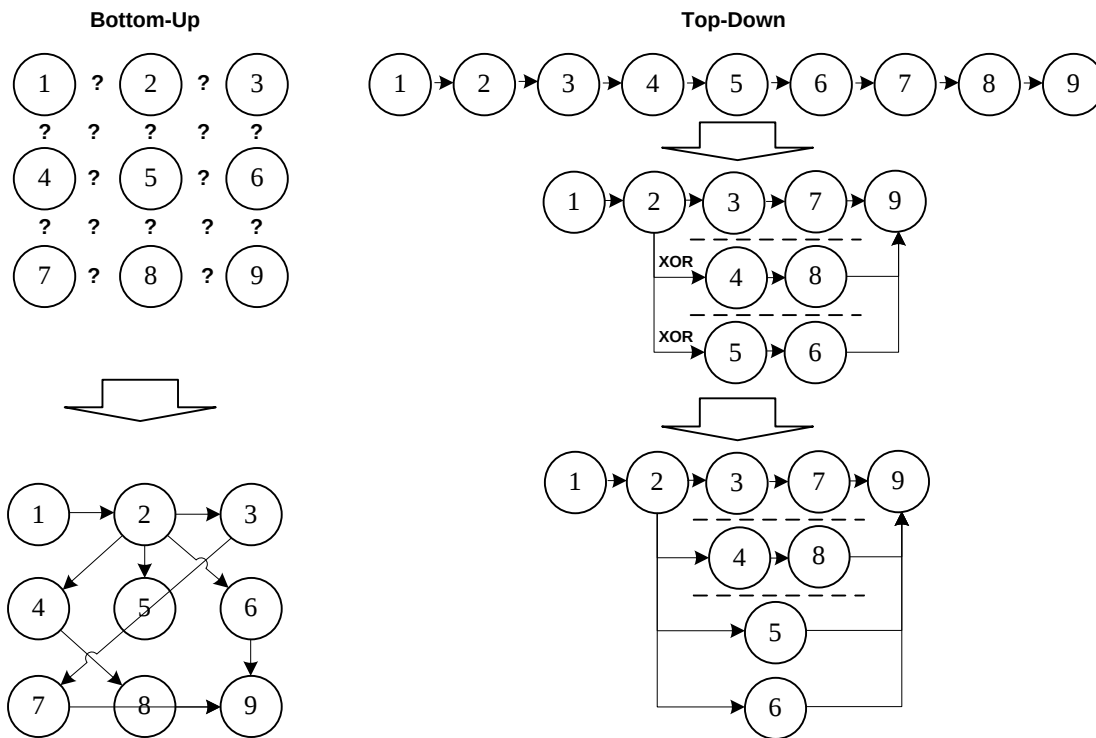


Abbildung 5.12: Vergleich des Top-Down und Bottom-Up Ansatzes zur Vorranggraphenentwicklung

Das grobe Vorgehen wird in Grafik 5.13 skizziert und in den folgenden Kapiteln im Detail beschrieben. Dabei wird zuerst die genutzte Datengrundlage definiert. Im Anschluss werden die verschiedenen Konzepte zur automatischen Generierung von Freiheitsgraden vorgestellt.

5.2.2 Datengrundlage

Das Konzept baut auf Stücklistendaten, Baubarkeitsregeln sowie bereits existierenden Arbeitszuordnungen auf und analysiert diese. Auch bei einer Erstinstallation ist eine Anwendung des Konzeptes denkbar, da selten ein komplett neues Produkt produziert wird, sondern häufig Daten von Vorgängerprodukten bzw. sehr ähnlichen Produkten verwendet werden können.²⁸⁵ Die Konzepte werden insbesondere unter Verwendung der offenen Variantenstückliste beschrieben.²⁸⁶ Die Grundstruktur dieser Stücklistenart und die notwendige Aufbereitung wird im Folgenden an einem Beispiel aus der Automobilindustrie erläutert.²⁸⁷

²⁸⁵Vgl. [Fal05]

²⁸⁶Vgl. Abschnitt 2.2.2.3

²⁸⁷Vgl. zum Stücklistenaufbau mit Code-Regeln [Sin03]

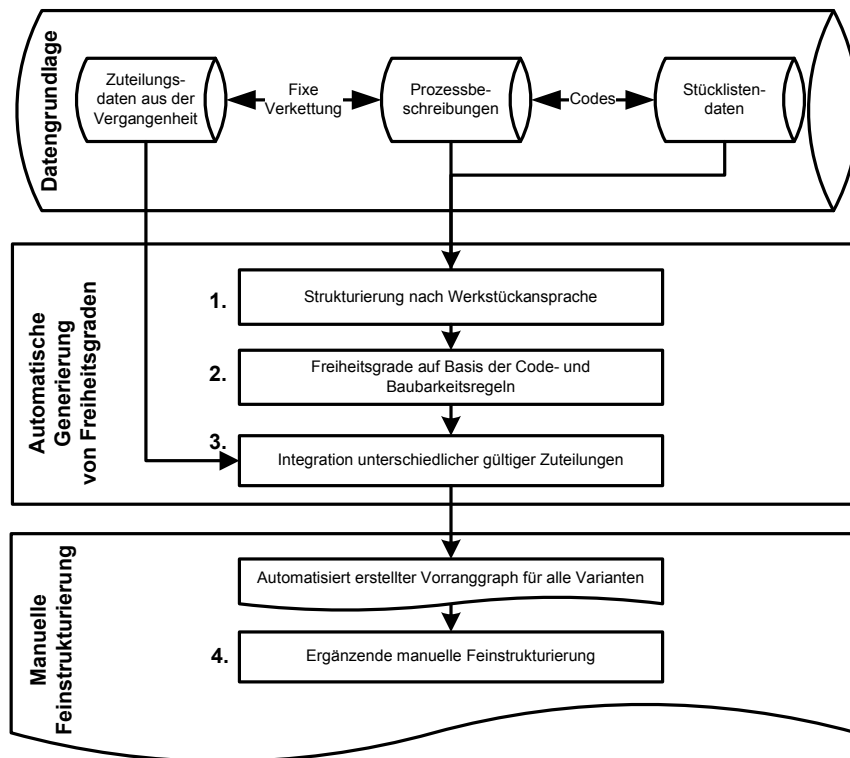


Abbildung 5.13: Vorgehen zu Vorranggraphenbildung

5.2.2.1 Produktdokumentation mit Hilfe der offenen Variantenstückliste

Wie in Abschnitt 2.2.2.3 beschrieben, wird ein Auftrag in der offenen Variantenstückliste durch die von einem Kunden festgelegten Eigenschaften definiert. In Tabelle 5.8 sind beispielhaft einige Eigenschaften, welche im Folgenden durch sog. Codes dargestellt werden, aufgeführt.

Eigenschaftscodes	Ausstattung
R1	Standardradio
R2	Radio mit kleinem Navigationsgerät
R3	Radio mit großem Navigationsgerät
K1	Klimaanlage
K2	Klimaautomatik
M1	V6 Ottomotor
M2	V8 Ottomotor
M3	V6 Dieselmotor

Tabelle 5.8: Eigenschaften zur Beschreibung eines Auftrages (Beispiel)

Ein Fahrzeug besteht aus Hauptmodulen, wie z.B. dem Cockpit oder dem Dach, die wiederum in Submodule unterteilt werden können. Innerhalb eines Submoduls unterscheidet man physikalische Positionen, die den Einbauort von Bauteilen identifizieren. Innerhalb einer Position können verschiedene Varianten von Bauteilen verwendet werden. Diese Bauteilvarianten werden als Positionsvarianten bezeichnet und jeweils mit einer Regel versehen, welche die durch den Kunden gewählten Eigenschaften beinhalten. Diese Regel wird im Folgenden Coderegel genannt. Für die Produktion wie auch die Logistik ist es interessant, welche Teile in einem Auftrag verbaut werden müssen. Erst die Kombination der Coderegeln einzelner Positionsvarianten determiniert durch Anwendung der booleschen Algebra den Verbau eines bestimmten Einzelteils. Abbildung 5.14 gibt eine Übersicht über den Aufbau einer offenen Variantenstückliste.

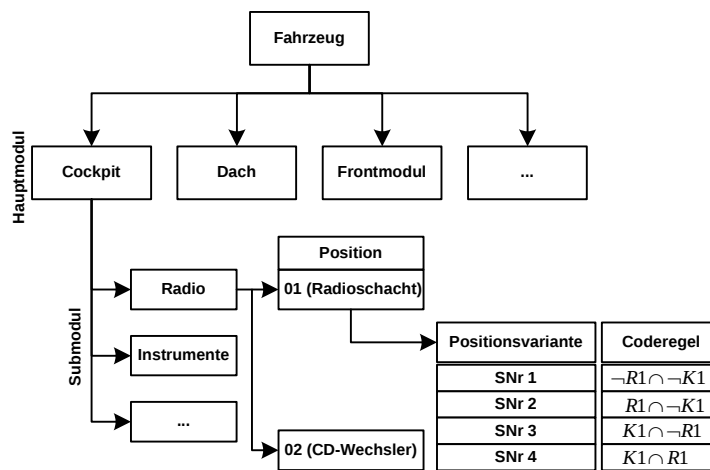


Abbildung 5.14: Beispielhafte Struktur einer offenen Variantenstückliste

Um die Pflege der Produktdokumentation zu vereinfachen, wird im Normalfall nur eine sog. kurze Coderegel hinterlegt.²⁸⁸ Diese Regel muss im Rahmen einer Teilebedarfsermittlung in eine lange Coderegel umgewandelt werden, da nur diese die Positionsvarianten vollständig voneinander abgrenzt. Genau diese Abgrenzung soll innerhalb des Konzeptes zur Vorranggraphenentwicklung verwendet werden. Daher wird im folgenden Abschnitt der Unterschied zwischen der kurzen und langen Coderegel verdeutlicht. Ein Beispiel findet sich in Tabelle 5.9.

Das Semikolon in der kurzen Coderegel der Positionsvariante 1 zeigt an, dass das zugehörige Teil immer verbaut wird. Dies ist jedoch nur der Fall, wenn keine der anderen Coderegeln innerhalb der Position wahr ist. Um eine eindeutige Regel zum Verbau jeder einzelnen Positionsvariante zu erhalten, bildet man das Komplement aller Codes aller anderen Positionsvarianten innerhalb einer Position und hängt dieses mit an die kurze Coderegel an. Das Ergebnis

²⁸⁸Vgl. dazu [Sin03]

Sachnummer	Modul	Position	Positionsvariante	Kurze Coderegel	Lange Coderegel
SNr 1	200	30	1	$;$	$\neg R1 \cap \neg K1$
SNr 2	200	30	1	$R1$	$R1 \cap \neg K1$
SNr 3	200	30	1	$K1$	$K1 \cap \neg R1$
SNr 4	200	30	1	$R1 \cap K1$	$R1 \cap K1$

Tabelle 5.9: Vergleich der kurzen und der langen Coderegel

ist die lange Coderegel, die in Form eines Venn-Diagramms, wie in Grafik 5.10 dargestellt, beschrieben werden kann.

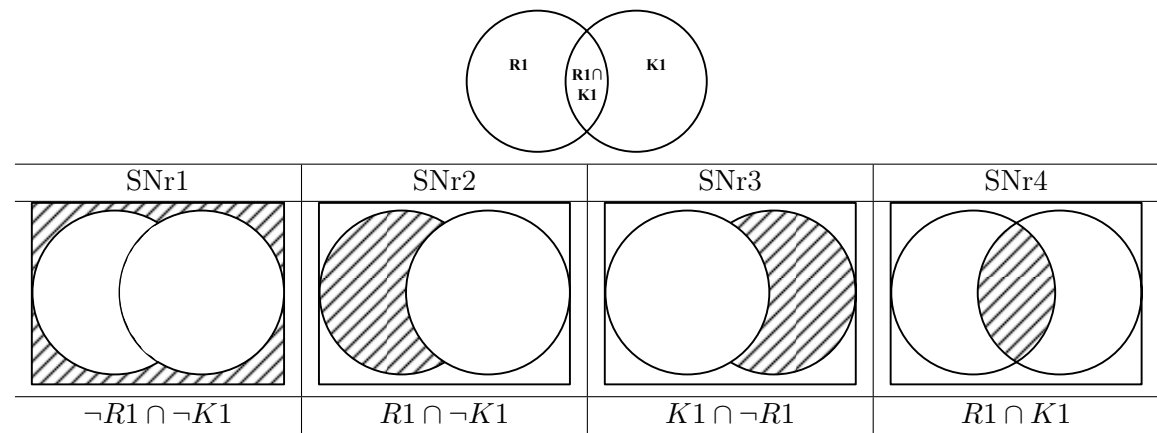


Tabelle 5.10: Darstellung der langen Coderegel als Venn-Diagramm

Die Entscheidung, welches Teil wann verbaut wird, hängt nicht nur von einem Code, sondern von einer Codekombination ab. Im Beispiel kann also der Einbau eines bestimmten Radios von dem verwendeten Klimagerät abhängen. Die Methode zur Erstellung der langen Coderegel wird im Folgenden beschrieben.²⁸⁹

5.2.2.2 Aufbereitung der Coderegeln

Die Aufbereitung der Coderegeln soll an dem kleinen Beispiel in Tabelle 5.11 mit 4 Positionsvarianten verdeutlicht werden. Für eine genauere Beschreibung der Zusammenhänge von Produktdokumentation mit Coderegeln sei auf Sinz²⁹⁰ verwiesen.

²⁸⁹Vgl. [Sin03]

²⁹⁰Vgl. [Sin03]

Positionsvariante	Coderegel
1	$a \vee a \wedge (b \vee c) \vee a \wedge b \wedge c$
2	b
3	c
4	e

Tabelle 5.11: Beispiel für Positionsvarianten und ihre Code-Regeln

Der erste Schritt zur Erzeugung der langen Coderegel ist die Überführung von allen Coderegeln in die disjunktive Normalform (DNF)²⁹¹, wie in Beispiel 5.12 gezeigt.

Positionsvariante	Coderegel
1	$a \vee a \wedge b \vee a \wedge c \vee a \wedge b \wedge c$
2	b
3	c
4	e

Tabelle 5.12: Beispiel für die Bildung der disjunktiven Normalform

Anschließend wird für jede Coderegel überprüft, ob sich darin Teilkonjunktionen²⁹² befinden, die alle anderen Teilkonjunktionen dominieren und somit eine Obermenge darstellen. Ist dies, wie im Beispiel 5.13 mit der Teilkonjunktion a der Fall, so werden diese in Unterpositionsvarianten ausgelagert.

Positionsvariante	Coderegel
1	$a \vee a \wedge b \vee a \wedge c \vee a \wedge b \wedge c$
1.1	a
2	b
3	c
4	e

Tabelle 5.13: Beispiel für die Bildung von Unterpositionsvarianten

Anschließend werden die Inhalte aller anderen Positionsvarianten negiert und mit einer UND Verknüpfung an die Coderegel der betrachteten Positionsvariante angefügt. Dieser Vorgang wird für sämtliche Positionsvarianten durchgeführt und sichert den gegenseitigen Ausschluss der Positionsvarianten. Bei dem Vorgehen werden ein paar Ausnahmen betrachtet, die im Folgenden aufgelistet werden. Dabei stellt A die ursprüngliche Positionsvariante dar und B die negativ anzufügende:

²⁹¹Eine Formel F der Aussagenlogik ist in disjunktiver Normalform, wenn sie eine Disjunktion von Konjunktionstermen ist. Ein Konjunktionsterm wird ausschließlich durch die konjunktive Verknüpfung von Literalen gebildet. Literale sind dabei nicht negierte oder negierte Variablen.

²⁹²**Teilkonjunktionen** sind eine Menge von Literalen, die UND, aber nicht ODER verknüpft sind.

- Die Literale, die in allen Teilkonjunktionen der betrachteten Positionsvariante A vorkommen, sowie alle negierten Literale werden im anzufügenden Teil B gestrichen.
- Eine Teilkonjunktion in B, die Obermenge von einer positiven Teilkonjunktion in A ist, wird nicht übernommen.
- Eine Teilkonjunktion in B, die Obermenge von einer negativen Teilkonjunktion in A ist, bewirkt ein gegenseitiges Aufheben und somit Entfernen der negativen Teilkonjunktion.
- Bei einer Unterpositionsvariante wird die dazugehörige Positionsvariante nicht betrachtet.

Übertragen auf das Beispiel ergeben sich die in Tabelle 5.14 dargestellten Regeln.

Positionsvariante	Coderegeln
1	$a \wedge b \vee a \wedge c \vee a \wedge b \wedge c \wedge \neg(e)$
1.1	$a \wedge \neg(b) \wedge \neg(c) \wedge \neg(e)$
2	$b \wedge \neg(a \vee a \wedge c \vee a \wedge c) \wedge \neg(a) \wedge \neg(c) \wedge \neg(e)$
3	$c \wedge \neg(a \wedge b \vee a \vee a \wedge b) \wedge \neg(a) \wedge \neg(b) \wedge \neg(e)$
4	$e \wedge \neg(a \wedge b \vee a \wedge c \vee a \wedge b \wedge c) \wedge \neg(a) \wedge \neg(b) \wedge \neg(c)$

Tabelle 5.14: Beispiel für die Bildung von sich gegenseitig ausschließenden Positionsvarianten

Die Coderegeln der Unterpositionsvarianten, die im ersten Schritt gebildet wurden, werden nun noch an die zugehörige Positionsvariante mit einem ODER verknüpft angehängt. Das Ergebnis, wiederum in DNF gebracht, ist in Tabelle 5.15 aufgezeigt. Damit ist die Erstellung der langen Coderegeln abgeschlossen.

Positionsvariante	Coderegeln
1	$(a \wedge b \vee a \wedge c \vee a \wedge b \wedge c) \wedge \neg e \vee a \wedge \neg b \wedge \neg c \wedge \neg e$
2	$b \wedge \neg a \wedge \neg c \wedge \neg e$
3	$c \wedge \neg a \wedge \neg b \wedge \neg e$
4	$e \wedge \neg a \wedge \neg b \wedge \neg c$

Tabelle 5.15: Beispiel für die Integration der Unterpositionsvarianten

5.2.2.3 Reihenfolge der Bearbeitung von Arbeitsvorgängen

Wie zuvor beschrieben wird mit einer existierenden Arbeitszuordnung begonnen. Diese Arbeitszuordnung beinhaltet primär nur die Zuordnung von Arbeitsvorgängen auf Arbeitsplätze durch die Variable $z_{avo,ap}^{Ist}$. Zusätzlich zu dieser Information wird noch die Reihenfolge der Abarbeitung der einzelnen Arbeitsvorgänge innerhalb einer Station benötigt. Mit Hilfe der Durchnummerierung der Stationen kann nun die interne Bearbeitungsreihenfolge über alle

Stationen hinweg entnommen werden, indem die einzelnen Bearbeitungsreihenfolgen der Stationen hintereinander gehängt werden. Ergebnis ist eine starre Bearbeitungsreihenfolge, wie in Grafik 5.15 beispielhaft aufgezeigt, welche streng genommen der Definition eines Vorranggraphen genügt und als Startpunkt dienen soll. Angewendet in einem Rekonfigurationsfall würde diese Startlösung keine einzige Verschiebung von Arbeitsvorgängen erlauben.

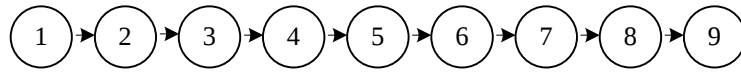


Abbildung 5.15: Starrer Vorranggraph als Startpunkt

5.2.3 Automatische Generierung von Freiheitsgraden

Im Folgenden werden in einem mehrstufigen Verfahren sukzessiv Freiheitsgrade im Vorranggraphen erzeugt, indem Vorrangbedingungen entfernt werden.

5.2.3.1 Strukturierung nach Werkstückansprache

Es wird davon ausgegangen, dass Arbeitsvorgänge an unterschiedlichen Werkstückansprachen²⁹³ $wa \in \mathcal{WA}$ keine Vorrangrestriktionen untereinander haben. Sie können parallel und somit unabhängig voneinander durchgeführt werden. Daher können alle Arbeitsvorgänge, die der Werkstückansprache $(a \in \mathcal{WA})$ zugewiesen sind, keine Arbeitsvorgänge beeinflussen, die $(b \in \mathcal{WA})$ zugeordnet sind. Die Arbeitsvorgänge werden nach ihrer zugewiesenen Werkstückansprache klassifiziert und für jede dieser Werkstückansprachen in einen Teilgraphen ausgelagert. Da zwischen diesen Teilgraphen keine Vorrangbeziehungen existieren können, werden sie im Graphen parallelisiert. Hierfür sind folgende Schritte notwendig:

1. Klassifizierung der Arbeitsvorgänge nach Werkstückansprache $wa \in \mathcal{WA}$
2. Hinzufügen von bisher transitiven Vorrangbedingungen zwischen Arbeitsvorgängen unterschiedlicher Werkstückansprachen
3. Entfernen sämtlicher Vorrangbedingungen, die zwischen Arbeitsvorgängen unterschiedlicher Werkstückansprachen existieren
4. Hinzufügen eines AND-Open und AND-Close Knotens, welche die Parallelität der Teilgraphen symbolisieren
5. Hinzufügen von Vorrangbedingungen zwischen dem AND-Open Knoten zu allen ersten Arbeitsvorgängen in jeder Werkstückansprache sowie von jedem letzten Arbeitsvorgang in jeder Werkstückansprache zum End-AND Knoten.

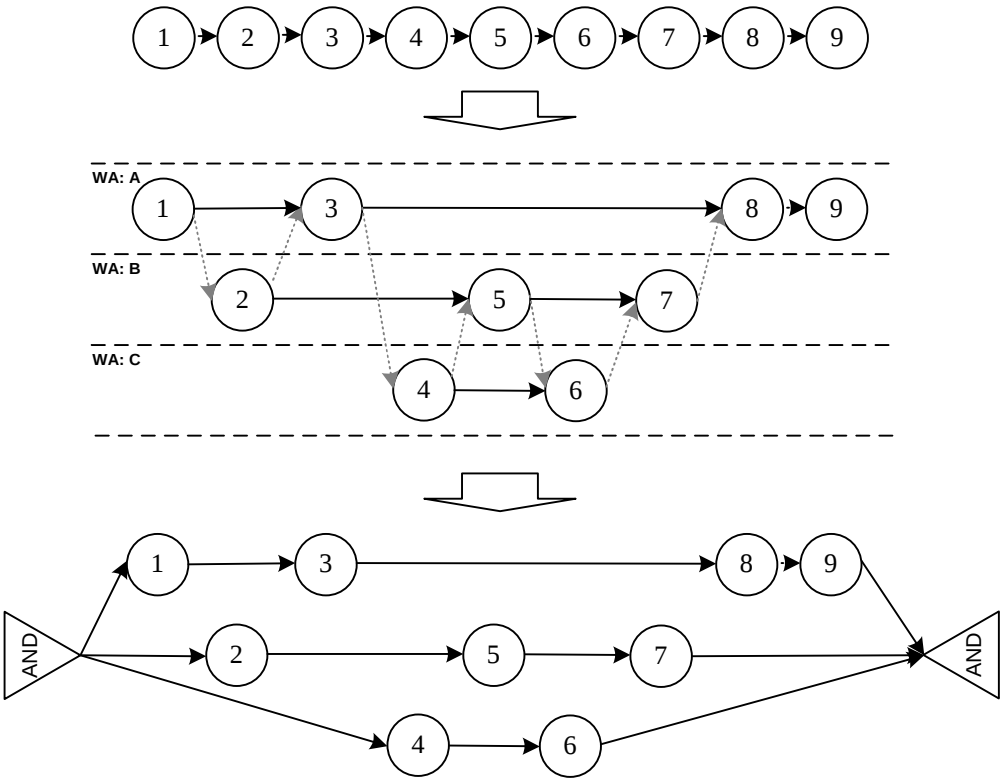


Abbildung 5.16: Beispiel für die Trennung in Teilgraphen mit Hilfe von unabhängigen Werkstückansprachen

In Abbildung 5.16 ist ein Beispiel für die Trennung dargestellt.

5.2.3.2 Freiheitsgrade auf Basis der Coderegel

Vorrangbeziehungen, die zwischen zwei Arbeitsvorgängen existieren, die nicht zusammen in einem Auftrag auftreten können, sind obsolet und können entfernt werden. Formal ist diese Bedingung in Gleichung (5.44) auf Basis der Notationserweiterung in Tabelle 5.16 definiert.

$$bp_{avo1,p} \neq bp_{avo2,p} \quad \forall p \in \mathcal{VAR} \tag{5.44}$$

Symbol	Definition
$\mathcal{VAR}$	Menge, die sämtliche bestellbaren Varianten beinhaltet

Tabelle 5.16: Erweiterung der Notation zur Definition von Freiheitsgraden auf Basis der Coderegel

²⁹³Vgl. Abschnitt 2.1.1

Der Identifikationsvorgang dieser obsoleten Vorrangbeziehungen wird anhand von zwei Wegen durchgeführt:

- Interpretationen von Baubarkeitsregeln
- Test der Coderegeln auf Ausschluss

5.2.3.2.1 Interpretation von Baubarkeitsregeln Baubarkeitsregeln definieren, welche Ausstattungsoptionen nicht zusammen mit anderen Ausstattungsoptionen auftreten können. In Tabelle 5.17 wird ein Beispiel aus der Automobilindustrie basierend auf den Ausstattungsoptionen aus Abschnitt 5.8 aufgegriffen, um den Sachverhalt zu verdeutlichen.

Code	Baubarkeitsregel
$M1$	$\neg M2 \vee \neg M3 \vee \neg M4$
$R1$	$\neg K2$
$M2$	$\neg K1$

Tabelle 5.17: Beispiele für Baubarkeitsregeln

Dabei bedeutet die erste Regel, dass ein Auftrag mit dem Motor $M1$ keinen weiteren Motor mit den Codes $M2$, $M3$, $M4$ beinhalten kann. Die zweite Regel besagt, dass das Standardradio mit dem Code $R1$ nicht zusammen mit der Klimaautomatik mit dem Code $K2$ verbaut werden kann. Die dritte Regel verhindert, dass es einen Auftrag gibt, bei dem ein V8-Ottomotor mit dem Code $M2$ und gleichzeitig die normale Klimaanlage $K1$ verbaut wird.

Mit Hilfe dieser Information können nun Coderegeln von Arbeitsvorgängen verglichen werden. Wenn in der Coderegel von Arbeitsvorgang A der nicht negierte Motorcode $M1$ auftritt, so sind sämtliche Arbeitsvorgänge, die einen nicht negierten Code $M2$, $M3$ oder $M4$ besitzen, von Arbeitsvorgang A unabhängig.

5.2.3.2.2 Test der Coderegeln auf Ausschluss Eine zweite Möglichkeit, sich ausschließende Arbeitsvorgänge zu identifizieren, ist der inhaltliche Vergleich der disjunktiven Normalformen der Coderegeln. Tritt ein Code bei Arbeitsvorgang A in allen Teilkonjunktionen positiv und bei Arbeitsvorgang B in allen Teilkonjunktionen negiert auf, so können die beiden Arbeitsvorgänge nicht gleichzeitig in einem Auftrag auftreten. Dieser Sachverhalt wird in dem Beispiel in 5.17 deutlich.

$$a \wedge b \leftrightarrow c \wedge \neg a$$

Abbildung 5.17: Beispiel für den gegenseitigen Ausschluss zweier Coderegeln

Bei der Generierung der langen Coderegeln werden, wie in Abschnitt 5.2.2.2 beschrieben, im letzten Schritt alle Coderegeln, die der gleichen Position zugeordnet sind, negiert hinter alle anderen Coderegeln einer Position gesetzt. Damit ist implizit gegeben, dass durch den Test der Coderegeln auf Ausschluss auch diese Arbeitsvorgänge erkannt werden.

5.2.3.2.3 Bildung von XOR-Trennungen Mit Hilfe der Information über sich ausschließende Arbeitsvorgänge werden im Folgenden sog. XOR-Trennungen²⁹⁴ gebildet. Hierfür werden XOR-Open und XOR-Close Knoten eingeführt, welche den Beginn und das Ende einer XOR-Trennung signalisieren.²⁹⁵ Arbeitsvorgänge, die sich innerhalb einer XOR-Trennung, aber in unterschiedlichen Zweigen befinden, sind unabhängig voneinander. Alle dazwischen existierenden Vorrangbedingungen können nicht gültig sein bzw. sind überflüssig und werden daher entfernt. Das Ergebnis wird in dem Beispiel in 5.18 unter Verwendung der Baubarkeitsregeln aus Tabelle 5.17 verdeutlicht. Die prinzipielle Vorgehensweise dafür wird durch Algorithmus 1 dargestellt.²⁹⁶

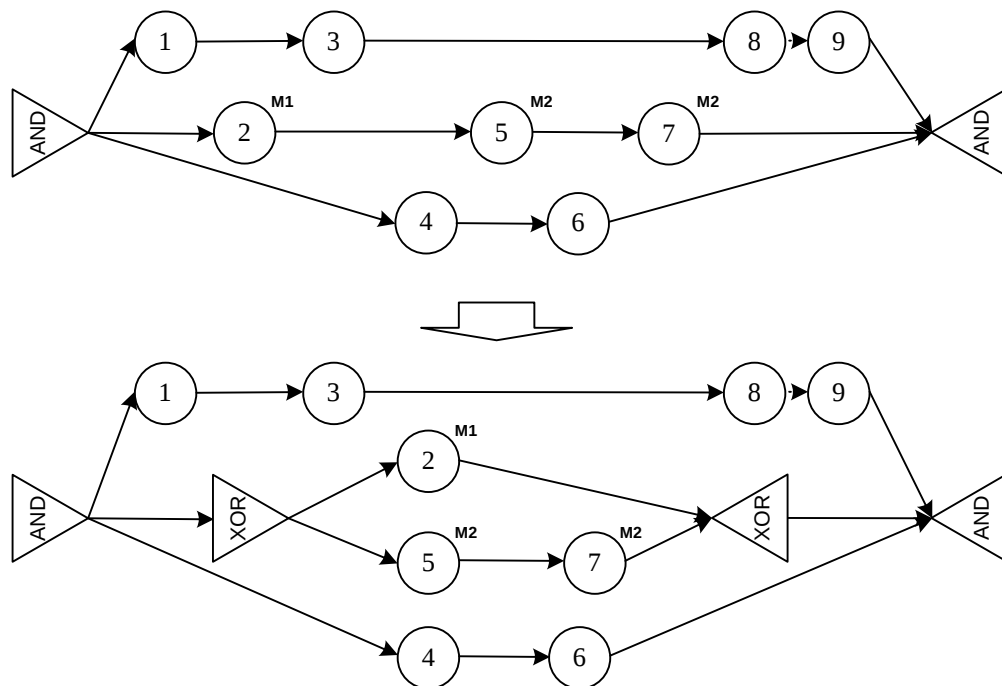


Abbildung 5.18: Vorgehen zur Trennung nach sich ausschließenden XOR-Zweigen

²⁹⁴Vgl. auch Boysen et al. in [BFS07a], welche den Vorranggraphen ebenfalls um AND/OR-Trennungen erweitern, um verschiedene Ausstattungsoptionen zu trennen und Wahrscheinlichkeiten bei der Erzeugung von Mischgraphen einfacher bilden zu können.

²⁹⁵Vgl. zur Einführung von XOR-Bereichen in Vorranggraphen auch [BFS07a], der diese Bereiche einführt um verschiedene Ausstattungsoptionen zu trennen und Wahrscheinlichkeiten bei der Erzeugung von Mischgraphen einfacher bilden zu können.

²⁹⁶Sonderfälle und weitere Einzelheiten werden in dem Algorithmus nicht betrachtet. Es wird die grundsätzliche Vorgehensweise dargestellt.

Zur Bildung der XOR-Trennungen wird die Arbeitsvorgangsliste je Werkstückansprache durchlaufen und jeweils jeder Arbeitsvorgang mit seinem Nachfolger verglichen (Algorithmus 1, Z.7). Der gegenseitige Ausschluss wird dabei auf zwei Wegen, wie in Abschnitt 5.2.3.2 beschrieben, erkannt.

Algorithmus 1 : Bildung von XOR-Trennungen

Data : AVOListe *al*

```

1 forall Elemente avo1 in al do
2   Set avo2 = Nachfolgeravo von avo1
3   Erzeuge XORTrennung Liste aller AVos die mit avo1 in einer Trennung sind
4   while (avo1 und avo2 schließen sich aus) oder (Codes avo1  $\supseteq$  avo2  $\neq$  „;1“) do
5     if not XORTrennung then
6       Bilde XORTrennung inkl. Kanten
7       Füge avo1 und avo2 zur Liste XORTrennung hinzu
8       if Code von avo2 ist Negation von Code avo1 then
9         Alle möglichen Pfade abgebildet durch avo1 und avo2
10        Lösche Kante: XOR-Open zu XOR-Close
11      Setze avo2 = den Nachfolger von avo2
12      forall i  $\in$  XORTrennung do
13        Prüfe Ausschluss von avo2 gegenüber i
14        if Ausschluss von avo2 gegenüber allen anderen i im XORTrennung then
15          Neuer Zweig für avo2 in Trennung einfügen
16      Liste für XOR-Trennung gefüllt oder keine XORTrennung
17      if XORTrennung then
18        Knoten und Kanten im XORTrennung ordnen
19        XORTrennung-ordnen(AVOListe XORTrennung)
  
```

Wenn zwei Arbeitsvorgänge gefunden wurden, die sich ausschließen, so wird eine XOR-Trennung mit den beiden Hilfsknoten XOR-Open und XOR-Close gebildet. Innerhalb der Schleife (Algorithmus 1, Z.7-Z.21) wird geprüft, ob weitere Arbeitsvorgänge in diese Trennung übernommen werden können. Nur wenn die beiden Arbeitsvorgänge alle möglichen Wege durch die XOR-Trennung darstellen, d.h. die eine Coderegeln die Negation der anderen ist, kann die direkte Kante vom XOR-Open zum XOR-Close entfernt werden, da einer der beiden Arbeitsvorgänge auf jeden Fall benötigt wird (Algorithmus 1, Z.12-Z.15). Dann wird der nächste zu überprüfende Arbeitsvorgang ausgewählt (Algorithmus 1, Z.16). Hierfür muss für die nächsten Arbeitsvorgänge eine der folgenden Bedingungen gelten:

- Der Code des neuen Arbeitsvorganges stimmt mit dem Code eines Arbeitsvorganges in der XOR-Trennung überein $\rightarrow$ Der Arbeitsvorgang kann mit in die XOR-Trennung unter dem anderen Arbeitsvorgang platziert werden.

- Der Code des neuen Arbeitsvorganges ist eine Untermenge von einem Code eines Arbeitsvorganges innerhalb der XOR-Trennung. -> Der Arbeitsvorgang kann mit in die XOR-Trennung unter dem anderen Arbeitsvorgang platziert werden.
- Es kann ein Ausschluss von dem neuen Arbeitsvorgang mit allen anderen Arbeitsvorgängen erkannt werden. -> Der Arbeitsvorgang kann als weiterer Zweig in der XOR-Trennung platziert werden.

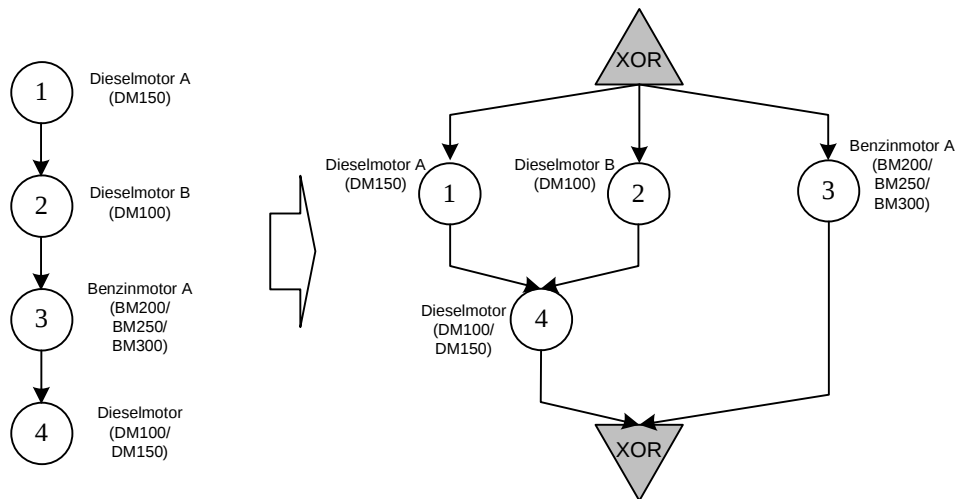


Abbildung 5.19: Beispiel für die Trennung in sich ausschließende XOR-Zweige

Dabei darf es sich bei dem betrachteten Arbeitsvorgang nicht um einen durch ein Semikolon gekennzeichneten Standardarbeitsvorgang handeln. Wenn die XOR-Trennung abgeschlossen ist, stellt ein weiterer Algorithmus sicher, dass die korrekten Kanten zwischen den Arbeitsvorgängen innerhalb einer Trennung gesetzt werden. Hierfür werden Arbeitsvorgänge, die die gleichen Coderegeln haben, untereinander gehängt (Algorithmus 2, Z.3-12) und überflüssige Kanten der Vorvorgänger gelöscht (Algorithmus 2, Z.14-20). Eine erfolgreiche Anwendung der beiden Algorithmen ist in dem Beispiel 5.19 dargestellt.

5.2.3.3 Integration unterschiedlicher gültiger Zuteilungen

Je länger eine Fließlinie existiert, desto mehr gültige Zuteilungen und somit Ablaufreihenfolgen von Arbeitsvorgängen wurden bereits erstellt. Durch jede Rekonfiguration wird eine weitere gültige Zuordnung generiert. Diese Daten können genutzt werden, um weitere Freiheitsgrade im Vorranggraphen zu identifizieren.

Hierfür sind zwei Schritte notwendig:

Algorithmus 2 : XOR-Trennung ordnen

```

1 AVOs mit gleichen Codes in Reihe bringen im XOR-Trennung
2 forall  $i \in \text{XORTrennung}$  do
3   forall  $j \in \text{XORTrennung}/1 - i$  do
4     if Code des AVO  $i$  wird vollständig durch Code von AVO  $j$  erfüllt then
5       Bilde Kante von  $i$  nach  $j$ 
6 Überflüssige Kanten zu AVOs löschen
7 forall  $k \in \text{XORTrennung}$  beginnend bei letztem AVO do
8   Ermittle alle Vorgängeravo  $VG$  von AVO  $k$ 
9   if  $TK$  von AVO  $k$  werden durch  $TK$  aus AVOS in  $VG$  abgedeckt then
10    Lösche Kante von allen Knoten vor den AVOs  $VG$  zu AVO  $k$ 
11    Füge Kante von allen  $VG$  zu AVO  $k$  hinzu

```

- Vereinigung der Graphen der unterschiedlichen Zuteilungen
- Zyklenbeseitigung und Identifikation von Freiheitsgraden

5.2.3.3.1 Vereinigung der Graphen Für jede Zuteilung werden die Methoden wie oben beschrieben durchgeführt. Darauf folgend werden die beiden entstehenden Graphen vereinigt. Formal wird diese Vereinigung von zwei Graphen in Gleichung 5.45 beschrieben.

$$\mathcal{G}_1 \cup \mathcal{G}_2 = (\mathcal{AVO}_1 \cup \mathcal{AVO}_2, \mathcal{V}_1 \cup \mathcal{V}_2) \quad | \quad \mathcal{G}_1 = (\mathcal{AVO}_1, \mathcal{V}_1) \wedge \mathcal{G}_2 = (\mathcal{AVO}_2, \mathcal{V}_2) \quad (5.45)$$

Werden gerichtete Graphen vereinigt, können Zyklen entstehen. Diese Problematik wird im vereinfachten Beispiel 5.20 dargestellt.

5.2.3.3.2 Zyklenbeseitigung und Identifikation von Freiheitsgraden Ein Zyklus lässt einen Vorranggraphen ungültig werden, da es zu einer Verklemmung zwischen den im Zyklus enthaltenen Arbeitsvorgängen kommt.²⁹⁷ Im ersten Schritt müssen die Zyklen identifiziert werden, um sie im zweiten Schritt auflösen zu können. Die Identifikation von Zyklen kann mit Hilfe einer Tiefensuche durch den Graphen realisiert werden.²⁹⁸

²⁹⁷ Als Verklemmung bezeichnet man einen "Zustand eines nichtsequentiellen Prozesses, in dem sequentielle Teilprozesse (z.B. einzelne Fäden (threads)) derart blockiert sind, dass diese Blockade nie wieder aufgelöst werden kann, unabhängig davon, wie die noch aktiven Teile des Prozesses sich verhalten.

²⁹⁸ Vgl. im Gegensatz dazu die Vorgehensweisen zur Zyklenauflösung bei der Bildung von Mischgraphen, z.B. in [Bra95], wo die beteiligten Knoten dupliziert werden. Dies ist in diesem Zusammenhang nicht gewünscht.

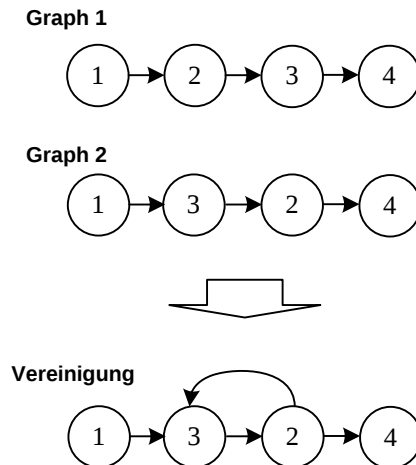


Abbildung 5.20: Entstehung von Zyklen in vereinigten Graphen

Ein Zyklus entsteht genau dann, wenn in einem Vorranggraphen verschiedene Ablaufreihenfolgen von gleichen Arbeitsvorgängen zusammengefügt wurden. Diese unterschiedlichen Ablaufreihenfolgen sind alle gültig. Abweichungen zwischen verschiedenen Ablaufreihenfolgen stellen Freiheitsgrade für die Verschiebung von Arbeitsvorgängen dar. Diese Freiheitsgrade werden, wie auch schon bei der Trennung nach Werkstückansprachen, mit Hilfe von AND-Open und AND-Close Knoten eingeführt, da jeder Zweig dieser Aufsplittung durchlaufen werden muss, dies jedoch unabhängig voneinander geschehen kann und somit die Reihenfolge frei wählbar ist. In dem Beispiel wurden, wie in Grafik 5.20 gezeigt, zwei verschiedene Ablaufreihenfolgen aus zwei Arbeitszuordnungen vereinigt. Dabei ist ein Zyklus mit den Knoten 2 und 3 aufgetreten. Dies bedeutet, dass es möglich ist, Arbeitsvorgang 3 sowohl hinter als auch vor Arbeitsvorgang 2 durchzuführen. Die Beseitigung dieses einfachen Zyklus ist im Beispiel 5.21 dargestellt.

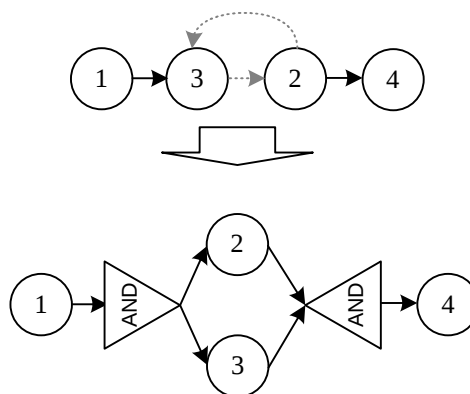


Abbildung 5.21: Vorgehen zur Beseitigung von Zyklen

Algorithmus 3 : Zyklensuche im Vorranggraphen**Data** : Knoten i , Knoten j , Pfad, Zyklen

```

1 if markiert( $j$ , "A") then
2   /*Zyklus gefunden*/
3   markiere Kante von  $i$  zu  $j$  als: "Rückkante"
4   Füge Pfad ab Knoten  $j$  als neuen Zyklus in Zyklen hinzu
5 else
6   /*Noch keinen Zyklus gefunden*/
7   markiere( $j$ , "A")
8   Füge  $j$  in Pfad ein
9   forall Nachfolger  $k$  von  $j$  do
10    markiere Kante von  $j$  zu  $k$  als "Vorwärtskante"
11    ZyklenSuche( $j, k, Pfad, Zyklen$ )
12   markiere( $j$ , "B")
13   lösche  $j$  aus Pfad

```

Es kann vorkommen, dass sich Zyklen überschneiden. Das heißt, dass die Schnittmenge der Knoten aus zwei Zyklen, wie in Gleichung 5.46 definiert, nicht leer ist. Ein Beispielfall ist in Abbildung 5.22 dargestellt. Hier wurde die Ablaufreihenfolge {12435} mit der Reihenfolge {13245} vereinigt. Dabei sind die Knoten 2, 3, 4 von mindestens einem Zyklus betroffen.

$$ZyklusA \cap ZyklusB \neq \{\}$$
 (5.46)

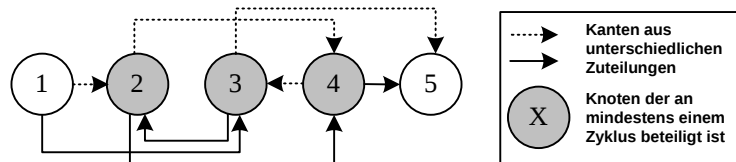


Abbildung 5.22: Beispiel für Zyklen mit Schnittmenge

Mehrere Zyklen, die sich auf die gleiche Knotenmenge beziehen, müssen zusammen aufgelöst werden. Für die Auflösung werden im ersten Schritt zusätzlich zu den expliziten alle impliziten Kanten identifiziert und in einer Matrix, wie in Tabelle 5.18, zusammengefasst.

Eine 1 bedeutet, dass es eine explizite oder implizite Vorrangbedingung zwischen dem Startknoten und Zielknoten gibt. Im nächsten Schritt können alle widersprüchlichen Bedingungen, also die, bei denen es eine Bedingung vom Startknoten zum Zielknoten sowie andersherum gibt, gelöscht werden. Das Ergebnis ist in Tabelle 5.19 abgebildet.

Die übrigen Vorrangbedingungen behalten ihre Gültigkeit. In der Beispielmatrix sind die Knoten 2 und 3 sowie 4 und 3 unabhängig voneinander, während 2 noch eine Beziehung zu

	Startknoten				
	1	2	3	4	5
1	0	0	0	0	0
2	1	0	1	0	0
3	1	1	0	1	0
4	1	1	1	0	0
5	1	1	1	1	0

Tabelle 5.18: Beispiel für eine Bedingungsmatrix

	Startknoten				
	1	2	3	4	5
1	0	0	0	0	0
2	1	0	0	0	0
3	1	0	0	0	0
4	1	1	0	0	0
5	1	1	1	1	0

Tabelle 5.19: Beispiel für eine aufgelöste Bedingungsmatrix

4 hat. Daraus folgt, dass der Arbeitsvorgang 3 unabhängig von den beiden Knoten 2 und 4 durchgeführt werden kann und daher innerhalb einer AND-Trennung, wie in Grafik 5.23 dargestellt, parallelisiert werden kann.

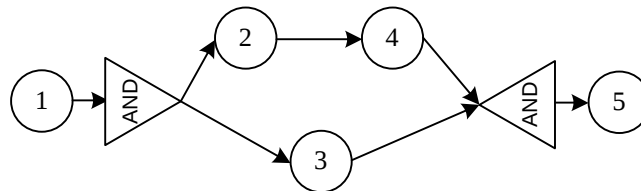


Abbildung 5.23: Beispiel für aufgelöste Zyklen mit Schnittmenge

5.2.3.3.3 Verwendung als selbst lernendes System Das hier vorgestellte Konzept zur Erstellung eines Vorranggraphen soll insbesondere im Kontext einer Rekonfiguration und der Integration eines Optimierungsansatzes²⁹⁹ dazu führen, dass der automatische Tauschalalgorithmus keine technisch ungültigen Lösungen generiert. Im Zuge einer regelmäßigen Rekonfiguration wird in jedem Planungszyklus eine neue gültige Abstimmungslösung gefunden. Wie in Abschnitt 5.2.3.3 beschrieben kann so nach jedem Planungszeitraum die neue Bearbeitungsreihenfolge der Arbeitsvorgänge mit dem Vorranggraphen verglichen werden, um weitere Freiheitsgrade zu identifizieren. Hierdurch wird ein selbst lernendes System realisiert,

²⁹⁹Vgl. dazu Abschnitt 5.1.3.2.3

welches sukzessiv weitere Freiheitsgrade erkennt, ohne Zusatzaufwand für den Montageplaner zu generieren.

5.2.4 Ergänzende manuelle Feinstrukturierung des Graphen

Nach der Durchführung aller vorgestellten Maßnahmen, wurde vollständig automatisch ein Vorranggraph generiert, welcher XOR und AND-Trennungen beinhalten. Sämtliche Freiheitsgrade im Graphen sind valide, so dass die Verwendung der Vorrangbedingungen im Rahmen einer Rekonfiguration immer zu gültigen Lösungen führt. In einer anschließenden manuellen Feinstrukturierung kann mit Hilfe des Wissens eines Montageplaners eine Feinstrukturierung von Teilbereichen vorgenommen werden. Hierdurch können weitere Freiheitsgrade im Graphen eingebaut werden. Die erweiterten Informationen durch die XOR und AND-Trennungen helfen dabei, die Komplexität zu reduzieren. Folgende Definition gilt für die Trennungen und wird verwendet:

Arbeitsvorgänge, die sich in verschiedenen Strängen einer AND-Trennung oder XOR-Trennung befinden, können keine Vorrangbeziehungen untereinander besitzen.

Durch diese Information kann die Komplexität für die Betrachtung durch einen Montageplaner deutlich reduziert werden. Die Teilbereiche der AND-Trennungen können unabhängig voneinander bearbeitet werden. Zusätzlich wird dem Planer bei der Feinstrukturierung durch die XOR-Trennungen aufgezeigt, zwischen welchen alternativen Teilbereichen keine Vorrangbeziehungen mehr betrachtet werden müssen. Zwei verschiedene Vorgehensweisen können für die manuelle Feinstrukturierung verfolgt werden:

1. Es wird eine manuelle Strukturierung jedes einzelnen “AND” Bereiches vorgenommen. Prinzipiell wird dabei je AND-Zweig ein einzelner Vorranggraph aufgebaut. Hierfür können die im Stand der Forschung beschriebenen manuellen Befragungsmethoden verwendet werden.³⁰⁰ Zusätzlich zu den von Bourjault et al.³⁰¹ eingeführten Methoden, um implizite Informationen aus bereits beantworteten Fragen wiederzuverwenden, kann die Fragenanzahl durch die XOR-Trennungen weiter reduziert werden. Es müssen keine Fragen mehr bezüglich Vorrangbeziehungen zwischen zwei Arbeitsvorgängen, die in zwei unterschiedlichen XOR-Zweigen liegen, gestellt werden. Problematik dieses Vorgehens ist, dass die existierenden Reihenfolgeinformationen innerhalb von AND-Zweigen verloren gehen, da jede Beziehung zwischen den Arbeitsvorgängen erneut in Frage gestellt wird. Hierbei können Freiheitsgrade generiert werden, die unter Umständen ungültig

³⁰⁰Vgl. Abschnitt 3.2

³⁰¹Vgl. [Bou84]

sind. Vorteil ist jedoch, dass so die maximal mögliche Menge an Freiheitsgraden generiert werden kann.

2. Die zweite Möglichkeit ist, die manuelle Feinstrukturierung an das Vorgehen der automatischen Generierung des Vorranggraphen anzulehnen. Dafür wird im ersten Schritt der Graph so verwendet, wie er nach der automatischen Strukturierung aufgebaut ist. Um zu vermeiden, dass in einer anschließenden manuellen Bearbeitung Freiheitsgrade erkannt werden, welche ungültig sind, kann eine alternative Vorgehensweise zur manuellen Bearbeitung durchgeführt werden. Statt wie üblich Vorrangbedingungen zu erfragen, werden existierende Vorrangbedingungen in Frage gestellt: “Ist die Vorrangbedingung zwischen *avo1* und *avo2* notwendig?” Wenn diese Frage verneint wird, kann eine “AND-Aufsplittung” zwischen den betroffenen Arbeitsvorgängen erfolgen. So bleiben die impliziten Bedingungen zu den weiteren Arbeitsvorgängen erhalten. Auf diese Art und Weise können sämtliche Vorrangbeziehungen im Vorranggraphen durchlaufen werden. Um alle Freiheitsgrade zu erkennen, muss dieses Vorgehen sehr häufig wiederholt werden. Entscheidender Vorteil ist jedoch, dass nach jeder Iteration gewährleistet ist, dass lediglich gültige Freiheitsgrade im Graphen existieren. So können bei komplexen Produkten auch verschiedene Teile des Graphen durch unterschiedliche Planer bearbeitet werden. Dies hat den Vorteil, dass keine Person benötigt wird, die den gesamten Produktionsprozess im Detail kennt. Besonders bei komplexen Produkten, wie z.B. in der Automobilindustrie, ist es in der Praxis üblich, dass die einzelnen Montageplaner nur einen Teilbereich des Produktionsprozesses beherrschen.

5.3 Belastungsorientierte Reihenfolgeplanung zur Reduzierung des Unterstützereinsatzes

5.3.1 Formalisierung der Reihenfolgeplanung

Im Folgenden wird das Problem der Reihenfolgeplanung auf Basis der Notation in (5.20) als Optimierungsmodell mathematisch formuliert.

Symbol	Definition
Mengen	
$\mathcal{A}$	Menge von Aufträgen a
$\mathcal{AP}$	Menge von Arbeitsplätzen ap
$\mathcal{AVO}$	Menge von Arbeitsvorgängen avo
BWP_{pz}	Menge von Werkstücken $p \in \mathcal{P}$, die zur Bewertung des Produktionszeitraumes $pz \in \mathcal{PZ}$ betrachtet werden müssen
$BWR\mathcal{F}_{pz}$	Menge von Reihenfolgepositionen $idx \in \mathcal{RF}$, die zur Bewertung des Produktionszeitraumes pz betrachtet werden müssen
$\mathcal{ST}$	Menge von Stationen st
$\mathcal{P}$	Menge von Werkstücken p
$\mathcal{P}_{pz}$	Menge von Werkstücken, die im Produktionszeitraum $pz \in \mathcal{PZ}$ produziert werden
$\mathcal{PZ}$	Menge von Planungszeiträumen pz
$\mathcal{RF}$	Menge von Reihenfolgepositionen idx
$\mathcal{RF}_{pz}$	Menge von Reihenfolgepositionen $idx \in \mathcal{RF}$ im Produktionszeitraum $pz \in \mathcal{PZ}$
$\mathcal{VG}_{ap}$	Menge von Vorgängerarbeitsplätzen $vg \in \mathcal{AP}$, von denen Arbeitsplatz $ap \in \mathcal{AP}$ abhängt
Parameter	
$bp_{a,avo}$	Binär: 1, wenn der Auftrag $a \in \mathcal{A}$ den Arbeitsvorgang $avo \in \mathcal{AVO}$ benötigt; 0, sonst (Bauplan)
dbl_{ap}	Größe des vorgelagerten „linken“ Driftbereiches des Arbeitsplatzes $ap \in \mathcal{AP}$ in Zeiteinheiten
dbr_{ap}	Größe des nachgelagerten „rechten“ Driftbereiches des Arbeitsplatzes $ap \in \mathcal{AP}$ in Zeiteinheiten
$\mathcal{K}^U$	Kostensatz für einen Unterstützer
$\mathcal{K}^N$	Kostensatz für Nacharbeit
pz	Produktionszeitraum $pz \in \mathcal{PZ}$
st_{ap}	Station $st \in \mathcal{ST}$, welcher der Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist

Symbol	Definition
sp_{ap}	Ansprechzone $sp \in \mathcal{SP}$, welcher der Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist
tz	Taktzeit der Fließlinie in Zeiteinheiten
t	Takt $t \in \mathcal{T}$
τ_{avo}	Bearbeitungszeit des Arbeitsvorganges $avo \in \mathcal{AVO}$ in Zeiteinheiten
w_{ap}	Anzahl an Werkern, die Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen sind
$z_{avo,ap}^{Ist}$	Binär: 1, wenn Arbeitsvorgang $avo \in \mathcal{AVO}$ dem Arbeitsplatz $ap \in \mathcal{AP}$ zugewiesen ist; 0, sonst (existierende Zuordnungslösung)
Variablen	
$\tau_{idx,ap}$	Bearbeitungszeit des Werkstückes $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ an Arbeitsplatz $ap \in \mathcal{AP}$ in Zeiteinheiten
$h_{a,p}$	Binär: 1, wenn Auftrag $a \in \mathcal{A}$ zu Werkstück $p \in \mathcal{P}$ zugeordnet ist; 0, sonst
$idx_{t,ap}$	Reihenfolgeposition des Werkstückes, welches sich im Takt $t \in \mathcal{T}$ an Arbeitsplatz $ap \in \mathcal{AP}$ befindet
$\mathcal{K}^{Ges}$	Gesamtkosten
$pos_{idx,ap}$	Endposition des Werkers an Arbeitsplatz $ap \in \mathcal{AP}$ nach Bearbeitung des Werkstückes $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ relativ zur linken Stationsgrenze von Arbeitsplatz $ap \in \mathcal{AP}$ in Zeiteinheiten
$u_{t,ap}^{AnzAP}$	Anzahl gleichzeitiger Unterstützereinsätze an Arbeitsplatz $ap \in \mathcal{AP}$ in Takt $t \in \mathcal{T}$
$u_{t,st}^{AnzST}$	Anzahl gleichzeitiger Unterstützereinsätze an Station $st \in \mathcal{ST}$ in Takt $t \in \mathcal{T}$
$u_{t,idx,ap}^{Belegt}$	Binär: 1, wenn ein Unterstützereinsatz, der an dem Werkstück $p \in \mathcal{P}$ mit der Reihenfolgeposition $idx \in \mathcal{RF}$ an Arbeitsplatz $ap \in \mathcal{AP}$ durchgeführt wird, in Takt $t \in \mathcal{T}$ noch nicht abgeschlossen ist; 0, sonst
u^E	Gesamtanzahl Unterstützereinsätze
$u_{idx,ap}^{Einsatz}$	Binär: 1, wenn ein Unterstützereinsatz an dem Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition idx an Arbeitsplatz $ap \in \mathcal{AP}$ durchgeführt wird; 0,sonst
$u_{t,idx,ap}^{Ende}$	Binär: 1, wenn ein Unterstützereinsatz, der an dem Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ an Arbeitsplatz $ap \in \mathcal{AP}$ durchgeführt wird, in oder nach Takt $t \in \mathcal{T}$ abgeschlossen wird; 0, sonst
u^{MAX}	Maximale Anzahl gleichzeitiger Unterstützereinsätze $\hat{=}$ Notwendige Anzahl an Unterstützereinsatzkräften
$u_{t,idx,ap}^{Start}$	Binär: 1, wenn ein Unterstützereinsatz, der an dem Werkstück an Reihenfolgeposition $idx \in \mathcal{RF}$ an Arbeitsplatz $ap \in \mathcal{AP}$ durchgeführt wird, in oder vor Takt $t \in \mathcal{T}$ angefangen hat; 0, sonst
u_t	Anzahl gleichzeitiger Unterstützereinsätze in Takt $t \in \mathcal{T}$
$u_{idx,ap}^{Zeit}$	Einsatzzeit für den Unterstützereinsatz, der an dem Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ an Arbeitsplatz $ap \in \mathcal{AP}$ durchgeführt wird

Symbol	Definition
$t_{idx,ap}$	Taktindex des Taktes, in dem sich das Werkstück $p \in \mathcal{P}$ mit der Reihenfolgeposition $idx \in \mathcal{RF}$ in der Station $st_{ap} \in \mathcal{ST}$ von Arbeitsplatz $ap \in \mathcal{AP}$ befindet
$vu_{idx,ap}$	Vorgängerüberhang, welcher die Startposition des Werkers an Arbeitsplatz $ap \in \mathcal{AP}$ bei der Bearbeitung des Werkstückes $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ beeinflusst.
$y_{p,idx}^{\mathcal{RF}}$	Binär: 1, wenn Werkstück $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ produziert wird; 0, sonst (Reihenfolgeplanung)
$wv_{idx,ap}$	Werkerverzug des Werkers an Arbeitsplatz $ap \in \mathcal{AP}$ vor der Bearbeitung des Werkstückes $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$
$wz_{vg,ap,idx}$	Wartezeit durch Vorgängerarbeitsplatz $vg \in \mathcal{VG}_{ap}$, welche die Startposition des Werkers an Arbeitsplatz $ap \in \mathcal{AP}$ bei der Bearbeitung des Werkstückes $p \in \mathcal{P}$ an Reihenfolgeposition $idx \in \mathcal{RF}$ beeinflusst.

Tabelle 5.20: Notation zur Formalisierung der Reihenfolgeplanung

Ziel der Reihenfolgeplanung ist es, die Werkstücke eines Planungszeitraumes so in eine Reihenfolge zu bringen, dass der Produktionsprozess möglichst effizient verläuft.³⁰² Zum Zeitpunkt der Reihenfolgeplanung steht die im Fließsystem zur Verfügung stehende Anzahl an Werkern bereits fest. Ebenfalls sind durch die vorhergehende Rekonfiguration die Arbeitsvorgänge bereits jeweils einem Arbeitsplatz zugeordnet. Einfluss kann nur noch auf die notwendigen Unterstützereinsätze ausgeübt werden, die durch Überlastung hervorgerufen werden.³⁰³ Hierbei ist insbesondere die Anzahl notwendiger vorzuhaltender Unterstützer von Interesse, die durch die maximale Anzahl an gleichzeitigen Unterstützereinsätzen definiert ist.³⁰⁴ Zusätzlich gilt es, die Anzahl Unterstützereinsätze zu minimieren, da hierdurch Nacharbeitskosten entstehen. Die Zielfunktion wird in Gleichung (5.47) definiert.

$$\text{Min } \mathcal{K}^{Ges} = \mathcal{K}^U \cdot u^{MAX} + \mathcal{K}^N \cdot u^E \quad (5.47)$$

Das Ergebnis der Reihenfolgeplanung wird mit Hilfe einer binären Variable modelliert, durch die definiert ist, ob ein bestimmtes Werkstück p einer bestimmten Reihenfolgeposition idx zugewiesen ist oder nicht. Gleichungen 5.48 und 5.49 stellen sicher, dass jedes Werkstück nur

³⁰²Vgl. Abschnitt 2.2.3.1

³⁰³Vgl. Abschnitt 2.1.2.2

³⁰⁴Vgl. Abschnitt 5.1.1.1

einer Reihenfolgeposition zugewiesen sowie jede Reihenfolgeposition nur durch ein Werkstück belegt wird.

$$\sum_{p \in \mathcal{P}} y_{p,idx}^{\mathcal{RF}} = 1 \quad \forall \quad idx \in \mathcal{RF}_{pz} \quad (5.48)$$

$$\sum_{idx \in \mathcal{RF}} y_{p,idx}^{\mathcal{RF}} = 1 \quad \forall \quad p \in \mathcal{P}_{pz} \quad (5.49)$$

Alle weiteren Bedingungen hängen von diesen Reihenfolgepositionen ab. Daher wird nachfolgend nicht mehr direkt auf die Werkstücke p selbst, sondern immer auf die Reihenfolgepositionen idx verwiesen.

Die Bearbeitungszeit $\tau_{idx,ap}$ des Werkstückes an der Reihenfolgeposition idx an Arbeitsplatz ap ist definiert in Gleichung (5.50).³⁰⁵ Dabei bildet die zuvor durchgeführte Rekonfiguration³⁰⁶ des Fließsystems mit der Zuweisung der Arbeitsvorgänge zu den Arbeitsplätzen $z_{avo,ap}^{Ist}$ den Ausgangspunkt für die Berechnung.

$$\tau_{idx,ap} = \sum_{p \in \mathcal{P}} \sum_{avo \in \mathcal{AVO}} (\tau_{avo} \cdot z_{avo,ap}^{Ist} \cdot bp_{a,avo}) \cdot y_{p,idx}^{\mathcal{RF}} \quad \forall ap \in \mathcal{AP}, p \in \mathcal{P}, a \in \mathcal{A} : h_{a,p} = 1 \quad (5.50)$$

5.3.1.1 Reihenfolgeabhängige Positionsbestimmung der Werker

Die Positionsvariable $pos_{idx,ap}$ definiert die Position eines Werkers an Arbeitsplatz ap nach Bearbeitung des Werkstückes an Reihenfolgeposition idx , relativ zur linken Stationsgrenze. Zusätzlich zu der Bearbeitungszeit eines Werkstückes $\tau_{idx,ap}$ muss eine unter Umständen verspätete Startposition des Werkers berücksichtigt werden. Diese Verspätung kann durch eine vorangegangene Überlastung des Werkers selbst (Werkerverzug) oder durch eine Verspätung eines Vorgängerarbeitsplatzes auftreten (Vorgängerüberhang). Auswirkungen hat dabei nur das Maximum aus diesen beiden Werten (5.51).

³⁰⁵Vgl. Erläuterungen in Abschnitt 2.1.2.1

³⁰⁶Vgl. zur Problemdefinition Abschnitt 2.2.1 und für die Methode Abschnitt 5.1

$$pos_{idx,ap} = \tau_{idx,ap} + \max \left\{ \begin{array}{l} \text{Eigener Verzug} \\ \underbrace{wv_{idx,ap}} \\ \underbrace{vu_{idx,ap}} \\ \text{Vorgängerüberhang} \end{array} \right\} \quad (5.51)$$

Startposition
Endposition

Der eigene Werkerverzug $wv_{idx,ap}$ wird definiert durch die Endposition nach dem letzten bearbeiteten Werkstück abzüglich des Abstandes zum jetzt zu bearbeitenden Werkstück. Falls einem Arbeitsplatz mehr als ein Werker zugewiesen ist, so entspricht das letzte bearbeitete Werkstück nicht dem aus dem letzten Takt. Daher muss, je nach Anzahl der zugewiesenen Werker w_{ap} , das letzte Werkstück durch $idx - w_{ap}$ ermittelt werden. Der Abstand zum letzten Werkstück entspricht der Taktzeit multipliziert mit der Anzahl eingesetzter Werker am Arbeitsplatz, da bei einem Mehrtakter nur jedes w_{ap} -te Werkstück von dem gleichen Werker übernommen wird. Der Werkerverzug kann dabei nie kleiner als die linke Driftgrenze werden³⁰⁷, daher wird aus diesen beiden Werten das Maximum bestimmt.

$$wv_{idx,ap} = \max \left\{ \begin{array}{l} pos_{idx-w_{ap},ap} - (tz \cdot w_{ap}) \\ -dbl_{ap} \end{array} \right\} \quad (5.52)$$

5.3.1.2 Modellierung von Abhängigkeiten zwischen Werkern

Der Vorgängerüberhang $vu_{idx,ap}$ gibt an, wie weit die Vorgängerarbeitsplätze, von denen die Arbeit des Werkers an Arbeitsplatz ap abhängt, in die Station eben dieses Arbeitsplatzes ap hineinarbeiten. Zur Definition dieser Abhängigkeiten existiert die Menge $\mathcal{VG}_{ap} \subset \mathcal{AP}$ an Arbeitsplätzen, die von Arbeitsplatz ap nicht überholt werden dürfen.³⁰⁸

Die Startposition des Werkers wird durch den Maximalwert aus den Wartezeiten wz , die sich durch die verschiedenen Vorgängerarbeitsplätze ergeben (5.53), bestimmt. Außerdem darf der Werker nicht weiter zurück, als es ihm die linke Driftgrenze erlaubt.

$$vu_{idx,ap} = \max[-dbl_{ap}; \max_{vg \in \mathcal{VG}_{ap}} \{wz_{vg,ap,idx}\}] \quad (5.53)$$

Zur Berechnung der Wartezeit $wz_{vg,ap,idx}$, die durch Vorgängerarbeitsplatz vg an Arbeitsplatz ap verursacht wird, müssen drei Fälle unterschieden werden. Der Vorgängerarbeitsplatz kann

³⁰⁷Ein weiteres Vorarbeiten als die Größe des linken Driftbereiches wird nicht erlaubt. Vgl. dazu auch die Systembeschreibung in Abschnitt 2.1

³⁰⁸Vgl. hierzu die Definitionen in Abschnitt 2.1.2.2.1

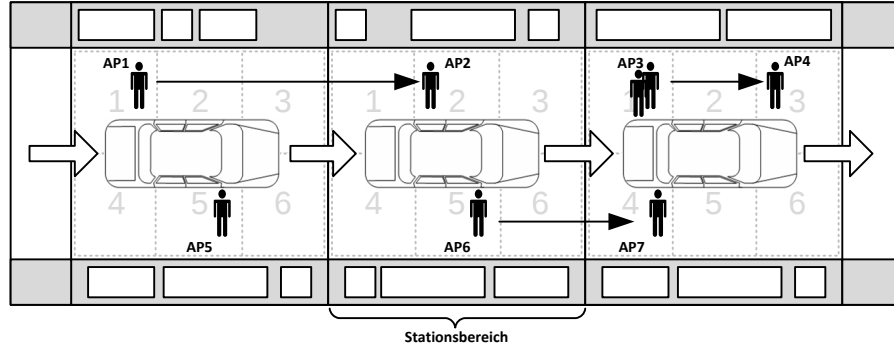


Abbildung 5.24: Beispiel für Werkerabhängigkeiten

sich in einer vorgelagerten Station $st_{vg} < st_{ap}$ oder aber innerhalb der eigenen Station $st_{vg} = st_{ap}$ befinden.³⁰⁹ Handelt es sich um eine vorgelagerte Station, so muss noch unterschieden werden, an welcher Ansprechzone sp_{vg} der Vorgängerarbeitsplatz zugewiesen ist. Die formale Beschreibung findet sich in Gleichung (5.54). Jeder Fall wird mit Hilfe der Arbeitsplätze im Beispiel 5.24 verdeutlicht.

$$wz_{vg,ap,idx} = \begin{cases} pos_{idx,vg} - \tau_{idx,ap} - \overbrace{(st_{ap} - st_{vg}) \cdot tz}^{\text{Entf. zu } vg} & \begin{array}{l} \text{Ansprechzone vor der eigenen und andere Station} \\ sp_{vg} < sp_{ap} \wedge st_{vg} \neq st_{ap} \end{array} \\ pos_{idx,vg} - (st_{ap} - st_{vg}) \cdot tz & \begin{array}{l} sp_{vg} \geq sp_{ap} \wedge st_{vg} \neq st_{ap} \\ \text{Ansprechzone hinter der eigenen und andere Station} \end{array} \\ pos_{idx,vg} - \tau_{idx,ap} & \begin{array}{l} sp_{vg} < sp_{ap} \wedge st_{vg} = st_{ap} \\ \text{Ansprechzone vor der eigenen und gleiche Station} \end{array} \end{cases} \quad (5.54)$$

Wenn die Ansprechzone eines Vorgängerarbeitsplatzes vg einen kleineren Wert³¹⁰ hat als die eigene, so kann ohne Überholvorgang parallel am selben Werkstück gearbeitet werden. Dies ist im Beispiel 5.24 bei Arbeitsplatz AP1 und AP2 der Fall. Während an AP1 noch gearbeitet wird, kann AP2 bereits zum gleichen Werkstück aufrücken und parallel arbeiten. Die Wartezeit, die sich für AP2 ergibt, kann ermittelt werden durch die Werkerendposition des Vorgängerarbeitsplatzes an dem betreffenden Werkstück $pos_{idx,vg}$ abzüglich der eigenen Bearbeitungszeit und der Entfernung zum Vorgänger. Für das Beispiel in Abbildung 5.25 gestaltet sich die Berechnung wie folgt:

$$wz_{AP1,AP2,\text{Werkstück1}} = 130 - 70 - 1 \cdot 50 = 10$$

³⁰⁹ Hierbei wird von einer durchgängigen Nummerierung der Stationen ausgegangen.

³¹⁰ Es wird davon ausgegangen, dass die einzelnen Ansprechzonen, wie in Abschnitt 2.1.1 dargestellt, durchnummeriert sind.

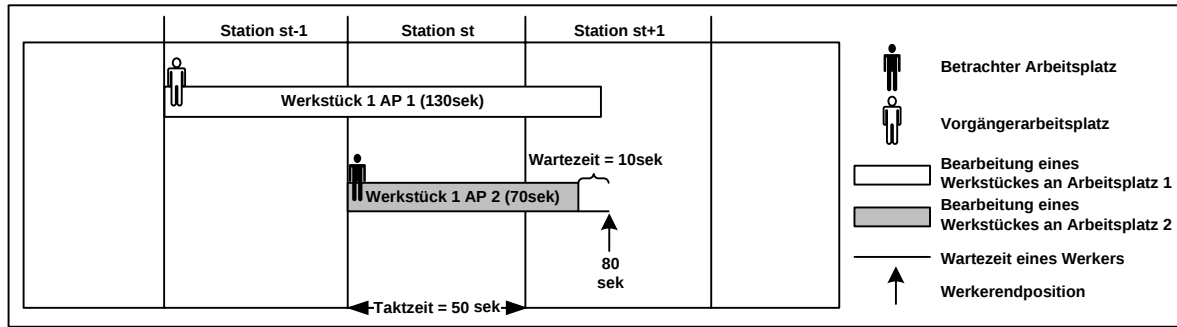


Abbildung 5.25: Beispiel für die Beeinflussung durch einen Vorgängerarbeitsplatz mit voran liegender Ansprechzone in anderer Station

Befindet sich die Ansprechzone des Vorgängerarbeitsplatzes hinter oder in der gleichen Ansprechzone des Arbeitsplatzes ap , so wie im Beispiel 5.24 mit den Arbeitsplätzen AP6 und AP7 der Fall, so ist keine parallele Arbeit an einem Werkstück möglich. Der Werker an Arbeitsplatz AP7 kann erst seine Ansprechzone erreichen, wenn der Vorgängerarbeitsplatz AP6 seine Arbeit vollständig beendet hat. In diesem Fall berechnet sich die Wartezeit durch die Werkerendposition des Vorgängerarbeitsplatzes $pos_{idx,vg}$ abzüglich des Abstandes zu der Station vom Vorgängerarbeitsplatz st_{vg} . Im Beispiel 5.25 ergibt sich folgende Berechnung:

$$wz_{AP6,AP7,Werkstück1} = 130 - 1 \cdot 50 = 80$$

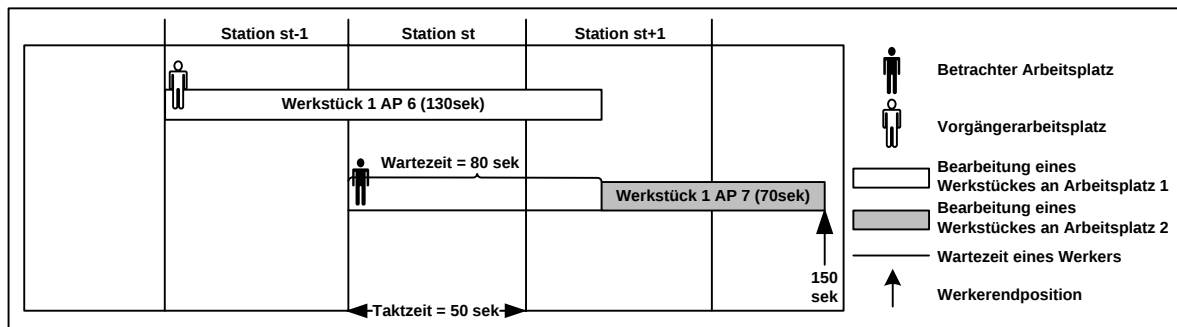


Abbildung 5.26: Beispiel für die Beeinflussung durch einen Vorgängerarbeitsplatz mit gleicher oder dahinter liegender Ansprechzone in anderer Station

Im dritten Fall befindet sich der Vorgängerarbeitsplatz vg an der gleichen Station und die Ansprechzone somit zwangsläufig vor der eigenen. Auch hier kann ein Verzug entstehen, wenn der Werker an dem Vorgängerarbeitsplatz länger zu arbeiten hat und ein Überholen nicht möglich ist. Die Wartezeit berechnet sich dann durch die Endposition des Vorgängerarbeitsplatzes $pos_{idx,vg}$ abzüglich der Bearbeitungszeit des Werkstückes am entsprechenden Arbeitsplatz ap . Dies ist im Beispiel 5.24 mit den Arbeitsplätzen AP3 und AP4 der Fall. Der Verzug ist illustriert in der Abbildung 5.27 und errechnet sich wie folgt:

$$wz_{AP3,AP4,Werkstück1} = 130 - 70 = 60$$

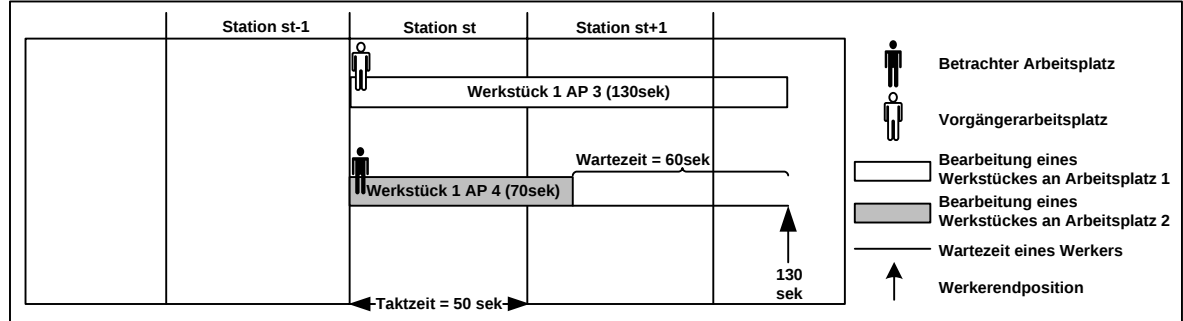


Abbildung 5.27: Beispiel für die Beeinflussung durch einen Vorgängerarbeitsplatz in der gleichen Station

5.3.1.3 Abbildung der Unterstützereinsätze

Überlastungen³¹¹ werden von Unterstützerarbeitskräften abgebaut. Die in Abschnitt 2.1.2.2.2 beschriebenen organisatorischen Alternativen sollen im Folgenden formal beschrieben werden.

5.3.1.3.1 Reaktiver Unterstützereinsatz Der Unterstützer übernimmt lediglich die Bearbeitungszeit des Werkstückes, die über der Driftgrenze liegt. Für diesen Fall wird die Einsatzzeit des Unterstützers an dem Werkstück formal in (5.55) definiert.

$$u_{idx,ap}^{Zeit} = \max[\tau_{idx,ap} + \max[wv_{idx,ap}, vu_{idx,ap}] - dbr_{ap} - (w_{ap} \cdot tz); 0] \quad (5.55)$$

Der Endzeitpunkt, an dem das Werkstück fertig bearbeitet ist, wird durch die Bearbeitungszeit des Werkstückes zuzüglich des Startzeitpunktes des Wokers, gegeben durch das Maximum aus Werkerverzug und Vorgängerüberhang, berechnet. Davon abgezogen wird die Größe des normalen Bearbeitungsbereiches an dem Arbeitsplatz inkl. Driftbereich. Ergebnis ist die über der Driftgrenze liegende Bearbeitungszeit des Werkstückes. Desweiteren wird in Gleichung 5.56 die Positionsrechnung des Wokers um die Unterstützereinsatzzeit bereinigt, da dieser unmittelbar mit dem nächsten Werkstück beginnen kann, sobald der Unterstützer das derzeitige Werkstück übernommen hat.

$$pos_{idx,ap} = \tau_{idx,ap} + \max[wv_{idx,ap}, vu_{idx,ap}] - (u_{p,ap}^{Zeit}) \quad (5.56)$$

³¹¹Überlastungen sind definiert als Bearbeitungszeiten, die über die erlaubte Driftgrenze hinaus entstehen. Vgl. auch Abschnitt 2.1.2.2.2

Im Beispiel 5.28 gestaltet sich die Berechnung wie folgt:

$$u_{\text{Werkstück1,AP2}}^{\text{Zeit}} = 85 + 40 - 40 - 50 = 35$$

$$pos_{\text{Werkstück1,AP2}} = 85 + 40 - 35 = 90$$

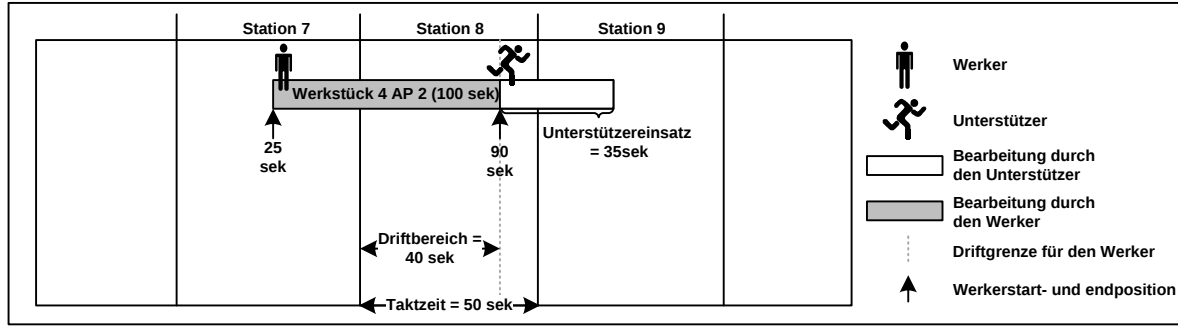


Abbildung 5.28: Beispiel für einen reaktiven Unterstützereinsatz

Zur späteren Ermittlung der Anzahl notwendiger Unterstützereinsätze wird eine binäre Variable in Gleichung 5.57 eingeführt, die definiert, ob an dem Werkstück an Reihenfolgeposition idx und Arbeitsplatz ap ein Unterstützer gearbeitet hat. Sobald eine Bearbeitungszeit auftritt, die den Werker über den Bearbeitungsbereich hinweg belasten würde, wird diese Variable auf 1 gesetzt.

$$u_{idx,ap}^{\text{Einsatz}} = \begin{cases} 1 & \text{wenn}(\tau_{idx,ap} + \max[wv_{idx,ap}, vu_{idx,ap}] - dbr_{ap} - (w_{ap} \cdot tz)) > 0 \\ 0 & \text{sonst} \end{cases} \quad (5.57)$$

Wie in Abschnitt 2.1.2.2.2 beschrieben, wird davon ausgegangen, dass ein Unterstützer, der innerhalb eines Taktes t die Aufgaben eines Arbeitsplatzes ap übernimmt, an keinem anderen Arbeitsplatz aushelfen kann. Dabei ist es unerheblich, wie lange er innerhalb des Taktes beschäftigt ist. Der Takt, und damit der Zeitpunkt eines Unterstützereinsatzes, kann bestimmt werden durch den Zusammenhang des Taktindex mit der aktuellen Reihenfolgeposition sowie der Station, welcher der aktuell betrachtete Arbeitsplatz zugehörig ist (5.58).³¹²

$$\begin{aligned} t_{idx,ap} &= st_{ap} + idx - 1 \\ idx_{t,ap} &= t - st_{ap} + 1 \end{aligned} \quad (5.58)$$

³¹²Vgl. dazu die Diskussion der Gleichzeitigkeit in Abschnitt 2.1.3.1.2

Ein Unterstützereinsatz, der in Takt i beginnt, kann für eine längere Belegung des Unterstützers über mehrere Takte hinweg sorgen, wie im Beispiel 5.28 der Fall. Die Ermittlung des Taktes, in dem der Unterstützereinsatz beginnt, ergibt in diesem Beispiel $t_{idx,ap} = 8 + 4 - 1 = 11$ auf Basis von Gleichung (5.58). Um zu bestimmen, ob ein Unterstützer durch einen Unterstützereinsatz, der an Reihenfolgeposition idx durchgeführt wird, noch in Takt t belegt ist, werden zwei binäre Statusvariablen eingeführt. Im ersten Schritt wird bestimmt, ob der Startzeitpunkt des Unterstützereinsatzes, definiert durch $pos_{idx,ap}$, innerhalb oder vor dem betrachteten Takt liegt (5.59). In diesem Fall wird die binäre Statusvariable $u_{t,idx,ap}^{Start}$ auf 1 gesetzt.

$$u_{t,idx,ap}^{Start} = \begin{cases} 1 & \text{wenn } ((t - t_{idx,ap}) \cdot tz - pos_{idx,ap}) \cdot u_{idx,ap}^{Einsatz} > 0 \\ 0 & \text{sonst} \end{cases} \quad (5.59)$$

Übertragen auf unser Beispiel gilt dies für alle Takte $t \geq 11$.

$$\begin{aligned} u_{10,11,AP2}^{Start} &= 0 \geq 0 - 40 \cdot 1 \\ u_{11,11,AP2}^{Start} &= 1 \geq 50 - 40 \cdot 1 \\ u_{12,11,AP2}^{Start} &= 1 \geq 100 - 40 \cdot 1 \\ u_{13,11,AP2}^{Start} &= 1 \geq 150 - 40 \cdot 1 \end{aligned}$$

Im zweiten Schritt wird durch Gleichung (5.60) errechnet, ob das Ende des Unterstützereinsatzes, definiert durch den Startzeitpunkt zuzüglich der Unterstützereinsatzzeit, innerhalb oder nach dem betrachteten Takt t liegt. Die binäre Statusvariable wird in diesem Fall auf 1 gesetzt.

$$u_{t,idx,ap}^{Ende} = \begin{cases} 1 & \text{wenn } (pos_{idx,ap} + u_{idx,ap}^{Zeit} - (t - t_{idx,ap}) \cdot tz) \cdot u_{idx,ap}^{Einsatz} > 0 \\ 0 & \text{sonst} \end{cases} \quad (5.60)$$

Übertragen auf unser Beispiel gilt dies für alle Takte $t \leq 12$.

$$\begin{aligned} u_{10,11,AP2}^{Ende} &= 1 \geq 40 + 35 - (-50) \cdot 1 \\ u_{11,11,AP2}^{Ende} &= 1 \geq 40 + 35 - (0) \cdot 1 \\ u_{12,11,AP2}^{Ende} &= 1 \geq 40 + 35 - (50) \cdot 1 \\ u_{13,11,AP2}^{Ende} &= 0 \geq 40 + 35 - (100) \cdot 1 \end{aligned}$$

Darauf aufbauend gibt $u_{t,idx,ap}^{Belegt}$ an, ob in Takt t noch ein Unterstützer belegt ist, der an dem Werkstück mit der Reihenfolgeposition idx an Arbeitsplatz ap einen Unterstützereinsatz begonnen hat (5.61).

$$u_{t,idx,ap}^{Belegt} = u_{idx,ap}^{Einsatz} \cdot u_{t,idx,ap}^{Start} \cdot u_{t,idx,ap}^{Ende} \quad (5.61)$$

Übertragen auf unser Beispiel gilt dies für die Takte $t = 11 \wedge t = 12$.

$$\begin{aligned} u_{10,11,AP2}^{Belegt} &= 0 = 1 \cdot 0 \cdot 1 \\ u_{11,11,AP2}^{Belegt} &= 1 = 1 \cdot 1 \cdot 1 \\ u_{12,11,AP2}^{Belegt} &= 1 = 1 \cdot 1 \cdot 1 \\ u_{13,11,AP2}^{Belegt} &= 0 = 0 \cdot 1 \cdot 1 \end{aligned}$$

5.3.1.3.2 Präventiver Unterstützereinsatz Der Unterstützer übernimmt das gesamte nächste Werkstück, falls dieses den Werker überlasten würde. Es wird somit präventiv eingegriffen. Der Werker überspringt das potentiell überlastende Werkstück und bearbeitet direkt das nächste. Für diesen Fall ist die Einsatzzeit des Unterstützers durch die Bearbeitungszeit des übernommenen Werkstückes in Gleichung (5.62) definiert.

$$u_{idx,ap}^{Zeit} = \tau_{idx,ap} \cdot u_{idx,ap}^{Einsatz} \quad (5.62)$$

Ein Unterstützereinsatz ist notwendig, wenn die prognostizierte Endposition des Werkers ausserhalb des Bearbeitungsbereiches liegen würde (5.63). In diesem Fall wird die binäre Statusvariable $u_{idx,ap}^{Einsatz}$ auf 1 gesetzt.

$$u_{idx,ap}^{Einsatz} = \begin{cases} 1 & \text{wenn} \quad \overbrace{\max[wv_{idx,ap}, vu_{idx,ap}] + \tau_{idx,ap}}^{\text{Prognostizierte Position nach Bearbeitung des Werkstückes}} - \overbrace{((w_{ap} \cdot tz) + dbr_{ap})}^{\text{Bearbeitungsbereich des Arbeitsplatzes}} > 0 \\ 0 & \text{sonst} \end{cases} \quad (5.63)$$

Die Werkerposition muss für den Fall, dass ein Unterstützereinsatz an dem Werkstück stattfindet, abweichend definiert werden. Dies wird in Gleichung (5.64) mit Hilfe einer Fallunterscheidung durchgeführt.

$$pos_{idx,ap} = \begin{cases} \tau_{idx,ap} + \max[wv_{idx,ap}, vu_{idx,ap}] & \text{wenn } u_{idx,ap}^{Einsatz} = 0 \\ \max[wv_{idx,ap}, vu_{idx,ap}] & \text{wenn } u_{idx,ap}^{Einsatz} = 1 \end{cases} \quad (5.64)$$

Die Endposition des Werkers entspricht dem Maximum aus Werkerverzug und Vorgängerüberhang ohne Einfluss der Bearbeitungszeit des Werkstückes. Die Berechnung soll an Beispiel 5.29 verdeutlicht werden. In diesem Fall existiert kein Vorgängerüberhang, so dass sich die Position ausschließlich durch den Werkerverzug definiert. Dieser entspricht, wie zuvor in Gleichung (5.52) definiert, der alten Position des Werkers, abzüglich des Abstandes zum nächsten Werkstück.

$$u_{\text{Werkstück5,AP2}}^{\text{Zeit}} = 90$$

$$pos_{\text{Werkstück5,AP2}} = \max[(95 - 50); 0] = 45$$

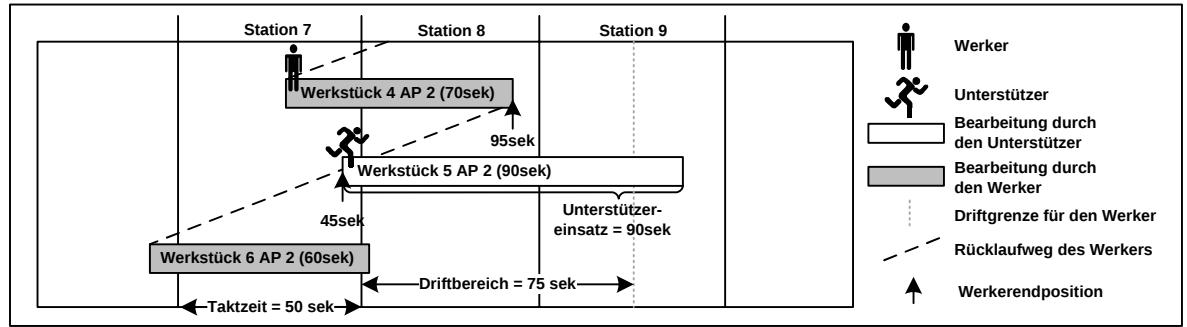


Abbildung 5.29: Beispiel für einen präventiven Unterstützereinsatz

Bei der präventiven Betrachtung von Unterstützereinsätzen wird davon ausgegangen, dass der Einsatz bereits im Vorfeld geplant und bekannt ist. Daher kann der Unterstützer schon mit dem Werkstück an der linken Stationsgrenze beginnen und muss nicht darauf warten, dass der Arbeiter ihm eine Überlastung signalisiert. Auch in diesem Fall kann der Unterstützereinsatz länger als einen Takt dauern. In Gleichung (5.65) wird die Statusvariable definiert, welche angibt, ob ein Unterstützereinsatz, der an dem Werkstück mit der Reihenfolgeposition idx durchgeführt wird, den Unterstützer in Takt t noch belegt. Die Statusvariable ist somit in jedem Takt t gültig, in dem noch ein Teil der Bearbeitungszeit des Werkstückes durch den Unterstützer durchgeführt werden muss.³¹³

$$u_{t,idx,ap}^{\text{Belegt}} = \begin{cases} 1 & \text{wenn } (u_{idx,ap}^{\text{Einsatz}} \cdot \tau_{idx,ap}) - ((t - t_{idx,ap}) \cdot tz) > 0 \\ 0 & \text{sonst} \end{cases} \quad (5.65)$$

³¹³Wie auch bei der Betrachtung von reaktivem Unterstützereinsatz wird davon ausgegangen, dass ein Unterstützer, der innerhalb eines Taktes t die Aufgaben eines Arbeitsplatzes ap übernimmt, an keinem anderen Arbeitsplatz aushelfen kann. Dabei ist es unerheblich, wie lange er innerhalb des Taktes beschäftigt ist. Vgl. Abschnitt 2.1.3.1.2

Bei beiden Alternativen der Gestaltung von Unterstützereinsätzen müssen noch Variablen für die Ermittlung der Zielgrößen eingeführt werden. Die Anzahl an Unterstützereinsätzen lässt sich ermitteln durch eine einfache Summierung der Statusvariable in (5.66).

$$u^E = \sum_{ap \in \mathcal{AP}} \sum_{idx \in \mathcal{RF}} u_{idx,ap}^{Einsatz} \quad (5.66)$$

Zur Ermittlung der vorzuhaltenden Unterstützer im System sind weitere Variablen notwendig. Durch Gleichung (5.67) wird die notwendige Anzahl an Unterstützern an Arbeitsplatz ap im Takt t ermittelt. Dies können mehrere sein, wenn ein Unterstützereinsatz, der an einem vorherigen Werkstück begonnen hat, noch nicht abgeschlossen ist und ein neuer Unterstützereinsatz am nächsten Werkstück notwendig wird. Es genügt hier, jeweils auf die Takte $i \leq t$ zu schauen, da nur in Takten, die vor t beziehungsweise in t liegen, ein Unterstützereinsatz existieren kann, der in Takt t noch bearbeitet wird.

$$u_{t,ap}^{AnzAP} = \sum_{i \in \mathcal{T} | i \leq t} u_{t,idx_i,ap,ap}^{Belegt} \quad (5.67)$$

Im nächsten Schritt wird eine Aggregation auf das Stationsniveau durchgeführt. Wenn an mehreren Arbeitsplätzen innerhalb einer Station Unterstützer belegt sind, müssen dementsprechend Unterstützer vorgehalten werden (5.68).

$$u_{t,st}^{AnzST} = \sum_{ap \in \mathcal{AP}} u_{t,ap}^{AnzAP} \mid st_{ap} = st \quad (5.68)$$

Darauf aufbauend wird in Gleichung 5.69 definiert, wie viele gleichzeitige Unterstützereinsätze im Takt t über alle Stationen notwendig sind.

$$u_t = \sum_{st \in \mathcal{ST}} u_{t,st}^{AnzST} \quad (5.69)$$

Die notwendige Anzahl vorzuhaltender Unterstützer im Fließsystem kann nun abgeleitet werden aus der maximalen Anzahl gleichzeitiger Unterstützereinsätze über alle Takte hinweg.³¹⁴ Die Zahl der vorzuhaltenden Unterstützer in einem Planungszeitraum muss konstant sein und definiert sich durch den Maximalfall, formal definiert in Gleichung (5.70).

$$u^{max} = \max_{t \in \mathcal{T}} \left[\sum_{st \in \mathcal{ST}} u_{t,st}^{AnzST} \right] \quad (5.70)$$

³¹⁴Vgl. Abschnitt 2.1.3.1.2

Im Folgenden werden die Zielfunktion und die Restriktionen in einem Modell mit zwei alternativen Bausteinen für die Organisationsformen der Unterstützereinsätze zusammengefasst.

$$\text{Min } \mathcal{K}^{Ges} = \mathcal{K}^u \cdot u^{MAX} + \mathcal{K}^N \cdot u^E$$

u.B.d.R.

$$\sum_{p \in \mathcal{P}} y_{p,idx}^{\mathcal{RF}} = 1 \quad \forall \quad idx \in \mathcal{RF}_{pz}$$

$$\sum_{idx \in \mathcal{RF}} y_{p,idx}^{\mathcal{RF}} = 1 \quad \forall \quad p \in \mathcal{P}_{pz}$$

$$\tau_{idx,ap} = \sum_{p \in \mathcal{P}} \sum_{avo \in \mathcal{AVO}} (\tau_{avo} \cdot z_{avo,ap}^{Ist} \cdot bp_{a,avo}) \cdot y_{p,idx}^{\mathcal{RF}} \quad \forall \quad ap \in \mathcal{AP}, idx \in \mathcal{RF}, a \in \mathcal{A} : h_{a,p} = 1$$

$$wv_{idx,ap} = \max \begin{cases} pos_{idx-w_{ap,ap}} - (tz \cdot w_{ap}) \\ -dbl_{ap} \end{cases} \quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP}$$

$$vu_{idx,ap} = \max[-dbl_{ap}; \max_{vg \in \mathcal{VG}_{ap}} \{wz_{vg,ap,idx}\}] \quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP}$$

$$wz_{vg,ap,idx} = \begin{cases} pos_{idx,vg} - \tau_{idx,ap} - (st_{ap} - st_{vg}) \cdot tz & sp_{vg} < sp_{ap} \wedge st_{vg} \neq st_{ap} \\ pos_{idx,vg} - (st_{ap} - st_{vg}) \cdot tz & sp_{vg} \geq sp_{ap} \wedge st_{vg} \neq st_{ap} \\ pos_{idx,vg} - \tau_{idx,ap} & sp_{vg} < sp_{ap} \wedge st_{vg} = st_{ap} \end{cases}$$

$$\forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP} \wedge vg \in \mathcal{VG}_{ap}$$

$$t_{idx,ap} = st_{ap} + idx - 1 \quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP}$$

$$idx_{t,ap} = t - st_{ap} + 1 \quad \forall \quad t \in \mathcal{T} \wedge ap \in \mathcal{AP}$$

$$u^E = \sum_{ap \in \mathcal{AP}} \sum_{idx \in \mathcal{RF}} u_{idx,ap}^{Einsatz}$$

$$u_{t,ap}^{AnzAP} = \sum_{i \in \mathcal{T} | i \leq t} u_{t,idx_{i,ap},ap}^{Belegt} \quad \forall \quad t \in \mathcal{T} \wedge ap \in \mathcal{AP}$$

$$u_{t,st}^{AnzST} = \sum_{ap \in \mathcal{AP}} u_{t,ap}^{AnzAP} \mid st_{ap} = st \quad \forall \quad t \in \mathcal{T} \wedge st \in \mathcal{ST}$$

$$u_t = \sum_{st \in \mathcal{ST}} u_{t,st}^{AnzST} \quad \forall \quad t \in \mathcal{T}$$

$$u^{MAX} = \max_{t \in \mathcal{T}} \left[\sum_{st \in \mathcal{ST}} u_{t,st}^{AnzST} \right]$$

(5.71)

Reaktiver Unterstützeinsatz

$$\begin{aligned}
pos_{idx,ap} &= \tau_{idx,ap} + \max[uv_{idx,ap}, vu_{idx,ap}] - (u_{p,ap}^{Zeit}) \quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP} \\
u_{idx,ap}^{Zeit} &= \max[\tau_{idx,ap} + \max[uv_{idx,ap}, vu_{idx,ap}] - dbr_{ap} - (w_{ap} \cdot tz); 0] \\
&\quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP} \\
u_{idx,ap}^{Einsatz} &= \begin{cases} 1 & \text{wenn } (\tau_{idx,ap} + \max[uv_{idx,ap}, vu_{idx,ap}] - dbr_{ap} - (w_{ap} \cdot tz)) > 0 \\ 0 & \text{sonst} \end{cases} \\
&\quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP} \\
u_{t,idx,ap}^{Start} &= \begin{cases} 1 & \text{wenn } ((t - t_{idx,ap}) \cdot tz - pos_{idx,ap}) \cdot u_{idx,ap}^{Einsatz} > 0 \\ 0 & \text{sonst} \end{cases} \\
&\quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP} \wedge t \in \mathcal{T} \\
u_{t,idx,ap}^{Ende} &= \begin{cases} 1 & \text{wenn } (pos_{idx,ap} + u_{idx,ap}^{Zeit} - (t - t_{idx,ap}) \cdot tz) \cdot u_{idx,ap}^{Einsatz} > 0 \\ 0 & \text{sonst} \end{cases} \\
&\quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP} \wedge t \in \mathcal{T} \\
u_{t,idx,ap}^{Belegt} &= u_{idx,ap}^{Einsatz} \cdot u_{t,idx,ap}^{Start} \cdot u_{t,idx,ap}^{Ende} \quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP} \wedge t \in \mathcal{T}
\end{aligned} \tag{5.72}$$

Präventiver Unterstützeinsatz

$$\begin{aligned}
pos_{idx,ap} &= \begin{cases} \tau_{idx,ap} + \max[uv_{idx,ap}, vu_{idx,ap}] & \text{wenn } u_{idx,ap}^{Einsatz} = 0 \\ \max[uv_{idx,ap}, vu_{idx,ap}] & \text{wenn } u_{idx,ap}^{Einsatz} = 1 \end{cases} \\
&\quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP} \\
u_{idx,ap}^{Zeit} &= \tau_{idx,ap} \cdot u_{idx,ap}^{Einsatz} \quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP} \\
u_{idx,ap}^{Einsatz} &= \begin{cases} 1 & \text{wenn } \max[uv_{idx,ap}, vu_{idx,ap}] + \tau_{idx,ap} - ((w_{ap} \cdot tz) + dbr_{ap}) > 0 \\ 0 & \text{sonst} \end{cases} \quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP} \\
u_{t,idx,ap}^{Belegt} &= \begin{cases} 1 & \text{wenn } (u_{idx,ap}^{Einsatz} \cdot \tau_{idx,ap}) - ((t - t_{idx,ap}) \cdot tz) > 0 \\ 0 & \text{sonst} \end{cases} \\
&\quad \forall \quad idx \in \mathcal{RF} \wedge ap \in \mathcal{AP} \wedge t \in \mathcal{T}
\end{aligned} \tag{5.73}$$

5.3.2 Vorgehensmodell zur Reihenfolgeplanung

Das Reihenfolgeplanungsproblem ist ein kombinatorisches Optimierungsproblem.³¹⁵ Auf Grund der Komplexität dieser Problemklasse ist die Ermittlung einer exakten Lösung in vertretbarer Zeit unmöglich und es gilt mit einem heuristischen Lösungsverfahren³¹⁶ eine Lösung zu ermitteln.³¹⁷ Das Verfahren ist zweigeteilt und wird in Abbildung 5.30 illustriert.

In einer Initialheuristik wird eine gleichmäßige Verteilung von Werkstücken mit hohen Bearbeitungszeiten und geringen Bearbeitungszeiten durchgeführt. Um die Komplexität zu reduzieren, wird dabei eine Aggregation mit Hilfe eines Clustering-Algorithmus vorgenommen. Ziel des Verfahrens ist die Minimierung der Anzahl notwendiger Unterstützereinsätze. Darauf aufbauend wird ein problemspezifisches iteratives Verbesserungsverfahren entwickelt, das die initiale Reihenfolge verwendet und insbesondere die gleichzeitigen Unterstützereinsätze sowie die Anzahl an Unterstützereinsätzen weiter minimiert.

Beide Verfahren nutzen ein System zur Fließliniensimulation, welches im nächsten Abschnitt vorgestellt wird. Anschließend wird der zweistufige Optimierungsansatz zur Reihenfolgeplanung beschrieben.

5.3.3 Konzept zur Fließliniensimulation

Im Folgenden wird eine Fließliniensimulation vorgestellt, welche im Wesentlichen aus der analytischen Bestimmung der Positionen von Werkern im Fließsystem besteht. Das Verfahren wird im Rahmen der Reihenfolgeoptimierung genutzt und im Anschluss zur Ermittlung der Leistungsfähigkeit der entwickelten Algorithmen in Kapitel 6 verwendet.

³¹⁵Vgl. [Mol96, S.112]

³¹⁶Vgl. [SM06, S.12] zu Heuristiken: „Heuristiken ermitteln in einem vertretbaren Rechenaufwand möglichst gute Lösungen. Allerdings kann nicht garantiert werden, dass eine im mathematischen Sinne optimale Lösung gefunden wird.“

³¹⁷Probleme dieser Art beschäftigen sich im Allgemeinen mit der Bestimmung einer besten Anordnung diskreter Objekte, in diesem Fall der Werkstücke, die jeweils einer Reihenfolgeposition zugeordnet werden. Diese Anordnung wird im Allgemeinen durch eine Permutation ganzer Zahlen dargestellt. Theoretisch könnte jede mögliche Reihenfolge der Werkstücke gebildet und der zugehörige Zielfunktionswert ermittelt werden (Vollständige Enumeration). Das Verfahren der vollständigen Enumeration lässt sich jedoch aufgrund der großen Anzahl möglicher Permutationen bei einer praxisrelevanten Problemgröße nicht durchführen. Die Berechnungsdauer wäre wesentlich länger als die verfügbare Zeit, die zwischen dem Zeitpunkt, zu dem die Planungsdaten verfügbar sind, und dem Entscheidungszeitpunkt liegt. Die Anzahl möglicher Reihenfolgen im vorliegenden Reihenfolgeproblem ist definiert durch $|\mathcal{RF}_{pz}|!$. Da bei der hohen Variantenvielfalt davon auszugehen ist, dass innerhalb eines Planungszeitraumes kein Werkstück dem anderen gleicht, kann jede dieser Reihenfolgen ein anderes Ergebnis ergeben. Wenn beispielsweise 70 Werkstücke in eine Produktionsreihenfolge gebracht werden müssten, so gilt es $70! \approx 1,2 \times 10^{100}$ verschiedene Reihenfolgen zu bewerten. In der Praxis müssen häufig hunderte von Werkstücken innerhalb eines Planungszeitraumes sequenziert werden. Ein Rechner, der in der Lage wäre, innerhalb einer Millisekunde eine Produktionsreihenfolge vollständig zu bewerten, würde schon bei 70 Werkstücken ca. 10^{87} Jahre für diese Berechnungen benötigen.

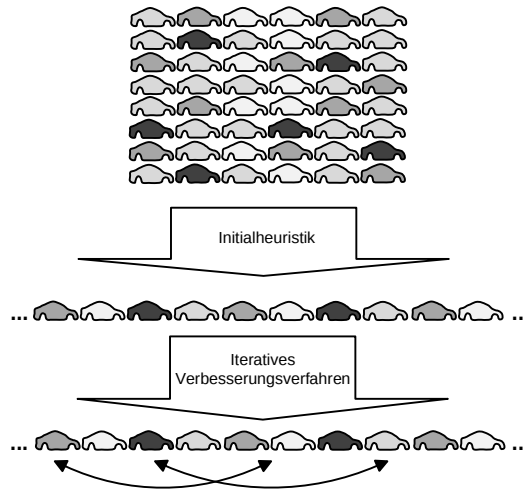


Abbildung 5.30: Vorgehen der zweistufigen Reihenfolgeheuristik

Symbol	Definition
$\mathcal{BEL}_{pz}$	Matrix mit den Bearbeitungszeiten der Werkstücke $p \in \mathcal{BWP}_{pz}$, die in dem Produktionszeitraum $pz \in \mathcal{PZ}$ bewertet werden.
$\mathcal{POS}_{pz}$	Matrix mit den Werkerendpositionen jeweils nach der Bearbeitung der Werkstücke $p \in \mathcal{BWP}$ an den Reihenfolgepositionen $idx \in \mathcal{BWRF}_{pz}$ je Arbeitsplatz $ap \in \mathcal{AP}$ im Produktionszeitraum $pz \in \mathcal{PZ}$.
$\mathcal{T}_{pz}$	Menge von Takten t , die im Produktionszeitraum pz stattfinden

Tabelle 5.21: Notationserweiterung für die Fließlinienbewertung

Bei der Berechnung werden keine Unsicherheiten betrachtet, d.h. sämtliche Eingangsdaten werden deterministisch vorgegeben. Störungen im Betriebsablauf werden nicht abgebildet, da diese unabhängig von den beiden zu bewertenden Planungsschritten Rekonfiguration und Reihenfolgeplanung sind. Es kann nicht davon ausgegangen werden, dass die hier berechneten Werkerpositionen zwingend ein genaues Abbild der Wirklichkeit darstellen. Dies liegt insbesondere daran, dass die Arbeitsgeschwindigkeiten von Werker zu Werker schwanken. Da es jedoch nicht möglich ist, diese Schwankungen vorherzusehen, wird durch die deterministische Betrachtungsweise gewährleistet, dass lediglich der Einfluss durch die zu bewertenden Planungsmethoden gemessen wird.

5.3.3.1 Ermittlung der Eingangsdaten für die Bewertung

Für eine vollständige Simulation des Produktionsablaufes im Fließsystem werden folgende Eingangsdaten benötigt:

- Reihenfolge, in der die Werkstücke in das Fließsystem eingeschleust werden $y_{p,idx}^{\mathcal{RF}}$

- Zuordnung der Arbeitsvorgänge zu den Arbeitsplätzen $z_{avo,ap}^{Ist}$
- Taktzeit tz des Fließsystems
- Bearbeitungszeiten der Arbeitsvorgänge τ_{avo}
- Zuordnung der notwendigen Arbeitsvorgänge je Auftrag $bp_{a,avo}$
- Zuordnung der Arbeitsplätze zu Ansprechzonen sp_{ap} und Stationen st_{ap}
- Organisationsform der Unterstützer (reaktiv oder präventiv)
- Größe der linken dbl_{ap} und rechten Driftbereiche dbr_{ap} je Arbeitsplatz
- Kostensatz für einen Unterstützer $\mathcal{K}^U$
- Kostensatz für die Nacharbeit durch einen Unterstützereinsatz $\mathcal{K}^N$
- Abhängigkeiten der Werker untereinander, definiert durch eine Menge von Vorgänger-arbeitsplätzen je Arbeitsplatz $\mathcal{VG}_{ap} \subset \mathcal{AP}$

Die Reihenfolge der Werkstücke sowie die Zuordnung der Arbeitsvorgänge zu den Arbeitsplätzen werden durch die Reihenfolgeplanung beziehungsweise die Rekonfiguration bestimmt. Die Taktzeit des Fließsystems, die Bearbeitungszeiten der Arbeitsvorgänge sowie die Zuordnung von den Arbeitsplätzen zu den Ansprechzonen und Stationen können den Informationen zum existierenden Fließsystem entnommen werden. Desweiteren muss eine Organisationsform der Unterstützer gewählt werden.³¹⁸

Die Bestimmung der notwendigen Arbeitsvorgänge zur Bearbeitung eines Werkstückes wird je nach Produktdokumentationsart unterschiedlich vorgenommen. Im Falle der offenen Variantenstückliste mit Code-Regeln³¹⁹ wird eine Art Stücklistenauflösung für die Produktionsstückliste durchgeführt.³²⁰

Die Driftgrenzen müssen für jeden Arbeitsplatz abhängig von seinen Driftmöglichkeiten hinsichtlich der technischen Hilfsmittel, dem Laufweg zur Materialbereitstellung und baulicher Grenzen definiert werden. Häufig wird in der Praxis ein Vordriften prinzipiell nicht erlaubt, um davor liegende Werker nicht unter Druck zu setzen. In diesem Fall wird die linke Driftgrenze auf 0 gesetzt. Eine untere Grenze für die rechte Driftgrenze kann mit Hilfe von Gleichung 5.74 abgeschätzt werden. Dafür wird die auf einen vollen Takt aufgerundete maximale Bearbeitungszeit an einem Arbeitsplatz verwendet, da davon ausgegangen werden kann, dass das Werkstück mit der maximalen Bearbeitungszeit ohne Unterstützereinsatz bearbeitet werden kann. Da die baulichen Gegebenheiten je Station wechseln, ist es sinnvoll, Driftbereiche

³¹⁸Vgl. Abschnitt 5.3.1

³¹⁹Vgl. hierzu Abschnitt 5.2.2.1

³²⁰Vgl. zu Verfahren der Stücklistenauflösung [Orl75, S.65ff]

immer auf eine Stationslänge aufzurunden, da sich innerhalb einer Station keine weiteren Begrenzungen ergeben.

$$dbr_{ap} \leftarrow \left\lceil \frac{\max_{idx \in \mathcal{RF}} [\tau_{idx,ap}]}{tz} \right\rceil \quad \forall ap \in \mathcal{AP} \quad (5.74)$$

Der Kostensatz für einen Unterstützer lässt sich aus den Jahreskosten für eine Unterstützerarbeitskraft ableiten. Hierfür wird die Relation zwischen der Länge des Planungszeitraumes und der Arbeitszeit eines Unterstützers je Jahr gebildet. Die Jahreskosten werden dann mit dem Quotienten multipliziert.

Der Kostensatz für die zu erwartende Nacharbeit durch einen Unterstützereinsatz kann aus historischen Daten abgeschätzt werden. Hierfür muss die Häufigkeit für eine Nacharbeit an Werkstücken, die von Unterstützern bearbeitet wurden, mit der normalen Häufigkeit für Nacharbeiten bei der Bearbeitung durch Werker verglichen werden. Die erhöhten Nacharbeitskosten, die den Unterstützereinsätzen zugerechnet werden können, werden dann durch die Anzahl Unterstützereinsätze dividiert, so dass man die durchschnittlich zu erwartenden Nacharbeitskosten durch einen Unterstützereinsatz erhält.³²¹

Für die Kalkulation der Werkerpositionen müssen zur Betrachtung der Abhängigkeiten zwischen den Werkern die Vorgängerarbeitsplätze definiert werden.³²²

5.3.3.2 Abbildung des rollierenden Bewertungshorizontes

In der Reihenfolgeplanung werden Werkstücke, die bereits durch die Programmplanung³²³ einem Produktionszeitraum PZ zugewiesen wurden, in eine Reihenfolge gebracht. Die Zuweisung bedeutet, dass diese Werkstücke in dem entsprechenden Produktionszeitraum, z.B. einer Produktionsschicht oder einem ganzen Produktionstag, in das Fließsystem eingelastet werden. Das letzte im Produktionszeitraum eingesteuerte Werkstück wird lediglich die Bearbeitung in der ersten Station erhalten. Alle restlichen durchzuführenden Arbeitsvorgänge werden im nächsten Produktionszeitraum durchgeführt. Umgekehrt bedeutet dies, dass im ersten Takt eines Produktionszeitraumes an $|ST| - 1$ Anzahl Werkstücken die Arbeit weitergeführt wird und nur an der ersten Station mit der Bearbeitung eines neuen Werkstückes begonnen wird. Grafik 5.31 zeigt diese Tatsache genauer.

Ein eventueller Werkerdrift kann sich nicht über einen Produktionszeitraum hinweg in den nächsten verschieben. Zwischen zwei Produktionszeiträumen wird das Fließband angehalten und die Arbeiten an jedem Arbeitsplatz zu Ende geführt. Dies bedeutet, dass die Werker

³²¹Hierbei wird implizit ein linearer Zusammenhang unterstellt.

³²²Vgl. Abschnitt 5.3.1

³²³Vgl. Abschnitt 2.3.3

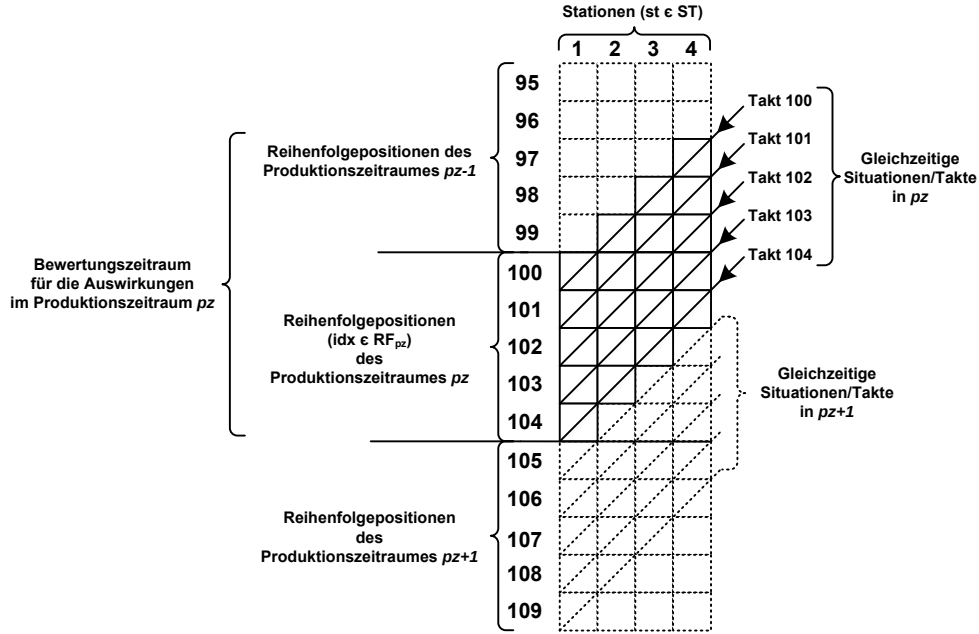


Abbildung 5.31: Darstellung des zu bewertenden Zeitraumes

in einem neuen Produktionszeitraum immer an der linken Stationsgrenze, d.h. Position 0, beginnen. Da auch die Bereitstellung von Unterstützerarbeitskräften je Produktionszeitraum vorgenommen wird, sollte eine Fließliniensimulation sich nicht ausschließlich auf die in einem Produktionszeitraum neu eingesteuerten Werkstücke konzentrieren, sondern muss auch die Restwerkstücke im Fließsystem betrachten. Zusammenfassend kann gesagt werden, dass alle Situationen, die innerhalb eines Produktionszeitraumes auftreten, bewertet werden müssen. Im Beispiel in Grafik 5.31 sind das die Takte 100 bis 105. In Algorithmus 4 werden die zu bewertenden Werkstücke $\mathcal{BWP}_{pz}$ und die dazugehörigen Reihenfolgepositionen $\mathcal{BWRF}_{pz}$ durch die Vereinigung der Reihenfolgepositionen und Werkstücke der zu betrachtenden Planungszeiträume bestimmt.

Algorithmus 4 : Bildung der zu bewertenden Werkstücke

```

1  $\mathcal{BWRF} := \mathcal{RF}_{pz}$ 
2  $\mathcal{BWP} := \mathcal{P}_{pz}$ 
3  $rp := |\mathcal{ST}| - 1$ 
4  $i := 1$ 
5 while  $rp > 0$  do
6    $i := i + 1$ 
7    $rp := rp - |\mathcal{RF}|_{pz-i}$ 
8    $\mathcal{BWRF}_{pz} := \mathcal{BWRF}_{pz} \cup \mathcal{RF}_{pz-i}$ 
9    $\mathcal{BWP}_{pz} := \mathcal{BWP}_{pz} \cup \mathcal{P}_{pz-i}$ 

```

In Zeile 1 wird die Menge der zu bewertenden Reihenfolgepositionen um alle Positionen des aktuellen Planungszeitraumes erweitert. Analog dazu wird in Zeile 2 mit den Werkstücken verfahren. Von Zeile 3 bis 9 werden frühere Planungszeiträume zu den Reihenfolgepositionen und Werkstücken hinzugefügt. Dafür wird in Zeile 3 definiert, wie viele Werkstücke aus früheren Planungszeiträumen, die für die Bewertung notwendig sind, noch nicht betrachtet wurden. In der Schleife von Zeile 5 bis 9 werden so lange frühere Planungszeiträume zu den Werkstücken und Reihenfolgepositionen hinzugefügt, bis alle zu bewertenden Werkstücke enthalten sind.³²⁴ Zusätzlich dazu muss eine Beschränkung der zu betrachtenden Takte vorgenommen werden. Lediglich die Fertigungstakte, die im Produktionszeitraum liegen, festgelegt durch die Definition (5.75), werden bewertet.

$$t \in \mathcal{T}_{pz} \quad \forall \quad t \in \mathcal{T} \quad | \quad \min(\mathcal{RF}_{pz}) \leq t \leq \max(\mathcal{RF}_{pz}) \quad (5.75)$$

5.3.3.3 Verfahren zur Ermittlung der Werkerpositionen

Im ersten Schritt wird erläutert, wie die Bearbeitungszeiten je Werkstück ermittelt werden. Im zweiten Schritt wird die Berechnung der Werkerpositionen über die einzelnen Takte hinweg durchgeführt.

5.3.3.3.1 Bestimmung der Belastungen der Arbeitsplätze je Werkstück Zur Bestimmung der Auslastungen, die je Werkstück auf den einzelnen Arbeitsplätzen auftreten, wird je Werkstück jeder Arbeitsvorgang durchlaufen und die Bearbeitungszeit dem jeweiligen Arbeitsplatz zugeordnet. Das Vorgehen ist in Algorithmus 5 beschrieben. Ergebnis ist eine Belastungsmatrix $\mathcal{BEL}_{pz}$ mit den Einträgen $\tau_{p,ap} \mid p \in \mathcal{P}_{pz}, ap \in \mathcal{AP}$, welche für jedes Werkstück eines Planungszeitraumes die Bearbeitungszeit an jedem Arbeitsplatz definiert. In dem Algorithmus werden alle Werkstücke durchlaufen (Zeile 1). Für jedes Werkstück wird der zugehörige Auftrag in Zeile 2 ermittelt. Dann werden alle Arbeitsvorgänge durchlaufen (Zeile 3). Wenn ein Arbeitsvorgang zur Montage des aktuell betrachteten Auftrages notwendig ist (Zeile 4), wird der zugehörige Arbeitsplatz ermittelt (Zeile 5) und die Ausführungszeit des Arbeitsvorganges zu dem entsprechenden Eintrag in der Matrix addiert (Zeile 6).

$$\mathcal{BEL}_{pz} = \begin{pmatrix} \tau_{1,1} & \tau_{1,2} & \dots & \tau_{1,|\mathcal{AP}|} \\ \tau_{2,1} & \tau_{2,2} & \dots & \tau_{2,|\mathcal{AP}|} \\ \vdots & \ddots & \ddots & \vdots \\ \tau_{|\mathcal{BWP}_{pz}|,1} & \tau_{|\mathcal{BWP}_{pz}|,2} & \dots & \tau_{|\mathcal{BWP}_{pz}|,|\mathcal{AP}|} \end{pmatrix}$$

³²⁴Hierbei werden immer vollständige Planungszeiträume hinzugefügt, obwohl dies für die Bewertung nicht unbedingt nötig ist. Durch die zusätzliche Einschränkung der zu bewertenden Takte wird jedoch eine korrekte Bewertung gewährleistet.

Algorithmus 5 : Berechnung der Bearbeitungszeiten

```

1 forall  $p \in \mathcal{P}$  do
2   set  $a \in \mathcal{A} \mid h_{a,p} = 1$ 
3   forall  $avo \in \mathcal{AVO}$  do
4     if  $bp_{a,avo}$  then
5       set  $ap \in \mathcal{AP} \mid z_{avo,ap}^{Ist} = 1$ 
6        $\tau_{p,ap} := \tau_{p,ap} + \tau_{avo}$ 

```

Auf Basis dieser Belastungsmatrix kann nachfolgend die Kalkulation der Werkerpositionen durchgeführt werden.

5.3.3.3.2 Berechnung der Werkerpositionen Wichtig für die Positionsbestimmung der Werker ist, dass diese Kalkulation möglichst schnell durchgeführt werden kann, um eine Verwendung in einer Heuristik zur Reihenfolgeoptimierung möglich zu machen. Daher wird ein Vorgehen vorgeschlagen, indem eine Positionsmatrix zeilenweise von links nach rechts nur einmal kalkuliert werden muss. Das Ergebnis ist eine Matrix $\mathcal{POS}_{pz}$ mit den Einträgen $pos_{idx,ap} \mid idx \in \mathcal{RF}, ap \in \mathcal{AP}$, welche für jede betrachtete Reihenfolgeposition die Endposition der Werker an dem jeweiligen Arbeitsplatz definiert. Außerdem sind nach Durchführung des Algorithmus die entsprechenden Variablen zur Bewertung der Unterstützereinsätze bestimmt.

$$\mathcal{POS}_{pz} = \begin{pmatrix} pos_{1,1} & pos_{1,2} & \dots & pos_{1,|\mathcal{AP}|} \\ pos_{2,1} & pos_{2,2} & \dots & pos_{2,|\mathcal{AP}|} \\ \vdots & \ddots & \ddots & \vdots \\ pos_{|\mathcal{BWRF}|_{pz},1} & pos_{|\mathcal{BWRF}|_{pz},2} & \dots & pos_{|\mathcal{BWRF}|_{pz},|\mathcal{AP}|} \end{pmatrix}$$

Für die reaktive Unterstützerbetrachtung wird die Berechnung in Algorithmus 6 durchgeführt. Das Verfahren läuft über alle Reihenfolgepositionen und Arbeitsplätze (Zeile 1-2). In Zeile 3 wird der Takt bestimmt, in dem das Werkstück an dem aktuellen Arbeitsplatz bearbeitet wird. Findet dieser Takt innerhalb des zu bewertenden Produktionszeitraumes pz statt (Zeile 4), so wird die Berechnung durchgeführt. Ist dies nicht der Fall, wird die Werkerendposition = 0 gesetzt (Zeile 21). In Zeile 5 wird das Werkstück ermittelt, welches an der zu bewertenden Reihenfolgeposition eingeschleust wird. Im nächsten Schritt wird überprüft, ob ein Unterstützereinsatz notwendig ist (Zeile 6). Ist dies der Fall, so werden in den Zeilen 7 bis 16 die notwendigen Variablen gesetzt. Dabei wird in Zeile 7 die Indikatorvariable für den Unterstützereinsatz gesetzt. In Zeile 8 wird der Unterstützereinsatz gezählt und in Zeile 9 die Länge des Einsatzes gespeichert. In den Zeilen 10 bis 16 werden die Belegungsvariablen der Unterstützer und die Anzahl Einsätze innerhalb eines Taktes berechnet. Hierfür wird der

aktuelle Takt und alle folgenden, in welchen der Unterstützereinsatz noch nicht abgeschlossen ist, durchlaufen (Zeile 11-16). Befindet sich der Startpunkt des Einsatzes vor oder in dem gerade betrachteten Takt, so ist der Unterstützer zu diesem Zeitpunkt belegt und die Variable für die Anzahl an Unterstützereinsätzen in dem betroffenen Takt wird um 1 erhöht (Zeilen 13 bis 15).

Algorithmus 6 : Berechnung der Werkerpositionen bei reaktivem Unterstützereinsatz

```

1 forall  $idx \in \mathcal{BWR}\mathcal{F}$  do
2   forall  $ap \in \mathcal{AP}$  do
3      $t := idx + st_{ap} - 1$ 
4     if  $t \in \mathcal{T}_{pz}$  then
5       set  $p \in \mathcal{P} \mid y_{p,idx}^{\mathcal{RF}} = 1$ 
6       if  $\tau_{p,ap} + \max[WV(idx, ap), VGU(idx, ap)] > (tz \cdot w_{ap}) + dbr_{ap}$  then
7          $u_{idx,ap}^{Einsatz} := 1$ 
8          $u^E := u^E + 1$ 
9          $u_{idx,ap}^{Zeit} := pos_{idx,ap} - (tz \cdot w_{ap}) + dbr_{ap}$ 
10         $i := 0$ 
11        while  $(i \cdot tz) < pos_{idx,ap}$  do
12           $k := idx + st_{ap} - 1 + i$ 
13          if  $(i + 1 \cdot tz) > tz \cdot w_{ap} + dbr_{ap}$  then
14             $u_{k,idx,ap}^{Belegt} := 1$ 
15             $u_k := u_k + 1$ 
16           $i := i + 1$ 
17         $pos_{idx,ap} := tz \cdot w_{ap} + dbr_{ap}$ 
18      else
19         $pos_{idx,ap} := \tau_{p,ap} + \max[WV(idx, ap), VGU(idx, ap)]$ 
20    else
21       $pos_{idx,ap} = 0$ 

```

Für die präventive Unterstützerbetrachtung wird die Berechnung in Algorithmus 7 durchgeführt. Der Algorithmus läuft ebenfalls über alle Reihenfolgepositionen und Arbeitsplätze (Zeile 1-2). In Zeile 3 wird der Takt bestimmt, in dem das Werkstück an dem aktuellen Arbeitsplatz bearbeitet wird. Findet dieser Takt innerhalb des zu bewertenden Produktionszeitraumes pz statt (Zeile 4), so wird die Berechnung durchgeführt. Ist dies nicht der Fall, wird die Werkerendposition gleich 0 gesetzt (Zeile 19). In Zeile 5 wird das Werkstück ermittelt, welches an der zu bewertenden Reihenfolgeposition eingeschleust wird. Im nächsten Schritt wird überprüft, ob ein Unterstützereinsatz an diesem Werkstück notwendig ist (Zeile 6). Ist dies der Fall, so werden von Zeile 7 bis 14 die Kennzahlen für die Unterstützereinsätze gepflegt. Dafür wird in Zeile 7 die Indikatorvariable für den Unterstützereinsatz gesetzt. In Zeile 8 wird die Einsatzanzahl hochgezählt und in Zeile 9 die Einsatzzeit des Unterstüt-

zers bestimmt. Von Zeile 10 bis 13 wird die Belegung des Unterstüters definiert sowie die Anzahl Einsätze je Takt hochgezählt. Die Werkerposition selbst ergibt sich dann ausschließlich durch das Maximum aus Werkerverzug und Vorgängerüberhang in Zeile 14. Wenn kein Unterstützereinsatz notwendig ist, wird in Zeile 17 die Werkerposition normal bestimmt.

Algorithmus 7 : Berechnung der Werkerpositionen bei präventivem Unterstützereinsatz

```

1 forall  $idx \in \mathcal{BWR}\mathcal{F}$  do
2   forall  $ap \in \mathcal{AP}$  do
3      $t := idx + st_{ap} - 1$ 
4     if  $t \in \mathcal{T}_{\mathcal{PZ}}$  then
5       set  $p \in \mathcal{P} \mid y_{p,idx}^{\mathcal{RF}} = 1$ 
6       if  $\max[WV(idx, ap), VGU(idx, ap)] + \tau_{idx,ap} - ((tz \cdot w_{ap}) + dbr_{ap}) > 0$  then
7          $u_{idx,ap}^{Einsatz} := 1$ 
8          $u^E := u^E + 1$ 
9          $u_{idx,ap}^{Zeit} := \tau_{p,ap}$ 
10         $i := 0$ 
11        while  $\tau_{p,ap} - (i \cdot tz) > 0$  do
12           $u_{t+i,idx,ap}^{Belegt} := 1$ 
13           $u_{t+1} := u_{t+1} + 1$ 
14           $i := i + 1$ 
15         $pos_{idx,ap} = \max[WV(idx, ap), VGU(idx, ap)]$ 
16      else
17         $pos_{idx,ap} := \tau_{idx,ap} + \max[WV(idx, ap), VGU(idx, ap)]$ 
18    else
19       $pos_{idx,ap} = 0$ 

```

In beiden Fällen, sowohl der präventiven als auch der reaktiven Unterstützerbetrachtung, werden die Funktionen WV für die Ermittlung des Werkerverzuges sowie VGU für die Ermittlung des Vorgängerüberhangs verwendet. Diese werden im Folgenden beschrieben.

In der Funktion WV wird der Werkerverzug in Zeile 1 durch die Endposition des Werkers am vorherigen Werkstück und der Kapazität am Arbeitsplatz ermittelt. Darauf folgend wird sichergestellt, dass der Werkerverzug nicht kleiner als die linke Driftgrenze werden kann (Zeile 2-3).

In der Funktion VGU wird der maximale gefundene Vorgängerüberhang im ersten Schritt gleich der linken Driftgrenze gesetzt. Daraufhin werden alle definieren Vorgänger durchlaufen (Zeile 1). Bei der Berechnung des Vorgängerüberhangs wird unterschieden, ob sich der Vorgängerarbeitsplatz in der gleichen Station befindet (Zeile 3-4), und wenn nicht, ob die Ansprechzone kleiner ist als die Ansprechzone des zu untersuchenden Arbeitsplatzes (Zeile 6-7) oder nicht (Zeile 8-9). Interessant für die Berechnung der Positionen ist nur der maximale

Funktion $WV(idx, ap)$

input : idx, ap **output** : $wv_{idx, ap}$

```

1  $wv_{idx, ap} := pos_{idx-w_{ap}, ap} - (tz \cdot w_{ap})$ 
2 if  $wv_{idx, ap} \leq -dbl_{ap}$  then
3    $wv_{idx, ap} := -dbl_{ap}$ 

```

Überhang, der durch Vorgänger erzeugt wird. Daher wird in den Zeilen 10 und 11 nach jedem Durchlauf der maximale Überhang aktualisiert.

Funktion $VGU(idx, ap)$

input : idx, ap **output** : $maxvu_{idx, ap}$

```

1  $maxvu_{idx, ap} := -dbl_{ap}$ 
2 forall  $vg \in \mathcal{VG}_{ap}$  do
3   if  $st_{vg} = st_{ap}$  then
4      $vu_{idx, ap} := pos_{idx, vg} - \tau_{idx, ap}$ 
5   else
6     if  $sp_{vg} < sp_{ap}$  then
7        $vu_{idx, ap} := pos_{idx, vg} - \tau_{idx, ap} - (st_{ap} - st_{vg}) \cdot tz$ 
8     else
9        $vu_{idx, ap} := pos_{idx, vg} - (st_{ap} - st_{vg}) \cdot tz$ 
10  if  $vu_{idx, ap} > maxvu_{idx, ap}$  then
11     $maxvu_{idx, ap} := vu_{idx, ap}$ 

```

5.3.3.4 Auswertung der Ergebnisse

Mit Hilfe der vorgestellten Algorithmen lassen sich die Positionen der Werker sowie die notwendigen Unterstützereinsätze realitätsnah bewerten. Daraus werden die notwendigen Einflussfaktoren u^{MAX} und u^E der Zielfunktion (5.47) ermittelt.

Die Informationen aus der Simulation lassen sich ebenfalls für eine Art Frühwarnsystem nutzen, indem die Zeitpunkte für notwendige Unterstützereinsätze abgeleitet werden. Diese Zeitpunkte können genutzt werden, um den Unterstützerarbeitskräften vor Produktionsbeginn als Arbeitsplan zu dienen. Dies ist insbesondere bei Einsatz des präventiven Unterstützereinsatzes notwendig.

5.3.4 Zweistufiger Optimierungsansatz zur Reihenfolgeplanung

5.3.4.1 Initialheuristik

Um die Komplexität des Problems beherrschbar zu machen, wird eine Aggregation vorgenommen. Hierfür werden die zu sequenzierenden Werkstücke im ersten Schritt nach ihrem Belastungsmuster geclustert. Die Repräsentanten der einzelnen Cluster werden im zweiten Schritt in eine optimale Reihenfolge gebracht. Diese Reihenfolge wird im dritten Teilschritt der Methode als Muster verwendet, indem in der ermittelten Reihenfolge reihum aus den Clustern Werkstücke „gepickt“ werden. Das Verfahren wird in Abbildung 5.32 grob dargestellt. Die Anforderungen und die Auswahl der einzelnen Teilmethoden werden im Folgenden diskutiert. Die Notation wird hierfür um die Bezeichner in Tabelle 5.22 erweitert.

Symbol	Definition
CL	Menge von Clustern CL_i
CL_i	Cluster $CL_i \in CL$
k	Parameter, der angibt, wie viele Cluster gebildet werden sollen
$\sigma_{p1,p2,ap}$	Abweichung der Bearbeitungszeiten der Werkstücke $p1 \in \mathcal{P}$ und $p2 \in \mathcal{P}$ an Arbeitsplatz $ap \in \mathcal{AP}$
Z	Menge von Clusterzentrioden z_i
z_i	Zentroid $z_i \in Z$ des Clusters $CL_i \in CL$

Tabelle 5.22: Notationserweiterung für die Reihenfolgeoptimierung

5.3.4.1.1 Clustering der Werkstücke Clustering ist zu verstehen als eine Methode zur Gruppierung von Objekten aus einer großen Menge hinsichtlich ihrer Ähnlichkeit.³²⁵ In unserem speziellen Fall wird das Clustering verwendet, um die Anzahl zu betrachtender Werkstücke im eigentlichen Sequenzierungsschritt zu reduzieren. Der Zentroid eines Clusters repräsentiert alle Aufträge innerhalb eines Clusters. Dafür ist eine möglichst homogene Füllung der Cluster notwendig. Um eine weitestgehend vollständige Abbildung aller Bereiche des Clusterraumes zu erreichen, muss zwischen den Clustern eine möglichst große Heterogenität herrschen. Zusätzlich zu diesen typischen Anforderungen an Clusterverfahren³²⁶ müssen in diesem speziellen Fall die Cluster annähernd gleich groß sein, damit eine Konstruktion der initialen Reihenfolge durch ein “Picking” aus den Clustern möglich wird.

Das Clustering ist als multivariates Verfahren in der Lage, Objekte mit einer Vielzahl von Merkmalen zu vergleichen. Hierfür wird, unabhängig vom Verfahren selbst, ein Ähnlichkeits-/Distanzmaß definiert.³²⁷ Jedes Werkstück wird als ein Vektor aus Bearbeitungszeiten verstanden.

³²⁵Vgl. [Bac94]

³²⁶Vgl. [BEPW06, S.489ff.]

³²⁷Vgl. [Kau96, S.440ff.]

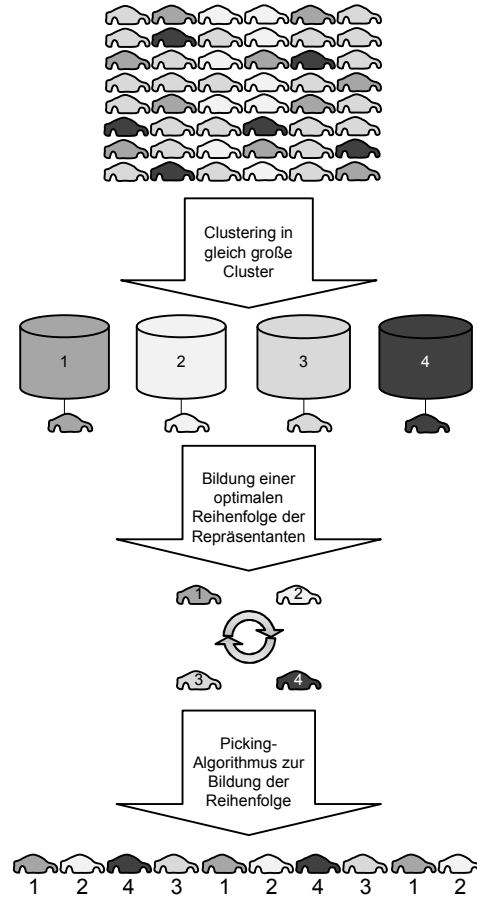


Abbildung 5.32: Vorgehen der dreiteiligen Initialheuristik

Die euklidische Distanz zweier Werkstücke ist definiert durch die Summe der absoluten Abweichungen in den einzelnen Bearbeitungszeiten (5.76).

$$\sum_{ap \in \mathcal{AP}} |\tau_{p1,ap} - \tau_{p2,ap}| \quad (5.76)$$

Um extreme Abweichungen stärker zu vermeiden, wird die Fehlerquadratsumme in Gleichung (5.77) als Distanzmaß verwendet.³²⁸

$$\sum_{ap \in \mathcal{AP}} (\tau_{p1,ap} - \tau_{p2,ap})^2 \quad (5.77)$$

Der Vektor aus Bearbeitungszeiten des Zentroiden eines Clusters z_i wird definiert durch die durchschnittliche Bearbeitungszeit der Werkstücke $p \in CL_i$ an jedem Arbeitsplatz (5.78).

³²⁸Vgl. [Bac94]

$$\tau_{z_i,ap} = \frac{\sum_{p \in CL_i} \tau_{p,ap}}{|CL_i|} \quad \forall \quad ap \in \mathcal{AP} \quad (5.78)$$

Das Clustering muss jedes Werkstück p einem Cluster CL_i zuweisen. Als Clustering-Verfahren wird ein Standard K-Means Algorithmus³²⁹ um problemspezifische Erweiterungen ergänzt. Das Standard-Vorgehen des K-Means Clustering beinhaltet folgende Schritte:³³⁰.

1. Zufällige Auswahl von k Startzentroiden $Z = \{z_1 \dots z_k\}$.
2. Für jedes $i \in \{1, \dots, k\}$ werden dem Cluster CL_i die Werkstücke aus $\mathcal{P}$ zugeordnet, welche näher an z_i als an allen anderen Zentren $z_j | j \neq i$ liegen.
3. Für jedes $i \in \{1, \dots, k\}$ wird das Zentrum neu definiert durch Gleichung (5.78).
4. Wiederhole Schritte 2 und 3 solange, bis sich die Zentren nicht mehr verändern.

Alternativ wird ein K-Means++ Algorithmus³³¹ verwendet, welcher statt einer zufälligen Auswahl der Startzentroiden (Random) eine explizite Auswahl dieser vornimmt, indem jeweils die entferntesten Bearbeitungszeitvektoren gewählt werden, um den Clusterraum möglichst gut abzudecken (K-Means++). Diese Zentrenauswahl zeigt in theoretischen Experimenten deutlich bessere Ergebnisse.³³²

Als dritte Möglichkeit wird eine Wahl der Startzentroiden anhand der summierten Fertigungszeit der Werkstücke durchgeführt. Geht man davon aus, dass ein Werkstück, das tendenziell eine hohe summierte Fertigungszeit zeigt, über alle Arbeitsplätze eine erhöhte Fertigungszeit aufweist, kann diese Zentrenauswahl eine homogene Clusterbildung begünstigen. Hierfür werden die Werkstücke nach ihrer summierten Fertigungszeit sortiert und dann absteigend in k gleich große Gruppen (± 1 Werkstück) geteilt. Aus diesen Gruppen wird jeweils das Zentrum definiert und als Startzentroid z_i für den K-Means verwendet. Dieses Verfahren wird im Folgenden „Sort&Group“ genannt.

Das Clusterverfahren muss für das anschließende „Picking“ annähernd gleich große Cluster erzeugen. Dies kann durch eine einfache Verhinderung der Zuweisung weiterer Werkstücke zu einem Cluster erreicht werden, wenn die Clustergröße bereits $\lceil \frac{\mathcal{P}}{k} \rceil$ erreicht hat. Durch die Verwendung dieser Beschränkung hat jedoch die Reihenfolge, in der die Werkstücke den Clustern zugewiesen werden, einen entscheidenden Einfluss. Daher werden zwei alternative Verfahren für die Reihenfolge verwendet.

1. Die Werkstücke werden den Clustern in zufälliger Reihenfolge zugewiesen. (Random)

³²⁹Vgl. [Ste56]

³³⁰Vgl. [AV06]

³³¹Vgl. [AV06]

³³²Vgl. [AV06, S.5]

2. Die Werkstücke werden vorher nach der summierten Fertigungszeit sortiert und in absteigender Reihenfolge zugewiesen. (Sort)

5.3.4.1.2 Sequenzierung der Cluster Im zweiten Schritt geht es darum, die Repräsentanten der gebildeten Cluster in eine optimale Reihenfolge zu bringen. Diese definiert, in welcher Reihenfolge im dritten Schritt Werkstücke aus den Clustern genommen werden. Demnach gilt es ein möglichst gutes Belastungsmuster zu bilden, welches immer wieder hintereinander gehängt ein gutes Ergebnis erzielt. Daher reicht es an dieser Stelle nicht, die Repräsentanten einmal zu bewerten. Hierdurch würden Reihenfolgen bevorzugt, in welchen die tendenziell bearbeitungszeitintensiven Werkstücke eine relativ späte Reihenfolgeposition einnehmen. Der Folgeeffekt auf die wiederholte Anwendung des Musters, welche im nachgelagerten Schritt durchgeführt wird, würde vernachlässigt. Daher werden die Repräsentanten der Cluster in der zu bewertenden Reihenfolge zweimal hintereinander gehängt, um auch die Endpositionen der Werker nach einmaliger Anwendung des Musters einzubeziehen. Dafür wird eine doppelte Zuweisung des gleichen Repräsentanten an zwei Reihenfolgepositionen zugelassen. Befindet sich der Repräsentant des dritten Clusters an der Reihenfolgeposition $idx = 4$, so wird er ebenfalls an der Reihenfolgeposition $idx = 4 + k$ zugewiesen (5.79).

$$y_{z_i, idx}^{\mathcal{RF}} = y_{z_i, idx+k}^{\mathcal{RF}} \quad \forall i \in \{1 \dots k\} \quad (5.79)$$

Es wird des Weiteren davon ausgegangen, dass alle Werker an ihrer linken Stationsgrenze beginnen können und sich vorher keine Werkstücke im Fließsystem befinden. Das Vorgehen der doppelten Sequenzbewertung wird in Abbildung 5.33 verdeutlicht. In dem Beispiel werden dadurch auch die Endpositionen nach dem Repräsentanten des dritten Clusters bewertet.

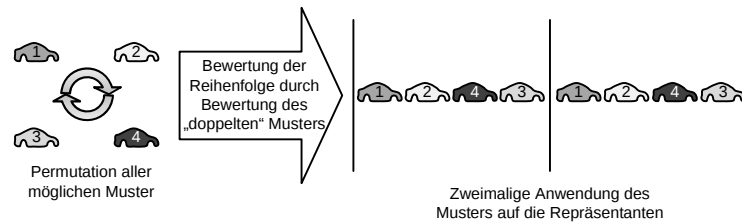


Abbildung 5.33: Vermeidung der Ausnutzung später Reihenfolgepositionen durch doppelte Bewertungssequenz

Auf Grund der stark reduzierten Anzahl Werkstücke können in diesem Schritt sämtliche möglichen Reihenfolgen mit Hilfe der Fließliniensimulation aus Abschnitt 5.3.3 bewertet werden. Die Vorgehensweise wird in der rekursiven Prozedur $Permut(i, \Phi, Z)$ aufgezeigt. Der Algorithmus ist im wesentlichen ein Permutationsalgorithmus, der um die Speicherung der besten Lösung erweitert wurde. Der Array Φ repräsentiert dabei die bisher beste Lösung. Beim Start

des Algorithmus beinhaltet Φ die durchnummerierten Reihenfolgepositionen $1..|Z|$. Die Menge Z beinhaltet alle Clusterrepräsentanten z_i . Grundprinzip des Algorithmus ist, dass alle möglichen Repräsentanten auf die momentan zu bestimmende Reihenfolgeposition i gesetzt und festgehalten werden (Zeile 6). Der Rest der Repräsentanten (Zeile 7) wird dann in einem weiteren rekursiven Aufruf der Prozedur zugeordnet (Zeile 8). Hierdurch werden alle möglichen Reihenfolgen gebildet. Wenn die Rekursion eine vollständige Reihenfolge gebildet hat und keine Repräsentanten mehr existieren, die noch nicht zugewiesen sind (Zeile 1), so wird die Lösung jeweils in ihrer Länge mit dem gleichen Muster verdoppelt und mit der ebenfalls verdoppelten bisherigen besten Lösung verglichen (Zeile 2). Für die Verdopplung des Musters wird die Funktion Double verwendet. Ist die neue Lösung besser als die alte, so wird diese als neue beste Lösung übernommen (Zeile 3).

Prozedur $\text{Permut}(i, \Phi, Z)$

input : i - Rekursionstiefe = Nächste zu fixierende Reihenfolgeposition

input : Φ - Bisher gebildete Reihenfolge bis Position $i - 1$

input : Z - Menge von Clusterrepräsentanten z_j

output : Ω - Reihenfolge mit der minimalen Anzahl notwendiger Unterstützereinsätze

```

1 if  $Z = \{ \}$  then
2    $\text{Simulation}(\Phi)$ 
3   if  $\text{SprAnz}(\text{Double}(\Phi)) < \text{SprAnz}(\text{Double}(\Omega))$  then
4      $\Omega := \Phi$ 
5 else
6   forall  $z_j \in Z$  do
7      $\Phi[i] := z_j$ 
8      $\tilde{Z} := Z \setminus z_j$ 
9      $\text{Permut}(i + 1, \Phi, \tilde{Z})$ 

```

Funktion $\text{Double}(\Omega)$

input : Ω - Zu verdoppelnde Reihenfolge

output : $\bar{\Omega}$ - Verdoppelte Reihenfolge

```

1  $\bar{\Omega} := \Omega$ 
2 for  $i \leftarrow 1$  to  $|\Omega|$  do
3    $\bar{\Omega}[i + \alpha] := \Omega[i]$ 

```

Ausschlaggebendes Zielkriterium für die Auswahl der optimalen Reihenfolge ist die Anzahl notwendiger Unterstützereinsätze. Diese wird zurückgegeben durch den Aufruf von $\text{SprAnz}()$. In diesem Schritt wird noch kein Fokus auf die gleichzeitige Anzahl Unterstützereinsätze gelegt, da eine korrekte Bewertung nur auf Basis der Clusterrepräsentanten nicht möglich ist. Eine Initiallösung mit einer minimalen Anzahl an Unterstützereinsätzen stellt jedoch die beste Voraussetzung für eine Minimierung der gleichzeitigen Unterstützereinsätze in der späteren Verbesserungsheuristik dar.

5.3.4.1.3 Picking-Algorithmus zur Konstruktion der Reihenfolge Im letzten Schritt der Initialheuristik werden die einzelnen Werkstücke aus den Clustern „gepickt“. Dabei steht die Reihenfolge der Cluster, aus denen gepickt wird, bereits durch die Sequenzierung der Repräsentanten fest. Jedoch muss in jedem Pick-Vorgang noch entschieden werden, welches Werkstück aus dem Cluster gewählt wird. Um das optimale Reihenfolgemuster nicht zu zerstören, wird dies mit Hilfe einer Funktion $Zf(CL_i)$ realisiert, die ein zufällig gewähltes Werkstück aus einem Cluster zurückgibt. Das zufällige Picking wird z mal durchgeführt und die beste Lösung, gemessen an der Anzahl an notwendigen Unterstützereinsätzen u^E , als initiale Reihenfolge zurückgegeben. Unter der Voraussetzung, dass die Cluster gleich groß sind, bis auf eine Abweichung von ± 1 , gestaltet sich der Picking-Algorithmus wie in Algorithmus (12) dargestellt. Zuerst wird in Zeile 1 der Vergleichswert für die beste Lösung auf den höchstmöglichen Wert gesetzt. Von Zeile 2 bis 17 wird z mal das Picking durchgeführt. Jedes Mal wird dafür in Zeile 3 die Clustervariable CL neu mit den ursprünglich gebildeten Clustern C initialisiert. Begonnen wird dann beim ersten Cluster (Zeile 4) und alle Reihenfolgepositionen werden durchlaufen (Zeile 5). Wenn der Cluster noch nicht leer ist (Zeile 6), wird ein zufälliges Werkstück aus dem Cluster gewählt, aus dem Cluster entfernt und der aktuellen Reihenfolgeposition zugewiesen (Zeilen 7-9). In den Zeilen (10-13) wird der nächste Cluster ermittelt. Wenn die Laufvariable i beim letzten Cluster angekommen ist (Zeile 10), ist der nächste Cluster wieder der erste (Zeile 11). In Zeile 14 wird die aktuelle Reihenfolge Φ aufgebaut. In den Zeilen 15-17 wird die aktuell gebildete Reihenfolge bewertet und die Lösung zwischengespeichert (Zeile 17), falls sie weniger Unterstützereinsätze erzeugt als die beste bisher gefundene Lösung Ω (Zeile 16).

5.3.4.2 Iteratives Verbesserungsverfahren

Die Reihenfolge, die mit Hilfe der Initialheuristik erstellt wurde, bietet die Grundlage für eine explizite Minimierung der gleichzeitigen Unterstützereinsätze mittels eines problemspezifischen iterativen Verbesserungsverfahrens. Die Grundidee des Verfahrens basiert darauf, gezielt die Unterstützereinsätze zu vermeiden, welche zu einer Erhöhung der gleichzeitigen Unterstützereinsätze führen. Die Methode verwendet die Fließliniensimulation, um eine Bewertung der initialen Reihenfolge vorzunehmen. Basierend auf den Ergebnissen werden Vertauschungen von Werkstücken durchgeführt, die vielversprechend für die Verbesserung des Zielfunktionswertes sind. Im ersten Schritt wird beschrieben, wie die Auswahl von zu tauschenden Werkstücken durchgeführt wird, um im zweiten Schritt potentielle Tauschpartner zu identifizieren. Zuletzt wird die Suchstrategie erläutert.

5.3.4.2.1 Identifikation problematischer Anordnungen Die maximale Anzahl gleichzeitiger Unterstützereinsätze definiert die notwendige Anzahl an Unterstützereinsatzkräften.

Algorithmus 12 : Picking Algorithmus zur Bildung der Initialreihenfolge

```

input  :  $z$  - Anzahl Durchläufe
input  :  $C$  - Menge von Clustern  $CL_i \in C$  mit Werkstücken
output :  $\Omega$  - Beste initiale Reihenfolge
1  $u^E(\Omega) := \infty$ 
2 for  $a \leftarrow 1$  to  $|z|$  do
3    $CL = C$ 
4    $i := 1$ 
5   forall  $idx \in \mathcal{RF}_{pz}$  do
6     if  $CL_i \neq \{ \}$  then
7        $p := Zf(CL_i)$ 
8        $CL_i := CL_i \setminus p$ 
9        $y_{p,idx}^{\mathcal{RF}} := 1$ 
10    if  $i = k$  then
11       $i := 1$ 
12    else
13       $j := i + 1$ 
14     $\Phi[idx] := p \mid y_{p,idx}^{\mathcal{RF}}$ 
15   $\text{Simulation}(\Phi)$ 
16  if  $u^E(\Phi) < u^E(\Omega)$  then
17     $\Omega := \Phi$ 

```

Selbst wenn diese nur in einem Takt im gesamten Planungszeitraum auftreten, müssen für diesen Fall genügend Unterstützereinsatzkräfte vorgehalten werden. Daher bringt lediglich die Vermeidung eines Unterstützereinsatzes, der innerhalb des entsprechenden Taktes stattfindet, die gewünschte Minimierung der Zielgröße. In der Abbildung 5.34 wird diese Tatsache verdeutlicht. Insgesamt finden im betrachteten Planungszeitraum sechs Unterstützereinsätze statt. Drei dieser Einsätze finden im selben Takt $t = 103$ statt, wodurch drei Unterstützerarbeitskräfte vorgehalten werden müssen. Die Reduzierung eines Unterstützereinsatzes innerhalb dieses Taktes wird direkt eine Verringerung der Anzahl notwendiger Unterstützer mit sich bringen. Hierfür werden die verantwortlichen Takte durch Gleichung 5.80 ermittelt.

$$t \in \mathcal{T} \quad | \quad \max[u_t] \quad (5.80)$$

Jedes Werkstück, das für eine Unterstützerbelegung in dem entsprechenden Takt t sorgt, kann für einen Tauschvorgang betrachtet werden. Der Unterstützereinsatz muss dabei nicht erst in dem entsprechenden Takt beginnen. Ebenso kann es sich dabei um ein Werkstück handeln, das über mehrere Takte hinweg von einem Unterstützer bearbeitet wird, wie in Abschnitt 5.3.1.3 definiert. In dem Beispiel 5.34 ist dies in Takt $t = 103$ der Fall. Der Unterstützereinsatz an dem Werkstück an Reihenfolgeposition $idx = 101$ beginnt bereits in Takt $t = 101$,

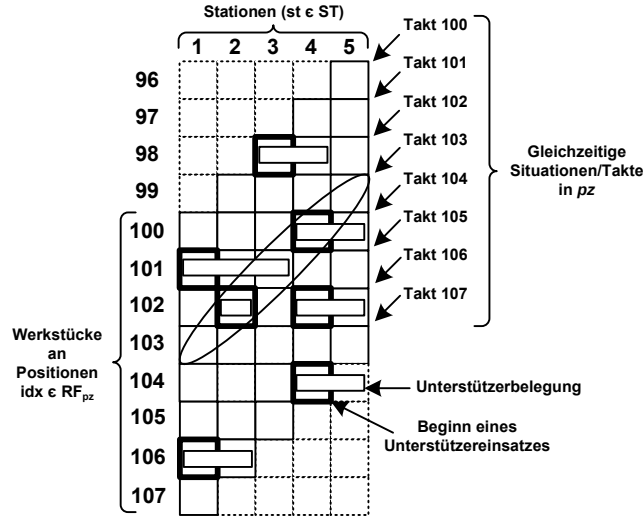


Abbildung 5.34: Identifikation von zu vermeidenden Unterstützereinsätzen

der Unterstützer ist aber bis zum Takt $t = 103$ belegt. Daher muss im ersten Schritt das verantwortliche Werkstück über die Unterstützerbelegungsvariable $u_{t,idx,ap}^{Belegt}$ gewählt werden (5.81). In Abbildung 5.34 sind dies die fett umrandeten Felder, welche einen beginnenden Unterstützereinsatz darstellen.

$$idx \in \mathcal{RF} \quad | \quad u_{t,idx,ap}^{Belegt} \quad (5.81)$$

Es können mehrere Takte existieren, die die maximale Anzahl an gleichzeitigen Unterstützereinsätzen notwendig machen. Eine Reduzierung der Zielgröße wird erst erreicht, wenn in allen entsprechenden Takten jeweils ein Unterstützereinsatz vermieden wurde. Die Auswahl des ersten zu bearbeitenden Taktes wird dabei zufällig vorgenommen.

5.3.4.2.2 Auswahl von Tauschpartnern Nachdem ein Werkstück $p1$ an einer Reihenfolgeposition $idx1$ zum Tausch ausgewählt wurde, muss ein passender Tauschpartner $p2$ identifiziert werden. Um einen Unterstützereinsatz zu vermeiden, der an Arbeitsplatz ap durch ein Werkstück $p1$ hervorgerufen wurde, muss der Tauschpartner $p2$ Bedingung 5.82 erfüllen.

$$\tau_{p2,ap} + pos_{idx1-w_{ap,ap}} \leq (w_{ap} \cdot tz) + dbr_{ap} \quad (5.82)$$

Zusätzlich zu dieser Bedingung sollte das Werkstück $p2$ hinsichtlich der restlichen Bearbeitungszeiten an den anderen Arbeitsplätzen möglichst ähnlich zu Werkstück $p1$ sein. Dies sorgt dafür, dass die vorher gültige Berechnung der Werkerpositionen durch den Tauschvorgang der beiden Werkstücke möglichst wenig beeinflusst wird. Je stärker die Abweichungen in

den Bearbeitungszeiten der beiden Werkstücke ist, desto eher muss damit gerechnet werden, dass die Berechnungen der Werkerpositionen ein falsches Bild widerspiegeln. Mit Rücksicht auf die Laufzeit wird es so möglich, mehrere Tauschvorgänge durchzuführen ohne jedes Mal eine vollständige Neubewertung der Positionen notwendig zu machen. Außerdem kann durch diese Bedingung vermieden werden, dass neue Unterstützereinsätze an unbekannten Stellen hervorgerufen werden. Die Abweichung der beiden Werkstücke $p1$ und $p2$ ohne Berücksichtigung des Arbeitsplatzes ap , an welchem der Unterstützereinsatz vermieden wurde, wird in Gleichung 5.83 definiert.

$$\sigma_{p1,p2,ap} = \sum_{i \in \mathcal{AP} | i \neq ap} |\tau_{p1,i} - \tau_{p2,i}| \quad (5.83)$$

Wurden Aufträge getauscht, werden die entsprechenden Reihenfolgepositionen für eine bestimmte Anzahl an Iterationen d fixiert, um zu verhindern, dass der Algorithmus die selben Positionen wieder vertauscht. Die Auswahl des optimalen Tauschpartners wird in der Funktion $TP(idx, ap, i, d)$ durchgeführt. Dafür wird zu Beginn die Menge aller Arbeitsplätze ohne den Arbeitsplatz, an dem der Unterstützereinsatz vermieden werden soll, definiert (Zeile 1). In Zeile 2 wird das passende Werkstück für die zu tauschende Reihenfolgeposition herausgesucht. Im nächsten Schritt werden alle Reihenfolgepositionen durchlaufen (Zeile 3). Für jede dieser Positionen wird jeweils das zugehörige Werkstück herausgesucht (Zeile 4). In Zeile 5 wird überprüft, ob der potentielle Tauschpartner an dem betroffenen Arbeitsplatz ap einen Unterstützereinsatz vermeiden würde und in den letzten d Durchläufen nicht in einen Tausch involviert war. Für alle Werkstücke, für die dies der Fall ist, werden die Abweichungen der Bearbeitungszeiten über alle Arbeitsplätze aufsummiert (Zeile 7). Dabei wird der Arbeitsplatz, an dem der Unterstützereinsatz vermieden werden soll, nicht mit betrachtet, da die Abweichung in der Bearbeitungszeit an dieser Stelle gewollt ist (Zeile 6). Falls das Werkstück keinen Unterstützereinsatz vermeiden würde, wird die Abweichung auf den höchstmöglichen Wert gesetzt (Zeile 8-9). In Zeile 10 wird der Tauschpartner mit der geringsten Abweichung ausgewählt. Dies ist das Werkstück, welches an Arbeitsplatz ap den Unterstützereinsatz wie gewünscht vermeidet, dem ursprünglichen Werkstück jedoch möglichst ähnlich ist.

Bezogen auf das Beispiel aus Abbildung 5.34 wird in Grafik 5.35 die Auswahl eines Tauschpartners dargestellt. Takt $t = 103$ wurde ausgewählt, da hier die maximale Anzahl an gleichzeitigen Unterstützereinsätzen auftritt. Demnach wird Reihenfolgeposition $idx = 100$ und damit das dazugehörige Werkstück p als einer der Werkstücke, die einen Unterstützer im entsprechenden Takt belegen, identifiziert. Für dieses Werkstück werden alle möglichen Tauschpartner im Planungszeitraum untersucht. Das Werkstück mit der Reihenfolgeposition $idx = 103$ würde den zu vermeidenden Unterstützereinsatz verhindern und besitzt die geringste Abweichung vom zu tauschenden Werkstück. Daher wird es ausgewählt und die Werkstücke werden getauscht.

Funktion $TP(idx, ap, i, d)$

input : idx - Zu tauschendes Werkstück
input : ap - Arbeitsplatz, an dem der Unterstützereinsatz vermieden werden soll
input : i - Aktuelle Iteration
input : d - Anzahl an Iterationen, für die vertauschte Reihenfolgepositionen gesperrt sind
output : $i\tilde{dx}$ - Optimaler Tauschpartner

```

1  $\tilde{\mathcal{AP}} := \mathcal{AP} \setminus ap$ 
2 set  $p \in \mathcal{P} \mid y_{p,idx}^{\mathcal{RF}} = 1$ 
3 forall  $i\tilde{dx} \in \mathcal{RF}_{pz}$  do
4   set  $\tilde{p} \in \mathcal{P} \mid y_{\tilde{p},i\tilde{dx}}^{\mathcal{RF}} = 1$ 
5   if  $(\tau_{\tilde{p},ap} + \max[WV(idx, ap), VGU(idx, ap)] \leq (w_{ap} \cdot tz) + dbr_{ap}) \ \& \ (iter_{i\tilde{dx}} + d < i)$ 
6     then
7       forall  $\tilde{ap} \in \tilde{\mathcal{AP}}$  do
8          $\sigma_{i\tilde{dx}} := \sigma_{i\tilde{dx}} + |\tau_{\tilde{p},ap} - \tau_{p,ap}|$ 
9       else
10         $\sigma_{i\tilde{dx}} := \infty$ 
11  $i\tilde{dx} := \min_{i\tilde{dx} \in \mathcal{RF}} [\sigma_{i\tilde{dx}}]$ 

```

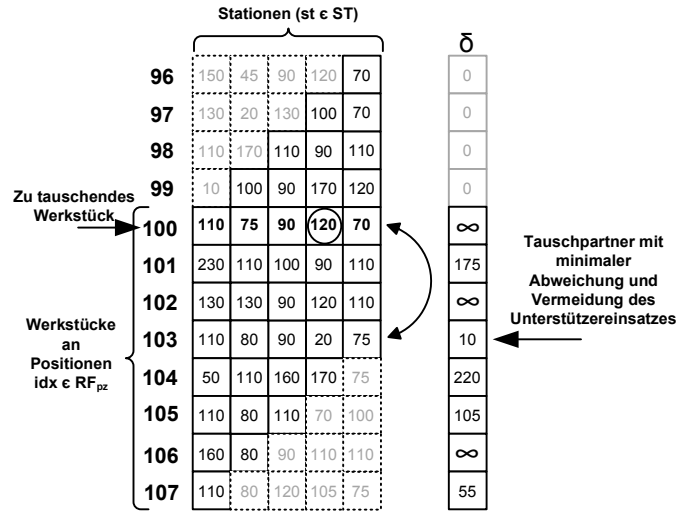


Abbildung 5.35: Identifikation potentieller Tauschpartner

5.3.4.2.3 Suchstrategie Für die Steuerung des Suchprozesses wird im Folgenden der Hauptalgorithmus des iterativen Verbesserungsverfahrens vorgestellt. Das Verfahren basiert dabei auf dem in 5.3.4.2.1 beschriebenen Vorgehen zur Erkennung von zu tauschenden Werkstücken sowie der Methode zur Auswahl des Tauschpartners im letzten Abschnitt 5.3.4.2.2. Um eine Bewertung nach jedem einzelnen Tauschvorgang mit Rücksicht auf die Laufzeit des Algorithmus zu verhindern, wird der Schwellenwert δ eingeführt. Bei jedem Tauschvorgang wird gemessen, wie stark die Werkstücke hinsichtlich ihrer Bearbeitungszeiten voneinander ab-

weichen. Ein Tauschvorgang hat nicht nur auf die beiden betroffenen Werkstücke Einfluss, sondern ändert auch die dahinter kalkulierten Werkerpositionen.³³³ Je größer die Abweichung in den Bearbeitungszeiten der beiden Werkstücke ist, desto mehr ist davon auszugehen, dass die ermittelten Werkerpositionen der Folgewerkstücke und damit auch die Informationen über Unterstützereinsätze nicht mehr korrekt sind. Wird der Schwellenwert überschritten, muss somit ein neuer Bewertungsvorgang initiiert werden. Das Vorgehen wird in Algorithmus 14 beschrieben. Zu Beginn wird die Reihenfolge aus den Binärvariablen zur Zuweisung eines Werkstückes zu einer Reihenfolgeposition gebildet (Zeile 1-2) und jeweils die letzte Iteration, in der eine Reihenfolgeposition in einen Tausch involviert war, mit $-d$ initialisiert (Zeile 3). In Zeile 4 wird die initiale Reihenfolge mit Hilfe der Fließliniensimulation aus Abschnitt 5.3.3 bewertet. Das Verfahren wird solange durchgeführt, bis über eine feste Anzahl von Iterationen z keine bessere Lösung mehr gefunden wurde (Zeile 6). In den Zeilen 7 und 8 wird die Variable für die summierte Abweichung in den Bearbeitungszeiten vertauschter Aufträge σ initialisiert und die Laufvariable für die aktuelle Iteration i hochgezählt. Die Repeat-Schleife von Zeile 9 bis 20 vertauscht so lange Werkstücke, bis die angegebene maximale summierte Abweichung der Bearbeitungszeiten δ überschritten wird. Dafür wird in Zeile 10 zufällig einer der Takte mit der maximalen Anzahl an gleichzeitigen Unterstützereinsätzen ausgewählt. In Zeile 11 wird zufällig eines der Werkstücke herausgesucht, das für eine Unterstützerbelegung in Takt t sorgt. In Zeile 12 wird der am besten geeignete Tauschpartner mit der Funktion TP gewählt.³³⁴ In den Zeilen 13 bis 15 wird die Vertauschung der beiden Werkstücke durchgeführt. In Zeile 16 wird für die durch den Tausch betroffenen Reihenfolgepositionen die letzte Iteration, in der sie getauscht wurden $iter_{idx}$, aktualisiert. In den Zeilen 17 bis 19 wird die summierte Abweichung der Bearbeitungszeiten σ berechnet. Wird die Repeat-Schleife verlassen, weil die Abweichungen größer geworden sind als die maximal erlaubte Abweichung (Zeile 20), wird die Reihenfolge aus den Binärvariablen gebildet (Zeile 21-22) und bewertet (Zeile 23). In den Zeilen 24 bis 30 wird die aktuelle Lösung mit der besten Lösung verglichen. Falls die aktuelle Lösung besser hinsichtlich der Anzahl notwendiger Unterstützer ist (Zeile 24) oder aber hinsichtlich der Anzahl Unterstützereinsätze besser und hinsichtlich der gleichzeitigen Unterstützer gleich gut ist (Zeile 28), so wird die aktuelle Lösung die beste Lösung (Zeile 25 und 29). Falls eine bessere Lösung gefunden wurde, wird das Abbruchkriterium des Verfahrens, welches die Anzahl an Durchläufen ohne Verbesserung zählt, wieder auf null gesetzt (Zeile 26 und 30).

³³³Die Auswahl des Tauschpartners wird aus diesem Grund bewusst nach der Prämisse, möglichst geringe Abweichung zu generieren, durchgeführt. Vgl. Abschnitt 5.3.4.2.2

³³⁴Vgl. Abschnitt 5.3.4.2.2

Algorithmus 14 : Iteratives Verbesserungsverfahren

input : δ - Maximale Abweichung der Bearbeitungszeiten, bevor neu bewertet wird
input : z - Maximale Anzahl an Durchläufen ohne Verbesserung
input : d - Anzahl an Iterationen, für die eine getauschte Reihenfolgeposition gesperrt ist
input : $\mathcal{RF}_{pz}$ - Initiale Reihenfolgepositionen
output : Ω - Verbesserte Reihenfolge

```

1 forall  $idx \in \mathcal{RF}_{pz}$  do
2    $\Omega[idx] := p \mid y_{p,idx}^{\mathcal{RF}}$ 
3    $iter_{idx} := -d$ 
4 Simulation( $\Omega$ )
5  $i := 0$ 
6 for  $a \leftarrow 0$  to  $z$  do
7    $\sigma := 0$ 
8    $i := i + 1$ 
9   repeat
10    set  $t = Zf(t \in \mathcal{T} \mid \max[u_t])$ 
11    set  $idx = Zf(idx \in \mathcal{RF}_{pz} \mid u_{t,idx,ap}^{Belegt})$ 
12     $\tilde{idx} := TP(idx, ap, i, d)$ 
13    set  $p \in \mathcal{P} \mid y_{p,idx}^{\mathcal{RF}} = 1$ 
14    set  $\tilde{p} \in \mathcal{P} \mid y_{\tilde{p},\tilde{idx}}^{\mathcal{RF}} = 1$ 
15     $y_{p,idx}^{\mathcal{RF}} := 0 \wedge y_{p,\tilde{idx}}^{\mathcal{RF}} := 1 \wedge y_{\tilde{p},idx}^{\mathcal{RF}} := 0 \wedge y_{\tilde{p},\tilde{idx}}^{\mathcal{RF}} := 1$ 
16     $iter_{idx} := i \wedge iter_{\tilde{idx}} := i$ 
17    forall  $\tilde{ap} \in \mathcal{AP} \setminus ap$  do
18       $\sigma_{\tilde{idx}} := \sigma_{\tilde{idx}} + |\tau_{\tilde{p},\tilde{ap}} - \tau_{p,\tilde{ap}}|$ 
19     $\sigma := \sigma + \sigma_{\tilde{idx}}$ 
20  until  $\sigma > \delta$ 
21  forall  $idx \in \mathcal{RF}_{pz}$  do
22     $\Phi[idx] := p \mid y_{p,idx}^{\mathcal{RF}}$ 
23  Simulation( $\Phi$ )
24  if  $u^{MAX}(\Phi) < u^{MAX}(\Omega)$  then
25     $\Omega := \Phi$ 
26     $a := 0$ 
27  else
28    if  $(u^{MAX}(\Phi) = u^{MAX}(\Omega)) \ \& \ (u^E(\Phi) < u^E(\Omega))$  then
29       $\Omega := \Phi$ 
30       $a := 0$ 

```

6 Validierung und Ergebnisse

Es ist nicht genug zu wissen,
man muss auch anwenden;
es ist nicht genug zu wollen,
man muss auch tun.

(Johann Wolfgang v. Goethe)

Die Validierung der entwickelten Methoden wird anhand von praxisnahen Daten durchgeführt, die im Rahmen dieser Arbeit anonymisiert und normiert wurden.

6.1 Validierung der Entscheidungsunterstützung zur Rekonfiguration

6.1.1 Testszenario

Zur Validierung und Bewertung wird das entwickelte Entscheidungsunterstützungssystem in der Praxis angewendet. Dies ist notwendig, da der Input des Planers zwingend für den Prozess erforderlich ist. Als Praxisbeispiel werden zwei unterschiedliche Endmontagelinien bei einem großen deutschen Automobilhersteller betrachtet und jeweils in einem Workshop optimiert. Zur Normierung der Daten wurde jeder Wert durch die Takteinheit te dividiert. Die Begebenheiten entsprechen den in Abschnitt 2 angenommenen Voraussetzungen. Vor beiden Workshops hat bereits eine Optimierung der Arbeitszuordnung mit traditionellen Methoden stattgefunden. Dabei haben die Planer ausschließlich die Durchschnittsauslastungen auf den Arbeitsplätzen betrachtet und diese „repariert“ sowie einen „Komprimierungsvorgang“ durchgeführt.³³⁵ Ein weiterer Handlungsbedarf für diese beiden Planungsschritte leitet sich nicht mehr ab.³³⁶ Daher wird im Rahmen der Validierung der Fokus auf den entscheidenden Antizipationsschritt gelegt. Implizit wird hierdurch das Ergebnis der traditionellen und der in

³³⁵Vgl. Abschnitt 5.1.3.1.1

³³⁶Vgl. Kennzahlensystem für den Handlungsbedarf in Abschnitt 5.1.3.1.1

dieser Arbeit vorgestellten Methode verglichen. Das Vorgehen im konkreten Praxisfall wird anhand von zwei Verschiebungen verdeutlicht.

Zur Bewertung der Ergebnisse wird unter anderem die in Abschnitt 5.3.3 vorgestellte Fließliniensimulation genutzt. Somit können die Kennzahlen vor und nach den Workshops verglichen sowie eine Quantifizierung der Auswirkungen auf die Betriebskosten mit Hilfe der Simulation durchgeführt werden. Zur Quantifizierung werden die Aufträge des zu planenden Produktionszeitraumes in 100 verschiedenen, zufällig ausgewählten Produktionsreihenfolgen simuliert. So ist gewährleistet, dass eine reihenfolgeunabhängige Bewertung durchgeführt wird.

Darüber hinaus werden die Kennzahlen zur Auswahl zu betrachtender Arbeitsplätze ebenfalls mit Hilfe der Fließliniensimulation bewertet.³³⁷ Hierfür wird eine ausreichend große Anzahl an Aufträgen aus der Vergangenheit verwendet, um eine Korrelationsanalyse zwischen den Kennzahlen und der hervorgerufenen Anzahl an Unterstützereinsätzen durchzuführen.

6.1.2 Ergebnisse

6.1.2.1 Kennzahlensystem zur Auswahl des Betrachtungsbereiches

Für die Korrelationsanalyse der Anzahl an Unterstützereinsätzen und der unterschiedlichen Kennzahlen zur Auswahl des Betrachtungsraumes werden 8900 Aufträge verwendet, die in der Vergangenheit produziert wurden. Mit Hilfe dieser Aufträge wird eine Simulation zur Bestimmung der Unterstützereinsätze je Arbeitsplatz durchgeführt und die jeweiligen Kennzahlen berechnet. Darauf folgt eine Korrelationsanalyse der Anzahl an Unterstützereinsätzen und jeder einzelnen Kennzahl. Die Definition der Kennzahlen wurde in Abschnitt 5.1.3.1.2 in Tabelle 5.4 durchgeführt. Für die Korrelationsanalyse wird der empirische Korrelationskoeffizient, definiert in Gleichung 6.1, verwendet.³³⁸

$$COR(X, Y) = \frac{\frac{1}{AP} \sum_{ap=1}^{AP} (u_{ap}^E - \overline{u_{ap}^E}) \cdot (\gamma_{ap} - \overline{\gamma_{ap}})}{\sqrt{\frac{1}{AP} \sum_{ap=1}^{AP} (u_{ap}^E - \overline{u_{ap}^E})^2} \cdot \sqrt{\frac{1}{AP} \sum_{ap=1}^{AP} (\gamma_{ap} - \overline{\gamma_{ap}})^2}} \quad (6.1)$$

Das aggregierte Ergebnis ist in Tabelle 6.1 dargestellt.

Die Analyse für die Kennzahl $\rho_{ap}^{QuadPos}$ zeigt den höchsten Korrelationskoeffizienten und damit die beste Eignung, potentiell zu bearbeitende Arbeitsplätze zu identifizieren.

³³⁷Vgl. Abschnitt 5.1.3.1.2

³³⁸Vgl. [Har99, S.561ff.]

Kennzahl γ	$COR(\gamma, u^E)$
Δ_{ap}^{LinPos}	0,699
ρ_{ap}^{LinPos}	0,449
$\Delta_{ap}^{QuadPos}$	0,560
$\rho_{ap}^{QuadPos}$	0,784

Tabelle 6.1: Ergebnis der Korrelationsanalyse für das Kennzahlensystem zur Auswahl des Betrachtungsbereiches

6.1.2.2 Anwendungsbeispiele

Im ersten Anwendungsbeispiel wird ein Abschnitt eines Montagebandes mit 107 Arbeitsplätzen betrachtet. Als Auftragsprogramm stehen für den kommenden Planungszeitraum 5347 Aufträge zur Verfügung, die zur Ermittlung der Bearbeitungszeiten auf den Arbeitsplätzen verwendet werden. Basierend auf den Empfehlungen und Regeln in Abschnitt 5.1.3.1.2 werden die beiden Arbeitsplätze in Abbildung 6.1 zur Bearbeitung ausgewählt. Die durchschnittlichen Bearbeitungszeiten auf den Arbeitsplätzen 1 und 2 sind 86% und 84%. Wie der Abbildung 6.1 zu entnehmen ist, treten auf Arbeitsplatz 1 etwa 600 Bearbeitungszeiten oberhalb der Taktzeit auf. Die detaillierten Kennzahlen können der Tabelle 6.2 entnommen werden.

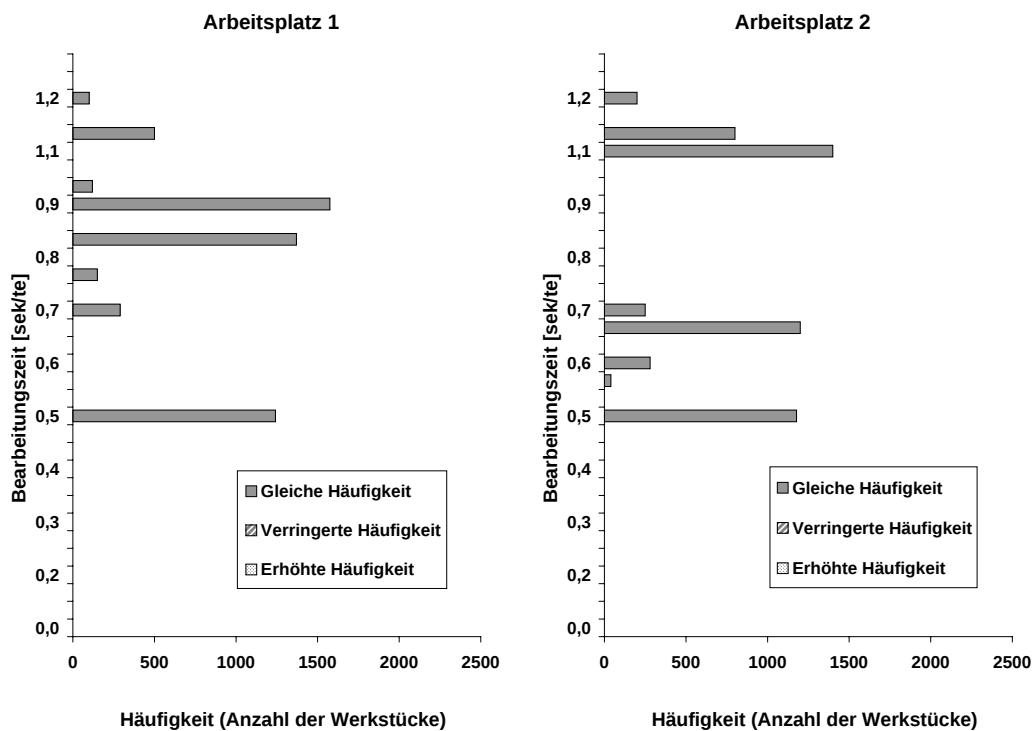


Abbildung 6.1: Zwei ausgewählte Arbeitsplätze zur Rekonfiguration

Arbeitsplätze	μ_{ap}	$\frac{\Delta_{LinPos}}{te}$	$\frac{\Delta_{QuadPos}}{te}$
Arbeitsplatz 1	86%	$0,015 \frac{sek}{te}$	$0,160 \frac{sek^2}{te}$
Arbeitsplatz 2	84%	$0,045 \frac{sek}{te}$	$0,370 \frac{sek^2}{te}$
$\emptyset / \sum$	85%	$0,060 \frac{sek}{te}$	$0,530 \frac{sek^2}{te}$

Tabelle 6.2: Kennzahlen für die ausgewählten Arbeitsplätze vor der Bearbeitung

Auf Basis dieser Daten wird nun ein Arbeitsvorgang gesucht, der die Übertaktzeit reduziert. Dafür wird die AIA-Kennzahl für alle Arbeitsvorgänge auf beiden Arbeitsplätzen berechnet.³³⁹ Ein Arbeitsvorgang mit einer Baurate von 43% mit $0,215 \frac{sek}{te}$ Bearbeitungszeit und einer errechneten AIA-Kennzahl von 1,013 wird gewählt. Auf dem Zielarbeitsplatz 2 hat der Arbeitsvorgang nur eine AIA von 0,856. Daraus folgt, dass durch eine Verschiebung des Arbeitsvorganges eine Reduzierung der Übertaktzeit auf Arbeitsplatz 1 erreicht wird und nur wenig bis keine Überlastung auf Arbeitsplatz 2 aufgebaut wird. Betrachtet man das Resultat mit Hilfe der Visualisierung aus Abschnitt 5.1.3.2.1, so wird die Übertaktzeit durch die Verschiebung auf Arbeitsplatz 1 vollständig beseitigt und keine Übertaktzeit auf Arbeitsplatz 2 aufgebaut. Dies bedeutet, dass der Arbeitsvorgang nur mit Arbeitsvorgängen auf Arbeitsplatz 2 korreliert, die in relativ niedrig auslastenden Aufträgen enthalten sind.³⁴⁰ Das Ergebnis ist in Abbildung 6.2 dargestellt.

Nach der Verschiebung beträgt die durchschnittliche Auslastung auf den beiden Arbeitsplätzen $\mu_{ap1} = 93\%$ und $\mu_{ap2} = 77\%$. Um die Differenz in den Durchschnittsauslastungen zu vermeiden, wird eine Verschiebung eines Arbeitsvorganges von Arbeitsplatz 2 zu 1 benötigt. Dafür wird ein Arbeitsvorgang mit einer Bearbeitungszeit von $0,114 \frac{sek}{te}$ und einer Baurate von 23% ausgewählt. Die AIA-Kennzahl beträgt 1.103 auf seinem Arbeitsplatz 2 und 0,767 auf dem alternativen Arbeitsplatz 1. Die Verschiebung des Arbeitsvorganges sorgt für eine deutliche Reduzierung der Übertaktzeit auf Arbeitsplatz 2 und erzeugt keine neue Übertaktzeit auf Arbeitsplatz 1. Das detaillierte Ergebnis ist in Abbildung 6.3 und Tabelle 6.3 dargestellt.

Arbeitsplätze	μ_{ap}	$\frac{\Delta_{LinPos}}{te}$	$\frac{\Delta_{QuadPos}}{te}$
Arbeitsplatz 1	86%	$0 \frac{sek}{te}$	$0 \frac{sek^2}{te}$
Arbeitsplatz 2	84%	$0,024 \frac{sek}{te}$	$0,159 \frac{sek^2}{te}$
$\emptyset / \sum$	85%	$0,024 \frac{sek}{te}$	$0,159 \frac{sek^2}{te}$

Tabelle 6.3: Kennzahlen für die beiden ausgewählten Arbeitsplätze nach der ersten Verschiebung

³³⁹Vgl. Gleichung (5.36) in Abschnitt 5.1.3.2.2³⁴⁰Vgl. dazu die Diskussion in 5.1.3.2.2

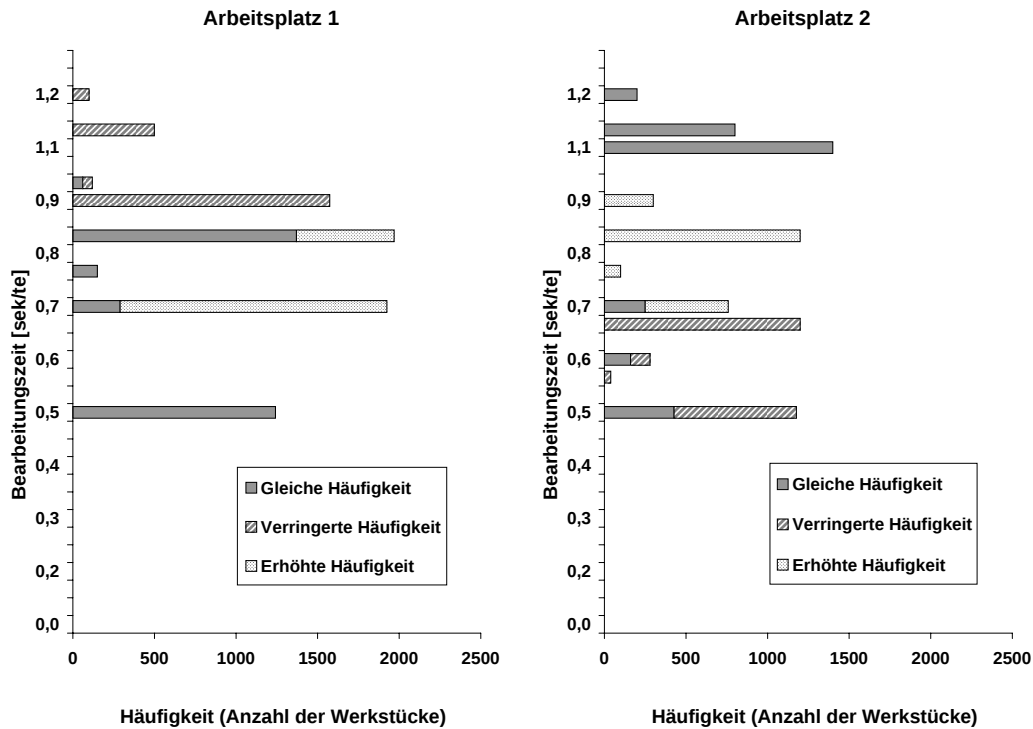


Abbildung 6.2: Zwei Arbeitsplätze nach einer Verschiebung in der Rekonfiguration

Mit nur zwei Verschiebungen war es in diesem Szenario möglich, die lineare Übertaktzeit um 60% und die quadratische Übertaktzeit um 70% zu reduzieren. Die geringe Anzahl an Verschiebungen sorgt für niedrige Rekonfigurationskosten und wird durch die direkte Identifizierung sinnvoller Verschiebungen möglich gemacht. Analog zu der vorgestellten Vorgehensweise wurden mehrere Betrachtungsbereiche untersucht, teilweise auch mit mehr als zwei Arbeitsplätzen im Fokus. Die detaillierten Ergebnisse des kompletten Fallbeispiels sind in Tabelle 6.4 zusammengefasst. In Summe konnte die lineare Übertaktzeit um $0,185 \frac{\text{sek}}{\text{te}}$ und damit um 34% reduziert werden. Die quadratische Übertaktzeit wurde in Summe sogar um 54% reduziert. Hierfür waren lediglich 21 Verschiebungen von Arbeitsvorgängen notwendig. Auf einigen Arbeitsplätzen konnte die Übertaktzeit vollständig beseitigt werden, so dass den Arbeitsplätzen eine vollständige Reihenfolgeunabhängigkeit attestiert werden kann.

Die Abstimmung der Montagelinie wurde vor und nach der Durchführung der acht Verschiebeszenarien aus Tabelle 6.4 mit Hilfe der Fließliniensimulation getestet. Hierbei wurden alle 107 Arbeitsplätze bewertet. Die genauen Ergebnisse sind in Tabelle 6.5 aufgeführt. Die Anzahl notwendiger Unterstützereinsätze u^E konnte im Durchschnitt um 43% und die Anzahl notwendiger Unterstützer u^{MAX} um 35% reduziert werden.

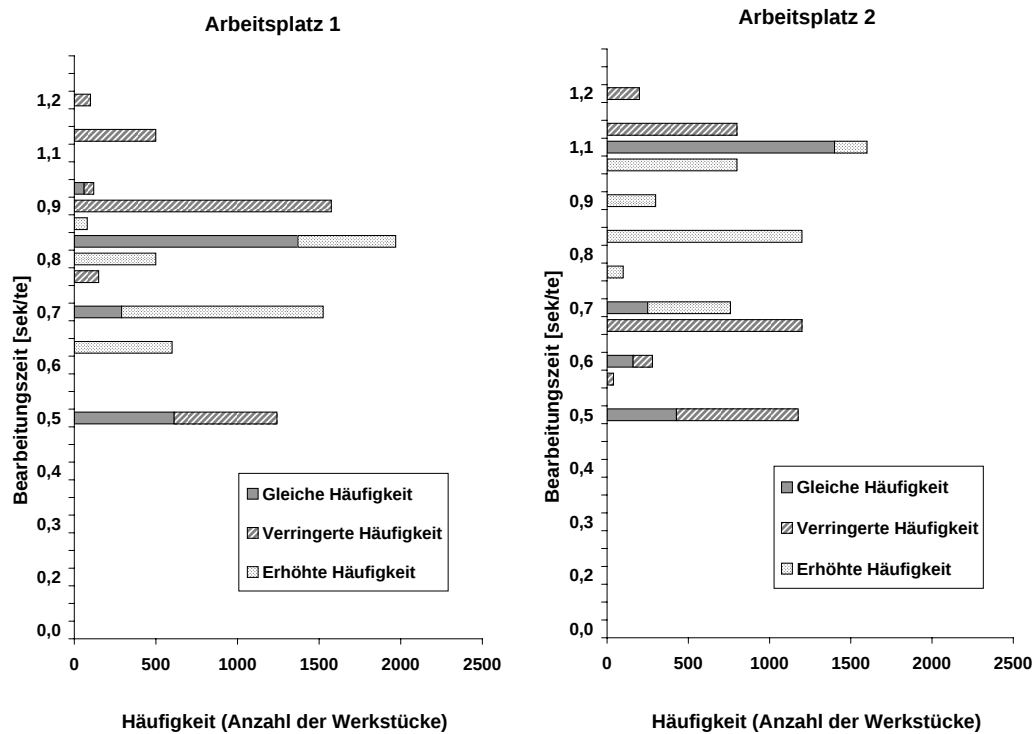


Abbildung 6.3: Zwei Arbeitsplätze nach der zweiten Verschiebung in der Rekonfiguration

Szenario	$\Delta^{LinPos}(\frac{sec}{te})$			Diff [%]	$\Delta^{QuadPos}(\frac{sec^2}{te})$			Diff [%]	Verschiebungen AVOs
	vor	nach	Diff		vor	nach	Diff		
1	0,060	0,024	-0,036	-60%	0,532	0,149	-0,383	-72%	2
2	0,088	0,057	-0,031	-36%	1,623	0,494	-1,129	-70%	2
3	0,034	0,012	-0,022	-65%	0,223	0,169	-0,055	-24%	2
4	0,010	0,010	0,000	4%	0,423	0,094	-0,330	-78%	4
5	0,113	0,086	-0,026	-24%	4,249	2,370	-1,879	-44%	6
6	0,051	0,051	0,001	2%	1,386	1,073	-0,313	-23%	1
7	0,124	0,059	-0,065	-53%	3,248	0,780	-2,468	-76%	2
8	0,063	0,058	-0,005	-8%	2,058	1,252	-0,806	-39%	2
$\Sigma / \oslash$	0,542	0,357	-0,185	-30%	13,742	6,379	-7,363	-53%	21

Tabelle 6.4: Ergebnisse des ersten Anwendungsbeispiels für die Rekonfiguration

Zuordnung	u^E	u^{MAX}
Vorher $\oslash$	487	20
Nachher $\oslash$	278	13
Δ	209	7

Tabelle 6.5: Ergebnisse der Simulation des ersten Anwendungsbeispiels für die Rekonfiguration

In einem zweiten Anwendungsbeispiel in einem etwas kleineren Montagelinienabschnitt mit 19 Stationen wurde ein ähnlicher Workshop durchgeführt. Die Ergebnisse sind in den Tabellen 6.6 und 6.7 aufgeführt. Die durchschnittliche Anzahl der Unterstützereinsätze konnte um 32% und die Anzahl notwendiger Unterstützer um 20% reduziert werden.

Szenario	$\Delta^{LinPos}(\frac{sec}{te})$			Diff [%]	$\Delta^{QuadPos}(\frac{sec^2}{te})$			Diff [%]	Verschiebungen AVOs
	vor	nach	Diff		vor	nach	Diff		
1	0,469	0,313	-0,156	-33%	57,942	33,187	-24,755	-43%	11
2	0,115	0,044	-0,071	-61%	2,168	1,552	-0,616	-28%	1
3	0,403	0,369	-0,035	-9%	52,860	44,373	-8,488	-16%	5
4	0,327	0,323	-0,004	-1%	43,062	38,691	-4,371	-10%	2
5	0,406	0,365	-0,041	-10%	40,508	39,829	-0,678	-2%	4
$\Sigma/\oslash$	1,72	1,41	-0,31	-23%	196,54	157,63	-38,91	-20%	21

Tabelle 6.6: Ergebnisse des zweiten Anwendungsbeispiels für die Rekonfiguration

Zuordnung	u^E	u^{MAX}
Vorher $\oslash$	122	5
Nachher $\oslash$	84	4
Δ	38	1

Tabelle 6.7: Ergebnisse der Simulation des zweiten Anwendungsbeispiels für die Rekonfiguration

Für jede der 100 zufällig gewählten Produktionsreihenfolgen, mit denen die Fließliniensimulation angestoßen wurde, ging die Anzahl an Unterstützern in ähnlichem Umfang zurück. Dies zeigt die Unabhängigkeit des Vorgehens zur Rekonfiguration von der Reihenfolgeplanung. Das Ergebnis der 100 durchgeführten Simulationen ist im Anhang in Tabelle A.1 aufgeführt.

6.2 Validierung der teilautomatisierten Vorranggraphenentwicklung

6.2.1 Testszenario

Zur Validierung der teilautomatisierten Vorranggraphenentwicklung sollen zwei unterschiedliche Wege verfolgt werden. Zuerst wird ein Vorranggraph aus Beispieldaten mit 472 Arbeitsvorgängen abgeleitet und hinsichtlich der erreichten Komplexitätsreduktion für eine manuelle Ergänzung des Graphen untersucht. In einem zweiten Schritt wird der Graph im Rahmen einer Rekonfiguration verwendet. Dabei wird die lokale Suchheuristik für Tauschkombinationen aus der Design-Phase der Rekonfiguration genutzt.³⁴¹ Der Betrachtungsbereich wird dafür auf vier Arbeitsplätze reduziert. Innerhalb dieser vier Arbeitsplätze kann die Heuristik bis zu fünf Verschiebungen in Kombination durchführen ($\omega = 5$). Verglichen wird die erreichte Reduzierung von Unterstützereinsätzen mit und ohne Betrachtung des Vorranggraphen. Dadurch kann das maximale theoretische Verbesserungspotential für die betrachteten Arbeitsplätze ermittelt werden. Die erreichten Lösungen sind ohne Beachtung eines Vorranggraphen in der Praxis allerdings nicht umsetzbar.

6.2.2 Ergebnisse

Der erzeugte Vorranggraph ist im Anhang in den Abbildungen A.1, A.2 und A.3 dargestellt.

6.2.2.1 Bestimmung der erreichten Komplexitätsreduktion

Die Menge an zu untersuchenden Vorrangbeziehungen vrb in einer nachgelagerten, manuellen Bearbeitung eines Graphen steigt mit der Anzahl der zu untersuchenden Knoten $\mathcal{AVO}$. Jede unabhängige Werkstückansprache kann einzeln, wenn gewollt auch von unterschiedlichen Personen, bearbeitet werden. Im Beispielfall wurden 24 unabhängige Werkstückansprachen $wa \in WA$ definiert. Die Arbeitsvorgänge wurden in diese unterschiedlichen Werkstückansprachen getrennt, definiert durch die Mengen $\mathcal{AVO}_{wa}$. Die maximale Anzahl der Knoten innerhalb einer Werkstückansprache stellt die größte Komplexität dar, die der Planer noch überblicken muss, um manuell Vorrangbeziehungen ergänzen zu können. Der Wert b_{wa} beschreibt die Größe des Betrachtungsraumes im Verhältnis zur Gesamtgröße des Graphen (6.2).

$$b_{wa} = \frac{|\mathcal{AVO}_{wa}|}{|\mathcal{AVO}|} \quad (6.2)$$

³⁴¹Vgl. Abschnitt 5.1.3.2

Die detaillierten Ergebnisse finden sich in Tabelle 6.8.

wa	$ \mathcal{AVO}_{wa} $	b_{wa}
1	14	2,97%
2	23	4,87%
3	12	2,54%
4	118	25,00%
5	16	3,39%
6	3	0,64%
7	15	3,18%
8	5	1,06%
9	8	1,69%
10	22	4,66%
11	6	1,27%
12	1	0,21%
13	27	5,72%
14	4	0,85%
15	2	0,42%
16	17	3,60%
17	12	2,54%
18	27	5,72%
19	134	28,39%
20	6	1,27%
21	21	4,45%
22	1	0,21%
23	35	7,42%
24	31	6,57%
25	8	1,69%
26	1	0,21%
27	1	0,21%
28	8	1,69%

Tabelle 6.8: Verbleibende Knoten im Vorranggraphen je Werkstückansprache

Werkstückansprache 19 beinhaltet nach der Trennung die meisten Arbeitsvorgänge. Die Menge der gleichzeitig zu betrachteten Knoten konnte im schlechtesten Fall um 71,61% reduziert werden.

Durch die Generierung von XOR-Trennungen wurden weitere Freiheitsgrade im Graphen erzeugt und die Anzahl der durch den Planer zu betrachtenden Beziehungen weiter reduziert. Jeder der erzeugten XOR-Knoten hat eine bestimmte Anzahl ausgehender Kanten, die weitere Freiheitsgrade darstellen. Sei $M_{v,x,wa}$ die Menge an Knoten, die von einer ausgehenden Kante $v \in V$ eines XOR-Aufspaltungsknotens $x \in X_{wa}$ in einer Werkstückansprache wa zugeordnet ist. Die Menge N_{wa} beinhaltet alle Knoten aus Werkstückansprache wa , die keiner XOR-Trennung zugeordnet sind und somit immer in einer Nachbearbeitung betrachtet werden müssen. Der Planer muss im schlimmsten Fall den längsten Pfad innerhalb einer Werkstückansprache betrachten. Die Reduktion des Betrachtungsbereiches rb_{wa} ergibt sich durch

das Verhältnis aus der Länge des längsten Pfades (Knotenmenge) und der Gesamtanzahl der Knoten in einer Werkstückansprache wa (6.3)

$$rb_{wa} = 1 - \frac{\sum_{x \in X_{wa}} (\max_v |M_{v,x,wa}|) + N_{wa}}{|\mathcal{AVO}_{wa}|} \quad (6.3)$$

Die Ergebnisse sind in Tabelle 6.9 dargestellt. Durch die XOR-Trennungen konnte der Betrachtungshorizont an Werkstückansprache 19 um weitere 9% verringert werden.

wa	$\sum_{x \in X_{wa}} (\max_v M_{v,x,wa})$	$ N_{wa} $	$ X_{wa} $	$ \mathcal{AVO}_{wa} $	rb_{wa}
1	5	9	2	14	0%
2	12	10	10	23	4%
3	7	4	7	12	8%
4	75	27	31	118	14%
5	11	4	3	16	6%
6	0	3	0	3	0%
7	10	2	9	15	20%
8	3	2	1	5	0%
9	3	3	3	8	25%
10	7	12	3	22	14%
11	4	2	2	6	0%
12	1	0	1	1	0%
13	11	12	5	27	15%
14	4	0	3	4	0%
15	2	0	2	2	0%
16	11	6	10	17	0%
17	6	3	3	12	25%
18	10	3	4	27	52%
19	92	30	34	134	9%
20	3	3	1	6	0%
21	16	3	4	21	10%
22	0	1	11	1	0%
23	23	4	2	35	23%
24	22	3	7	31	19%
25	3	5	2	8	0%
26	1	0	1	1	0%
27	0	1	0	1	0%
28	3	2	3	8	38%

Tabelle 6.9: Ergebnisse Reduktion des Betrachtungshorizontes durch XOR-Aufsplittungen

6.2.2.2 Verwendung des Vorranggraphen innerhalb einer Rekonfigurationsmethode

Im Folgenden wird der Vorranggraph im Rahmen einer Rekonfiguration beispielhaft mit Hilfe der Lösungsheuristik aus Abschnitt 5.1.3.2.3 verwendet. Dabei soll festgestellt werden, ob genügend Freiheitsgrade im Graphen erkannt werden, um einer Rekonfiguration Freiraum zur

Ermittlung besserer Zuordnungslösungen zu geben. Hierfür werden beispielhaft Arbeitsplätze herausgegriffen, die einmal unter Verwendung des Vorranggraphen und einmal unter Vernachlässigung sämtlicher Restriktionen optimiert werden. Basierend auf der Startlösung wird der Algorithmus mit folgenden Parametern angestoßen:

- χ = lineare Übertaktzeit
- $\beta = 20$ (Anzahl an Lösungen, die zur weiteren Untersuchung generiert werden sollen)
- $\omega = 5$ (Anzahl an erlaubten aufeinander folgenden Verschiebungen)
- $\kappa = 10$ (Anzahl an weiterverfolgten Lösungen je Verschiebungsstufe)

Fixierungen von Arbeitsvorgängen sowie Unter- und Obergrenzen der durchschnittlichen Auslastungen auf den Arbeitsplätzen werden nicht verwendet. Die entstandenen Rekonfigurationen werden mit Hilfe der Fließliniensimulation³⁴² bewertet. Die Ergebnisse beider Durchläufe sind in Abbildung 6.4 dargestellt. Für jeden Durchlauf sind die 20 besten gefundenen Lösungen aufgezeigt, sortiert nach der durchschnittlichen Unterstützereinsatzzeit. Bei der Durchführung der Rekonfiguration ohne Vorranggraphen konnte eine Reduzierung der Anzahl Unterstützereinsätze um 10,5% erreicht werden. Bei Beachtung des Vorranggraphen wurde eine Reduktion um 10,4% erreicht. Die Tatsache, dass der Unterschied in der Lösungsqualität minimal ist, zeigt, dass genügend Freiheitsgrade im Graphen vorhanden sind. Dem Algorithmus wurde allerdings nur eine maximale Anzahl von fünf aufeinanderfolgenden Verschiebungen erlaubt, um dem Planer noch die Möglichkeit zu geben, die Übersicht über die Lösung zu behalten.³⁴³ Darüber hinaus werden hierdurch implizit auch die Rekonfigurationskosten gering gehalten.

Zusätzlich zu den durchgeführten Tests am Vorranggraphen und der beispielhaften Rekonfiguration wurde die Methode zur teilautomatisierten Vorranggraphenentwicklung im Rahmen der Rekonfigurationstests in Abschnitt 6.1 verwendet.

³⁴²Vgl. Abschnitt 5.3.3

³⁴³Vgl. dazu die Ausführungen in Abschnitt 5.1.3.2.3

Durchlauf	ohne Vorranggraph			Durchlauf	mit Vorranggraph		
	u^E	u^{MAX}	$\oslash u^{Zeit}$		u^E	u^{MAX}	$\oslash u^{Zeit}$
START	848	6	70,11	START	848	6	70,11
1	763	5	64,78	1	768	5	65,37
2	762	5	64,79	2	767	5	65,38
3	763	5	64,79	3	767	5	65,38
4	762	5	64,79	4	770	5	65,48
5	762	5	64,81	5	770	5	65,48
6	762	5	64,81	6	759	5	65,60
7	766	5	64,81	7	768	5	65,61
8	766	5	64,81	8	767	5	65,61
9	764	5	64,82	9	767	5	65,62
10	764	5	64,82	10	767	5	65,62
11	761	5	64,83	11	767	5	65,64
12	761	5	64,86	12	767	5	65,64
13	759	5	65,66	13	770	5	65,71
14	759	5	65,67	14	769	5	65,72
15	759	5	65,70	15	769	5	65,72
16	759	5	65,91	16	769	5	65,73
17	763	6	65,91	17	770	5	65,74
18	763	6	65,91	18	769	5	65,74
19	765	6	66,58	19	769	5	65,76
20	765	6	66,59	20	760	5	65,76
Beste Werte	759	5	64,78	Beste Werte	760	5	65,37

Abbildung 6.4: Ergebnisse der Lösungsheuristik mit und ohne Vorranggraphen im Vergleich

6.3 Validierung der belastungsorientierten Reihenfolgeplanung zur Reduzierung des Unterstützereinsatzes

6.3.1 Testszenario

Zur Validierung der belastungsorientierten Reihenfolgeplanung wird im ersten Schritt die Fließliniensimulation getestet. Dabei wird ermittelt, in wie weit Abhängigkeiten zwischen den Werkern Einfluss auf die Werkerpositionen und somit auf die Unterstützereinsätze haben. Des Weiteren werden die Auswirkungen durch reaktiven und präventiven Unterstützereinsatz verglichen. Um das Potential der Heuristik zur Reihenfolgeplanung zu ermitteln, wird zuerst eine Parametrisierung der Clusteringmethode für die initiale Reihenfolgebildung durchgeführt. Darauf folgen verschiedene Anwendungsbeispiele. Die Testdaten beinhalten sieben verschiedene Auftragsprogramme mit jeweils ca. 200 Aufträgen. Jeder dieser Aufträge beinhaltet einige tausend Arbeitsvorgänge. In einem vorgelagerten Schritt wird eine Ermittlung der Arbeitsvorgänge je Auftrag im Rahmen einer Stücklistenauflösung³⁴⁴ durchgeführt. Hierdurch ergeben sich die Bearbeitungszeiten der Aufträge auf jedem der 162 Arbeitsplätze. Die

³⁴⁴Vgl. zu Verfahren der Stücklistenauflösung [Orl75, S.65ff]

Driftgrenzen wurden wie in Abschnitt 5.3.3.1 beschrieben, mit Hilfe des Werkstücks ermittelt, das die größte Bearbeitungszeit aufweist (6.4).

$$dbr_{ap} = \left\lceil \frac{\max_{idx \in \mathcal{RF}} [\tau_{idx,ap}]}{tz} \right\rceil \quad \forall ap \in \mathcal{AP} \quad (6.4)$$

Eine Vordrift wurde im untersuchten Fallbeispiel nicht erlaubt ($dbl_{ap} = 0$).

6.3.2 Ergebnisse

6.3.2.1 Einfluss der Abhängigkeit zwischen Werkern

Im gegebenen Praxisfall konnten die Abhängigkeiten zwischen den Werkern nicht vollständig für alle Arbeitsplätze ermittelt werden. Um eine Aussage zu treffen, in wie weit sich diese Abhängigkeiten auf die Bewertung von Unterstützereinsätzen auswirken, werden die Kennzahlen für zwei verschiedene Fälle ermittelt. Im besten Fall (Best-Case) wird davon ausgegangen, dass alle Werker unabhängig voneinander arbeiten können und sich nicht gegenseitig verzögern, wie in Abbildung 6.5 oben dargestellt. Im Gegensatz dazu wird der schlechteste Fall (Worst-Case) berechnet, indem davon ausgegangen wird, dass alle Werker auf einer Seite des Bandes voneinander abhängen. Jeder Werker wird durch den Werker davor beeinflusst und beeinflusst wiederum den nächsten Werker, wie in Abbildung 6.5 unten dargestellt.

Die Ergebniskennzahlen werden für sieben verschiedene Beispielprogramme je Organisationsalternative der Unterstützer einmal im Best- und im Worst-Case ermittelt und sind in den Tabellen 6.10 und 6.11 angegeben.

$\mathcal{P}$	$ \mathcal{P} $	u^E			u^{MAX}		
		BestCase	WorstCase	Diff [%]	BestCase	WorstCase	Diff [%]
1	200	188	2872	1427,66	12	45	275,00
2	193	231	2882	1147,62	13	48	269,23
3	187	217	2790	1185,71	12	49	308,33
4	194	224	2831	1163,84	10	48	380,00
5	203	246	3172	1189,43	13	48	269,23
6	194	251	3131	1147,41	15	50	233,33
7	174	233	2856	1125,75	14	49	250,00
$\odot$	192,14	227,14	2933,43	1198,20	12,71	48,14	283,59

Tabelle 6.10: Vergleich der Simulation ohne Werkerabhängigkeiten (Best-Case) und mit maximalen Werkerabhängigkeiten (Worst-Case) bei reaktivem Unterstützereinsatz

Die Ergebnisse zeigen, dass die Abhängigkeiten zwischen den Werkern einen starken Einfluss auf die Notwendigkeit für Unterstützereinsätze haben. Im Worst-Case wurden bei präventivem Unterstützereinsatz im Durchschnitt 20-mal so viele Unterstützereinsätze erzeugt und 2,5-mal so viele Unterstützer benötigt. Beim reaktiven Unterstützereinsatz treten 12-mal so viele

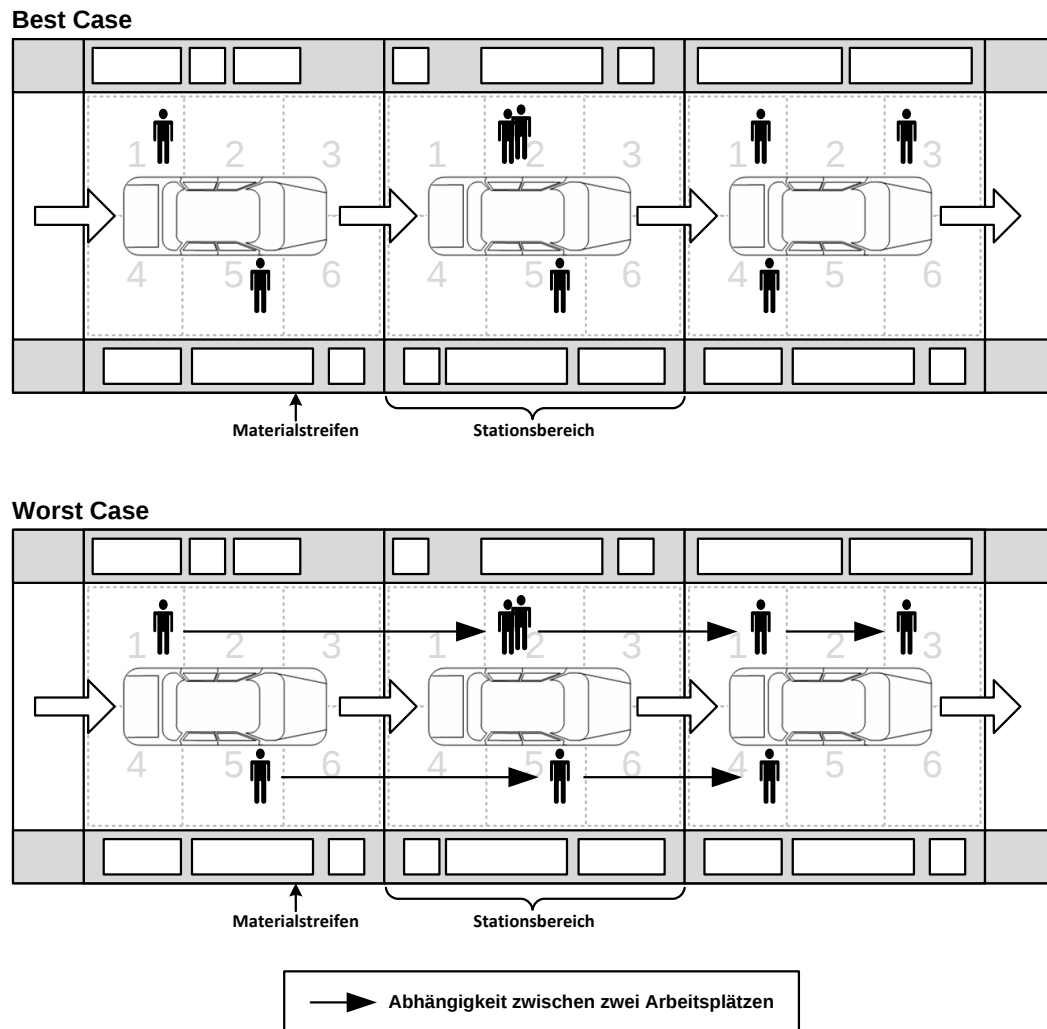


Abbildung 6.5: Vergleich Abhängigkeiten im Best-Case und im Worst-Case

$\mathcal{P}$	$ \mathcal{P} $	u^E			u^{MAX}		
		BestCase	WorstCase	Diff [%]	BestCase	WorstCase	Diff [%]
1	200	117	2949	2420	14	46	228
2	193	149	2900	1846	14	48	242
3	187	142	2805	1875	13	50	284
4	194	138	2856	1970	13	51	292
5	203	140	3147	2148	14	46	229
6	194	150	3087	1958	14	51	264
7	174	137	2827	1963	16	52	225
Ø	192,14	139,00	2938,71	2026	14,00	49,14	252

Tabelle 6.11: Vergleich der Simulation ohne Werkerabhängigkeiten (Best-Case) und mit maximalen Werkerabhängigkeiten (Worst-Case) bei präventivem Unterstützereinsatz

Unterstützereinsätze auf und 2,8 mal so viele Unterstützer werden benötigt. Hierdurch wird deutlich, dass in einem realen Planungsfall einer Montagelinie unabhängige Arbeitsplätze

hintereinander platziert werden sollten, um den Effekt des Durchreichens von Werkerverzug zu minimieren. Wenn Abhängigkeiten nicht vermieden werden können, sollten sie bei einer Fließliniensimulation in jedem Fall betrachtet werden.

6.3.2.2 Einfluss der Organisationsform von Unterstützereinsätzen

Im Tabelle 6.12 werden die beiden Organisationsformen des Unterstützereinsatzes, reaktiv und präventiv, im Best-Case verglichen. Die Zahlen verhalten sich im Worst-Case ähnlich. Als Basis wurde der reaktive Unterstützereinsatz gewählt. Im Vergleich dazu verringert sich die Anzahl an Unterstützereinsätzen bei der präventiven Organisationsform durchschnittlich um etwa 38,68%. Dies ist dadurch zu erklären, dass beim präventiven Unterstützereinsatz jeweils ein ganzer Auftrag vom Unterstützer übernommen wird und somit nachhaltig Drift reduziert wird. Im Gegensatz dazu erhöht sich die Unterstützereinsatzzeit durchschnittlich um 163,50%. Durch die verlängerte Einsatzzeit der Unterstützer erhöht sich auch die Wahrscheinlichkeit, dass zwei Einsätze gleichzeitig stattfinden. Hierdurch hat sich die Anzahl notwendiger Unterstützereinsätze um 12,71% erhöht.

$\mathcal{P}$	$ \mathcal{P} $	u^E			$\frac{u^{Zeit}}{ \mathcal{P} }$			u^{MAX}		
		A	B	Diff [%]	A	B	Diff [%]	A	B	Diff [%]
1	200	188	117	-37,77	85,41	231,34	170,86	12	14	16,67
2	193	231	149	-35,50	119,41	319,16	167,28	13	14	7,69
3	187	217	142	-34,56	142,21	325,14	128,63	12	13	8,33
4	194	224	138	-38,39	113,72	310,87	173,36	10	13	30,00
5	203	246	140	-43,09	109,96	289,65	163,41	13	14	7,69
6	194	251	150	-40,24	115,21	315,87	174,17	15	14	-6,67
7	174	233	137	-41,20	126,10	336,39	166,76	14	16	14,29
○	192,14	227,14	139	-38,68	116,00	304,06	163,50	12,71	14,00	11,14

Tabelle 6.12: Vergleich der Auswirkungen von reaktivem (A) und präventivem (B) Unterstützereinsatz

6.3.2.3 Parametrisierung der Clusteringmethode

Zur Parametrisierung der Clusteringmethode muss im ersten Schritt getestet werden, welches Verfahren für die Startzentroidenwahl (Random, K-Means ++ oder Sort&Group) und die Sortierung der Aufträge bei der Zuordnung (Random oder Sort) gewählt wird.³⁴⁵ Hierdurch ergeben sich sechs verschiedene Konfigurationen, die jeweils mit sieben Auftragsprogrammen getestet werden. Für die zufallsabhängigen Verfahren wird jeweils eine ausreichend große Anzahl Durchläufe angestoßen und der Mittelwert bestimmt. Die erreichte Homogenität in den Clustern wird mit Hilfe der Fehlerquadratsumme bei einer Clusteranzahl ($k = 6$) ermittelt.³⁴⁶

³⁴⁵Vgl. Abschnitt 5.3.4.1.1

³⁴⁶Vgl. Abschnitt 5.3.4.1.1

In Tabelle 6.13 sind jeweils die Durchschnittswerte aus den Szenarien, auf die Anzahl an Aufträgen normiert, angegeben. Die detaillierten Ergebnisse der einzelnen Szenarien finden sich in Tabelle A.2 im Anhang.

Reihenfolge der Zuordnung	Auswahl der Startzentroiden		
	Random	K-Means ++	Sort&Group
Random	105,18	105,85	104,02
Sort	108,46	108,65	107,85

Tabelle 6.13: Durchschnittliche Homogenität der Cluster je Auftrag in Abhängigkeit von der Auswahl der Startzentroiden und der Reihenfolge der Zuordnung über alle Szenarien

Bei der Kombinationen der Verfahren zeigt die Auswahl der Startzentroiden nach der Sort&Group Methode, in Verbindung mit der zufälligen Zuordnungsreihenfolge, die besten Ergebnisse. Daher wird dieses Verfahren im Folgenden verwendet.

Im nächsten Schritt muss die Anzahl zu bildender Cluster festgelegt werden. Hierfür werden wiederum Testläufe mit sieben Auftragsprogrammen durchgeführt. Die Clusteranzahl wird dabei zwischen 2 und 22 variiert. Die genauen Ergebnisse für die Entwicklung der Homogenität sind in Tabelle 6.14 und in der dazugehörigen Abbildung 6.6 zusammengefasst. Zusätzlich zu der resultierenden Fehlerquadratsumme wird auch die Laufzeit des Verfahrens bestimmt. In Tabelle 6.15 und Abbildung 6.7 sind die Ergebnisse der Laufzeitberechnung dargestellt.

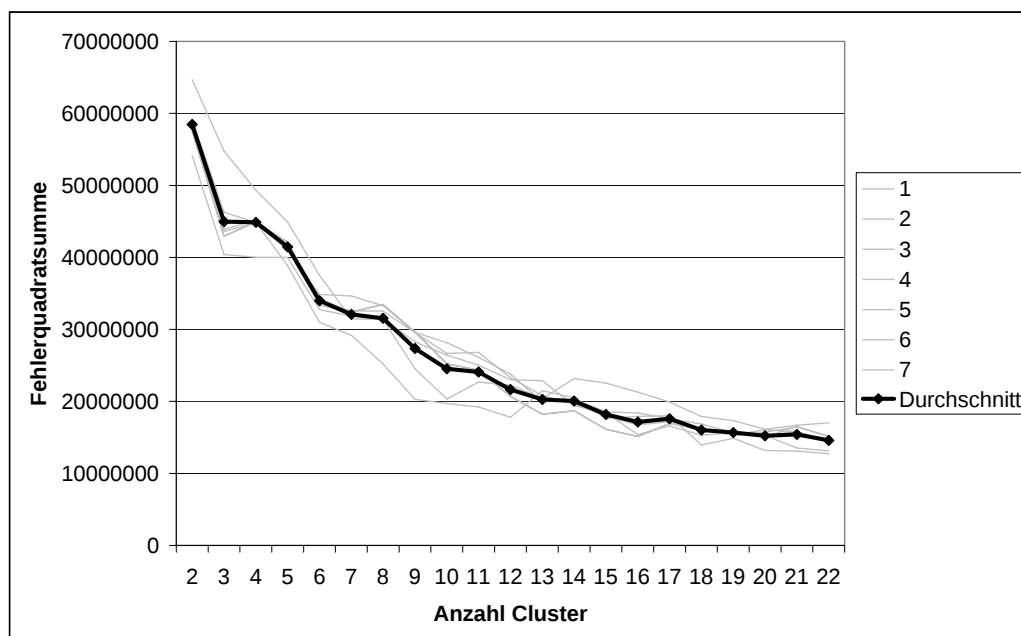


Abbildung 6.6: Entwicklung der Homogenität (Quadratsumme) bei unterschiedlicher Clusteranzahl

k	Fehlerquadratsumme $\sum_{ap \in AP} (\tau_{p1,ap} - \tau_{p2,ap})^2$							
	1	2	3	4	5	6	7	$\emptyset$
2	57628736	59124402	54123335	57455923	57628736	64706047	58546769	58459135
3	42940203	46279478	40390176	43882022	42940203	54768549	43605728	44972337
4	44939818	44845185	40010716	45198582	44939818	49336098	44850738	44874422
5	42094324	40976443	40000514	41261385	42094324	44909302	38860050	41456620
6	34199131	34857864	32750872	33329267	34199131	37506211	30981103	33974797
7	32415658	34646016	31853647	32575378	32415658	31465500	29185865	32079674
8	33460210	33288194	31453475	32532647	33460210	31374635	25189654	31537003
9	29647734	29597093	28264543	29544386	29647734	24530302	20288130	27359988
10	25226555	28162638	26438519	26675079	25226555	20344318	19708397	24540294
11	24333684	26130760	25032353	26797000	24333684	22683833	19234997	24078044
12	20687690	23804126	23029254	23294331	20687690	22248079	17782286	21647637
13	18217000	19908211	22884566	20796998	18217000	20406944	21488195	20274131
14	18702102	20285733	19420194	19677908	18702102	23163348	20509710	20065871
15	16134449	18535526	18422018	17903075	16134449	22547281	17617178	18184854
16	15134124	18388677	15387919	16744791	15134124	21287104	17925084	17143118
17	16957163	17596716	16606248	17178924	16957163	19908336	17995000	17599936
18	16079241	16853100	15286868	16014188	16079241	17905171	13956198	16024858
19	15354224	15676963	15671855	15397011	15354224	17329058	14871222	15664937
20	16051793	15064304	15263843	15721625	15244745	16117356	13191836	15236500
21	15791955	16505835	13532749	16447247	15791955	16686294	13096069	15407443
22	14442199	15125757	13148661	15152947	14442199	17016602	12740949	14581331

Tabelle 6.14: Entwicklung der Homogenität in Abhängigkeit von der Clusteranzahl

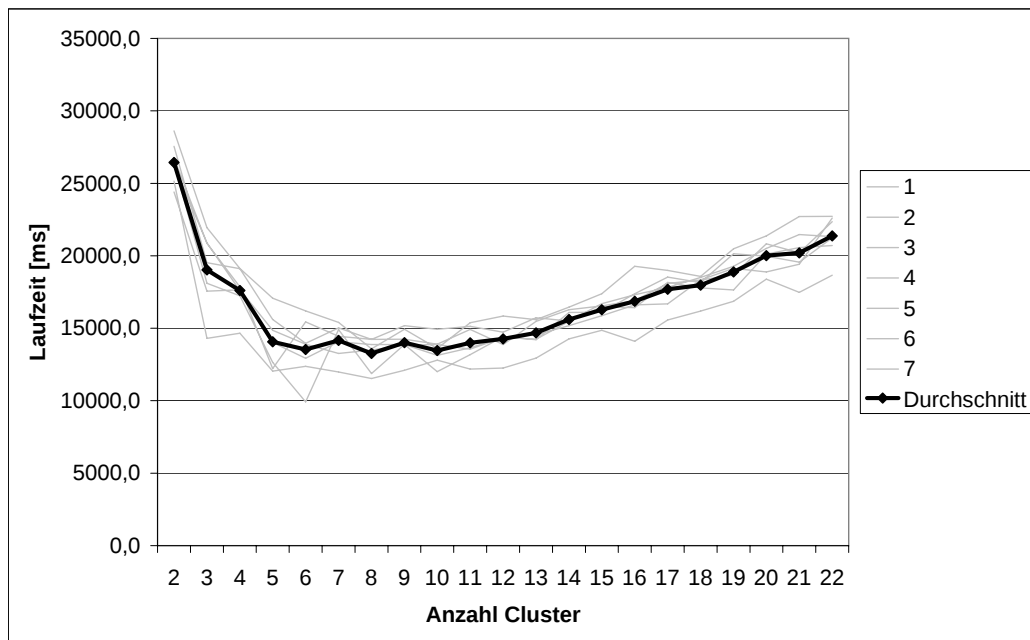


Abbildung 6.7: Entwicklung der Laufzeit in ms bei unterschiedlicher Clusteranzahl

Die Entwicklung der Homogenität zeigt, dass eine möglichst große Clusteranzahl gewählt werden sollte. Im Gegenzug dazu zeigt das Laufzeitverhalten die geringsten Werte zwischen

k	Laufzeit in ms							
	1	2	3	4	5	6	7	$\emptyset$
2	27547,6	26250,6	24399,4	26734,0	28616,9	26371,2	25149,4	26438,4
3	19528,4	20884,4	17565,0	20862,2	21941,6	18105,8	14309,0	19028,1
4	19109,8	17593,6	17655,2	17849,8	19125,9	17237,1	14655,9	17603,9
5	15609,4	14015,4	14858,4	12218,6	17069,5	12643,4	12043,4	14065,5
6	13962,2	12924,8	13905,4	15443,6	16200,7	9918,5	12377,8	13533,3
7	15021,6	14071,6	13261,4	14449,1	15400,7	14902,7	11980,9	14155,4
8	14237,4	13871,8	13492,8	14230,6	13578,7	11874,7	11537,2	13260,5
9	15181,2	13874,8	13845,8	14246,2	14925,7	13871,5	12102,8	14006,9
10	14924,8	13134,0	13874,2	13905,5	13603,7	12002,8	12796,5	13463,1
11	15134,2	13646,6	13552,2	14914,2	15378,8	13184,0	12187,2	13999,6
12	14731,2	14371,4	14364,6	13873,1	15844,5	14421,5	12252,8	14265,6
13	15728,0	14249,6	14561,6	15476,4	15591,3	14205,9	12940,3	14679,0
14	15497,0	15480,8	15177,0	16279,6	16460,1	16087,1	14265,3	15606,7
15	16700,0	16477,6	15867,0	16541,8	17375,8	16099,6	14865,2	16275,3
16	17315,4	16418,4	16607,6	16904,2	19275,9	17380,8	14105,9	16858,3
17	17815,8	18162,0	16680,2	18094,9	18991,5	18524,5	15568,4	17691,0
18	18506,2	17780,8	18336,6	18326,1	18579,0	18146,4	16180,8	17979,4
19	19281,4	17637,2	19136,6	18676,0	20485,3	20137,0	16858,9	18887,5
20	20509,4	20121,2	18879,8	20826,1	21372,8	19971,4	18393,3	20010,6
21	21465,6	20562,0	19413,6	20207,2	22710,4	19558,9	17474,6	20198,9
22	21319,2	20712,2	22601,6	22354,1	22726,0	21237,0	18655,8	21372,3

Tabelle 6.15: Entwicklung der Laufzeit in Abhängigkeit von der Clusteranzahl

5 und 12 Clustern. Im nächsten Schritt wird die optimale Zentroidensequenzierung mit Hilfe der vollständigen Enumeration aller möglichen Reihenfolgen vorgenommen. Jede dieser Reihenfolgen wird mit Hilfe der Fließliniensimulation in doppelter Ausprägung bewertet.³⁴⁷ Dabei wird das Laufzeitverhalten in Abhängigkeit von der Anzahl an Clustern ermittelt und in Abbildung 6.8 dargestellt. Die detaillierten Ergebnisse finden sich in Tabelle 6.16.

k	Laufzeit in sek							
	1	2	3	4	5	6	7	$\emptyset$
2	0,1	0,1	0,1	0,0	0,1	0,1	0,0	0,1
3	0,2	0,2	0,2	0,2	0,2	0,2	0,2	0,2
4	1,1	1,2	1,2	1,1	1,1	1,1	1,1	1,1
5	7,0	7,2	7,3	7,2	7,1	7,2	7,1	7,2
6	52,3	51,9	53,1	51,5	50,5	51,3	51,1	51,7
7	432,9	431,7	437,5	429,6	421,8	426,2	419,8	428,5
8	3.915,4	3.858,5	3.886,5	3.809,2	3.816,8	3.810,8	3.808,4	3.843,7

Tabelle 6.16: Entwicklung der Laufzeit für die optimale Zentroidensequenzierung bei unterschiedlicher Clusteranzahl je getesteter Instanz

Um den Parameter für die Anzahl Cluster endgültig zu bestimmen, wird für jede Clusteranzahl zwischen zwei und acht eine Reihenfolge mit den drei Schritten Clustering, Zentroidensequenzierung und Picking durchgeführt. Die resultierenden Reihenfolgen werden hinsichtlich

³⁴⁷Vgl. Abschnitt 5.3.4.1.2

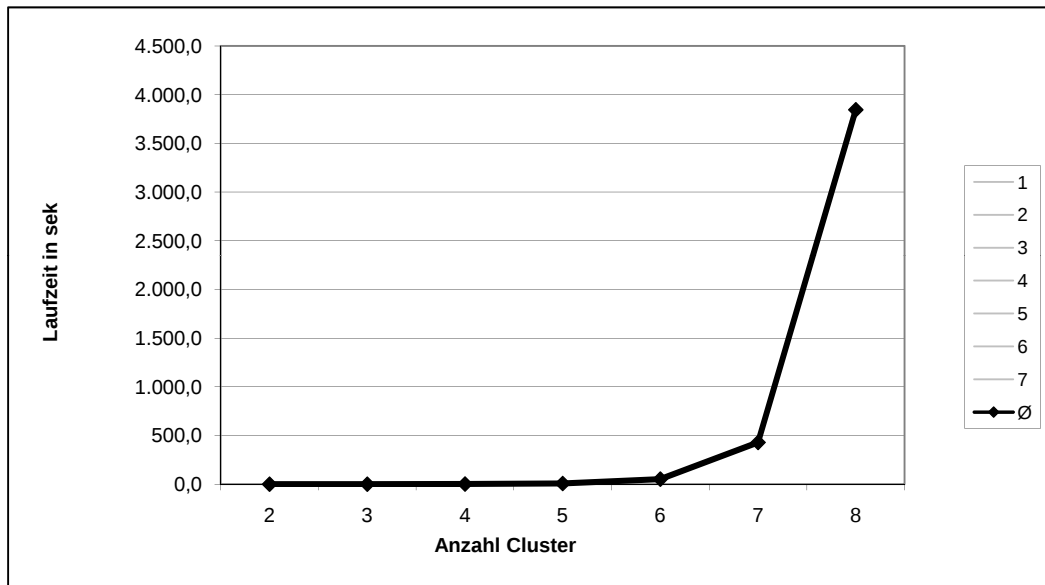


Abbildung 6.8: Entwicklung der Laufzeit für die optimale Zentroidensequenzierung bei unterschiedlicher Clusteranzahl in sek

der Unterstützereinsätze bewertet. Da das Picking im Anschluss an die Zentroidensequenzierung zufällig Aufträge aus den Clustern wählt, wird diese Zufallsauswahl mehrfach durchgeführt und ein Mittelwert gebildet, um eine generelle Aussage der Leistungsfähigkeit des Verfahrens in Abhängigkeit von der Clusteranzahl treffen zu können. Die Kennzahlen werden für alle sieben Auftragsprogramme ermittelt und sind in Tabelle 6.17 im Durchschnitt dargestellt. Das beste Ergebnis erzielt die Parametrisierung bei $k = 6$ mit einer durchschnittlichen Anzahl an Unterstützereinsätzen von 120,68. Auch die Anzahl notwendiger Unterstützer erreicht so den besten Wert von 12,95. Eine Erhöhung der Clusteranzahl hat schlechtere Ergebnisse und deutlich höhere Laufzeiten zur Folge.

k	$\oslash u^E$	$\oslash u^{MAX}$
2	194,77	19,31
3	125,50	14,47
4	164,53	17,40
5	164,47	16,95
6	120,68	12,95
7	121,69	14,41
8	128,22	15,55

Tabelle 6.17: Durchschnittliche Unterstützerkennzahlen je Clusteranzahl in der Initialreihenfolge

6.3.2.4 Anwendungsbeispiele

In dem folgenden Anwendungsbeispiel wird sowohl die Initialheuristik mit den drei Schritten Clustering, Sequenzierung der Zentroiden und Picking als auch das iterative Verbesserungsverfahren getestet. Die Ergebnisse werden mit sieben Auftragsprogrammen mit jeweils ca. 200 Aufträgen generiert. Die Aufträge erzeugen dabei Bearbeitungszeiten auf 176 Arbeitsplätzen. Für das iterative Verbesserungsverfahren wird der Parameter d auf 20 Iterationen gesetzt und δ auf null, so dass nach jeder Veränderung eine neue Bewertung durchgeführt wird. Die entstehenden Reihenfolgen werden mit Hilfe des Fließliniensimulators hinsichtlich ihres notwendigen Unterstützereinsatzes geprüft und im Anschluss verglichen mit den Kennzahlen einer zufällig gewählten (Random) und einer mit Hilfe von Car-Sequencing Kriterien gesteuerten Reihenfolge (Car-Seq). Die Kriterien für das Car-Sequencing wurden über einen längeren Zeitraum von Planern angepasst und ermittelt. Für den Vergleich mit der zufälligen Reihenfolge wird eine ausreichend große Anzahl an Reihenfolgen gebildet, jeweils simuliert und daraus der Mittelwert gebildet. In Tabelle 6.18 werden die Ergebnisse im Best-Case, also ohne Werkerabhängigkeiten, und in Tabelle 6.19 mit maximaler Werkerabhängigkeit jeweils mit reaktivem Unterstützereinsatz zusammengefasst.

$\mathcal{P}$	$ \mathcal{P} $	u^E				u^{MAX}			
		R	CS	IH	IH+IVV	R	CS	IH	IH+IVV
1	200	335	189	185	155	12	10	6	
2	193	347	231	192	184	13	9	7	
3	187	302	217	157	125	12	8	6	
4	194	305	224	215	138	10	8	6	
5	203	339	246	263	193	13	9	8	
6	194	387	252	329	277	15	12	9	
7	174	328	233	193	165	14	10	8	
$\bigcirc$	192,14	334,71	227,43	219,14	176,71	12,71	9,43	7,14	

Tabelle 6.18: Ergebnisse für reaktiven Unterstützereinsatz (A) im Best-Case (R = Zufallsreihenfolge, CS = Car Sequencing, IH = Initialheuristik, IVV = Iteratives Verbesserungsverfahren)

Dieselbe Testreihe wird noch einmal durchgeführt unter Berücksichtigung des präventiven Unterstützereinsatzes. Die Ergebnisse finden sich in den Tabellen 6.20 und 6.21

Vergleicht man die ermittelten Werte, so zeigt sich eine durchschnittliche Verbesserung der Anzahl Unterstützereinsätze um 13,71 % und der Anzahl notwendiger Unterstützereinsätze um 29,55% im Vergleich zur Car-Sequencing Reihenfolge. Es zeigt sich, dass die Ergebnisse ohne Werkerabhängigkeiten (Best-Case) deutlich besser sind als mit maximalen Werkerabhängigkeiten (Worst-Case). Die detaillierten Ergebnisse im Vergleich mit der zufälligen und der Car-Sequencing Reihenfolge finden sich in Tabelle 6.22 und in Abbildung 6.9.

$\mathcal{P}$	$ \mathcal{P} $	u^E				u^{MAX}			
		R	CS	IH	IH+IVV	R	CS	IH	IH+IVV
1	200	2003	1843	1865	1867	30	29	28	
2	193	1985	1836	1818	1816	30	29	28	
3	187	1890	1784	1769	1772	31	29	28	
4	194	1891	1816	1758	1757	31	28	27	
5	203	2171	2137	2053	2057	32	29	28	
6	194	2196	2103	2036	2045	34	31	30	
7	174	1996	1927	1946	1941	35	29	29	
$\bigcirc$	192,14	2018,86	1920,86	1892,14	1893,57	31,86	29,14	28,29	

Tabelle 6.19: Ergebnisse für reaktiven Unterstützereinsatz (A) im Worst-Case (R = Zufallsreihenfolge, CS = Car Sequencing, IH = Initialheuristik, IVV = Iteratives Verbesserungsverfahren)

$\mathcal{P}$	$ \mathcal{P} $	u^E				u^{MAX}			
		R	CS	IH	IH+IVV	R	CS	IH	IH+IVV
1	200	173	117	120	93	13	10	6	
2	193	181	149	127	90	14	10	7	
3	187	167	142	102	85	13	9	5	
4	194	174	138	120	91	13	9	8	
5	203	179	140	111	104	14	11	8	
6	194	198	150	142	131	15	11	11	
7	174	171	137	117	103	16	11	9	
$\bigcirc$	192,14	177,57	139,00	119,86	99,57	14,00	10,14	7,71	

Tabelle 6.20: Ergebnisse für präventiven Unterstützereinsatz (B) im Best-Case (R = Zufallsreihenfolge, CS = Car Sequencing, IH = Initialheuristik, IVV = Iteratives Verbesserungsverfahren)

$\mathcal{P}$	$ \mathcal{P} $	u^E				u^{MAX}			
		R	CS	IH	IH+IVV	R	CS	IH	IH+IVV
1	200	1672	1643	1626	1623	30	27	25	
2	193	1656	1634	1596	1596	30	27	26	
3	187	1586	1590	1575	1565	32	28	26	
4	194	1602	1598	1537	1535	34	27	25	
5	203	1814	1837	1781	1781	31	28	28	
6	194	1810	1818	1784	1784	34	30	28	
7	174	1648	1686	1595	1595	35	30	27	
$\bigcirc$	192,14	1684,00	1686,57	1642,00	1639,86	32,29	28,14	26,43	

Tabelle 6.21: Ergebnisse für präventiven Unterstützereinsatz (B) im Worst-Case (R = Zufallsreihenfolge, CS = Car Sequencing, IH = Initialheuristik, IVV = Iteratives Verbesserungsverfahren)

Die Laufzeit der Reihenfolgeoptimierung, inklusive der Initialheuristik und dem iterativen Verbesserungsverfahren, liegt im Durchschnitt bei 6,17 min.³⁴⁸

³⁴⁸Für die Tests wurde ein handelsüblicher PC mit einem Intel Core™ 2 CPU T5600, 1,83 GHz und 1GB Arbeitsspeicher verwendet.

Abhängigkeiten	Unterstützer	Vergleich zu R		Vergleich zu CS	
		u^E	u^{MAX}	u^E	u^{MAX}
Best-Case	reaktiv(A)	-47,89%	-50,96%	-22,85%	-43,92%
	präventiv(B)	-44,05%	-63,63%	-28,22%	-45,30%
Worst-Case	reaktiv(A)	- 6,21%	-26,62%	-1,35%	-11,05%
	präventiv(B)	- 2,32%	-30,76%	-2,44%	-17,91%
	○	-25,12%	-42,99%	-13,71%	-29,55%

Tabelle 6.22: Durchschnittliche Verbesserung der Heuristik (IH+IVV) im Vergleich zur zufälligen Reihenfolge (R) und zum Car Sequencing (CS)

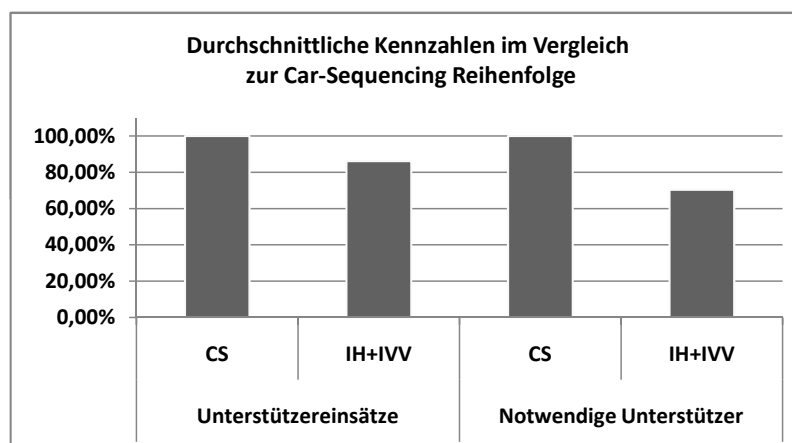
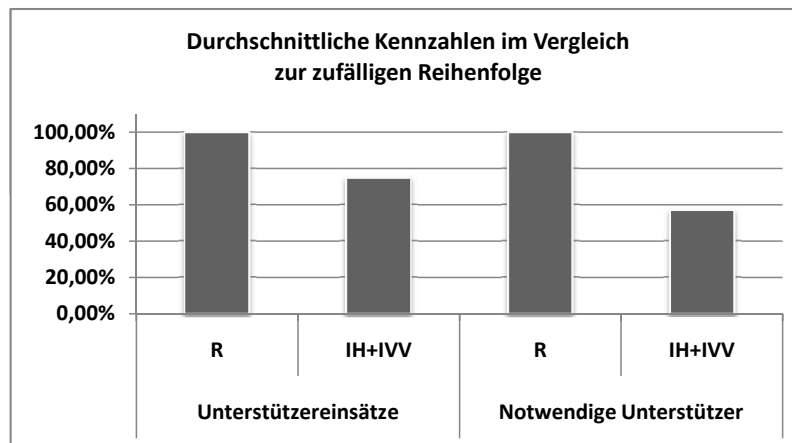


Abbildung 6.9: Vergleich der durchschnittlichen Kennzahlen für die Reihenfolgeheuristik

7 Zusammenfassung und Ausblick

So eine Arbeit wird eigentlich nie fertig,
man muß sie für fertig erklären,
wenn man nach Zeit und Umständen
das möglichste getan hat.

(Johann Wolfgang von Goethe)

In dieser Arbeit wurde ein Verfahren zur kostenoptimalen Kapazitätsabstimmung in der getakteten Variantenfließlinie entwickelt. Dabei wurden insbesondere der Unterstützereinsatz in der Fließlinie sowie die unterschiedlichen Planungszeiträume der zu treffenden Entscheidungen betrachtet. So wurde im ersten Schritt das Problem der Rekonfiguration einer Variantenfließlinie bei gegebenem Auftragsprogramm mit Hilfe eines Entscheidungsunterstützungssystem gelöst. Dabei wird in dem Verfahren erstmalig beachtet, dass Rekonfigurationskosten bei der Verschiebung von Arbeitsvorgängen entstehen und bereits eine Zuordnungslösung existiert. Im zweiten Schritt wurden Algorithmen für die Vorranggraphenerstellung entwickelt, die im Gegensatz zu existierenden Ansätzen einen Top-Down Ansatz verfolgen und somit garantieren können, dass lediglich valide Freiheitsgrade im Vorranggraphen enthalten sind. Im dritten Schritt wurden ein Simulator für getaktete Variantenfließlinien und eine Heuristik für die Optimierung der Produktionsreihenfolge vorgestellt. Hierbei wurde erstmals versucht, die Anzahl notwendiger Unterstützereinsatzkräfte explizit zu bewerten und zu minimieren. Alle drei Verfahren wurden implementiert und sind in der Praxis in Planungsmethoden eingeflossen. Die Testergebnisse zeigen eine durchschnittliche Reduzierung der Unterstützereinsätze um 37% durch die Rekonfiguration und weitere 25% durch die Reihenfolgeplanung. Die Anzahl an notwendigen Unterstützern konnte um 27% durch die Rekonfiguration und weitere 43% durch die Reihenfolgeplanung reduziert werden.

Zur Beschreibung des Problems wurde mit einer allgemeinen formalen Systembeschreibung begonnen, welche den Betrachtungsgegenstand der Arbeit beschreibt. Dabei wurde sowohl der Produktionsablauf als auch die Kostenstruktur in der getakteten Variantenfließlinie beschrieben. Darauf aufbauend wurden die einzelnen Teilaufgaben systematisiert.

Die Rekonfiguration stellte das erste Teilproblem dar, in welchem die Arbeitsvorgänge zu den Arbeitsplätzen und damit die Anzahl an Werkern und Betriebsmitteln bestimmt sowie

ihre Zuordnung zu den Arbeitsplätzen vorgenommen wird. Dabei wurde der Fokus auf den Zielkonflikt zwischen der Reduzierung von Betriebs- und Umstellungskosten gelegt. Das Ziel, die Restriktionen sowie die Voraussetzungen des Rekonfigurationsproblems wurden in Kapitel 2.2.1 definiert. Die konkreten Anforderungen an eine Problemlösung wurden daraufhin in Abschnitt 2.3.1 dargestellt. In Abschnitt 3.1 wurde der Stand der Forschung auf existierende Methoden für die Rekonfiguration hin untersucht. Der Bedarf für ein Verfahren zur Rekonfiguration wurde von einigen Autoren explizit konstatiert. Des Weiteren wurde die Übertragbarkeit von Zielsetzungen aus der Erstinstallation einer Fließlinie auf die Problemstellung der Rekonfiguration überprüft. Dabei wurden einige übertragbare Teile identifiziert und übernommen. In Abschnitt 5.1 wurde zuerst eine vollständige Formalisierung des Problems inklusive der Übertragung von Zielkriterien aus dem Stand der Forschung vorgenommen. Auf diesem mathematischen Modell basierend wurde dann ein Entscheidungsunterstützungssystem vorgestellt, welches den Planer in den Prozess integriert, um die Umstellungskosten für jede Verschiebung bewerten zu können. Durch dieses Konzept kann auf einer basierenden Planungslösung eine kostenoptimale Anpassung der Zuordnung für die nächste Periode vorgenommen werden. In Abschnitt 6.1 wurde das Verfahren erfolgreich angewendet und das resultierende Potential bewertet.

Der Vorranggraph stellt ein notwendiges Eingangsdatum für viele existierende Ansätze zur Fließlinienabstimmung dar. Auch in dem vorgestellten Verfahren zur Rekonfiguration kann der Vorranggraph verwendet werden, um den Suchraum der Algorithmen einzuschränken. In Abschnitt 2.2.2 wurde das Problem der Vorranggraphenerstellung diskutiert. Insbesondere bei variantenreichen Produkten scheitert die Vorranggraphenerstellung häufig am hohen Aufwand. Die Anforderungen an eine Vorranggraphenentwicklung, die auch bei variantenreichen Produkten den Aufwand gering hält, wurden in Abschnitt 2.3.2 definiert. Der Stand der Forschung zu manuellen sowie automatisierten Methoden der Vorranggraphenentwicklung wurde in Abschnitt 3.2 strukturiert und vorgestellt. Insbesondere der Aufbau auf einer existierenden Produktdokumentation wie der Stückliste wurde dort als viel versprechendes Vorgehen identifiziert. In Abschnitt 5.2 wurde ein solches Verfahren entwickelt. Um sicherzustellen, dass lediglich valide Freiheitsgrade in den Vorranggraphen integriert werden, wurde ein Ansatz entwickelt, der Top-Down vorgeht und auf Basis von alten Zuteilungslösungen sowie Code-Regeln aus einer offenen Variantenstückliste arbeitet. Das Verfahren wurde in Abschnitt 6.2 getestet und innerhalb der Rekonfigurationsworkshops aus 6.1 verwendet. Die Methode lässt sich für jedes Problem anwenden, in dem ein Vorranggraph benötigt wird und bereits eine Stücklistenstruktur bzw. alte Zuordnungslösungen und damit Arbeitsreihenfolgen existieren.

Die kurzfristiger zu treffenden Entscheidungen, wie die Bestimmung und Allokation der notwendigen Unterstützereinsatzkräfte sowie die Bestimmung der Produktionsreihenfolge wurden in Abschnitt 2.2.3 als drittes Teilproblem identifiziert und definiert. Die Anforderungen

an ein solches Verfahren wurden in Abschnitt 2.3.3 beschrieben. In Abschnitt 3.3 wurden existierende Ansätze aus dem Mixed-Model-Sequencing auf ihre Eignung zur Lösung des beschriebenen Problems untersucht. Dabei wurde kein Verfahren gefunden, welches die Anzahl an notwendigen Unterstützerarbeitskräften beachtet bzw. die praktischen Gegebenheiten ausreichend detailliert abbildet. Insbesondere der Bedarf einer Bewertungsmethode für die Reihenfolgeplanungen wurde von einigen Autoren erwähnt. In Abschnitt 5.3 wurde daher zuerst das Problem als mathematisches Optimierungsproblem formalisiert und darauf aufbauend ein Fließliniensimulator entwickelt, der den Unterstützerbedarf ermittelt. Daraufhin wurde eine Heuristik zur Reduzierung des Unterstützereinsatzes entwickelt, die die Bewertungsmethode nutzt. Das Verfahren wurde in Abschnitt 6.3 getestet und validiert.

Die Verfahren können unabhängig voneinander eingesetzt werden. Während sich die Testfälle auf die Automobilindustrie beziehen, ist eine Übertragbarkeit immer dann gegeben, wenn die Anforderungen aus Kapitel 2 übereinstimmen. Dies ist z.B. bei der Herstellung von „weißer Ware“ wie z.B. Waschmaschinen der Fall.

In dieser Arbeit wurden die Auswirkungen der Rekonfiguration auf den jeweils folgenden Planungszeitraum bewertet. Die längerfristigen Auswirkungen wurden dabei außer Acht gelassen und könnten in zukünftigen Forschungsvorhaben in Betracht gezogen werden. Dabei ist es insbesondere untersuchenswert, wie konstant die Korrelation von Arbeitsvorgängen und damit das Bestellverhalten der Kunden über die Zeit ist.

Der vorgestellte Ansatz zur Erstellung eines Vorranggraphen macht es erstmals möglich, mit geringem Aufwand einen validen Graphen abzuleiten. Weitere Forschung ist jedoch notwendig, um sämtliche Freiheitsgrade in den Graphen zu integrieren. Dabei sollten insbesondere aufwandsreduzierende Methoden zur manuellen Feinstrukturierung des Vorranggraphen entwickelt werden.

Im Bereich der Reihenfolgeplanung fällt in der Praxis auf, dass eine geplante Reihenfolge häufig kurz vor Produktionsbeginn angepasst werden muss, weil kurzfristige Störungen wie z.B. Teilesperren auftreten. Hierfür sollten in Zukunft leistungsfähige belastungsorientierte Resequenzierungsmethoden entwickelt werden, die mit diesen Störungen umgehen können, dabei jedoch die ursprünglich erstellte Reihenfolge möglichst intakt halten.

Mit dem weiter zunehmenden Konkurrenzdruck in den Märkten müssen auch die bestehenden Produktionskonzepte überdacht werden, um weitere Potentiale zur Effizienzsteigerung zu verwirklichen. Möglichkeiten dafür sind z.B. die Modularisierung von Produkten, die weitere Konzentration auf das Kerngeschäft oder auch die Verfolgung von Gleichteilestrategien. Die Auswirkungen dieser strategischen Entscheidungen auf die operative Produktionsebene können auf Basis der erstellten Fließliniensimulation bewertet werden. Zum Beispiel die Frage, welche Module innerhalb einer Nebenlinie vorbereitet werden und welche Arbeitsvorgänge in einer Hauptlinie verbleiben sollten, um die Notwendigkeit für Unterstützereinsätze und

Schwankungen in den Bearbeitungszeiten weiter zu reduzieren, könnte den Fokus für folgende Forschungsvorhaben bilden. Die teilautomatisierte Vorranggraphenbildung kann dazu genutzt werden, mögliche Produktstrukturierungen zu identifizieren.

Literaturverzeichnis

- [AH94] AHMADI, R.H. und W. HERNAN: *Design for Synchronized Flow Manufacturing*. Management Science, 40(11):1469–1483, 1994.
- [Ame96] AMEN, M.: *Ein exaktes Verfahren zur kostenorientierten Fließbandabstimmung*. Operations Research Proceedings, Seiten 224–229, 1996.
- [Ame98] AMEN, M.: *Heuristic methods for cost-oriented assembly line balancing-A survey*. International Journal of Production Economics, 68(1):1–14, 1998.
- [Ame00] AMEN, M.: *An exact method for cost-oriented assembly line balancing*. International Journal of Production Economics, 64:187–195, 2000.
- [Ame01] AMEN, M.: *Heuristic methods for cost-oriented assembly line balancing: A comparison on solution quality and computing time*. International Journal of Production Economics, 69:255–264, 2001.
- [Ame06] AMEN, M.: *Cost-oriented assembly line balancing: model formulations, solution difficulty, upper and lower bounds*. European Journal of Operational Research, 168:747–770, 2006.
- [Amm85] AMMER, E.D.: *Rechnerunterstützte Planung von Montageablaufstrukturen für Erzeugnisse der Serienfertigung*. Springer-Verlag, 1985.
- [Arc66] ARCUS, A.L.: *COMSOAL: A computer method of sequencing operations for assembly lines*. International Journal of Production Research, 4:259–277, 1966.
- [Asi87] ASIM, R.: *From what if to what's best*. Decision Support Systems, 3(1):27–35, 1987.
- [AV06] ARTHUR, D. und S. VASSILVITSKII: *K-Means++: The Advantages of Careful Seeding*. Technischer Bericht, Stanford University, 2006.
- [Bac94] BACHER, J.: *Clusteranalyse*. Oldenbourg Verlag GmbH, München, 1994.
- [BAL⁺91] BALDWIN, D.F., T.E. ABELL, M.-C. LUI, T.L. DE FAZIO und D.E. WHITNEY: *An integrated computer aid for generating and evaluating assembly sequences for mechanical products*. IEEE Trans. Robot. and Automat., 7:78–94, 1991.

- [Bay86] BAYBARS, I.: *A survey of exact algorithms for the simple assembly line balancing problem*. Management Science, 32, 1986.
- [BDES92] BARD, J.F., E. DAR-EL und A. SHTUB: *An analytic framework for sequencing mixed model assembly lines*. International Journal of Production Research, 30:35–48, 1992.
- [BE96] BARTHOLDI, J.J. und D.D. EISENSTEIN: *A Production Line that balances itself*. Operations Research, 44(1):21–35, January-February 1996.
- [Bec07] BECKER, H.: *Auf Crashkurs - Automobilindustrie im globalen Verdrängungswettbewerb* -, Band 2. Springer Berlin Heidelberg, 2007.
- [Ben08] BENOIST, T.: *Soft car sequencing with colors: Lower bounds and optimality proofs*. European Journal of Operational Research, 191:957–971, 2008.
- [BEPW06] BACKHAUS, K., B. ERICHSON, W. PLINKE und R. WEIBER: *Multivariate Analysemethoden - Eine anwendungsorientierte Einführung*. Springer, 2006.
- [BFS06] BOYSEN, N., M. FLIEDNER und A. SCHOLL: *A classification of assembly line balancing problems*. European Journal of Operational Research, 2006.
- [BFS07a] BOYSEN, N., M. FLIEDNER und A. SCHOLL: *Assembly line balancing: Joint precedence graphs under high product variety*. Technischer Bericht, Friedrich-Schiller-University Jena, 2007.
- [BFS07b] BOYSEN, N., M. FLIEDNER und A. SCHOLL: *Production Planning of Mixed-Model Assembly Lines: Overview and Extensions*. Technischer Bericht, Friedrich-Schiller-University Jena, 2007.
- [BFS07c] BOYSEN, N., M. FLIEDNER und A. SCHOLL: *Produktionsplanung bei Variantenfließfertigung: Planungshierarchie und Elemente einer Hierarchischen Planung*. Zeitschrift für Betriebswirtschaft, 77:759–793, 2007.
- [BFS07d] BOYSEN, N., M. FLIEDNER und A. SCHOLL: *Sequencing Mixed-Model Assembly Lines: Survey, Classification and Model Critique*. Technischer Bericht, Friedrich-Schiller-University Jena, 2007.
- [BFS07e] BOYSEN, N., M. FLIEDNER und A. SCHOLL: *Sequencing mixed-model assembly lines to minimize part inventory cost*. OR Spectrum, 2007.
- [BFS08] BOYSEN, N., M. FLIEDNER und A. SCHOLL: *Assembly Line balancing Which model to use when*. International Journal of Production Economics, 2008.

- [BGLS95] BULLINGER, H.J., M. GOMMEL, C.-U. LOTT und M. SCHMAUDER: *Arbeitsgestaltung: Personalorientierte Gestaltung marktgerechter Arbeitssysteme*. Teubner, Stuttgart, 1995.
- [BNM08] BRIANT, O., D. NADDEF und G. MOUNIÉ: *Greedy approach and multi-criteria simulated annealing for the car sequencing problem*. European Journal of Operational Research, 191:993–1003, 2008.
- [Boc99] BOCK, S.: *Modelle und verteilte Algorithmen zur Planung getakteter Fließlinien*. Doktorarbeit, University of Paderborn, 1999.
- [Bol97] BOLAT, A.: *Stochastic procedures for scheduling minimum job sets on mixed model assembly lines*. Journal of the Operational Research Society, 48:490–501, 1997.
- [Bou84] BOURJAUULT, A.: *Contribution a une Approch Methodology de L’assemblage Automatisé: Elaboration Automatique des Sequences Operatoires*. Doktorarbeit, Universite de Franche-Comte, 1984.
- [Boy05a] BOYSEN, N.: *Reihenfolgeplanung bei Variantenfließfertigung: Ein integrativer Ansatz*. Zeitschrift für Betriebswirtschaft, 75:135–156, 2005.
- [Boy05b] BOYSEN, N.: *Variantenfließfertigung*. Doktorarbeit, Universität Hamburg, Juli 2005.
- [BR06] BUKCHIN, Y. und I. RABINOWITCH: *A branch-and-bound based solution approach for the mixed-model assembly line-balancing problem for minimizing stations and task duplication costs*. European Journal of Operational Research, 174:492–508, 2006.
- [Bra95] BRAUN, W.J.: *Beitrag zur Festlegung der Arbeitsteilung in manuellen Arbeitssystemen*. Doktorarbeit, Universität Karlsruhe, 1995.
- [BRvB06] BOCK, S., O. ROSENBERG und T. VAN BRACKEL: *Controlling mixed-model assembly lines in real-time by using distributed systems*. European Journal of Operational Research, 168:880–904, 2006.
- [Brä03] BRÄUTIGAM, L.P.: *Ansätze zu einer Theorie des Kostenverhaltens bei Variantenproduktion*. Doktorarbeit, Universität Jena, 2003.
- [BS93] BAUMGARTEN, H. und F. STRAUBE: *Gezähmte Individualität*. Logistik heute, 9:60 – 64, 1993.
- [BS03] BECKER, C. und A. SCHOLL: *A survey on problems and methods in generalized assembly line balancing*. Technischer Bericht, Friedrich-Schiller-Universität Jena, 2003.

- [BS06] BECKER, C. und A. SCHOLL: *A survey on problems and methods in generalized assembly line balancing*. European Journal of Operational Research, 168:694–715, 2006.
- [BSAF94] BOLAT, A., M. SAVSAR und M.A. AL-FAWZAN: *Algorithms for the real-time scheduling of jobs on mixed model assembly lines*. Computers & Operations Research, 21:487–498, 1994.
- [BSJ94] BARD, J.F., A. SHTUB und S.B. JOSHI: *Sequencing mixed-model assembly lines to level parts usage and minimize line length*. International Journal of Production Research, 32:2431–2454, 1994.
- [Buk98] BUKCHIN, J.: *A comparative study of performance measures for throughput of a mixed model assembly line in a JIT environment*. International Journal of Production Research, 36(10):2669–2685, 1998.
- [Bul86] BULLINGER, H.J.: *Systematische Montageplanung*. Carl Hanser Verlag, München, 1986.
- [BY92a] BOLAT, A. und C. YANO: *Scheduling algorithms to minimize utility work at a single station on a paced assembly line*. Production Planning & Control, 3:393–405, 1992.
- [BY92b] BOLAT, A. und C. YANO: *A surrogate objective for utility work in paced assembly lines*. Production Planning & Control, 3:406–412, 1992.
- [Che89] CHEN, C.L.: *Automatic Assembly Sequences Generation by Pattern-Matching*. Technischer Bericht, School of Engineering and Technology, Electrical Engineering and CAD/CAM Center, Purdue University, Indianapolis, 1989.
- [CLP08] CORDEAU, J.F., G. LAPORTE und F. PASIN: *Iterated tabu search for the car sequencing problem*. European Journal of Operational Research, 191:945–956, 2008.
- [CNY03] CHUTIMA, P., W. NIMMANO und C. YIANGKAMOLSING: *Application of fuzzy genetic algorithm for sequencing in mixed-model assembly line with processing time*. International Journal of Industrial Engineering, 10:325–331, 2003.
- [CSC93] CHO, Y., C.K. SHIN und H.S. CHO: *Automated inference on stable robotics assembly sequences based upon the evaluation of base assembly motion instability*. Robotica, 11:351–362, 1993.
- [Dan03] DANGELMAIER, W.: *Produktion und Information - System und Modell*. Springer, 2003.

- [Dan08] DANGELMAIER, W.: *Produktionsplanung und -steuerung*. Vorlesungsskript, März 2008.
- [Dan09] DANGELMAIER, W.: *Theorie der Produktionsplanung und -steuerung - Im Sommer keine Kirschpralinen?* Springer, Berlin, 2009.
- [DE78] DAR-EL, E.M.: *Mixed-model assembly line sequencing problems*. OMEGA, 6:313–323, 1978.
- [DEC75] DAR-EL, E.M. und R.F. COTHER: *Assembly line sequencing for model mix*. International Journal of Production Research, 13:463–477, 1975.
- [DEC77] DAR-EL, E.M. und S. CUCUY: *Optimal model-mixed sequencing for balanced assembly lines*. OMEGA, 5:333–341, 1977.
- [Dec92] DECKER, M.: *Planning of buffers within mixed-model lines*. OR Spectrum, 14:201–209, 1992.
- [Dec93] DECKER, M.: *Variantenfließfertigung*. Physica-Verlag, Heidelberg, 1993.
- [DEN81] DAR-EL, E.M. und A. NADIVI: *A mixed-model sequencing application*. International Journal of Production Research, 19:69–84, 1981.
- [Deu71] DEUTSCH, D.F.: *A branch and bound technique for mixed-product assembly line balancing*. Doktorarbeit, Arizona State University, 1971.
- [DFW87] DE FAZIO, T.L. und D.E. WHITNEY: *Simplified generation of all mechanical assembly sequences*. IEEE Journal of Robotics and Automation, RA-3(6):640–658, 1987.
- [DH08] DING, F.-Y. und J. HE: *A heuristic procedure for the automobile assembly-line sequencing problem considering multiple product options*. International Journal of Production Research, 46(20):5827 – 5847, 2008.
- [Die95] DIECKHOFF, C.: *Darstellung und Bewertung alternativer Automobil-Montagestrukturen mit einem parallelrechnerunterstützten Simulationssystem*. Doktorarbeit, Institut für Fabrikbetriebslehre und Unternehmensforschung, 1995. Düsseldorf.
- [DKS96] DOMSCHKE, W., R. KLEIN und A. SCHOLL: *Antizipative Leistungsabstimmung bei moderner Variantenfließfertigung*. Zeitschrift für Betriebswirtschaft, 66:1465–1490, 1996.
- [DS92] DINI, G. und M. SANTOCHI: *Automated sequencing and subassembly detection in assembly planning*. Annals CIRP, 41:1–4, 1992.

- [DSV93] DOMSCHKE, W., A. SCHOLL und S. VOSS: *Produktionsplanung*. Springer Verlag, Berlin, 1993.
- [DSV97] DOMSCHKE, W., A. SCHOLL und S. VOSS: *Produktionsplanung - Ablauforganisatorische Aspekte*. Springer, Berlin, 2 Auflage, 1997.
- [DW97] DANGELMAIER, W. und H.-J. WARNEKE: *Fertigungslenkung*. Springer Verlag, Berlin, 1997.
- [Eck03] ECKSTEIN, P.P.: *Repetitorium Statistik*. Gabler, 2003.
- [EG99] EREL, E. und H. GOKCEN: *Shortest-route formulation of mixed-model assembly line balancing problem*. European Journal of Operational Research, 116:194–204, 1999.
- [Fal05] FALKENAUER, E.: *Line Balancing in the Real World*. In: *International Conference on Product Lifecycle Management*, 2005.
- [Fel80] FELBECKER, O.: *Ein Beitrag zur Reihenfolgeplanung der Mehrprodukt-Linienfertigung*. Doktorarbeit, Rheinisch-Westfälische Technische Hochschule Aachen, 1980.
- [FF89] FANDEL, G. und P. FRANCOIS: *Just-in-Time-Produktion und -Beschaffung, Funktionsweise, Einsatzvoraussetzungen und Grenzen*. Zeitschrift für Betriebswirtschaft, 59:531–544, 1989.
- [Fin89] FINLAY, P.N.: *Introducing Decision Support Systems*. NCC Blackwell, Oxford, 1989.
- [Fle88] FLEISCHMANN, B.: *Operations-Research-Modelle und -verfahren in der Produktionsplanung*. Zeitschrift für Betriebswirtschaft, 58:347–372, 1988.
- [Gav08] GAVRANOVIC, H.: *Local search and suffix tree for car-sequencing problem with colors*. European Journal of Operational Research, 191:972–980, 2008.
- [GBT97] GOLDSCHMIDT, O., J.F. BARD und A. TAKVORIAN: *Complexity results for mixed-model assembly lines with approximation algorithms for the single station case*. The International Journal of Flexible Manufacturing Systems, 9:251–272, 1997.
- [GGR06] GAMBERINI, R., A. GRASSI und B. RIMINI: *A new multi-objective heuristic algorithm for solving the stochastic assembly line re-balancing problem*. International Journal of Production Economics, 102:226–243, 2006.
- [Got06a] GOTTSCHALK, S.: *Lean in der Einzel-, und Kleinserienfertigung*. In: *Lean Management Summit*, Band 3, Seite S. 173 ff., 2006.

- [Got06b] GOTTSCHALK, S.F.: *Dedicated Flexibility - Komplexitätsoptimale Gestaltung manueller Serienmontage*. Doktorarbeit, Rheinisch-Westfälische Technische Hochschule Aachen, 2006.
- [GT04] GÜNTHER, H.-O. und H. TEMPELMEIER: *Produktion und Logistik*, Band 6. Springer, Berlin, September 2004.
- [Gör78] GÖRKE, M.: *Rechnerunterstütztes Verfahren zur Leistungsabstimmung von Mehrmodell-Montagesystemen*. Doktorarbeit, Universität Stuttgart - Fak. Fertigungstechnik, Inst. für Industrielle Fertigung und Fabrikbetrieb, Mainz, 1978.
- [Hac89] HACKSTEIN, R.: *Produktionsplanung und -steuerung*. VDI-Verlag, Düsseldorf, 2. Auflage, 1989.
- [Har74] HARDECK, W.: *Rechnerunterstützte Austaktung von Fließbandlinien*. Zeitschrift für Operations Research, 18:B 237–B 249, 1974.
- [Har99] HARTUNG, J.: *Statistik*, Band 12. Oldenbourg Verlag, 1999.
- [HB91] HENRIOUD, J.M. und A. BOURJAULT: *LEGA: A computer-aided generator of assembly plans*. Computer-Aided Mechanical Assembly Planning, Seiten 191–215, 1991.
- [HdMS91] MELLO, L.S. HOMEM DE und A.C. SANDERSON: *A correct and complete algorithm for the generation of mechanical assembly sequences*. IEEE Transactions on Robotics and Automation, 7:228–240, 1991.
- [Hel93] HELLBERG, K.: *Methoden zur automatischen Erzeugung von Arbeitsgangfolgen*. VDI Verlag, 1993.
- [HL89] HUANG, F. und C.S.G. LEE: *Precedence knowledge in feature mating operation assembly planning*. IEEE International Conference on Robotics and Automation, Seiten 62–68, May 1989.
- [Hot06] HOTMAN, E.: *Decision Support Expert System - for Process Selection*. Doktorarbeit, Faculty of Electrical Engineering, Computer Science and Mathematics - University of Paderborn, 2006.
- [HSMZ00] HALLER, E., E.F. SCHILLER, R. MÜLLER und G. ZÜLCH: *Leistungsabstimmung von PKW-Montagelinien*. FB/IE Zeitschrift für Unternehmensentwicklung und Industrial Engineering, 49:4–11, 2000.
- [JK84] JENTSCH, W. und F. KADEN: *Automatic Generation of Assembly Sequences*. Artificial Intelligence and Information-Control Systems of Robots, Seiten 197–200, 1984.

- [JWC97] JONES, R.E., R.H. WILSON und T.L. CALTON: *Constraint-Based Interactive Assembly Planning*. Proceedings of the 1997 IEEE International Conference on Robotics and Automation, Seiten 913–920, 1997.
- [Kau96] KAUFMANN, H.: *Multivariate statistische Verfahren*, Kapitel Clusteranalyse, Seiten 437–536. de Gruyter, 1996.
- [Ker62] KERN, W.: *Die Messung industrieller Fertigungskapazitäten und ihre Ausnutzung*. Westdeutscher Verlag, 1962.
- [KHK96] KIM, Y.K., C.J. HYUN und Y. KIM: *Sequencing in mixed model assembly lines: A genetic algorithm approach*. Computers & Operations Research, 23:1131–1145, 1996.
- [KKK00a] KIM, Y.K., J.Y. KIM und Y. KIM: *Balancing and sequencing mixed-model U-lines with a co-evolutionary algorithm*. Production Planning & Control, 11(8):754–764, 2000.
- [KKK00b] KIM, Y.K., J.Y. KIM und Y. KIM: *A Coevolutionary Algorithm for Balancing and Sequencing in Mixed Model Assembly Lines*. Technischer Bericht 3, Hingham, MA, USA, 2000.
- [KKK06] KIM, Y.K., J.Y. KIM und Y. KIM: *An endosymbiotic evolutionary algorithm for the integration of balancing and sequencing in mixed-model U-lines*. European Journal of Operational Research, 168:838–852, 2006.
- [KMSO06] KIENER, S., N. MAIER-SCHEUBECK und R. OBERMAIER: *Produktionsmanagement*. Oldenbourg, 2006.
- [Koe85] KOETHER, R.: *Verfahren zur Verringerung von Modell-Mix-Verlusten in Fließmontagen*. Doktorarbeit, Universität Stuttgart, 1985.
- [KOP07] KARA, Y., U. OZCAN und A. PEKER: *An approach for balancing and sequencing mixed-model JIT U-lines*. The International Journal of Advanced Manufacturing Technology, 32(11-12):1218–1231, May 2007.
- [KSW96] KERN, W., H.H. SCHRÖDER und J. WEBER (Herausgeber): *Handwörterbuch der Produktionswirtschaft*. Schäffer-Poeschel, 1996.
- [KWJC96] KAUFMAN, S.G., R.H. WILSON, R.E. JONES und T.L. CALTON: *The Archimedes 2 Mechanical Assembly Planning System*. Proceedings of the 1996 IEEE International Conference on Robotics and Automation, Seiten 3361–3368, 1996.
- [Lan90] LANCASTER, K.: *The economics of product variety: A survey*. Marketing Science, 9:189–206, 1990.

- [Lau99] LAUTENSCHLAEGER, M.: *Mittelfristige Produktionsprogrammplanung mit auslastungsabhängigen Vorlaufzeiten*. Doktorarbeit, Technische Hochschule Darmstadt, 1999.
- [LC93] LIN, C. und T.C. CHANG: *3D MAPS: Three-dimensional mechanical assembly planning system*. Journal of Manufacturing Systems, 12:437–456, 1993.
- [Leo97] LEOPOLD, N.: *Ein Planungsverfahren zur Kapazitätsabstimmung für Modell-Mix-Montagelinien am Beispiel einer Automobil-Endmontage*. Doktorarbeit, Universität Stuttgart, 1997.
- [Mac72] MACASKILL, J.L.C.: *Production-Line Balances for mixed-model lines*. Management Science, 19:423–434, 1972.
- [Mei07] MEISSNER, R. UND MÜLLER, M.: *Herausforderungen an die Integration von Produktentwicklungs- & Produktionsplanungsprozess zur Entwicklung serienprozessfähiger Produkte*. Symposium Design for X, 18:183–190, 2007.
- [Mey04] MEYR, H.: *Supply chain planning in the German automotive industry*. OR Spectrum, 26:447–470, 2004.
- [MF98] MCMULLEN, P.R. und G.V. FRAZIER: *Using Simulation and Data Envelopment Analysis to Compare Assembly Line Balancing Solutions*. Journal of Productivity Analysis, 11:149–168, 1998.
- [Mil93] MILLING, P.: *Betriebswirtschaftliche Grundlagen moderner Produktionsstrukturen*, Kapitel Personalwirtschaft in der Fabrik der Zukunft. Herne, 1993.
- [Mil02] MILTENBURG, J.: *Balancing and Scheduling Mixed-Model U-Shaped Production Lines*. International Journal of Flexible Manufacturing Systems, 14:119–151(33), April 2002.
- [MNP99] MERENGO, C., F. NAVA und A. POZZETTI: *Balancing and sequencing manual mixed-model assembly lines*. International Journal of Production Research, 37(12):2835–2860, 1999.
- [Mol96] MOLLEMEIER, A.: *Integrierte Steuerung getakteter Variantenfließlinien*. Doktorarbeit, Universität Paderborn, 1996.
- [Mon98] MONDEN, Y.: *Toyota Production System*. B&T, 3. Auflage, 1998.
- [MT03] MCMULLEN und P. TARASEWICH: *Using ant techniques to solve the assembly line balancing problem*. IIE Transactions, 35:605–617, 2003.

- [MY65] MOODIE, C.L. und H.H. YOUNG: *A heuristic method of assembly line balancing for assumptions of constant or variable work element times*. Journal of Industrial Engineering, 16:23–29, 1965.
- [MY01] MATANACHAI, S. und C.A. YANO: *Balancing mixed-model assembly lines to reduce work overload*. IIE Transactions, 33:29–42, 2001.
- [Mül02] MÜLLER, R.: *Planung hybrider Montagesysteme auf Basis mehrschichtiger Vorranggraphen*. Doktorarbeit, Universität Karlsruhe, Februar 2002.
- [NM93] NEUMANN, K. und M. MORLOCK: *Operations Research*. Carl Hanser Verlag, 1993.
- [Orl75] ORLICKY, J.: *Materials Requirements Planning*. McGraw-Hill, 1975.
- [OY79] OKAMURA, K. und H. YAMASHINA: *A heuristic algorithm for the assembly line model-mix sequencing problem to minimize the risk of stopping the conveyor*. International Journal of Production Research, 17:233–247, 1979.
- [Par88] PARRELLO, B.D.: *CAR WARS: The (almost) birth of an expert system*. AI Expert, 3(1):60–64, 1988.
- [Phi88] PHILIPPAKIS, A.S. (Herausgeber): *Structured what if analysis in DSS models*, Band 3, Kailua-Kona, HI, USA, 1988. Coll. of Bus., Arizona State Univ., Tempe, AZ; System Sciences - Decision Support and Knowledge Based Systems Track, Proceedings of the Twenty-First Annual Hawaii International Conference.
- [Pic75] PICHLER, FRANZ: *Mathematische Systemtheorie - Dynamische Konstruktionen*. De Gruyter Lehrbuch, 1975.
- [PKW86] PARRELLO, B., W. KABAT und L. WOS: *Job-shop scheduling using automated reasoning: A case study of the car-sequencing problem*. Journal of Automated Reasoning, 2:1–42, 1986.
- [PR08] PRANDTSTETTER, M. und G.R. RAIDL: *An integer linear programming approach and a hybrid variable neighborhood search for the car sequencing problem*. European Journal of Operational Research, 191:1004–1022, 2008.
- [PW97] PINNOI, A. und W.E. WILHELM: *A Family of hierarchical Models for the design of Deterministic Assembly Lines*. International Journal of Production Research, 35(1):253–280, 1997.
- [RAN⁺08a] RIBEIRO, C.C., D. ALOISE, T.F. NORONHA, C. ROCHA und S. URRUTIA: *An efficient implementation of a VNS/ILS heuristic for a real-life car sequencing problem*. European Journal of Operations Research, 191:596–611, 2008.

- [RAN⁺08b] RIBEIRO, C.C., D. ALOISE, T.F. NORONHA, C. ROCHA und S. URRUTIA: *A hybrid heuristic for a multi-objective real-life car sequencing problem with painting and assembly line constraints*. European Journal of Operational Research, 191:981–992, 2008.
- [RDDB02] REKIEK, B., A. DOLGUI, A. DELCHAMBRE und A. BRATCU: *State of art of optimization methods for assembly line design*. Annual Reviews in Control, 26:163–174, 2002.
- [REF84] REFA: *Methodenlehre des Arbeitsstudiums: Teil 1 Grundlagen*. Hanser, München, 7. Auflage, 1984.
- [Ros74] ROSEN, S.: *Hedonic prices and implicit markets: Product differentiation in pure competition*. Journal of Political Economics, 82:34–55, 1974.
- [RT91] RACHAMADUGU, R. und B. TALBOT: *Improving the equality of workload assignments in assembly lines*. International Journal of Production Research, 29:619–633, 1991.
- [RV70] ROBERTS, S.D. und C.D. VILLA: *On a multiproduct assembly line - balancing problem*. AIIE Transactions, 2:361–364, 1970.
- [Röd06] RÖDER, A.: *A methodology for modeling inter-company supply chains and for evaluating a method of integrated product and process documentation*. European Journal of Operational Research, 169:1010–1029, March 2006.
- [Sal55] SALVESON, M.E.: *The assembly line balancing problem*. Journal of Industrial Engineering, 6:18–25, 1955.
- [Saw02] SAWIK, T.: *Monolithic vs. hierarchical balancing and scheduling of a flexible assembly line*. European Journal of Operational Research, 143(1):115–124, November 2002.
- [SBF06] SCHOLL, A., N. BOYSEN und M. FLIEDNER: *The sequence-dependent assembly line balancing problem*. OR Spectrum, 2006.
- [SC82] SPRAGUE, R.H. und E.D. CARLSON: *Building Effective Decision Support Systems*. University of Hawaii IBM Research Laboratory, San Jose, 1 Auflage, February 1982.
- [Sch89] SCHULTE, C.: *Produzieren Sie zu viele Varianten?* Harvard Manager, 11:60–66, 1989.
- [Sch95] SCHOLL, A.: *Balancing and sequencing of assembly lines*. Physica-Verlag, Heidelberg, 2. Auflage, February 1995.

- [Sch99] SCHULTE, C.: *Lexikon der Logistik*. Oldenbourg, München, 1999.
- [Sch01] SCHOLL, A.: *Robuste Planung und Optimierung*. Physica-Verlag, Heidelberg, 2001.
- [Sch06] SCHUH, G.: *Produktionsplanung und -steuerung*. Springer, 2006.
- [SCNA08] SOLNON, C., V.D. CUNG, A. NGUYEN und C. ARTIGUES: *The car sequencing problem: overview of state-of-the-art methods and industrial case-study of the ROADEF'2005 challenge problem*. European Journal of Operational Research, 191:912–927, 2008.
- [SD92] SANTOCHI, M. und G. DINI: *Computer-aided planning of assembly operations: the selection of assembly sequences*. Robotics and Computer-Integrated Manufacturing, 9:439–446, 1992.
- [Sei98] SEIDEL, U.A.: *Verfahren zur Generierung und Gestaltung von Montageablaufstrukturen komplexer Serienerzeugnisse*. Doktorarbeit, Universität Stuttgart, 1998.
- [SH98] SARKER, B.R. und PAN H.: *Designing a Mixed-Model assembly line to minimize the costs of Idle and Utility Times*. Computers & Industrial Engineering, 34(3):609–628, 1998.
- [SHdMZ90] SANDERSON, A.C., L.S. HOMEM DE MELLO und H. ZHANG: *Assembly Sequence Planning*. AI Magazine, 11:62–81, 1990.
- [Sin03] SINZ, C.: *Verifikation regelbasierter Konfigurationssysteme*. Doktorarbeit, Fakultät für Informations- und Kognitionswissenschaften der Eberhard-Karls-Universität Tübingen, 2003.
- [SKD98] SCHOLL, A., R. KLEIN und W. DOMSCHKE: *Pattern Based Vocabulary Building for Effectively Sequencing Mixed-Model Assembly Lines*. Journal of Heuristics, 4(4):359–381, Dezember 1998.
- [SM98] SPARLING, D. und J. MILTENBURG: *The mixed-model U-line balancing problem*. International Journal of Production Research, 36:485–501, 1998.
- [SM06] SUHL, L. und T. MELLOULI: *Optimierungssysteme*. Springer Verlag, 2006.
- [Smi98] SMITH, C.L.: *Computer-Supported Decision Making - Meeting the Decision Demands of Modern Organizations*. Ablex Publishing Corporation, 1998.
- [SOC00] SUMICHRIST, R.T., K.A. OXENRIDER und E.R. CLAYTON: *An Evolutionary Algorithm for Sequencing Production on a Paced Assembly Line*. Decision Sciences, 31:149–172, March 2000.

- [Sol08] SOLNON, C.: *Combining two pheromone structures for solving the car sequencing problem with Ant Colony Optimization*. European Journal of Operational Research, 191:1043–1055, 2008.
- [SP01] SARKER, B.R. und H. PAN: *Designing a mixed-model, open-station assembly line using mixed-integer programming*. Journal of Operational Research Society, 52:545–558, 2001.
- [SS91] SCHNEEWEISS, C. und V. SÖHNER: *Kapazitätsplanung bei moderner Fließfertigung*. Link Physica, 1991.
- [Ste56] STEINHAUS, H.: *Sur la division des corp materiels en parties*. Bull. Acad. Polon. Sci., IV:801–804, 1956.
- [Ste77] STEFFEN, R.: *Produktionsplanung bei Fließbandfertigung*. Doktorarbeit, Wiesbaden, 1977.
- [Sys07] SYSKA, A.: *Produktionsmanagement - Das A — Z wichtiger Methoden und Konzepte für die Produktion von heute*, Kapitel Kundenentkopplungspunkt, Seiten 83–84. Gabler, 2007.
- [TA98] TURBAN, E. und J. E. ARONSON: *Decision Support Systems and Intelligent Systems*. Prentice-Hall, 5. Auflage, 1998.
- [Tho67] THOMOPOULOS, N.T.: *Line Balancing-Sequencing for Mixed-Model Assembly*. Management Science, 14(2):59–75, Oktober 1967.
- [Tho70] THOMOPOULOS, N.T.: *Mixed Model Line Balancing with smoothed station Assignments*. Management Science, 16:593–605, 1970.
- [Tsa95] TSAI, L.-H.: *Mixed-model sequencing to minimize utility work and the risk of conveyor stoppage*. Management Science, 41:485–495, 1995.
- [VS02] VILARINHO, P.M. und A.S. SIMARIA: *A two-stage heuristic method for balancing mixed-model assembly lines with parallel workstations*. International Journal of Production Research, 40(6):1405–1420, 2002.
- [VS06] VILARINHO, P.M. und A.S. SIMARIA: *ANTBAL: an ant colony optimization algorithm for balancing mixed-model assembly lines with parallel workstations*. International Journal of Production Research, 44:291–303, 2006.
- [Wil95] WILSON, R.H.: *Minimizing User Queries in Interactive Assembly Planning*. IEEE Transactions on Robotics and Automation, 11:308–312, 1995.

- [WK64] WESTER, L. und M. KILBRIDGE: *The assembly line model-mix sequencing problem*. Proceedings of the third international conference on Operations Research, Seiten 247–260, 1964.
- [Wol89] WOLTER, J.D.: *On the Automatic Generation of Assembly Plans*. Proc. IEEE Intl. Conf. on Robotics & Automation, Scottsdale, AZ, 14 – 19 May 1989, 1989.
- [XO94] XIAOBO, Z. und K. OHNO: *A sequencing problem for a mixed-model assembly line in an JIT production system*. Computers & Industrial Engineering, 27:71–74, 1994.
- [XO97] XIAOBO, Z. und K. OHNO: *Algorithms for sequencing mixed models on an assembly line in a JIT production system*. Computers & Industrial Engineering, 32:47–56, 1997.
- [XO00] XIAOBO, Z. und K. OHNO: *Properties of a sequencing problem for a mixed model assembly line with conveyor stoppages*. European Journal of Operational Research, 124:560–570, 2000.
- [YB89] YANO, C.A. und A. BOLAT: *Survey, development, and application of algorithms for sequencing paced assembly lines*. Journal of Manufacturing and Operations Management, 2:172–198, 1989.
- [YR91] YANO, C.A. und R. RACHAMADUGU: *Sequencing to minimize work overload in assembly lines with product options*. Management Science, 37:572–586, 1991.
- [YR92] YOKOTA, Y. und D.R. ROUGH: *Assembly/disassembly sequence planning*. Assembly Automation, 12:31–38, 1992.
- [YSH05] YOO, J.K., Y. SHIMIZU und R. HINO: *A sequencing problem for mixed-model assembly line with the aid of relief-man*. JSME International Journal, 48:15–20, 2005.

A Anhang

A.1 Ergebnisse Rekonfiguration

Reihenfolge	u^E		u^{MAX}	
	vorher	nachher	vorher	nachher
0	98	63	6	4
1	125	92	4	4
2	116	88	4	4
3	122	85	4	3
4	119	76	5	4
5	125	83	4	3
6	121	81	5	5
7	119	79	5	4
8	129	89	6	4
9	123	87	4	4
10	115	88	4	4
11	121	85	4	3
12	124	88	5	5
13	122	81	5	4
14	124	87	5	5
15	124	84	5	5
16	119	87	5	5
17	119	87	5	4
18	127	91	5	5
19	118	83	4	3
20	114	75	5	5
21	117	87	4	4
22	120	85	5	5
23	123	87	5	5
24	119	81	3	3
25	124	88	6	4
26	117	80	3	3
27	110	77	5	4
28	114	79	5	5
29	122	89	5	5
30	115	78	3	3
31	122	88	4	4
32	131	92	5	4
33	120	81	4	3
34	121	83	5	5

Reihenfolge	u^E		u^{MAX}	
	vorher	nachher	vorher	nachher
35	127	81	4	3
36	119	82	5	4
37	124	85	5	5
38	120	81	4	3
39	123	80	4	4
40	116	78	4	4
41	124	86	5	4
42	115	78	5	4
43	133	88	5	5
44	120	79	4	3
45	122	85	4	3
46	126	86	4	3
47	122	87	6	4
48	123	84	4	3
49	123	83	5	4
50	121	85	5	4
51	127	90	5	4
52	122	91	6	4
53	130	89	5	4
54	116	81	5	4
55	116	76	5	4
56	124	82	5	4
57	123	88	5	4
58	131	90	3	3
59	134	87	5	4
60	131	88	5	4
61	123	84	6	4
62	116	76	4	4
63	119	83	4	4
64	121	90	3	3
65	125	80	5	4
66	128	84	5	4
67	119	84	5	4
68	129	93	4	3
69	112	79	4	4
70	131	96	4	4
71	122	84	5	5
72	117	86	4	4
73	126	89	5	4
74	117	82	5	4
75	115	85	5	4
76	127	81	5	5
77	109	78	4	4
78	127	94	5	6
79	121	87	6	4
80	123	87	6	4

Reihenfolge	u^E		u^{MAX}	
	vorher	nachher	vorher	nachher
81	117	82	4	4
82	126	85	6	5
83	121	82	4	3
84	122	88	5	4
85	127	81	4	3
86	113	77	5	3
87	118	81	5	4
88	120	81	4	4
89	123	84	4	4
90	110	80	4	3
91	131	83	4	4
92	136	90	5	4
93	128	84	4	3
94	120	86	5	5
95	116	78	4	4
96	133	93	5	4
97	115	77	4	5
98	118	81	5	4
99	120	90	4	3

Tabelle A.1: Unterstützereinsätze vor und nach der Rekonfiguration für 100 zufällige Reihenfolgen

A.2 Ergebnisse Vorranggraphenbildung

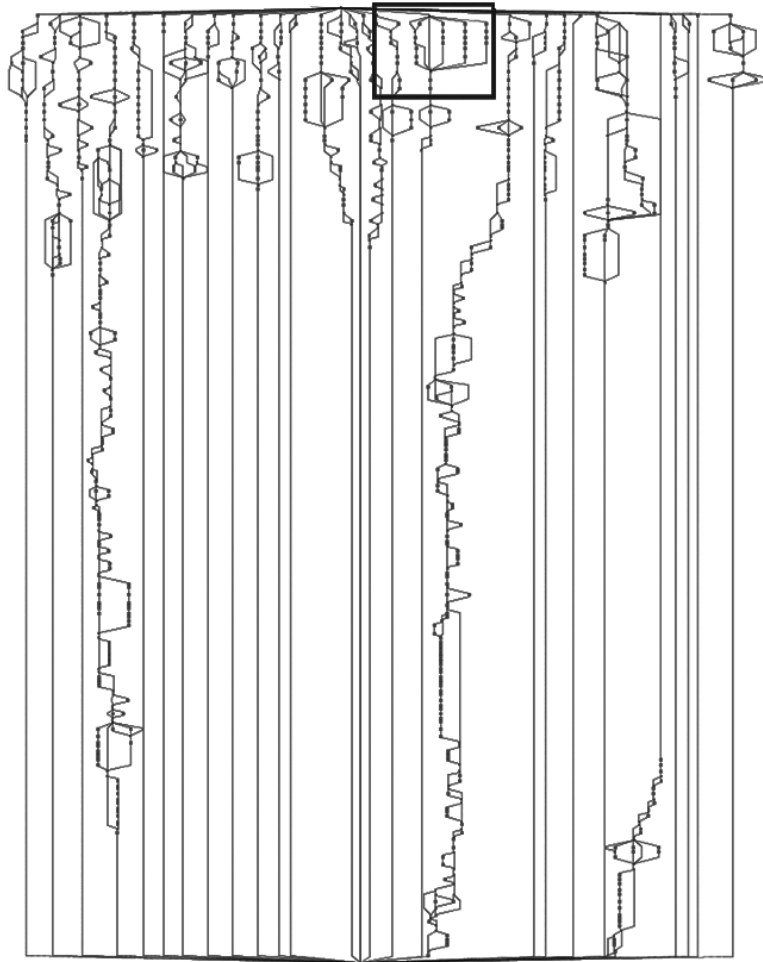


Abbildung A.1: Beispiel für einen vollständigen Vorranggraphen mit 472 Arbeitsvorgängen

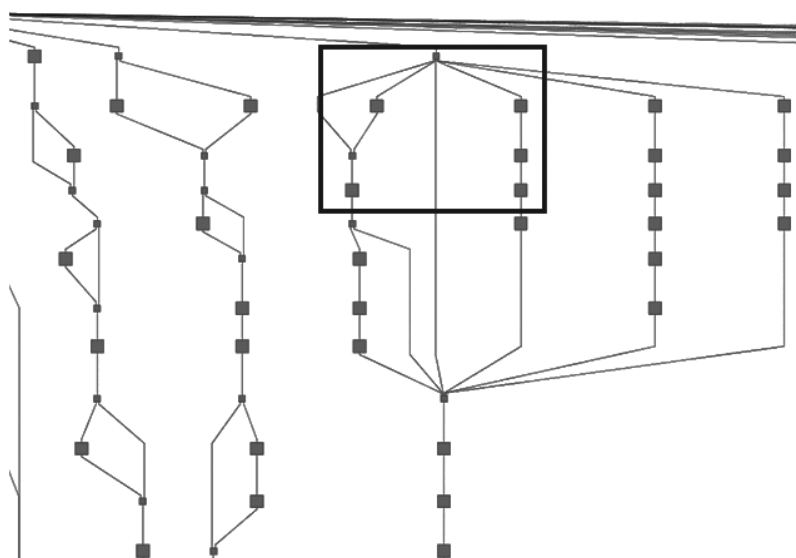


Abbildung A.2: Vergrößerter Ausschnitt aus dem Beispiel-Vorranggraphen

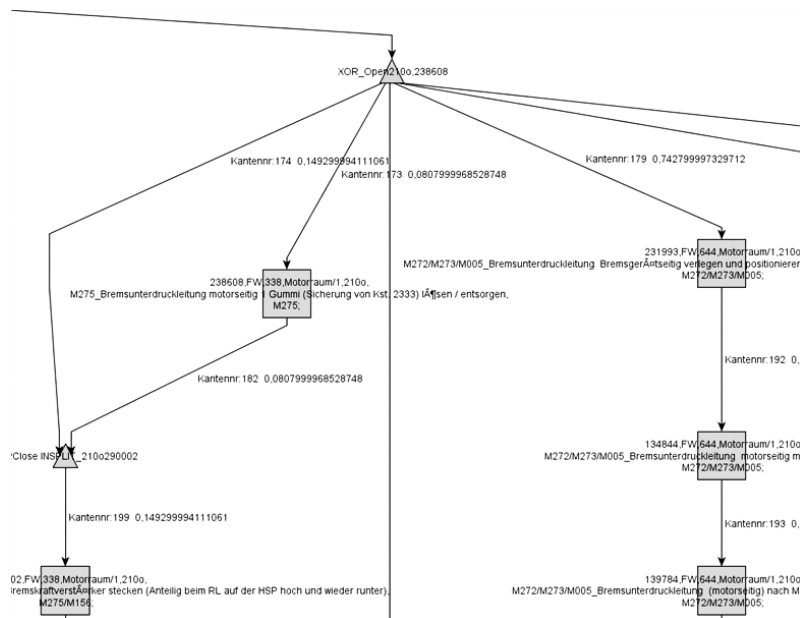


Abbildung A.3: Vergrößerter Ausschnitt aus dem Beispiel-Vorranggraphen

A.3 Ergebnisse Reihenfolgeoptimierung

Szenario	Reihenfolge der Zuordnung	Startzentroidenauswahl		
		Random	K-Means ++	Sort&Group
1	Random	108.51	108.80	111.05
	Sort	109.50	110.80	110.68
2	Random	109.42	109.32	105.48
	Sort	113.37	114.43	113.82
3	Random	103.51	106.38	103.21
	Sort	109.48	108.39	110.36
4	Random	111.56	112.56	104.94
	Sort	114.24	113.41	108.79
5	Random	103.06	102.42	102.69
	Sort	105.87	107.41	105.12
6	Random	105.66	106.55	106.07
	Sort	107.48	107.60	109.49
7	Random	94.52	94.89	94.70
	Sort	99.27	98.53	96.72

Tabelle A.2: Durchschnittliche Homogenität der Cluster je Auftrag in Abhängigkeit von der Startzentroidenauswahl und der Reihenfolge der Zuordnung