

Drexl, Andreas; Jordan, Carsten

**Working Paper — Digitized Version**

## Materialflußorientierte Produktionssteuerung bei Variantenfließfertigung

Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel, No. 362

**Provided in Cooperation with:**

Christian-Albrechts-University of Kiel, Institute of Business Administration

*Suggested Citation:* Drexl, Andreas; Jordan, Carsten (1994) : Materialflußorientierte Produktionssteuerung bei Variantenfließfertigung, Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel, No. 362, Universität Kiel, Institut für Betriebswirtschaftslehre, Kiel

This Version is available at:

<https://hdl.handle.net/10419/155430>

**Standard-Nutzungsbedingungen:**

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

**Terms of use:**

*Documents in EconStor may be saved and copied for your personal and scholarly purposes.*

*You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.*

*If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.*

Nr. 362

**Materialflußorientierte Produktionssteuerung  
bei Variantenfließfertigung**

Andreas Drexl, Carsten Jordan

Januar 1994

Professor. Dr. *Andreas Drexl*, Dipl.-Wirtschafts-Ingenieur *Carsten Jordan*, Institut für Betriebswirtschaftslehre, Christian-Albrechts-Universität zu Kiel, Olshausenstr. 40, 24118 Kiel.

## 1 Problemstellung und Abgrenzung

Als grundlegende Konzepte der Produktionssteuerung werden in der Literatur neben klassischen Vorgehensweisen der Materialbestandsoptimierung (Produktion auf Lager) in jüngster Zeit zunehmend Ansätze der Materialflußoptimierung (Produktion auf Abruf) diskutiert.<sup>1</sup>

Im Rahmen der Materialflußoptimierung sind kurze Durchlaufzeiten, schnelle Reaktionsmöglichkeiten auf Kundenwünsche, hohe Liefertermintreue und niedrige Bestände die wichtigsten Zielsetzungen.<sup>2</sup> Zu den materialflußorientierten<sup>3</sup> Konzepten der Produktionssteuerung zählen neben der produktionssynchronen Beschaffung vor allem Kanban-Systeme<sup>4</sup> sowie die Just-in-time-Produktion<sup>5</sup>.

Bei der Verfolgung der im Rahmen des Paradigmas "Materialflußoptimierung" relevanten Ziele kommt dem Konzept der Variantenfließfertigung eine besondere Rolle zu. Die Variantenfließfertigung läßt sich als kundenauftragsorientierte Fertigung ("Losgröße 1") mit serieller, taktgebundener Montage kennzeichnen.<sup>6</sup> Hierbei ist die Zusammenfassung gleichartiger Varianten zu Losen nicht vorteilhaft. Wichtige Beispiele sind die Motoren-, Traktoren- und PKW-Fertigung.

Insbesondere in der Kraftfahrzeugwirtschaft ist die Variantenfließfertigung von herausragender Bedeutung. Hier verlassen teilweise täglich mehr als 1000 Fahrzeuge das Band, wobei kaum zwei Wagen völlig identisch sind.<sup>7</sup> Da die unterschiedlichen Varianten i.d.R. in Konkurrenz um die verfügbare Kapazität der Fertigungsanlagen stehen, ist ein erhöhter Koordinationsaufwand in der Ablaufplanung und Produktionssteuerung zur Vermeidung von Über-/Unterauslastung der Fertigung und zur Verstetigung des Materialflusses erforderlich.

In dieser Situation werden in der Literatur v.a. Modelle und Verfahren zum "Level Scheduling" für "Mixed-Model Assembly Lines" vorgeschlagen. Diesen Modellen ist gemeinsam, daß sie als *Zielsetzung* die Minimierung der quadrierten oder absoluten Abweichung der tatsächlichen von der angestrebten Produktionsrate<sup>8</sup> anstreben. Bei dieser Betrachtung gelingt es jedoch nicht, den durch das Level Scheduling induzierten

---

<sup>1</sup> Knolmayer (1987) spricht in diesem Zusammenhang von einem Paradigma-Wechsel.

<sup>2</sup> Vgl. hierzu insbesondere Zäpfel (1991).

<sup>3</sup> Zu den systemtechnischen Grundlagen von Materialflußsystemen siehe Jünemann (1989).

<sup>4</sup> Siehe hierzu z.B. Krajewski et al. (1987) sowie die umfassende Übersicht in Berkley (1992).

<sup>5</sup> Zur Funktionsweise und zu den Einsatzvoraussetzungen vgl. z.B. Fandel / Francois (1989); zur Abgrenzung von anderen Produktionssegmenten (wie z.B. Werkstattfertigung) vgl. vor allem Drexel et al. (1994).

<sup>6</sup> Vgl. hierzu ausführlicher Decker (1993); siehe auch Ziegler (1990).

<sup>7</sup> Vgl. Decker (1993).

<sup>8</sup> Zur Definition der Produktionsrate (bzw. Nachfragerate) vgl. Abschnitt 2.

Materialfluß in der gewünschten Weise zu verstetigen sowie die von der Reihenfolge der Varianten abhängige Belastung der Arbeitsstationen zu berücksichtigen.<sup>9</sup> Als Alternative bietet sich das sogenannte "Car Sequencing" an, bei dem die Anforderungen in Form von *Nebenbedingungen* formuliert werden.

Der vorliegende Beitrag ist wie folgt aufgebaut: In Abschnitt 2 formulieren und analysieren wir verschiedenen Varianten von Level Scheduling-Modellen. Abschnitt 3 ist dem Car Sequencing-Problem gewidmet: Es wird zunächst als gemischt-ganzzahliges Programmierungsmodell und dann als konzeptionelles Modell formuliert. Während ersteres unmittelbar von Softwarepaketen zur gemischt-ganzzahligen Programmierung gelöst werden kann, ist das zweite direkt in deklarativen Sprachen codierbar und vom Inferenzmechanismus auswertbar. In Abschnitt 4 berichten wir über Rechenerfahrungen bei der Lösung des "Car Sequencing"-Problems mit beiden Modellierungs- und Lösungskonzepten. Ferner vergleichen wir die beiden sehr unterschiedlichen Vorgehensweisen. Ein Ausblick auf weiterführende Forschungsarbeiten in Abschnitt 5 beschließt den Beitrag.

## 2 Level Scheduling

Zur Variantenfließfertigung sind sogenannte "Mixed-Model Assembly Lines", d.h. Fließlinien für die kundenauftragsorientierte Herstellung zahlreicher Varianten eines Produktes geeignet.<sup>10</sup> Voraussetzung für den effizienten Einsatz derartiger Fließlinien ist die Lösung folgender Probleme:<sup>11</sup>

- (P1) Ermittlung der Zykluszeit bzw. Bandgeschwindigkeit,
- (P2) Vorgabe von Anzahl und Reihenfolge der Bandstationen,
- (P3) Durchführung des Bandabgleichs sowie
- (P4) Bestimmung der Reihenfolge, in der die Varianten "auf's Band gelegt" werden.

Das Level Scheduling behandelt die Problemstellung (P4), eine Fragestellung der (operativen) Produktionsdurchführungsplanung, das immer wieder (d.h. in der Regel täglich) unter Zeitdruck zu lösen ist und für das damit eine leistungsfähige EDV-Unterstützung besonders wichtig ist.

Als Zielsetzungen kommen beim Level Scheduling prinzipiell folgende in Frage:

- (Z1) Nivellierung der Arbeitslast (Summe der Bearbeitungszeiten) für jede Bandstation bzw.

---

<sup>9</sup> Detaillierte Ausführungen hierzu finden sich in Abschnitt 3.

<sup>10</sup> Vgl. z.B. Miltenburg (1989).

<sup>11</sup> Vgl. zu diesen Problembereichen insbesondere Domschke et al. (1993).

(Z2) Aufrechterhaltung einer im Zeitablauf konstanten Bedarfsrate (d.h. eines konstanten Teilebedarfs).

Im folgenden wird die Problemstellung (P4) unter der Zielsetzung (Z2) betrachtet. Gegenstand der Planung ist also die Bestimmung der Reihenfolge, in der die Varianten "auf's Band gelegt" werden, sodaß die Aufrechterhaltung einer im Zeitablauf konstanten Bedarfsrate gesichert ist.

Zur Formalisierung gehen wir von folgenden Annahmen aus:<sup>12</sup> Gegeben sei mit

$V$  : die Anzahl an Varianten eines Produktes sowie mit

$d_v$  : die Nachfrage nach Variante  $v$  ( $v = 1, \dots, V$ ) im Planungszeitraum.

Damit ergibt sich die Produktionsmenge

$$T := \sum_{v=1}^V d_v$$

und die Produktions- bzw. Nachfragerate  $r_v$  für Variante  $v$

$$r_v := d_v / T.$$

Die Linie soll nun so belegt werden, daß der von Variante  $v$  gefertigte Anteil möglichst der Nachfragerate  $r_v$  entspricht.

Zur Formulierung des Entscheidungsproblems verwenden wir Variablen:<sup>13</sup>

$x_{vt}$  : kumulierte Produktionsmenge von Variante  $v$  bis "Takt"  $t$

Für die Entscheidungsvariablen  $x_{vt}$  gilt, daß sie größer gleich Null und ganzzahlig sein müssen, d.h.  $x_{vt} \in \mathbb{N}_0$ .

Da in jedem Takt nur eine Variante aufgelegt wird, muß  $\sum_{v=1}^V x_{vt} = t$  ( $t = 1, \dots, T$ ) gelten. Wir erhalten damit folgendes *ganzzahlige, quadratische Optimierungsmodell*:<sup>14</sup>

$$\min \sum_{t=1}^T \sum_{v=1}^V (x_{vt} - t \cdot r_v)^2 \quad (1)$$

unter Beachtung der Nebenbedingungen

<sup>12</sup> Vgl. vor allem Miltenburg (1989), Miltenburg/Sinnamon (1989), (1992) sowie Miltenburg et al. (1990); siehe auch Thomopoulos (1967) sowie Yano/Rachamadugu (1991).

<sup>13</sup> Die Anführungszeichen bei "Takt" weisen darauf hin, daß die Fließlinie nicht im eigentlichen Sinne getaktet ist; der "Takt" ergibt sich vielmehr aus der Produktionsmenge. Aus Vereinfachungsgründen wird im folgenden "Takt" ohne Anführungszeichen verwendet.

<sup>14</sup> Eine Reformulierung dieses Modells, die sich sehr effizient als lineares Zuordnungsproblem lösen läßt, wird in Kubiak/Sethi (1991) vorgestellt.

$$\sum_{v=1}^V x_{vt} = t \quad (t = 1, \dots, T) \quad (2)$$

$$0 \leq x_{vt} - x_{v,t-1} \leq 1 \quad (v = 1, \dots, V; t = 2, \dots, T) \quad (3)$$

$$x_{vt} \in \mathbb{N}_0 \quad (v = 1, \dots, V; t = 1, \dots, T) \quad (4)$$

Die Zielfunktion (1) minimiert die Summe der Abweichungen der aktuell produzierten Anzahl an Einheiten ( $x_{vt}$ ) von der angestrebten ( $t \cdot r_v$ ). Die Nebenbedingungen (2) und (3) stellen sicher, daß jeweils die erforderliche Anzahl an Varianten hergestellt wird. Man beachte, daß Nebenbedingungen  $x_{vT} \leq d_v$  ( $v = 1, \dots, V$ ) nicht erforderlich sind, weil sie im Optimum ohnehin eingehalten werden.

Als *alternative Zielsetzungen* werden in der Literatur vorgeschlagen:<sup>15</sup>

$$\min \sum_{t=1}^T \sum_{v=1}^V |x_{vt} - t \cdot r_v| \quad (1')$$

$$\min \sum_{t=1}^T \sum_{v=1}^V (x_{vt}/t - r_v)^2 \quad (5)$$

$$\min \sum_{t=1}^T \sum_{v=1}^V |x_{vt}/t - r_v| \quad (5')$$

(1) bzw. (1') gleichen die aktuell produzierte Anzahl an Einheiten ( $x_{vt}$ ) an die angestrebte ( $t \cdot r_v$ ) an. (5) bzw. (5') gleichen den aktuellen Anteil des Varianten-Mix ( $x_{vt}/t$ ) an den angestrebten ( $r_v$ ) an. Während (1) und (5) "Ausreißer" eliminieren, minimieren (1') und (5') die Summe der Abweichungen.

Eine alternative Formulierung des Level Scheduling-Problems, die sich sehr einfach lösen läßt, kann folgendermaßen hergeleitet werden:<sup>16</sup> Bezeichne  $z_{vi}$  mit

$$z_{vi} := [(i - 1/2) T] / d_v \quad (v = 1, \dots, V; i = 1, \dots, d_v)$$

den Takt (= Fälligkeitstermin), in dem die  $i$ -te Einheit von Variante  $v$  benötigt wird. Sei ferner als Entscheidungsvariable

$$y_{vi} : \text{ der Takt, in dem die } i\text{-te Einheit von Variante } v \text{ tatsächlich produziert wird} \\ (y_{vi} \in \mathbb{N} \text{ bzw. } y_{vi} \in [1, \dots, T]),$$

<sup>15</sup> Steiner/Yeomans (1993) betrachten diese Problemstellung darüber hinaus unter der Annahme, daß die maximale Abweichung minimiert werden soll (Minimax-Zielsetzung).

<sup>16</sup> Vgl. hierzu Inman/Bulfin (1991).

definiert, so erhalten wir das folgende Modell:

$$\min \sum_{v=1}^V \sum_{i=1}^{d_v} (y_{vi} - z_{vi})^2 \quad (6)$$

unter Beachtung der Nebenbedingungen

$$y_{vi} \neq y_{sj} \quad (1 \leq v \leq s \leq V; i = 1, \dots, d_v; j = 1, \dots, d_s; i \neq j) \quad (7)$$

$$y_{vi} \in \mathbb{N} \quad (v = 1, \dots, V; i = 1, \dots, d_v) \quad (8)$$

Die Zielfunktion (6) minimiert die Summe der Terminabweichungen. Die Nebenbedingungen (7) stellen sicher, daß alle Varianten in verschiedenen Takten eingeplant werden. Zusammen mit der Definition der Entscheidungsvariablen  $y_{vi} \in \mathbb{N}$  in (8) ist sichergestellt, daß jede Variante in genau einem Takt vorgesehen wird.

Als Alternative zu (6) kann auch die Zielsetzung (6') verwendet werden:

$$\min \sum_{v=1}^V \sum_{i=1}^{d_v} |y_{vi} - z_{vi}| \quad (6')$$

Die Level Scheduling-Modelle (6)-(8) bzw. (6')-(8) haben den Vorteil, daß sie sehr einfach lösbar sind. Es genügt, die  $z_{vi}$  ( $v = 1, \dots, V; i = 1, \dots, d_v$ ) gemäß obiger Formel zu berechnen, nach monoton steigenden Werten zu sortieren und anschließend die Varianten den Takten  $t = 1, \dots, T$  in der Reihenfolge steigender Werte zuzuordnen. Diese Vorgehensweise liefert im übrigen auch für die Modelle (1)-(4), (1')-(4), (2)-(5) und (2)-(5') sehr gute Näherungslösungen.<sup>17</sup>

In Abschnitt 3 werden wir mit Hilfe eines Beispiels zeigen, daß es mit diesen Modellen nicht gelingt, eine obere Schranke für die Arbeitsbelastung einzelner Stationen einzuhalten sowie eine "ausreichende" Materialversorgung zu gewährleisten, wenn Anzahl und Mix des Materialbedarfs der Varianten (die beim Level Scheduling gar nicht explizit betrachtet werden) voneinander abweichen.

Zur Vermeidung der in diesem Abschnitt zunächst nur angedeuteten Probleme kommt als Alternative das in Abschnitt 3 behandelte "Car Sequencing-Modell" in Frage. Dabei werden eine obere Schranke für die Arbeitsbelastung einzelner Stationen sowie Höhe und Mix des Materialbedarfs der Varianten explizit betrachtet.

<sup>17</sup> Vgl. hierzu erneut Inman/Bulfin (1991).

### 3 Car Sequencing

Zur Verdeutlichung betrachten wir folgende Ausgangssituation eines Automobilherstellers:<sup>18</sup> Die an einem Tag auf einer Linie herzustellenden Fahrzeuge sind inhomogen (d.h. sie erfordern unterschiedliche Optionen). Stellt man nun die Bandgeschwindigkeit so niedrig ein, daß in jedem Takt jede Option in jedes Fahrzeug eingebaut werden kann, so arbeitet die Montage unwirtschaftlich. Die Forderung lautet vielmehr, daß die Linie bei "wirtschaftlicher" Bandgeschwindigkeit eine vorgegebene Anzahl an Fahrzeugen für jede Option herstellen können muß.

Wir betrachten folgendes Beispiel: Nicht mehr als 60% der Fahrzeuge erhalten die Option "Schiebedach". Fünf Fahrzeuge passieren in der Zeit für die Installation genau eines Schiebedaches den Abschnitt der Linie, in dem Schiebedächer installiert werden. Damit sind drei Teams zur Installation von Schiebedächern im entsprechenden Abschnitt der Linie erforderlich, und die Linie hat die Kapazitätsbeschränkung '3:5' für die Option Schiebedach. Ganz analog nehmen wir an, daß für die Option "Klimaanlage" die Kapazitätsbeschränkung '2:6' einzuhalten ist. Ferner sei für die Option "Heckscheibenwischer" die Kapazitätsbeschränkung '2:4' gegeben. Für die Option "Alu-Felgen" gelte schließlich die Kapazitätsbeschränkung '2:3'.

Varianten entstehen nun dadurch, daß in jedes Fahrzeug eine oder mehrere der verfügbaren Optionen einzubauen sind. Als Konsequenz ergibt sich, daß die Kapazitätsbelastung an den einzelnen Arbeitsstationen eine Funktion der Reihenfolge ist, in der die Varianten auf das Band gelegt werden.<sup>19</sup> Selbstverständlich ist damit auch der Materialbedarf unmittelbar von der Reihenfolge abhängig. Damit entsteht ein kombinatorisches Optimierungsproblem, dessen Ziel die Vermeidung von Kapazitätsüberlastungen an den einzelnen Arbeitsstationen und damit unmittelbar einhergehend die Steuerung des Materialflusses ist. Der formalen Behandlung dieses kombinatorischen Optimierungsproblems wenden wir uns im folgenden zu.

Zur Formalisierung des Problems gehen wir von folgenden Annahmen aus: Gegeben sei

$O$  : die Anzahl verschiedener Optionen  $o$  ( $o = 1, \dots, O$ ),

<sup>18</sup> Vgl. hierzu Dincbas et al. (1988), Parello (1988) sowie Parello et al. (1986).

<sup>19</sup> Man kann sich leicht folgendes überlegen: Bezieht man den Materialbedarf und die Arbeitsbelastungen, die sich aus Reihenfolgeentscheidungen bzgl. der Optionen für die einzelnen Stationen ergeben, nicht ein, so müßte jeder Abschnitt der Linie so ausgelegt werden, daß die Maximalbelastung zu bewältigen ist. Im Beispiel würde das die (unwirtschaftliche) Auslegung '5:5', '6:6', '4:4' sowie '3:3' implizieren.

- $H_o : N_o$  : von  $N_o$  aufeinanderfolgenden Fahrzeugen dürfen höchstens  $H_o$  die Option  $o$  besitzen, sowie
- $V$  : die Anzahl an Varianten  $v$  ( $v = 1, \dots, V$ ).

Sei ferner

- $a_{vo}$  := 1, falls Variante  $v$  Option  $o$  erfordert  
:= 0, sonst

und

- $d_v$  die herzustellende Anzahl an Fahrzeugen von Variante  $v$ .

Wir erhalten damit (wie in Abschnitt 2) aus  $T = \sum_{v=1}^V d_v$  die Gesamtzahl zu produzierender Fahrzeuge. Ferner bezeichne

- $M_o$  : die Menge der Varianten  $v$ , die Option  $o$  erfordern.

Offensichtlich gilt  $M_o := \{o \mid a_{vo} = 1, v = 1, \dots, V\}$  für den Zusammenhang zwischen  $a_{vo}$  und  $M_o$ .

Unter Verwendung der Entscheidungsvariablen

- $w_{vt}$  := 1, falls Variante  $v$  im Takt  $t$  eingeplant wird  
:= 0, sonst

erhalten wir folgendes *Entscheidungsmodell* der gemischt-ganzzahligen (binären) Optimierung:

$$\sum_{v=1}^V w_{vt} = 1 \quad (t = 1, \dots, T) \quad (9)$$

$$\sum_{t=1}^T w_{vt} = d_v \quad (v = 1, \dots, V) \quad (10)$$

$$\sum_{\tau=t}^{t+N_o} \sum_{v=1}^V w_{v\tau} a_{vo} \leq H_o \quad (o = 1, \dots, O; t = 1, \dots, T-N_o) \quad (11)$$

$$w_{vt} \in \{0, 1\} \quad (v = 1, \dots, V; t = 1, \dots, T) \quad (12)$$

Gesucht ist also eine Belegung der Entscheidungsvariablen  $w_{vt}$  so, daß alle Restriktionen eingehalten werden. (9) bzw. (10) entsprechen den Zuordnungsrestriktionen, (11) repräsentiert die Kapazitätsrestriktionen der Fließlinie.

Zur Veranschaulichung betrachten wir das Car Sequencing-Problem in *Abbildung 1* mit den vier oben eingeführten Optionen, die durch die Symbole

"■" (Option "Alu-Felgen")

"□□" (Option "Heckscheibenwischer")

"////" (Option "Schiebedach")

"■" (Option "Klimaanlage")

dargestellt werden (je "Zeile" eine Option). In *Abbildung 1* ist ebenfalls jede der sechs zu fertigenden Varianten  $v = 1, \dots, V$  dargestellt (je "Spalte" eine Variante). Ferner ist in der "Kopfzeile" die Stückzahl  $d_v$  abzulesen. Insgesamt sind  $T = 14$  Fahrzeuge zu fertigen.

*Abbildung 1: Beispiel*

$o$	$H_o : N_o$	4	1	2	2	2	3	$d_v$
1	2 : 3		■	■	■		■	
2	2 : 4			□□	□□			
3	3 : 5		////			////		
4	2 : 6	■			■			
		1	2	3	4	5	6	$v$

Eine zulässige Lösung dieses Problems ist in *Abbildung 2* veranschaulicht. Wie man leicht nachvollziehen kann, ist von jeder Variante die herzustellende Anzahl an Fahrzeugen vorgesehen, jede Variante ist mit den erforderlichen Optionen ausgestattet, und die Kapazitätsbeschränkungen ' $H_o : N_o$ ' sind erfüllt.

*Abbildung 2: Car Sequencing-Lösung für das Beispiel*

		■	■		■		■	■		■	■		■
			□□		□□		□□						□□
		////		////					////				
■	■					■	■					■	■
1	1	2	3	5	3	1	4	6	5	6	6	1	4

Betrachten wir dieses Beispiel nun (unter Vernachlässigung der Kapazitätsrestriktionen  $H_o : N_o$ ) als Level Scheduling Problem, so liefert das in Abschnitt 2 skizzierte Verfahren die Fälligkeitstermine  $z_{vi}$  in *Tabelle 1* und schließlich die Optimallösung von *Tabelle 2* (eingetragen in den ersten beiden Zeilen).

*Tabelle 1: Fälligkeitstermine  $z_{vi}$  für das Beispiel*

$v$	$d_v$	$i$			
1	4	1.75	5.25	8.75	12.25
2	1	7			
3	2	3.5	10.5		
4	2	3.5	10.5		
5	2	3.5	10.5		
6	3	2.3	7	11.6	

*Tabelle 2: Level Scheduling-Lösung für das Beispiel*

$t$	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$v$	1	6	3	4	5	1	2	6	1	3	4	5	6	1
$o=1$		■	■	■			■	■		■	■		■	
$o=2$			□□	□□						□□	□□			
$o=3$					▨		▨					▨		
$o=4$	■			■		■			■		■			■

Aus *Tabelle 2* ist unmittelbar folgendes ersichtlich: Die Minimierung der Abweichung von Nachfragerate und Produktionsrate wurde erreicht. Allerdings hat diese optimale Annäherung zweier "Raten" den fundamentalen Nachteil, daß sie von der Materialverfügbarkeit und der Arbeitsbelastung einzelner Stationen vollständig abstrahiert. Das bedeutet: Bei Option  $o=1$  kommt es im Bereich  $t=2, \dots, 4$  zu einer Überschreitung der Kapazitätsrestriktion  $H_1 : N_1 = 2:3$ . Auch bei Option  $o=4$  wird die Kapazitätsrestriktion  $H_4 : N_4 = 2:6$  überschritten, und zwar im Bereich  $t=1, \dots, 6$ , im Bereich  $t=4, \dots, 10$  und im Bereich  $t=9, \dots, 14$ .

Zusammenfassend zeigt der exemplarische Vergleich der Vorgehensweisen, daß der Car Sequencing-Ansatz eine wesentlich genauere Abbildung der Gegebenheiten in der Produktion erlaubt und damit a priori realistischere und damit leichter umsetzbare Pläne zu generieren gestattet als der Level Scheduling-Ansatz.

Auf der Basis des Modells (9)-(12) sind wir damit also prinzipiell in der Lage, eine geeignete Reihenfolge der Varianten zu bestimmen. Wie wir jedoch weiter unten zeigen werden, hat der "traditionelle" Modellierungsweg der gemischt-ganzzahligen Programmierung gravierende Nachteile. Diese lassen sich durch Reformulierung des Problems als sogenanntes "konzeptionelles Modell", Codierung in deklarativen Sprachen und Lösung mittels Inferenz vermeiden.

Die Entscheidungsvariablen des konzeptionellen Modells sehen wie folgt aus: Bezeichne

$\mathcal{V}_t$  : die Variante  $v$ , die im Takt  $t$  eingeplant ist,

so erhalten wir  $\mathcal{V}^v$  als die Menge der Takte  $t$ , in denen Variante  $v$  vorgesehen ist:

$$\mathcal{V}^v = \{ t \mid \mathcal{V}_t = v \}$$

Unter Verwendung dieser Hilfsvariablen  $\mathcal{V}^v$  läßt sich das *konzeptionelle Modell* wie folgt sehr kompakt formulieren.

$$|\mathcal{V}^v| = d_v \quad (v = 1, \dots, V) \quad (13)$$

$$|\{ \tau \mid \mathcal{V}_\tau \in M_o, \tau = t, \dots, t+N_o \}| \leq H_o \quad (o = 1, \dots, O; t = 1, \dots, T-N_o) \quad (14)$$

Zur Erläuterung: (13) sorgt dafür, daß die erforderliche Anzahl an Varianten gefertigt wird. (14) sichert, daß die Kapazitätsrestriktionen der Fließlinie beachtet werden; dabei wird für jeweils  $N_o$  aufeinanderfolgende Takte sichergestellt, daß höchstens  $H_o$  aufeinanderfolgende Varianten die Option  $o$  erfordern.

Das Modell (13)-(14) kann unmittelbar in deklarativen Sprachen wie CHARME<sup>20</sup> oder CHIP<sup>21</sup> codiert und anschließend vom Inferenzmechanismus ausgewertet werden.

Bevor wir in Abschnitt 4 über exemplarische Rechenergebnisse berichten, die bei der Lösung des Car Sequencing-Problems auf der Basis der Formulierungen (9)-(12) bzw. (13)-(14) erzielt wurden, soll zunächst eine weitere wichtige Verallgemeinerung dargestellt werden, und zwar anhand eines Beispiels. Betrachten wir den Fall, daß sich die Belastung einer Arbeitsstation nicht nur aus einer spezifischen Option aufeinanderkommender Fahrzeuge ergibt. Vielmehr seien zwei Optionen gemeinsam zu beachten.

Zur Verdeutlichung betrachten wir das Car Sequencing-Problem von *Abbildung 3* mit  $O = 5$  Optionen und  $V = 5$  Varianten.

<sup>20</sup> Zu den Grundlagen von CHARME vgl. BULL (1990), Drexl et al. (1995) sowie Pesch et al. (1994); der CHARME-Code für das Modell (13)-(14) ist in Drexl/Jordan (1994) zu finden.

<sup>21</sup> Vgl. Dinçbas et al. (1988).

Abbildung 3: Beispiel

$o$	$H_o : N_o$	1	2	3	4	5	$v$
1	2 : 3	■	■			■	
2	3 : 4		◇◇	◇◇	◇◇	◇◇	
3	3 : 7	□□		□□		□□	
4	2 : 6		▷◁		▷◁		
5	4 : 5				■	■	
6	2 : 8			◇□		◇□	
7	2 : 9		◇◁		◇◁		

Soll nun die sich aus den Optionen  $o = 2$  und  $o = 3$  gemeinsam ergebende Belastung betrachtet werden, so können wir das dadurch bewerkstelligen, daß wir eine weitere Option  $o = 6$

"◇□" (Option "2 & 3")

mit der Kapazitätsbeschränkung  $H_6 : N_6 = 2 : 8$  hinzufügen<sup>22</sup> (ebenfalls in Abbildung 3 eingetragen). Analog erhalten wir bei gemeinsamer Betrachtung der Optionen  $o = 2$  und  $o = 4$  die Zusatzoption  $o = 7$

"◇◁" (Option "2 & 4")

mit der Kapazitätsbeschränkung  $H_7 : N_7 = 2 : 9$ , die ebenfalls in Abbildung 3 eingetragen ist. Ein Eintrag erfolgt dabei jeweils genau dann, wenn eine Variante beide Optionen gleichzeitig erfordert.

Zusammenfassend bedeutet das, daß sich die Berücksichtigung von Arbeitsbelastungen durch mehrere Optionen auf den bereits behandelten Fall einer Option zurückführen läßt. Mithin sind dann lediglich Änderungen der Eingabedaten erforderlich, um auch diesen Fall abzudecken.

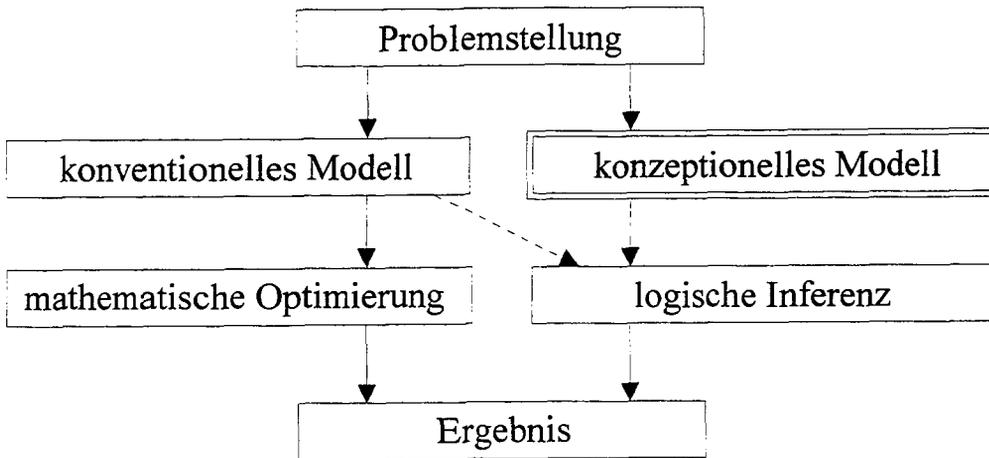
#### 4 Numerische Ergebnisse

Das Modell (9)-(12) wurde "konventionell" (d.h. prozedural) mit Software zur gemischt-ganzzahligen Programmierung (LINGO als Modellierungssprache und LINDO als general purpose optimizer) gelöst. Ferner wurde das konzeptionelle Modell (13)-(14) in

<sup>22</sup> Die Parameter  $H : N$  bei Doppeloptionen können beispielsweise die Verfügbarkeit von Springern bzw. Werkern, die an mehreren Stationen eingesetzt werden, widerspiegeln.

CHARME implementiert. *Abbildung 4* zeigt im Überblick, wie man auf unterschiedlichen Wegen von der Problemstellung zum Ergebnis gelangt, d.h. von demselben Ausgangspunkt (Problemstellung) ausgehend an dasselbe Ziel (Ergebnis).

*Abbildung 4: Von der Problemstellung zum Ergebnis*<sup>23</sup>



Von besonderem Interesse ist in *Abbildung 4* das konzeptionelle Modell (13)-(14) (doppelt umrandet). Die gestrichelte Linie deutet an, daß man auch das konventionelle Modell in deklarativen Sprachen kodieren und mit dem Inferenzmechanismus auswerten könnte.<sup>24</sup>

*Tabelle 3* gibt ausgewählte Rechenergebnisse wieder. Rechenzeiten sind in CPU-sec auf einer IBM RISC 6000 Modell 550 angegeben. In den ersten drei Spalten ist die Größe des jeweils gelösten Problems durch  $O$ ,  $V$  und  $T$  angegeben. Der Unterschied zwischen beiden Datensätzen besteht darin, daß das erste Beispiel eine zulässige Lösung ( $z$ ) besitzt, während das zweite Beispiel (mit modifizierten Kapazitätsrestriktionen) keine zulässige Lösung ( $n$ ) hat. In der mit CHARME überschriebenen Spalte steht die vom Inferenzmechanismus jeweils benötigte CPU-Zeit. In der Spalte LINGO steht die für die Generierung des MPS-Datenfiles erforderliche Rechenzeit, unter LINDO die für das Bestimmen einer zulässigen Lösung (bzw den Nachweis, daß keine existiert) benötigte CPU-Zeit.

*Tabelle 3: Ausgewählte Rechenergebnisse*

$O$	$V$	$T$	CHARME	LINGO	LINDO	
4	9	50	45	48	353	(z)
4	9	50	1	48	10	(n)

<sup>23</sup> In Anlehnung an *Abbildung 3* aus Drexl et al. (1995).

<sup>24</sup> Eine Möglichkeit, auf die aus den in Drexl et al. (1995) genannten Gründen auch hier verzichtet wird (i.d.R. weniger effizient etc.).

## 5 Bewertung und Ausblick

Abschließend stellen wir den Car Sequencing- und den Level Scheduling-Ansatz einander gegenüber (vgl. *Tabelle 4*) und vergleichen die verschiedenen Modellierungsmöglichkeiten und die dadurch implizierten Inferenzmechanismen (vgl. *Tabelle 5*).

*Tabelle 4: Level Scheduling vs. Car Sequencing*

	Level Scheduling	Car Sequencing
Arbeitsbelastung einzelner Stationen erfaßbar ?	nein	ja
Zusammenhang zwischen Varianten (Optionen) und Materialfluß abbildbar ?	nein	ja

Tabelle 4 verdeutlicht, daß der Car Sequencing- dem Level Scheduling-Ansatz hinsichtlich zweier wichtiger Eigenschaften überlegen ist, nämlich der Möglichkeit, die Arbeitsbelastungen einzelner Stationen explizit zu erfassen und der Abbildung des Zusammenhangs zwischen Reihung der Varianten und dem durch die Optionen induzierten Materialbedarf. Der Car Sequencing-Ansatz ist daher klar vorzuziehen, auch wenn der erforderliche Rechenzeitbedarf i.d.R. höher ist.

Ein summarischer Vergleich der Modelle (9)-(12) und (13)-(14) mit den zugehörigen Inferenzmechanismen ist *Tabelle 5* zu entnehmen. Es zeigt sich, daß für den Car Sequencing-Ansatz das konzeptionelle Modelle mit logischer Inferenz dem konventionellen Modell mit mathematischer Programmierung überlegen ist, und zwar im Hinblick auf vier Aspekte:

- Wie aus *Tabelle 3* bereits ersichtlich wurde, kann bereits der Aufwand für die Generierung des MPS-Datenfiles erheblich sein. Für größere praktische Probleme dürfte er sich i.a. sowohl hinsichtlich Rechenzeit- als auch Speicherplatzanforderungen als undurchführbar erweisen.
- Der für die Bestimmung einer zulässigen Lösung (bzw. dem Nachweis, daß keine existiert) erforderliche Rechenaufwand ist bei mathematischer Programmierung teilweise erheblich, v.a. aber auch stark schwankend. Auch bei logischer Inferenz

kann sehr viel Rechenzeit erforderlich sein, allerdings schwanken die Rechenzeiten i.a. weniger stark.

Tabelle 5: Vergleich der Modelle (9)-(12) und (13)-(14)

	Modell (9)-(12)	Modell (13)-(14)
Aufwand Generierung MPS-Datenfile	erheblich	entfällt
Rechenzeitbedarf bei Inferenz bzw. math. Programmierung	z.T. groß, stark schwankend	z.T. groß, eher stabil
Teillösungen brauchbar?	nein	ja
alle zulässigen Lösungen generierbar?	nein	ja

- Bricht man die Berechnung vorzeitig ab, so liegen i.a. nur teilweise zulässige Lösungen vor. Bei mathematischer Programmierung sind diese unbrauchbar, da für beliebige Variablen  $x_{vt}$  die Binäritätsbedingung  $x_{vt} \in \{0, 1\}$  noch nicht erfüllt sein kann, d.h.  $0 < x_{vt} < 1$  gilt. Bei logischer Inferenz sind auch Teillösungen brauchbar, da die Inferenz so gesteuert werden kann, daß nur Variablen "in der Nähe" des Planungshorizontes  $T$  noch nicht fixiert sind. Damit liegen auch bei vorzeitigem Abbruch verwertbare Teilpläne vor.
- Schließlich sind mit logischer Inferenz "per Knopfdruck" alle zulässigen Lösungen generierbar. Damit bieten sich Möglichkeiten der Evaluation bzgl. qualitativer Kriterien. Die mathematische Programmierung bietet diesen "Komfort" der Bestimmung aller zulässigen Lösungen i.a. nicht.

Zusammenfassend ist also die Modellierung (13)-(14) derjenigen von (9)-(12) eindeutig vorzuziehen. Weiterführende Forschungsarbeiten werden daher diese Richtung beschreiben und verallgemeinerte Fragestellungen der Variantenfließfertigung im Kontext einer konzeptionellen Modellierung mit logischer Inferenz analysieren.

## Zusammenfassung

Kurze Durchlaufzeiten trotz kundenorientierter, variantenreicher Produktion sind in vielen Branchen Bestandteil einer unverzichtbaren Wettbewerbsstrategie. Da die unterschiedlichen Varianten i.d.R. in Konkurrenz um die verfügbare Kapazität der Fertigungsanlagen stehen, ist ein erhöhter Koordinationsaufwand in der Ablaufplanung und Produktionssteuerung erforderlich. In dieser Situation werden in der Literatur v.a. Modelle und Verfahren zum "Level Scheduling" für "Mixed-model Assembly Lines" vorgeschlagen. Mit ihrer Hilfe gelingt es jedoch nicht, den Materialfluß zu verstetigen sowie die Belastung der Arbeitsstationen zu erfassen. Als Alternative bietet sich das sogenannte "Car Sequencing" an, das im vorliegenden Beitrag mathematisch formuliert wird, und zwar als gemischt-ganzzahliges Programmierungsmodell und als konzeptionelles Modell. Während ersteres unmittelbar von MIP-Solvern gelöst werden kann, ist das zweite direkt in deklarativen Sprachen codier- und vom Inferenzmechanismus auswertbar. Die Vorteile dieses Ansatzes werden an Beispielen veranschaulicht.

## Summary

For diversified small-lot production many companies use mixed-model assembly lines. This way they satisfy customers' demand without holding large inventories, for a variety of products. Products compete for the scarce capacities and make effective line balancing necessary. Traditionally, in this situation level schedules for just-in-time production systems are suggested. But, level schedules do not succeed in levelling the material flow and balancing the stations' workload. As an alternative the so-called "car sequencing" has been suggested. In this paper we model the car sequencing problem as a mixed-integer programming problem. Moreover, it is modelled as a conceptual model which is well-suited to be encoded in constraint-logic programming solvers. We provide insight in the advantages of the second approach.

## Literaturverzeichnis

- Berkley, B.J. (1992): A review of the kanban production control research literature. *Production and Operations Management*, Vol. 1, S. 393-411.
- BULL (1990): CHARME VI user's guide and reference manual. Artificial Intelligence Development Centre, BULL S.A. France.
- Decker, M. (1993): Variantenfließfertigung. Physica, Heidelberg.
- Dincbas, M.; H. Simonis und P. van Hentenryck (1988): Solving the car-sequencing problem in constraint logic programming, in: *Proceedings of the European Conference on Artificial Intelligence (ECAI-88)*, München, S. 290-295.
- Domschke, W.; A. Scholl und S. Voß (1993): *Produktionsplanung - Ablauforganisatorische Aspekte*. Springer, Berlin.
- Drexel, A.; B. Fleischmann; H.-O. Günther; H. Stadtler und H. Tempelmeier (1994): Konzeptionelle Grundlagen kapazitätsorientierter PPS-Systeme. *Zeitschrift für betriebswirtschaftliche Forschung*, Jg. 46, S. 1022-1045.

- Drexl, A. und C. Jordan (1994): Modellierung von Materialflußsystemen mit neuen Softwaretools. 1. Karlsruher Workshop über Lean-Techniken in der Kraftfahrzeugwirtschaft.
- Drexl, A.; E. Pesch und F. Salewski (1995): Zur Bedeutung der Modellbildung für die Entwicklung wissensbasierter Systeme. Erscheint in Zeitschrift für Betriebswirtschaft.
- Fandel, G. und P. Francois (1989): Just-in-Time-Produktion und -Beschaffung - Funktionsweise, Einsatzvoraussetzungen und Grenzen. Zeitschrift für Betriebswirtschaft, 59. Jg., S. 531-544.
- Inman, R.R. und R.L. Bulfin (1991): Sequencing JIT mixed-model assembly lines. Management Science, Vol. 37, S. 901-904.
- Jünemann, R. (1989): Materialfluß und Logistik - Systemtechnische Grundlagen mit Praxisbeispielen. Springer, Berlin u.a.
- Knolmayer, G. (1987): Materialflußorientierung statt Materialbestandsoptimierung: Ein Paradigmawechsel in der Theorie des Produktions-Managements?, in: Baetge, J. et al. (Hrsg.): Logistik - eine Aufgabe der Unternehmenspolitik. Duncker & Humblot, Berlin, S. 53-69.
- Krajewski, L.J.; B.E. King; L.P. Ritzman und D.S. Wong (1987): Kanban, MRP, and shaping the manufacturing environment. Management Science, Vol. 33, S. 39-57.
- Kubiak, W. und S. Sethi (1991): A note on "level schedules for mixed-model assembly lines in just-in-time production systems". Management Science, Vol. 37, S. 121-122.
- Miltenburg, J. (1989): Level schedules for mixed-model assembly lines in just-in-time production systems. Management Science, Vol. 35, S. 192-207.
- Miltenburg, J. und G. Sinnamon (1989): Scheduling mixed-model multi-level just-in-time production systems. International J. of Production Research, Vol. 37, S. 1487-1509.
- Miltenburg, J. und G. Sinnamon (1992): Algorithms for scheduling multi-level just-in-time production systems. IIE Transactions, Vol. 24, S. 121-130.
- Miltenburg, J.; G. Steiner und S. Yeomans (1990): A dynamic programming algorithm for scheduling mixed-model, just-in-time production systems. Mathl. Comput. Modelling, Vol. 13, S. 57-66.
- Parello, B.D. (1988): CAR WARS - the (almost) birth of an expert system. AI Expert, Vol. 3, S. 60-64.
- Parello, B.D.; W.C. Kabat und L. Wos (1986): Job-shop scheduling using automated reasoning: a case study of the car-sequencing problem. Journal of Automated Reasoning, Vol. 2, S. 1-42.
- Pesch, E.; A. Drexl und A. Kolen (1994): Modellbasierte Inferenz in CHARME. OR Spektrum, Bd. 16, S. 193-202.
- Steiner, G. und S. Yeomans (1993): Level schedules for mixed-model, just-in-time processes. Management Science, Vol. 39, S. 728-735.
- Thomopoulos, N.T. (1967): Line balancing-sequencing for mixed-model assembly. Management Science, Vol. 14, S. B59-B75.
- Yano, C.A. und R. Rachamadugu (1991): Sequencing to minimize work overload in assembly lines with product options. Management Science, Vol. 37, S. 572-586.
- Zäpfel, G. (1991): Produktionslogistik - Konzeptionelle Grundlagen und theoretische Fundierung. Zeitschrift für Betriebswirtschaft, 61. Jg., S. 209-235.
- Ziegler, H. (1990): Produktionssteuerung bei Mehrproduktfließlinien. Habilitationsschrift, Universität-GH-Paderborn, Fachbereich Wirtschaftswissenschaften.