

Universität Würzburg
Institut für Informatik
Research Report Series

**Vergleich zweier Analysemethoden
zur Leistungsbewertung
von Kanban Systemen**

A. K. Schömig und **M. Kahnt**

Report Nr. 105

Februar 1995

*Bayerische Julius–Maximilians Universität Würzburg,
Institut für Informatik, Lehrstuhl für Informatik III
Am Hubland, D-97074 Würzburg, Federal Republic of Germany
Tel.: +49-931-8885511, Fax: +49-931-8884601,
e-mail: schoemig@informatik.uni-wuerzburg.de*

Abstract: *Kanban–Systeme sind eine Möglichkeit um die Just–In–Time Philosophie als Fertigungssteuerungssystem zu implementieren. Wesentliches Ziel hierbei ist die Reduzierung der Lagerbestände innerhalb des eigentlichen Produktionsprozesses.*

Wir vergleichen zwei analytische Methoden zur Leistungsbewertung von Kanban–Systemen. Diese beruhen zum einen auf der direkten stochastischen Analyse des Zustandsraumes einer Kanban–Zelle, zum anderen auf einem Verfahren zur Analyse von Warteschlangennetze ohne Produktformeigenschaft. Abschließend diskutieren wir die Komplexität der vorgestellten Verfahren.

Keywords: Kanban, Scheduling, Warteschlangennetze, Methode von Marie, Markovketten, Simulation

1 Grundlagen

1.1 Zentrale Produktionsplanung

Bei vielen Produktionsplanungs- und steuerungssystemen (PPS-Systeme) handelt es sich um zentrale Informationssysteme der Produktionsplanung. Sie ermitteln bei mehrstufigen Produktionsprozessen ausgehend von den Erzeugnissen und deren Lieferterminen wann welche Teile in welcher Menge zur Verfügung stehen müssen. Der wichtigste Bestandteil zentraler PPS-Systeme ist eine computerbasierte Variante der Materialbedarfsplanung, die im Englischen *Material Requirements Planning* (MRP) genannt wird. Ein damit gesteuertes Produktionssystem wird auch als *Push-System*¹ bezeichnet, da die Teile, die zur Erfüllung eines Auftrags nötig sind, von der ersten bis zur letzten Produktionsstufe von dem aus der MRP resultierenden Produktionsplan durch die Produktion geschoben werden. Man spricht in diesem Zusammenhang auch von *Bringpflicht*: Die einzelnen Stufen müssen von ihnen zu fertigende Teile bis zu einem gegebenen Termin anliefern.

Sowohl bei PPS-Systemen, als auch bei manuellen Produktionsplanungssystemen, sind große Lagerbestände und ein unangemessen hoher Anteil der Wartezeit an den Durchlaufzeiten der Aufträge bzw. Fertigungslose festzustellen. Dieser Anteil beträgt nach Hestermann [10] über 75%.

1.2 Just-in-Time

Das Ziel der JIT Produktion ist sowohl die Reduktion der Lagerhaltung, als auch die Verringerung des Bestandes, der sich insgesamt im Produktionsprozeß befindet. Jeder Bestand, der nicht zur Produktion des aktuellen Bedarfs dient, wird als überflüssig angesehen und genauso schlecht wie Lagerhaltung beurteilt.

Die Verringerung des Bestands wird mit sogenannten *Pull-Systemen* verwirklicht. Ein *Pull-System* wird, anders als ein *Push-System*, nicht durch einen zentralen Plan, sondern durch den aktuellen Bedarf der einzelnen Produktionsstufen gesteuert. Dies bedeutet, daß im Fall von Lagerhaltung die Entnahme eines Fertigteils aus dem Endlager oder der Auftrag selbst, die Produktion eines Neuteils anstößt. Im anschließenden Produktionsverfahren löst jeweils die konkrete Nachfrage nach Vorprodukten in einer Produktionsstufe die Fertigung in den direkt vorgelagerten Stufen aus. Es besteht also auf Seiten der nachgelagerten Stufe eine *Holpflicht* für Vorprodukte. Die Teile werden gemäß der Endnachfrage durch den Produktionsprozeß gezogen, so daß nur die Teile gefertigt werden, die aktuell benötigt werden. Falls ausschließlich einzelne Teile produziert und zwischen den Stufen ausgetauscht werden, entspricht dies dem Idealfall der Losgröße eins. Doch auch bei anderen Losgrößen soll die kleinstmögliche verwendet werden, um so die Wartezeiten der Teile innerhalb der Lose zu minimieren.

1.3 Kanban-gesteuerte Fertigungssysteme

Die *Kanban* Steuerung stellt keine eigenständige Variante der Produktionssteuerung dar, sondern ist lediglich eine Möglichkeit die Prinzipien der JIT Produktion in einem Fertigungssystem zu verwirklichen. Der Begriff *Kanban* kommt aus dem Japanischen und

¹Siehe z.B. [6]

bedeutet soviel wie Schild, oder Karte. Die Hauptmerkmale eines Kanban-gesteuerten Fertigungssystems (oder kurz: Kanban-Systems), wie es von Toyota eingeführt worden ist, sind nach Esparrago [5]:

- ◇ Aufteilung des Produktionssystems in einzelne *Produktionsstellen*.
- ◇ *Eingangs-* und *Ausgangslager* für jede Produktionsstelle.
- ◇ *Transport-Kanban-Karten* zur Steuerung des Transports von Teilen zwischen den Produktionsstellen.
- ◇ *Produktions-Kanban-Karten* zur Steuerung der Produktion in den Produktionsstellen.
- ◇ *Ablagen* für die Transport- und Produktions-Karten.
- ◇ Standardisierte *Container* für den Transport einer festen Anzahl von Teilen.

Berkley [1] bietet einen Überblick und eine Klassifizierung der gegenwärtig sich im Einsatz befindlichen Kanban-Systeme. Die Verwendung von Kanban-Karten zur Steuerung der Produktion und des Transports von Teilen dient in gleichem Maß der Bestandskontrolle, wie die ausschließliche Benutzung genormter Container für den Transport der Teile.

Bei den hier verwendeten Modellen wird das *Singlecard Kanban-System* betrachtet, dessen Steuerung mit nur einer Art von Kanban-Karte verwirklicht wird. Dies ist möglich, falls keine räumliche Distanz zwischen den Produktionsstufen existiert.

2 Analyse nach Mitra und Mitrani

2.1 Modellbeschreibung

Entsprechend dem Modell von Mitra und Mitrani [9], setzt sich eine solche Kanban gesteuerte Fertigungslinie aus mehreren Zellen zusammen. Es handelt sich um ein einfaches Kanban-System, d.h. jede Kanban-Zelle besteht aus nur einer Bedieneinheit mit Warteschlange, der die Teile gemäß der FIFO Disziplin entnommen werden, dem *Bulletin Board*, einem Puffer für freie Kanban-Karten und dem *Output Hopper*, einem Puffer für fertig bearbeitete Teile mit Kanban-Karte. Die Kapazität dieser Puffer ist jeweils gleich der festen Anzahl von Kanban-Karten, die jeder Zelle zur Verfügung steht.

Der Fluß der Teile und Kanban-Karten durch die Zellen des Kanban-Systems (siehe Abbildung 1) kann nun folgendermaßen beschrieben werden: Befindet sich im Output Hopper in Zelle $(k - 1)$ ein fertiges Teil mit Kanban-Karte und im Bulletin Board in Zelle k eine freie Kanban-Karte, so wird die Kanban-Karte vom Teil in Zelle $(k - 1)$ gelöst und in das Bulletin Board von Zelle $(k - 1)$ eingereiht. Das Teil kann nun in Zelle k übergehen, bekommt dort die freie Kanban-Karte zugeteilt und wandert in den Puffer der Bedieneinheit. Wenn das Teil fertig bedient ist, geht es in den Output Hopper der Zelle k . Falls das Bulletin Board in Zelle $(k + 1)$ leer ist, wartet das Teil im Output Hopper bis eine freie Kanban-Karte im Bulletin Board von Zelle $(k + 1)$ verfügbar ist. Ist dies der Fall, dann wird die zugehörige Kanban-Karte vom Teil getrennt, das Teil wechselt in Zelle

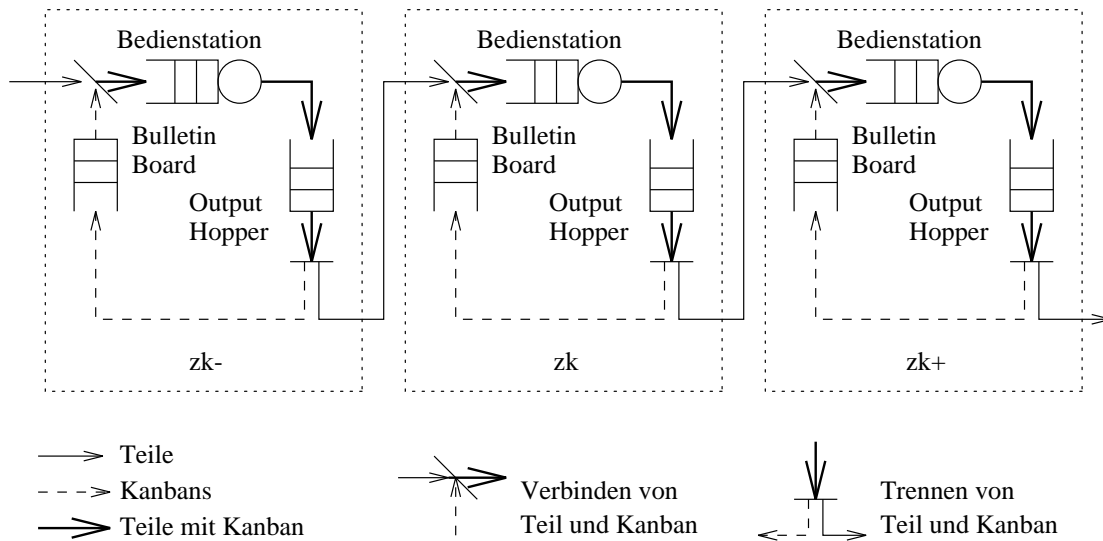


Abbildung 1: *Fluß der Teile und Kanban-Karten in einem einfachen Kanban-System*

$(k + 1)$ und die nun freie Kanban-Karte in Zelle k geht in das Bulletin Board von Zelle k , wo sie auf ein fertig bedientes Teil im Output Hopper von Zelle $(k - 1)$ wartet.

Aus der oben beschriebenen Verfahrensweise folgt, daß innerhalb einer Zelle die Summe der in der Bedienstation² und im Output Hopper vorhandenen Teile und der freien Kanban-Karten im Bulletin Board die Gesamtzahl der Kanban-Karten dieser Zelle nicht überschreiten kann. Auch ist das gleichzeitige Vorhandensein von freien Kanban-Karten im Bulletin Board einer Zelle k und von bedienten Teilen im Output Hopper der Zelle $(k - 1)$ nicht möglich; mindestens einer dieser beiden Puffer ist immer leer. Folglich kann der Zustand einer Zelle durch Angabe der wartenden Kanban-Karten im Bulletin Board und der wartenden Teile im Output Hopper vollständig beschrieben werden.

Da ein *saturiertes* System (unendliche Menge an Rohteilen und unendlicher Bedarf an Fertigteilen) betrachtet wird, muß niemals eine freie Kanban-Karte im Bulletin Board der ersten Zelle, oder ein Fertigteil im Output Hopper der letzten Zelle warten. Folglich werden diese beiden Puffer im Modell weggelassen.

2.2 Betrachtung der isolierten Zelle

Beschreibt man den Zustand eines Kanban-Systems durch Vektoren, die die Anzahlen der Kanban-Karten in den Bulletin Boards und der Teile in den Output Hoppers der einzelnen Zellen enthalten, so kann der darauf aufbauende Markov-Prozeß mit angemessenen Methoden analysiert werden. Sein Zustandsraum wächst jedoch exponentiell mit der Anzahl der Kanban-Karten und Zellen. Da sich nach Mitra und Mitrani [9] schon bei nur einer Kanban-Karte und N Zellen $0,447(2,618)^{N+1}$ Zustände, für große N ergeben, muß eine Analyse approximativ erfolgen.

Hierzu betrachtet man zunächst die isolierte Zelle k (vgl. Abb. 2), die durch einen Poissonstrom mit mittlerer Ankunftsrate ρ_k mit Rohteilen versorgt wird und von der durch

²Mit Bedienstation soll hier die Bedieneinheit einschließlich Warteschlange bezeichnet werden.

einen Poissonstrom mit mittlerer Ankunftsrate σ_k Fertigteile angefordert werden. Beide Ströme werden auf dieser Stufe der Analyse als bekannt angenommen. Die Bedienzeiten der Bedieneinheit der Zelle sind exponentiell verteilt mit mittlerer Bedienrate μ_k . In der Zelle befinden sich C_k Kanban-Karten. Ferner gibt es einen *External Parts Buffer* mit Kapazität C_{k-1} , in dem ankommende Rohteile gepuffert werden und einen *External Demands Buffer* mit Kapazität C_{k+1} , in dem neu eintreffende Anforderungen zwischengespeichert werden. Da die beiden Puffer beschränkt sind, gehen ankommende Rohteile bzw. Anforderungen verloren, falls der External Parts Buffer, bzw. der External Demands Buffer voll ist.

Der Fluß der Teile entspricht dem im Kanban-System. Sind im Bulletin Board eine freie Kanban-Karte und im External Parts Buffer ein Rohteil vorhanden, so wandern beide in die Bedienstation. Falls sich im Output Hopper ein Teil und eine Anforderung im External Demands Buffer befinden, verläßt das Teil die Zelle, die Anforderung den Puffer und die freigewordene Kanban-Karte geht in das Bulletin Board.

Die Zustände des Zustandsraums S_k der isolierten Zelle k können durch die Tupel $(I_{k,t}, J_{k,t})$, mit $-C_{k-1} \leq I_{k,t} \leq C_k$, $-C_{k+1} \leq J_{k,t} \leq C_k$ und $I_{k,t} + J_{k,t} \leq C_k$ beschrieben werden. Falls $I_{k,t}$ negativ ist befinden sich $-I_{k,t}$ Rohteile zum Zeitpunkt t im External Parts Buffer, sonst $I_{k,t}$ Kanban-Karten im Bulletin Board. Ist $J_{k,t}$ negativ, so sind $J_{k,t}$ Anforderungen zur Zeit t im External Demands Buffer, andernfalls befinden sich $J_{k,t}$ Teile im Output Hopper. Die Anzahl der Teile in der Bedienstation ist gleich der Gesamtzahl der Kanban-Karten der Zelle, abzüglich der Teile im Output Hopper ($J_{k,t} > 0$) und der Kanban-Karten im Bulletin Board ($I_{k,t} > 0$). Folglich erhält man den endlichen Markov-Prozeß $\mathcal{M} = \{(I_{k,t}, J_{k,t}); t \geq 0\}$, der nach Mitra und Mitrani aufgrund

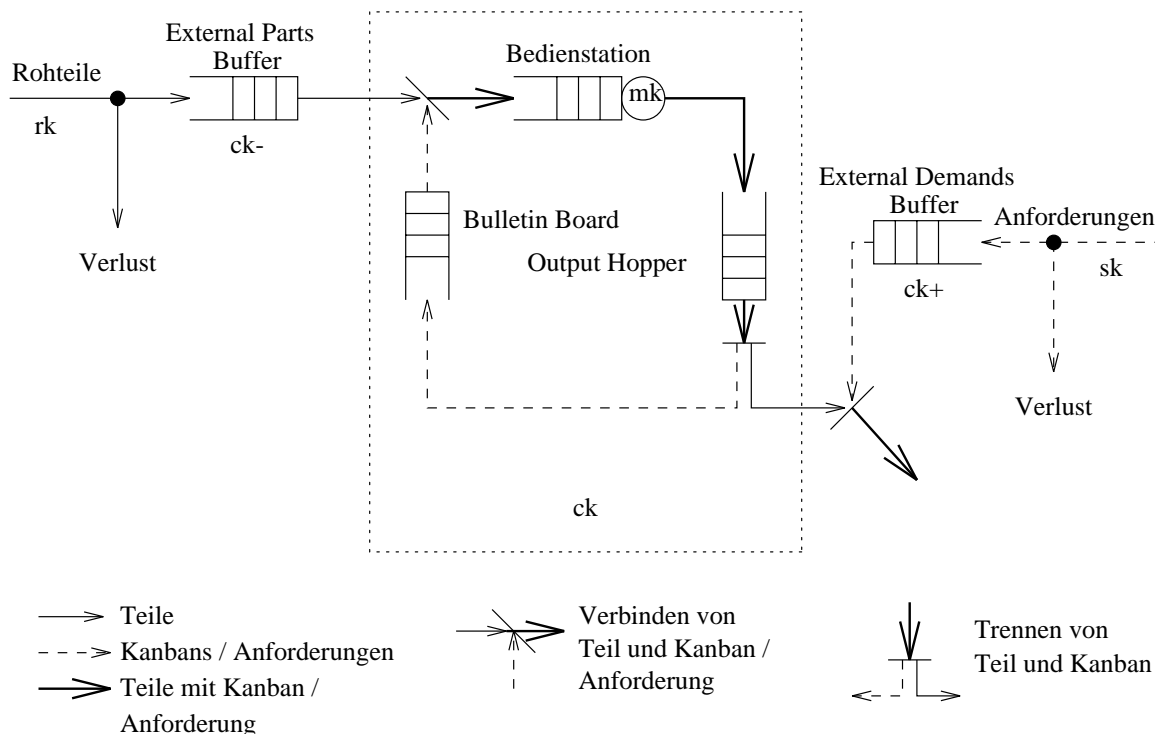


Abbildung 2: Modell der isolierten Zelle k

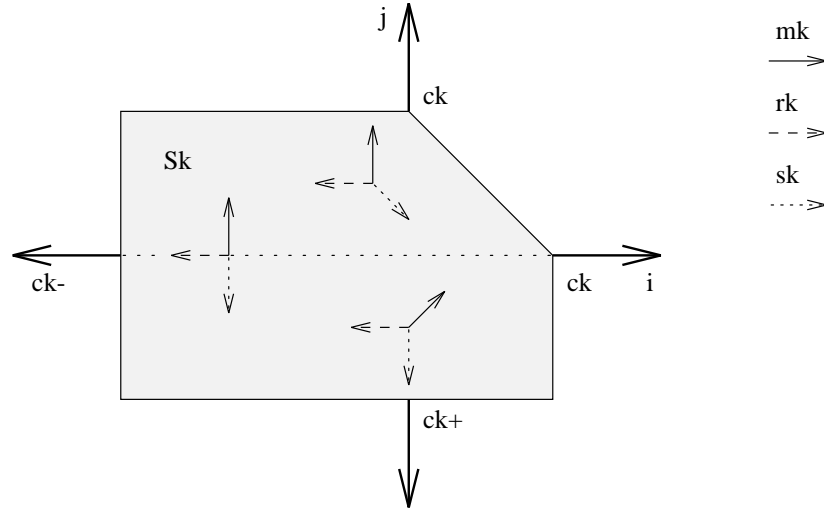


Abbildung 3: Allgemeines Zustandsübergangsdiagramm der isolierten Zelle k

seiner Irreduzibilität die Zustandsverteilung $\{p_{ij}\}$, mit $p_{ij} = \lim_{t \rightarrow \infty} P(I_{k,t} = i, J_{k,t} = j)$; $(i, j) \in S_k$ besitzt (siehe Abbildung 3).

Zur Berechnung der Zustandswahrscheinlichkeiten ergeben sich die Gleichgewichtsgleichungen

$$\begin{aligned}
 p_{ij} [\mu_k \theta(i < C_k \wedge j < C_k \wedge i + j < C_k) + \rho_k \theta(i > -C_{k-1}) + \sigma_k \theta(j > -C_{k+1})] = \\
 \mu_k [p_{i,j-1} \theta(j > 0) + p_{i-1,j-1} \theta(j \leq 0 \wedge j > -C_{k+1} \wedge i > -C_{k-1})] \\
 + \sigma_k [p_{i-1,j+1} \theta(j \geq 0 \wedge j < C_k \wedge i > C_{k-1}) + p_{i,j+1} \theta(j < 0)] \\
 + \rho_k p_{i+1,j} \theta(i < C_k \wedge i + j < C_k), \quad (i, j) \in S_k \quad (1)
 \end{aligned}$$

mit der Nebenbedingung:

$$\sum_{s \in S_k} p_s = 1. \quad (2)$$

Dabei sei

$$\theta(a) = \begin{cases} 1, & \text{falls } a \text{ wahr} \\ 0, & \text{sonst.} \end{cases}$$

Gemäß den obigen Voraussetzungen für die erste und letzte Zelle eines Kanban-Systems mit N Zellen wird $\rho_1 = \sigma_N = \infty$ und $C_0 = C_{N+1} = 1$ gesetzt. Für die erste Zelle existieren nur Zustände $(-1, j)$, für die letzte nur Zustände $(i, -1)$. Dies ermöglicht eine direkte Berechnung der Zustandswahrscheinlichkeiten aus Gleichung (1) für die Zellen 1 und N zu:

$$p_j = \frac{(1 - \alpha)\alpha^{j+C_2}}{1 - \alpha^{C_1+C_2+1}}, \quad j = -C_2, \dots, 0, \dots, C_1 \quad (3)$$

$$p_i = \frac{(1 - \beta)\beta^{i+C_{N-1}}}{1 - \beta^{C_{N-1}+C_N+1}}, \quad i = -C_{N-1}, \dots, 0, \dots, C_N \quad (4)$$

mit $\alpha = \mu_1/\sigma_1$ und $\beta = \mu_N/\rho_N$.

Der Durchsatz T_k der isolierten Zelle im Gleichgewicht kann auf drei Arten angegeben werden.

1. Als die mittlere Anzahl von Rohteilen, die die Zelle pro Zeiteinheit betreten, also nicht verloren gehen:

$$T_k = \rho_k [1 - P(I_k = -C_{k-1})], \quad k = 2, 3, \dots, N. \quad (5)$$

2. Als die mittlere Anzahl von Anforderungen, die pro Zeiteinheit erfüllt werden können, also nicht verloren gehen:

$$T_k = \sigma_k [1 - P(J_k = -C_{k+1})], \quad k = 1, 2, \dots, N - 1. \quad (6)$$

3. Und als mittlere Anzahl von Teilen, die pro Zeiteinheit bearbeitet werden, also nicht im Bulletin Board oder Output Hopper warten:

$$T_k = \mu_k \{1 - [P(I_k = C_k) + P(J_k = C_k) + P(I_k + J_k = C_k)]\}, \quad k = 1, 2, \dots, N. \quad (7)$$

2.3 Analyse des Gesamtsystems

Der Übergang von der Betrachtung der isolierten Zellen zur Analyse des gesamten Systems (siehe Abb. 4) kann unter Einbeziehung des folgenden Zusammenhangs geschehen: Zelle k wird von Zelle $(k - 1)$ mit Rohteilen versorgt und der Anforderungsstrom kommt aus Zelle $(k + 1)$. Die Rolle des External Parts Buffer übernimmt der Output Hopper von Zelle $(k - 1)$ und die des External Demands Buffer das Bulletin Board in Zelle $(k + 1)$, weshalb auch die Kapazitäten dieser beiden Puffer oben mit C_{k-1} und C_{k+1} angegeben wurden.

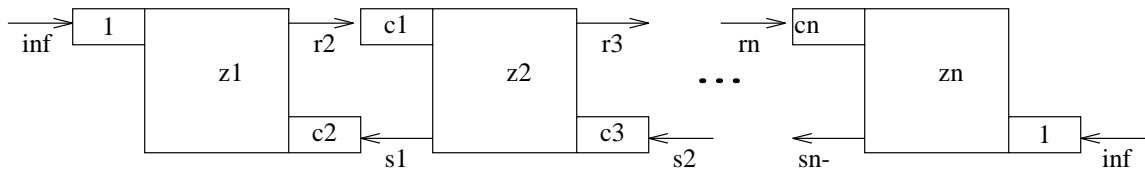


Abbildung 4: Übergang von der isolierten Zelle zum gesamten Kanban-System

Die Vektoren $\vec{\rho} = (\rho_2, \rho_3, \dots, \rho_N)$ und $\vec{\sigma} = (\sigma_1, \sigma_2, \dots, \sigma_{N-1})$ müssen jetzt noch so bestimmt werden, daß die Vorgänge innerhalb des Kanban-Systems richtig beschrieben

sind. (Aufgrund der Sonderstellung der ersten und letzten Zelle wird $\rho_1 = \sigma_N = \infty$ gesetzt.)

Da der Durchsatz durch alle Zellen des Kanban-Systems im Gleichgewicht identisch ist und die Entsprechung zwischen External Parts Buffer und Output Hopper zweier benachbarter Zellen existiert, erhalten Mitra und Mitrani aus (5) und (6) nach verschiedenen Umformungen mehrere Gleichungen der Form

$$\vec{\rho} = f(\vec{\rho}, \vec{\sigma}), \quad \vec{\sigma} = g(\vec{\rho}, \vec{\sigma})$$

zur Berechnung von $\vec{\rho}$ und $\vec{\sigma}$. Diese Fixpunktgleichungen können mit einem Iterationsverfahren numerisch gelöst werden (siehe Algorithmus Seite 10), jedoch ohne Garantie für die Existenz und Eindeutigkeit der Lösung, auch wenn die Autoren intuitive Argumente für die Eindeutigkeit geben.

Von diesen Gleichungen liefern den Versuchen von Mitra und Mitrani zufolge

$$\rho_k = \frac{T_{k-1}}{1 - P(J_{k-1} = C_{k-1})}, \quad k = 2, 3, \dots, N \quad (8)$$

$$\sigma_k = \frac{T_{k+1}}{1 - P(J_k = -C_{k+1})}, \quad k = 1, 2, \dots, N - 1 \quad (9)$$

Ergebnisse, die mit den simulierten Werten am besten übereinstimmen.

Sind die Vektoren $\vec{\rho}$, $\vec{\sigma}$ und somit auch der Durchsatz des Gesamtsystems T , der für alle Zellen gleich ist ($T = T_1 = T_2 = \dots = T_N$), berechnet, können die anderen Leistungsgrößen bestimmt werden:

- ◇ Als mittlere Anzahl von Teilen in Zelle k , L_k und im Gesamtsystem, L erhält man

$$L_k = C_k - \sum_{i=1}^{C_k} iP(I_k = i), \quad k = 1, 2, \dots, N \quad (10)$$

und

$$L = \sum_{k=1}^N L_k \quad (11)$$

- ◇ Nach Little berechnet sich die mittlere Durchlaufzeit W eines Teils durch das System zu

$$W = L / T \quad (12)$$

Zusammenfassend läßt sich der nachfolgende Algorithmus formulieren.

Algorithmus 1: *Approximative Analyse eines einfachen Kanban-Systems nach Mitra und Mitrani.*

Schritt 0: Initialisierung

- ◇ setze $\rho_k = \mu_{k-1}$, für $k = 2, 3, \dots, N$;
- ◇ setze $\sigma_k = \mu_{k+1}$, für $k = 1, 2, \dots, N - 1$;

Schritt 1: Berechnung der Zustandswahrscheinlichkeiten

- ◇ Berechnung der p_{ij} nach (1) und (2), für die Zellen 2 bis $N - 1$;
- ◇ Berechnung der p_j nach (3), für Zelle 1;
- ◇ Berechnung der p_i nach (4), für Zelle N ;

Schritt 2: Berechnung des Durchsatzes

- ◇ Berechnung der Durchsätze T_k nach (7), für $k = 1, 2, \dots, N$;

Schritt 3: Berechnung der neuen externen Ankunftsraten

- ◇ Berechnung der Ankunftsraten der Rohteile ρ_k nach (8), für $k = 2, 3, \dots, N$;
- ◇ Berechnung der Ankunftsraten der Anforderungen σ_k nach (9), für $k = 1, 2, \dots, N - 1$;

Schritt 4: Test des Abbruchkriteriums

- ◇ Falls sich die Vektoren $\vec{\rho}$ und $\vec{\sigma}$ zweier aufeinanderfolgender Iterationsschritte um mehr als ein vorgegebenes ϵ unterscheiden, gehe zu Schritt 1;

Schritt 5: Berechnung der Leistungsgrößen

1. Berechnung der Aufträge in Zelle k , L_k nach (10), für $k = 1, 2, \dots, N$;
2. Berechnung der Aufträge im System, L nach (11);
3. Berechnung der Durchlaufzeit W nach (12);

(Für diesen und alle weiteren Algorithmen wird $\epsilon = 1 \cdot 10^{-5}$ gesetzt.)

3 Analyse nach DiMascolo, Frein und Dallery

3.1 Modellbeschreibung und Dekomposition

Modellbeschreibung

Im Gegensatz zu Mitra und Mitrani, deren Modell eines Kanban-System aus interagierenden, aber streng abgegrenzten Zellen besteht, betrachten DiMascolo, Frein und Dallery in [4] das gesamte System als ein Warteschlangennetz Y . Diesem Modell liegt ebenfalls eine Produktionslinie zugrunde, die aus N Produktionsstufen besteht. Eine mit Kanban-Karten gesteuerte Produktionsstufe wird im folgenden, wie auch bei Mitra und Mitrani, als Zelle und ein Kanban-System mit mehreren Bedieneinheiten in einer Zelle als Sektor Kanban-System bezeichnet. Jede Zelle k enthält C_k Kanban-Karten und eine Anzahl von Maschinen m_k , die den Produktionsprozeß der Zelle darstellen. Der Übergang zwischen den Zellen wird durch die Kanban-Karten geregelt. Nur wenn in Zelle k eine Kanban-Karte frei ist, darf ein Teil aus Zelle $(k - 1)$ überwechseln.

Zum Vergleich mit der Analyseverfahren von Mitra und Mitrani wird hier die Untersuchung des saturierten Systems vorgestellt. Das Warteschlangennetzmodell des Kanban-Systems, wie es in Abb. 5 gezeigt ist, setzt sich aus den Unternetzen X_1 bis X_N , die den Produktionsstufen entsprechen, und je einer Synchronisationsstation J_k zwischen X_k und X_{k+1} zusammen. Jede Bedienstation i eines Unternetzes X_k steht für eine Maschine der Produktionsstufe k . J_k enthält die Puffer P_k , für Fertigteile der Zelle k , und F_{k+1} , für freie Kanban-Karten der Zelle $(k + 1)$. Dabei werden J_{k-1} und J_k auch als *upstream* und *downstream* Synchronisationsstationen der Zelle k bezeichnet. In F_k , X_k und P_k befinden sich insgesamt C_k Kanban-Karten bzw. Teile mit Kanban-Karte. Die Kanban-Steuerung ist hier folgendermaßen realisiert: Sind in P_{k-1} und F_k gleichzeitig ein Fertigteile und eine freie Kanban-Karte vorhanden, wird die Kanban-Karte aus Zelle $(k - 1)$ vom Teil getrennt und geht in F_{k-1} . Dem Teil selbst wird die freie Kanban-Karte aus Zelle k zugeteilt und es wechselt zur Bearbeitung in die erste Bedienstation in X_k . Hat das Teil X_k durchlaufen, wartet es in P_k auf eine freie Kanban-Karte in F_{k+1} , um unter Freiwerden der Kanban-Karte aus Zelle k in X_{k+1} überzugehen.

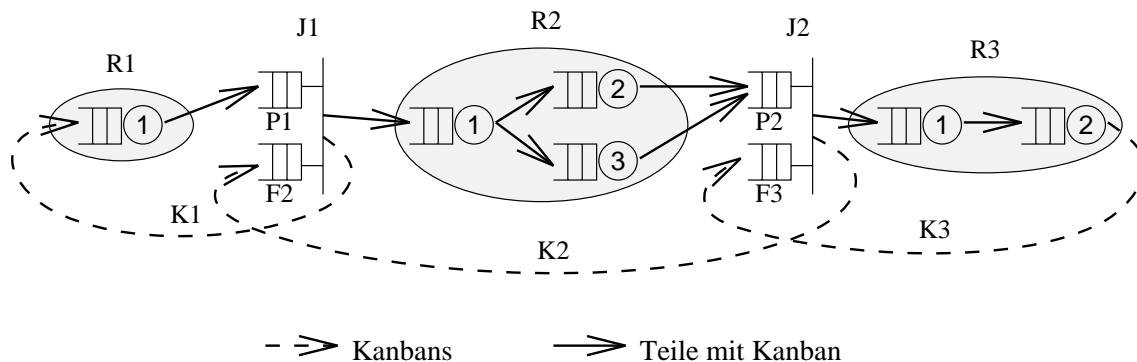


Abbildung 5: Sektor Kanban-System als Warteschlangennetz Y

Dekomposition des Warteschlangennetzes Y

Für die spätere Analyse werden Bedieneinheiten mit mittlerer Bedienrate μ_k^i verwendet, deren Bedienzeitverteilung durch eine Phasenverteilung darstellbar ist. Die Übergänge zwischen den Bedienstationen der X_k erfolgen zufällig. Eine direkte Untersuchung des gesamten Netzes Y mit einem Markov-Prozeß, dessen Zustände durch die Angabe der Zahl der Teile in den einzelnen Stationen und der Phase in der sich die Bedieneinheit befindet, gegeben sind, führt bei größeren Systemen zu einem aufgrund seiner Größe nicht mehr handhabbaren Zustandsraum. Deshalb soll eine approximative Methode Anwendung finden, zu deren Verwendung eine Aufteilung von Y in leichter analysierbare Subsysteme Y_k , $k = 1, 2, \dots, N$, notwendig ist.

Durch die Dekomposition des Warteschlangennetzes Y entstehen Subsysteme Y_k , $k = 1, 2, \dots, N$, die jeweils von einer upstream Synchronisationsstation I_k , einem Unternetz X_k und einer downstream Synchronisationsstation O_k gebildet werden (vgl. Abbildung 6). Die Beschränkung auf das saturierte System führt zum Wegfall der Synchronisationsstationen I_1 in Y_1 und O_N in Y_N . In allen Subsystemen Y_k dient I_k der Synchronisation zwischen dem Ankunftsstrom der Rohteile von Zelle k und deren Kanban-Karten. Es handelt sich um einen Poissonstrom mit mittleren lastabhängigen Ankunftsraten $\lambda_k^u(n_k^u)$. O_k synchronisiert die Fertigteile der Zelle k mit dem Poissonstrom der Anforderungen an Zelle k , der eine mittlere, ebenfalls lastabhängige, Ankunftsrate $\lambda_k^d(n_k^d)$ besitzt. Dabei steht n_k^u , $0 \leq n_k^u \leq C_{k-1}$, für die Anzahl von Rohteilen in I_k und n_k^d , $0 \leq n_k^d \leq C_{k+1}$, für die bereits in O_k vorhandenen Anforderungen. Auf der Ebene des einzelnen Subsystems wird davon ausgegangen, daß alle externen Ankunftsraten $\lambda_k^u(n_k^u)$ und $\lambda_k^d(n_k^d)$ bekannt sind.

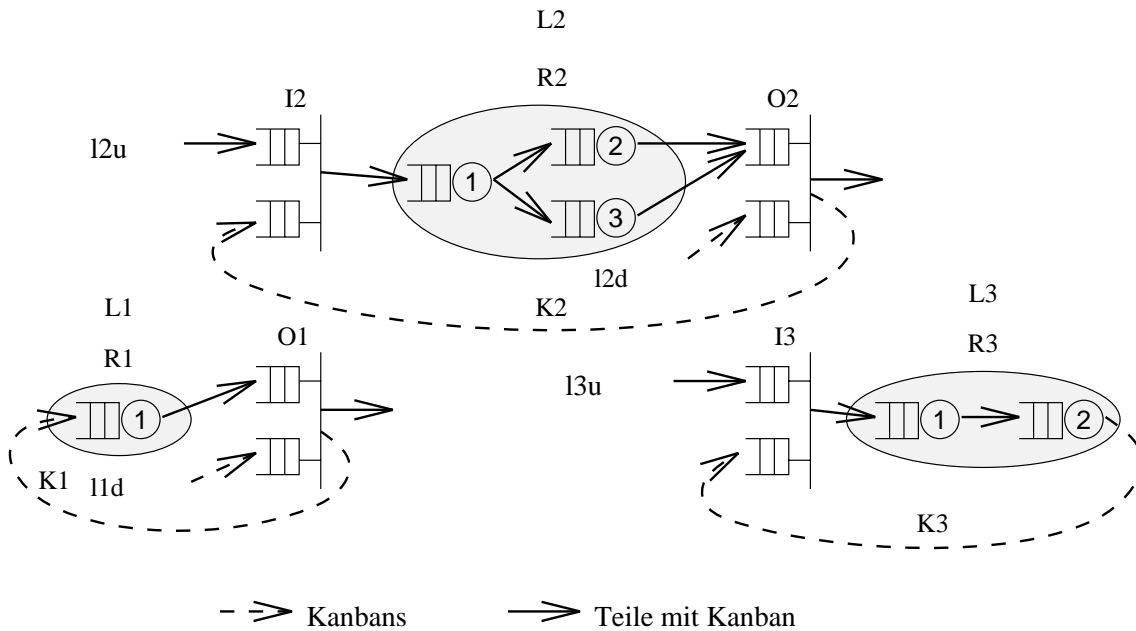


Abbildung 6: Dekomposition des Warteschlangennetzes Y der vorherigen Abbildung in die Subsysteme Y_1 , Y_2 und Y_3

Wie Abb. 8 verdeutlicht, können folgende Gleichgewichtsgleichungen bestimmt werden:

$$p_{n_k^I,0} \lambda_k^u(0) = p_{n_k^I-1,0} \lambda_k^I(n_k^I - 1), \quad n_k^I = 1, 2, \dots, C_k \quad (13)$$

$$p_{0,n_k^u} \lambda_k^I(0) = p_{0,n_k^u-1} \lambda_k^u(n_k^u - 1), \quad n_k^u = 0, 1, \dots, C_{k-1} \quad (14)$$

Die Wahrscheinlichkeit $p_k^I(n_k^I)$, daß sich n_k^I Aufträge (Kanban-Karten) in Station I_k befinden, kann aus den Gleichungen

$$p_k^I(n_k^I) = p_{n_k^I,0}, \quad n_k^I = 1, 2, \dots, C_k \quad (15)$$

$$p_k^I(0) = \sum_{n_k^u=0}^{C_{k-1}} p_{0,n_k^u} \quad (16)$$

errechnet werden. Mit (14) und (16) ergibt sich dann für $p_k^I(0)$:

$$p_k^I(0) = p_{0,0} \left(1 + \sum_{n_k^u=1}^{C_{k-1}} \frac{\prod_{n=0}^{n_k^u-1} \lambda_k^u(n)}{\lambda_k^I(0)^{n_k^u}} \right) \quad (17)$$

Die allgemeine Formel zur Berechnung des bedingten Durchsatzes eines Knotens aus [7] präsentiert sich, angewendet auf Station I_k , in der Form:

$$\nu_k^I(n_k^I) = \lambda_k^I(n_k^I - 1) \frac{p_{n_k^I-1,0}}{p_{n_k^I,0}}, \quad n_k^I = 1, 2, \dots, C_k \quad (18)$$

Der bedingte Durchsatz von I_k kann zusammenfassend mit den Gleichungen (18), (13) und (15) als

$$\nu_k^I(n_k^I) = \lambda_k^I(0), \quad n_k^I = 2, 3, \dots, C_k \quad (19)$$

sowie mit (18), (13), (15) und (17) als

$$\nu_k^I(1) = \lambda_k^I(0) \left(1 + \sum_{n_k^u=1}^{C_{k-1}} \frac{\prod_{n=0}^{n_k^u-1} \lambda_k^u(n)}{\lambda_k^I(0)^{n_k^u}} \right) \quad (20)$$

angegeben werden.

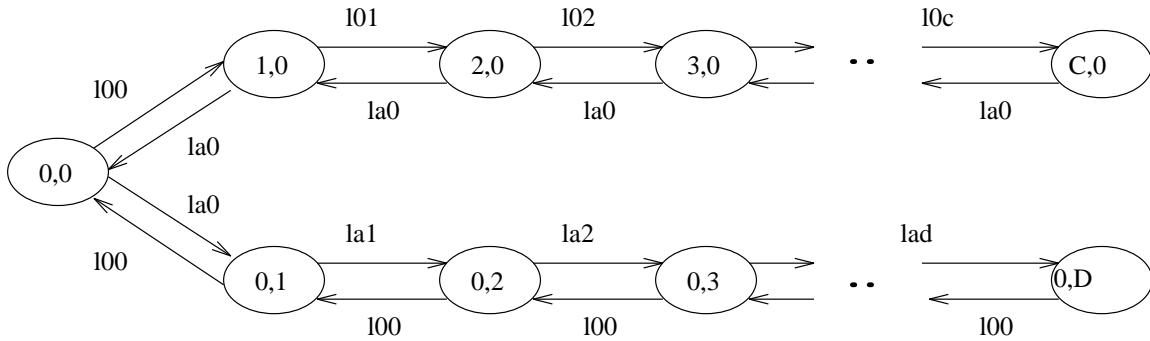


Abbildung 8: Zustandsübergangsdigramm der Synchronisationsstation I_k

Die Formeln der bedingten Durchsätze $\nu_k^O(n_k^O)$ der Synchronisationsstation O_k lauten

$$\nu_k^O(n_k^O) = \lambda_k^O(0), \quad n_k^O = 2, 3, \dots, C_k \quad (21)$$

$$\nu_k^O(1) = \lambda_k^O(0) \left(1 + \sum_{n_k^d=1}^{C_{k+1}} \frac{\prod_{n=0}^{n_k^d-1} \lambda_k^d(n)}{\lambda_k^O(0)^{n_k^d}} \right) \quad (22)$$

und können auf gleiche Weise hergeleitet werden.

Verwendet man in der Methode von Marie zusätzlich die Gleichungen (19) und (20), bzw. (21) und (22) zur Berechnung der bedingten Durchsätze der beiden Synchronisationsstationen, dann kann das gesamte Subsystem Y_k analysiert werden. Dabei sind $\lambda_k^I(n_k^I)$ und $\lambda_k^O(n_k^O)$ die lastabhängigen Ankunftsrate in den Stationen I_k und O_k , wie sie sich bei der Analyse nach Marie aus dem Faltungsalgorithmus [2] ergeben. Für dessen Anwendung müssen alle J_k zu Beginn der Untersuchung mit Bedienraten μ_k^I und μ_k^O initialisiert werden. Hierfür bieten sich λ_{k-1}^O und λ_{k+1}^I (siehe Algorithmus 2) an.

3.3 Analyse des Gesamtsystems

Wenn man von der Betrachtung eines einzelnen Subsystems Y_k zur Analyse des ursprünglichen Warteschlangennetzes Y zurückgeht, ist es notwendig sich nochmals die Zusammenarbeit der Subsysteme vor Augen zu halten. Wie bereits ausgeführt, wird jede Synchronisationsstation J_k im dekomponierten System durch die downstream Station O_k der Zelle k und die upstream Station I_{k+1} der Zelle $(k+1)$ repräsentiert. Die Versorgung der Zelle $(k+1)$ mit Rohteilen findet aus Zelle k statt, die Anforderungen an Zelle k stammen wiederum aus Zelle $(k+1)$. Daraus ergibt sich, daß der Strom der externen Rohteile nach I_{k+1} gleich dem Strom der Fertigteile von X_k nach O_k ist. Außerdem ist der Strom der externen Anforderungen nach O_k gleich dem der freien Kanban-Karten im Subsystem Y_{k+1} von O_{k+1} nach I_{k+1} . Dieser Zusammenhang kann allgemein mit den Gleichungen

$$\lambda_k^u(n) = \lambda_{k-1}^O(n), \quad n = 0, 1, \dots, C_{k-1} - 1, \quad k = 2, 3, \dots, N \quad (23)$$

$$\lambda_k^d(n) = \lambda_{k+1}^I(n), \quad n = 0, 1, \dots, C_{k+1} - 1, \quad k = 1, 2, \dots, N - 1 \quad (24)$$

beschrieben werden.

Wie aus (23) und (24) ersichtlich ist, können die upstream Ankunftsrate $\lambda_k^u(n)$ aus der Analyse von Y_{k-1} gewonnen werden, die downstream Raten $\lambda_k^d(n)$ ergeben sich bei der Untersuchung von Y_{k+1} . Dies legt ein iteratives Vorgehen nahe, da jedes Subsystem erst dann vollständig untersucht werden kann, wenn alle Größen der benachbarten Systeme bereits bestimmt sind. Das Iterationsverfahren besteht aus zwei Schritten: Im ersten Schritt werden die Subsysteme Y_k , $k = 1, 2, \dots, N - 1$ analysiert. Stehen dabei die $\lambda_k^O(n)$ der Zelle k fest, wird $\lambda_{k+1}^u(n) = \lambda_k^O(n)$ gesetzt und Y_{k+1} kann betrachtet werden. Die Werte der $\lambda_k^I(n)$ ergeben sich im zweiten Schritt durch die Analyse der Subsysteme Y_k , $k = N, N - 1, \dots, 2$. Auch hier wird nach der Untersuchung von Y_k $\lambda_{k-1}^d(n) = \lambda_k^I(n)$ gesetzt und mit Y_{k-1} fortgefahren.

Sind alle unbekannt Parameter des Warteschlangennetzes Y mit Hilfe des Iterationsverfahrens bestimmt, können die Leistungsgrößen Durchsatz, Anzahl der Teile im System und Durchlaufzeit berechnet werden:

- ◇ Der Durchsatz T , der für alle Zellen dieses seriellen Systems gleich sein muß, kann mit einer der Gleichungen

$$T = \begin{cases} \sum_{r=1}^{C_k} p_k^I(r) \nu_k^I(r), & k = 1, 2, \dots, N-1 \\ \sum_{i=1}^{m_k} \lambda_k^i(C_k) q_{i,0}, & k = N \end{cases} \quad (25)$$

ermittelt werden.

- ◇ Die Anzahl der Teile in Zelle k , L_k ist gleich

$$L_k = \begin{cases} C_k, & k = 1 \\ C_k - \sum_{j=1}^{C_k} j p_k^I(j), & k = 2, 3, \dots, N \end{cases} \quad (26)$$

Aus deren Summe kann die Anzahl der Teile im System L errechnet werden.

$$L = \sum_{k=1}^N L_k \quad (27)$$

- ◇ Die mittlere Durchlaufzeit W ist nach Little:

$$W = L / T \quad (28)$$

Den Aussagen der Autoren zufolge konvergiert der im Anschluß angegebene Algorithmus zur Analyse des Gesamtsystems in den allermeisten Fällen. Wenn auch kein Beweis dafür gegeben wird, so kann die Konvergenz durch die Ergebnisse dieser Arbeit nur bestätigt werden. DiMascolo, Frein und Dallery weisen außerdem darauf hin, daß, falls der Algorithmus nicht konvergiert, die Konvergenz durch leichte nicht näher erläuterte Modifikation erreicht werden könne.

Algorithmus 2: *Approximative Analyse eines Sektor Kanban-Systems nach Di Mascolo, Frein und Dallery.*

Schritt 0: Initialisierung

- ◇ setze die downstream Raten $\lambda_k^d(n_k^I) = \mu_{k+1}^1$, für $n_k^i = 0, 1, \dots, C_{k+1} - 1$ und $\lambda_k^d(C_{k+1}) = 0$, für $k = 1, 2, \dots, N-1$;
- ◇ setze die Bedienraten der Synchronisationsstationen O_k im Hilfsnetz $\mu_k^O = \lambda_k^d(0)$, für $k = 1, 2, \dots, N-1$;

Schritt 1: Berechnung der upstream Parameter

- ◇ für $k = 1, 2, \dots, N - 1$:
 1. Analyse des Subsystems Y_k mit der um die Untersuchung von Synchronisationsstationen erweiterten Methode von Marie (Gleichungen (19)–(22)) zur Berechnung der $\lambda_k^O(n)$, für $n = 0, 1, \dots, C_k - 1$;
 2. Bestimmung der upstream Parameter $\lambda_{k+1}^u(n)$ nach (23), für $n = 0, 1, \dots, C_k - 1$;
 3. setze $\mu_{k+1}^I = \lambda_k^O(0)$;

Schritt 2: Berechnung der downstream Parameter

- ◇ für $k = N, N - 1, \dots, 2$:
 1. Analyse des Subsystems Y_k mit der um die Untersuchung von Synchronisationsstationen erweiterten Methode von Marie (Gleichungen (19)–(22)) zur Berechnung der $\lambda_k^I(n)$, für $n = 0, 1, \dots, C_k - 1$;
 2. Bestimmung der downstream Parameter $\lambda_{k-1}^d(n)$ nach (24), für $n = 0, 1, \dots, C_k - 1$;
 3. setze $\mu_{k-1}^O = \lambda_k^I(0)$;

Schritt 3: Test des Abbruchkriteriums

- ◇ Falls sich nicht alle $\lambda_k^d(n)$ und $\lambda_k^u(n)$ zweier aufeinanderfolgender Iterationsschritte um weniger als ein vorgegebenes ϵ unterscheiden, gehe zu Schritt 1.

Schritt 4: Berechnung der Leistungsgrößen

1. Berechnung des Durchsatzes T nach (25);
2. Berechnung der Aufträge in Zelle k , L_k nach (26), für $k = 1, 2, \dots, N$;
3. Berechnung der Aufträge im System L nach (27);
4. Berechnung der Durchlaufzeit W nach (28);

4 Vergleich der Analysemethoden

4.1 Numerische Ergebnisse

Im folgenden werden die beiden approximativen Analyseverfahren nach Mitra und Mitrani und nach DiMascolo, Frein und Dallery einem Vergleich unterzogen, der ihre Anwendbarkeit und die Qualität ihrer Resultate für verschiedene Fälle aufzeigen soll. Den Ergebnissen der approximativen Verfahren werden dabei simulativ ermittelte Werte gegenübergestellt, die als Maßstab für ihre Güte gelten. Diese Werte werden aus zehn unabhängigen Simulationsläufen des jeweiligen Systems gewonnen. Jeder Lauf hat eine Dauer von 10 000

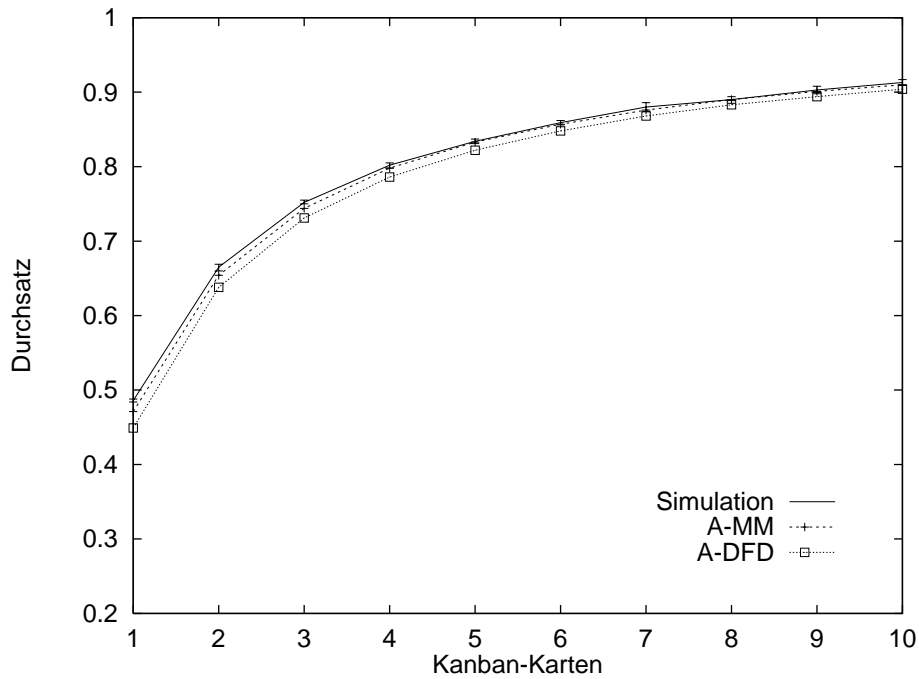


Abbildung 9: *Durchsatz bei Variation der Anzahl der Kanban-Karten im einfachen Kanban-System*

Zeiteinheiten. Die statistischen Daten der transienten Phase zu Beginn der Simulation werden verworfen.

Zur Vereinfachung der Unterscheidung der beiden approximativen Analyseverfahren wird das Verfahren von Mitra und Mitrani mit A-MM und das von DiMascolo, Frein und Dallery mit A-DFD bezeichnet.

Einfaches Kanban-System

Die einfachen Kanban-Systeme, die in diesem Abschnitt betrachtet werden, bestehen aus identischen Zellen, deren Bedieneinheiten exponentiell verteilte Bedienzeiten mit Mittelwert eins besitzen.

Abbildung 9 zeigt das bereits aus [4] und [9] bekannte Ansteigen des Durchsatzes bei gleichzeitiger Erhöhung der Kanban-Karten der Zellen, für ein System aus fünf Zellen. Die Durchsatzsteigerung ist umso größer je weniger Kanban-Karten vorhanden sind. Beide Analyseverfahren zeigen dieses Verhalten und stimmen in ihren Ergebnissen gut mit der Simulation überein. Der relative Fehler bei A-DFD liegt zwischen acht Prozent bei einer und einem Prozent bei zehn Kanban-Karten, fällt also mit steigender Anzahl von Karten. Der relative Fehler in A-MM sinkt ebenfalls mit steigender Kartenzahl, liegt mit nur einer Karte bei drei Prozent und fällt ab drei Karten auf unter ein Prozent. Die Anzahl der Aufträge im System, oder *WIP* (Work In Process) steigt, wie auch die Durchlaufzeit, linear mit wachsender Kartenzahl (siehe Abbildung 13). Die relativen Fehler des WIP und der Durchlaufzeit liegen bei A-MM zwischen unter einem und zwei Prozent. Dabei ist die Abweichung zwischen Analyse und Simulation unabhängig von der Anzahl der Kanban-

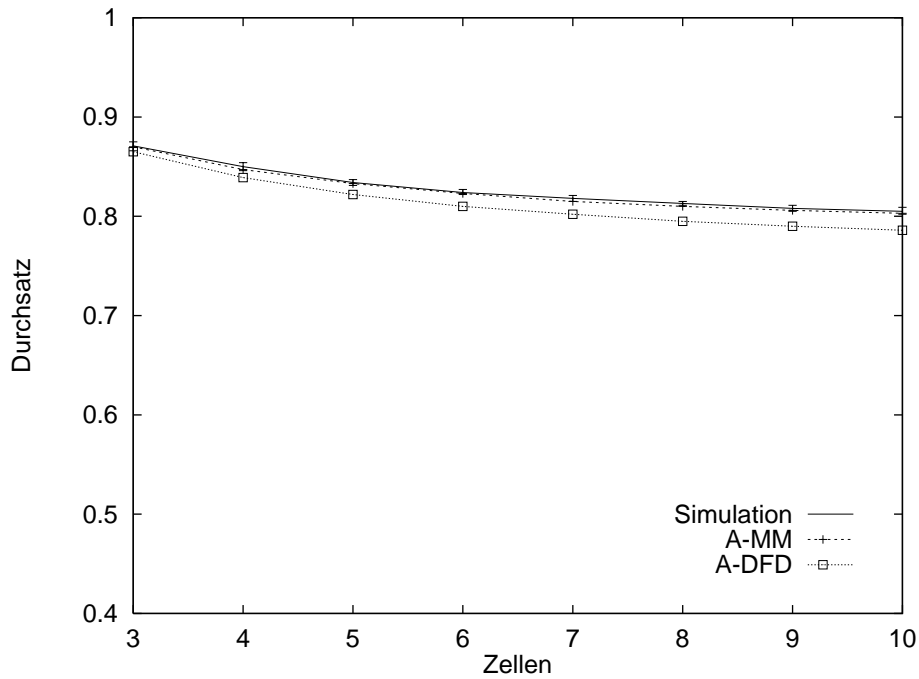


Abbildung 10: *Durchsatz bei Variation der Anzahl der Zellen im einfachen Kanban-System*

Karten. Bei A-DFD bewegt sich der relative Fehler des WIP zwischen einem und fünf Prozent, der der Durchlaufzeit zwischen unter einem und vier Prozent. Der relative Fehler ist auch hier für beide Größen weniger stark von der Kartenzahl abhängig als der des Durchsatzes.

Eine Erhöhung der Anzahl der Zellen führt zu einer Verringerung des Durchsatzes. Abbildung 10 stellt diesen Zusammenhang für ein Kanban-System mit fünf Karten in jeder Zelle dar und stimmt darin mit den Ergebnissen in [4] und [9] überein. Wie ebenfalls aus der Abbildung zu erkennen ist, liefert A-MM Werte für den Durchsatz, die fast mit denen der Simulation zusammenfallen. Die relativen Fehler sind kleiner als ein Prozent. Die Werte von A-DFD werden mit steigender Zellenzahl ungenauer, ihr relativer Fehler wächst von unter einem Prozent bei drei Zellen auf drei Prozent bei zehn Zellen an. WIP und Durchlaufzeit steigen wiederum linear mit wachsender Größe des Kanban-Systems (vgl. Abbildung 14). Bei beiden Analysemethoden ist sowohl in Bezug auf die Anzahl von Aufträgen im System, als auch auf die Durchlaufzeit eine mit der Zellenzahl zunehmende Abweichung zwischen Simulation und Analyse bemerkbar. Die relativen Fehler des WIP nach A-MM liegen zwischen unter einem und drei Prozent, nach A-DFD zwischen unter einem und fünf Prozent. Die relativen Fehler der Durchlaufzeit entwickeln sich bei A-MM und A-DFD von unter einem auf zwei Prozent.

Die Leistungsgrößen eines einfachen Kanban-Systems werden von den vorgestellten Analyseverfahren gut approximiert. Die Analyse nach Mitra und Mitrani liefert aber durchwegs genauer mit der Simulation übereinstimmende Ergebnisse, was mit dem Umstand begründet werden kann, daß in diesem Verfahren die einzelnen Zellen exakt und nur die Interaktion der Zellen approximativ untersucht wird. DiMascolo, Frein und Dallery analysieren sowohl die Sektoren, als auch das gesamte System approximativ.

Sektor Kanban-System

Das betrachtete Sektor Kanban-System besteht aus drei Zellen mit dem in Abbildung 11 gezeigten Netz in Zelle 2. In Zelle 1 und 3 befindet sich jeweils eine Maschine mit exponentiell verteilten Bedienzeiten und Mittelwert eins. Zur Durchführung dieser Untersuchung ist eine Erweiterung des Verfahrens von Mitra und Mitrani auf die Analyse von Kanban-Systemen mit mehreren Bedieneinheiten in den Zellen notwendig. Dies kann durch die Verwendung von *Flow-Equivalent-Server* geschehen.

Mit der Steigerung der Anzahl der Kanban-Karten steigt auch der Durchsatz an. Die Durchsatzerhöhung ist wiederum umso größer je weniger Karten vorhanden sind. A-MM und A-DFD verhalten sich, wie aus Abbildung 12 ersichtlich ist, entsprechend. A-MM weicht aber mit zunehmender Kartenzahl stärker von den Simulationsergebnissen ab. Der relative Fehler beträgt bei einer Karte sieben Prozent, verringert sich jedoch auf gleichbleibende vier Prozent. Die Ergebnisse von A-DFD sind mit einem relativen Fehler behaftet, der mit zunehmender Kartenzahl von neun auf unter ein Prozent fällt. WIP und Durchlaufzeit nehmen in Abhängigkeit von der Anzahl der Kanban-Karten linear zu (vgl. Abbildung 15). Der relative Fehler des approximativ berechneten WIP liegt bei A-MM zwischen einem und drei Prozent, bei A-DFD im Bereich von unter einem bis drei Prozent. Bei den Werten des Durchsatzes ergibt sich für den relativen Fehler ein Intervall von zwei bis sechs Prozent für A-MM und von unter einem bis sieben Prozent für A-DFD. Die relativen Fehler werden jeweils mit zunehmender Kartenzahl geringer.

Im Fall von Sektor Kanban-Systemen liegen die Resultate von A-DFD deutlich näher an den simulierten Werten, als die mit A-MM erzeugten. A-MM analysiert zuerst isoliert das Netz der Bedienstationen, berechnet die lastabhängigen Bedienraten des Flow Equivalent Server und untersucht anschließend das Kanban-System. Dadurch wird eine ungenauere Approximation der Kanban-Zelle mit mehreren Bedienstationen erreicht, als bei A-DFD die in jedem Schritt die Subsysteme mit der Methode von Marie untersucht.

Ergebnis

Sowohl die Ergebnisse, die mit den Analyseverfahren nach Mitra und Mitrani gewonnen werden, als auch jene nach DiMascolo, Frein und Dallery stimmen für die beiden betrachteten Varianten gut mit den simulativ gewonnenen Werten überein. Die maximalen relativen Fehler betragen bei A-MM sieben Prozent, bei A-DFD neun Prozent. Dabei werden Durchsatz und WIP von beiden Verfahren unterschätzt. Die approximativen Verfahren errechnen die Durchlaufzeiten nach dem Gesetz von Little aus dem Durchsatz und der Anzahl von Aufträgen im System. Daher können diese Größen je nach Qualität der

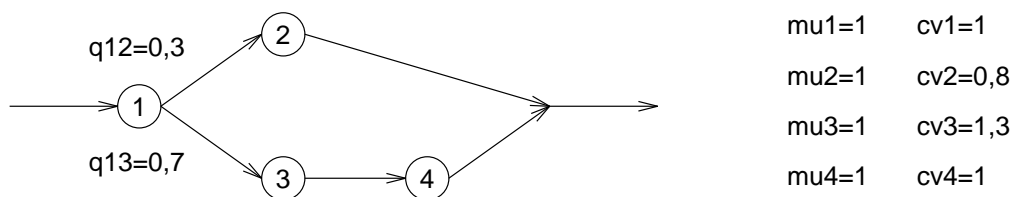


Abbildung 11: Netz in Zelle 2

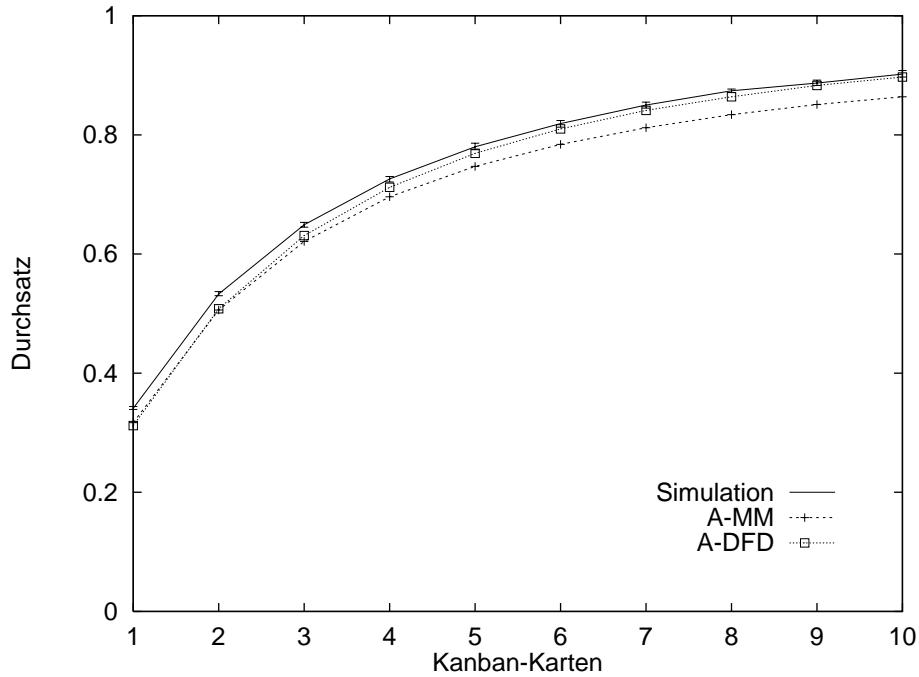


Abbildung 12: *Durchsatz bei Variation der Anzahl der Kanban-Karten im Sektor Kanban-System*

beiden anderen Ergebnisse über-, oder unterschätzt werden.

Allgemein liegen die mit A-MM erzeugten Ergebnisse in den Fällen näher an den Simulationsergebnissen, in denen die isolierte Zelle exakt untersucht werden kann und auch die externen Ankunftsrate nicht, wie bei Sektor Kanban-Systemen, aufgrund der Verwendung der Flow-Equivalent-Server auf ungenaue mittlere Bedienraten beruhen. In solchen komplexeren Kanban-Systemen ist A-DFD besser zur Untersuchung geeignet.

4.2 Komplexität

Bei dem vorgestellten Iterationsverfahren von Mitra und Mitrani werden die Zustandswahrscheinlichkeiten direkt durch Lösung des Systems der Gleichgewichtsgleichungen bestimmt. Dazu muß in jedem Iterationsschritt für jede Zelle das lineare Gleichungssystem

$$\vec{p} Q = \vec{0}$$

mit dem Wahrscheinlichkeitsvektor \vec{p} und der Generatormatrix Q mit Hilfe eines angemessenen numerischen Verfahrens berechnet werden. Diese Berechnung trägt entscheidend zur Komplexität der Analysemethode bei und ist selbst wiederum in ihrem Aufwand hauptsächlich von der Größe der Generatormatrix, die dem Quadrat der Zustände der isolierten Zelle des jeweiligen Systems entspricht, abhängig. Die Anzahl der Zustände wächst schnell mit der Größe der betrachteten Systeme und führt zu umfangreichen Gleichungssystemen, deren Speicherung und Berechnung bei sehr großen Systemen erheblichen Aufwand verursacht (Matrizen mit mehreren hunderttausend Einträgen). Desweiteren kann auch,

je nach Wahl der Parameter, eine Vielzahl (mehrere hundert) von Iterationen bis zum Erreichen der gewünschten Konvergenzgenauigkeit notwendig werden.

Der hauptsächlichliche Aufwand des iterativen Analyseverfahrens von DiMascolo, Frein und Dallery liegt in der wiederholten Analyse der Subsysteme Y_k . Jeder Iterationsschritt benötigt zur Berechnung der upstream und der downstream Parameter eines Kanban-Systems mit N Zellen $2(N - 1)$ solcher Analysen der Subsysteme. Die Subsysteme werden mit der ebenfalls iterativen Methode von Marie untersucht. Der Algorithmus konvergiert unabhängig von der Wahl der Parameter sehr schnell, meist in deutlich weniger als zehn Schritten.

Bei einem direkten Vergleich der Laufzeiten der beiden Algorithmen auf dem gleichen Rechner kann festgestellt werden, daß die Analyse nach DiMascolo, Frein und Dallery Werte im Sekunden-, bzw. Minutenbereich liefert, wohingegen die Analyse nach Mitra und Mitrani teilweise Laufzeiten von mehreren Stunden aufweist.

5 Zusammenfassung und Ausblick

In dieser Arbeit verglichen wir zwei analytische Methoden zur Leistungsbewertung von Kanban Systemen. Das Verfahren von Mitra und Mitrani beruht auf der direkten stochastischen Analyse des Zustandsraumes einer Kanban-Zelle. Die Arbeitsgruppe DiMascolo, Frein und Dallery dagegen benutzt ein Verfahren zur Analyse von Warteschlangennetze ohne Produktformeigenschaft, welches an die Aufgabenstellung angepaßt wird. Der Vergleich beider Methoden zeigt für große Parameterbereiche eine gute Übereinstimmung mit simulativ ermittelten Ergebnissen. Die Vorgehensweise von Mitra und Mitrani ist von ihrer Komplexität her allerdings schwerer zu handhaben als das alternative Verfahren. Für größere Systeme ist deshalb der Ansatz von DiMascolo, Frein und Dallery vorzuziehen.

Beide Methoden sind unseres Erachtens gut geeignet für Erweiterungen. Beispielsweise kann daran gedacht werden, statt einer einfachen Bedieneinheit in einer Kanban-Zelle eine Gruppenbedieneinheit mit variabler Startschranke zu verwenden. Gruppenbedieneinheiten sind in Halbleiterfertigungsumgebungen gang und gäbe. Des weiteren kann der Fall von mehreren Produkten berücksichtigt werden. Hier ist eine andere Analysemethode für Warteschlangennetze anzuwenden, die dann wiederum analog zur Methode von Marie zu erweitern ist.

6 Anhang: Vergleich der Ergebnisse

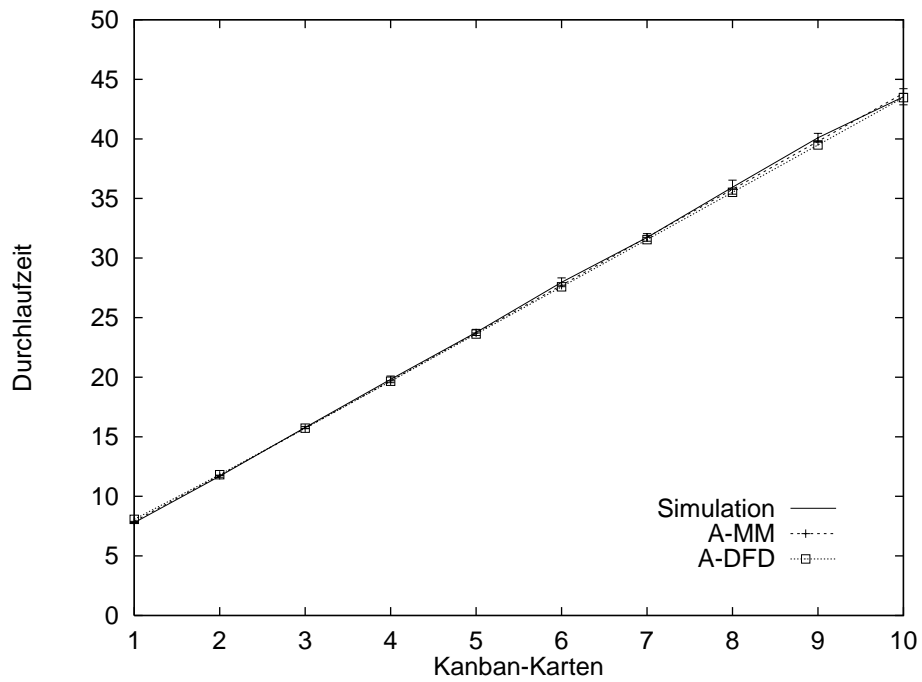
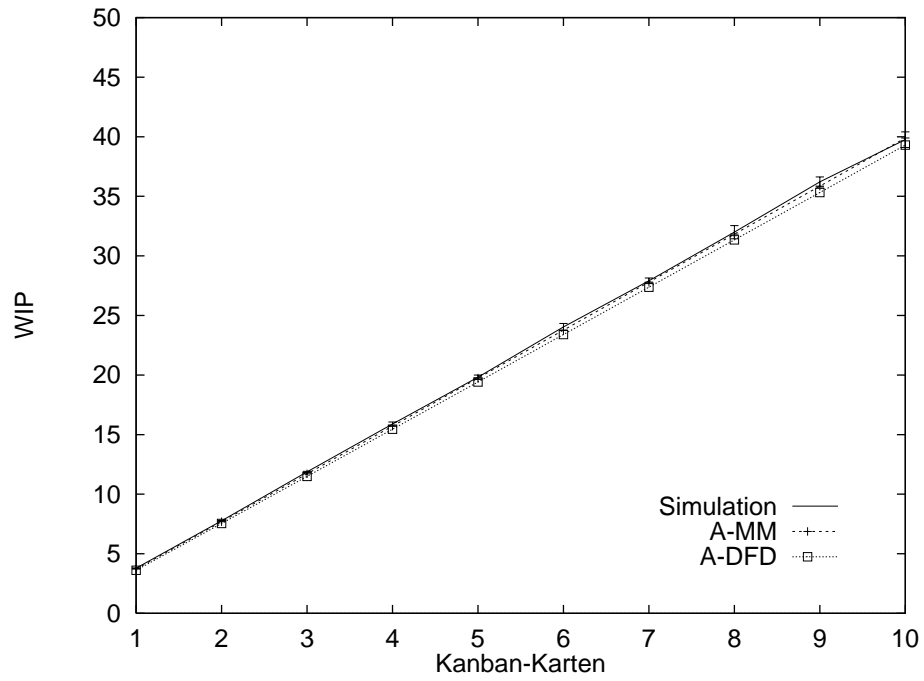


Abbildung 13: *Lineares Ansteigen von WIP und Durchlaufzeit bei zunehmender Anzahl der Kanban-Karten im einfachen Kanban-System*

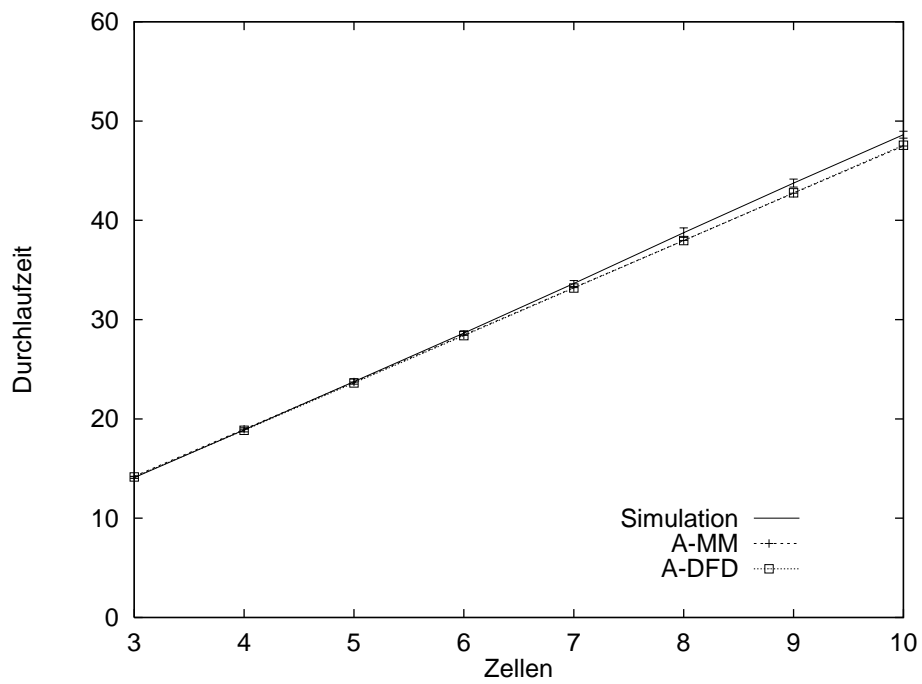
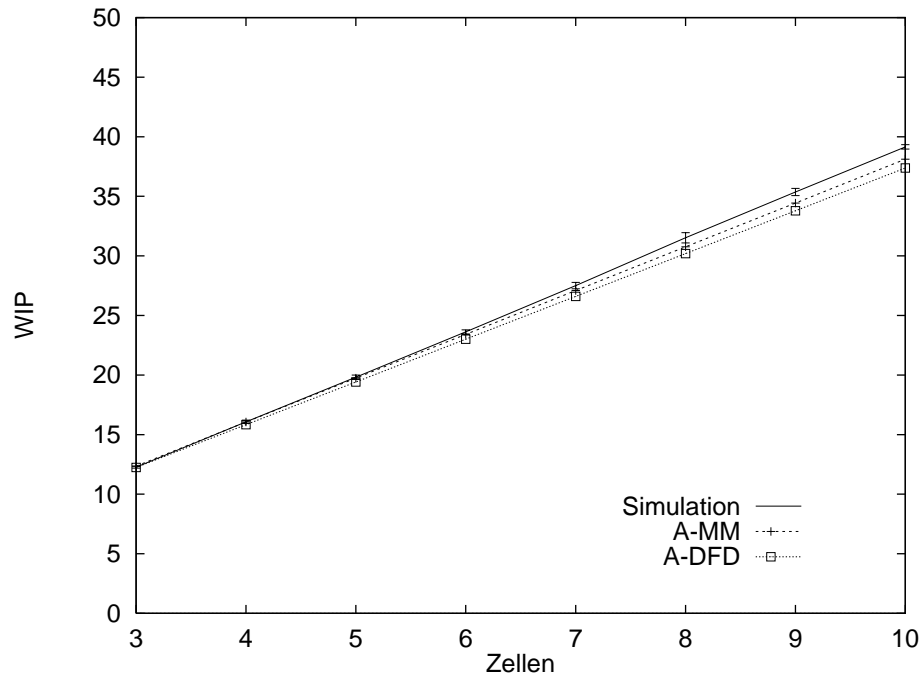


Abbildung 14: *Wachsende Werte für WIP und Durchlaufzeit bei größerer Anzahl von Zellen im einfachen Kanban-System*

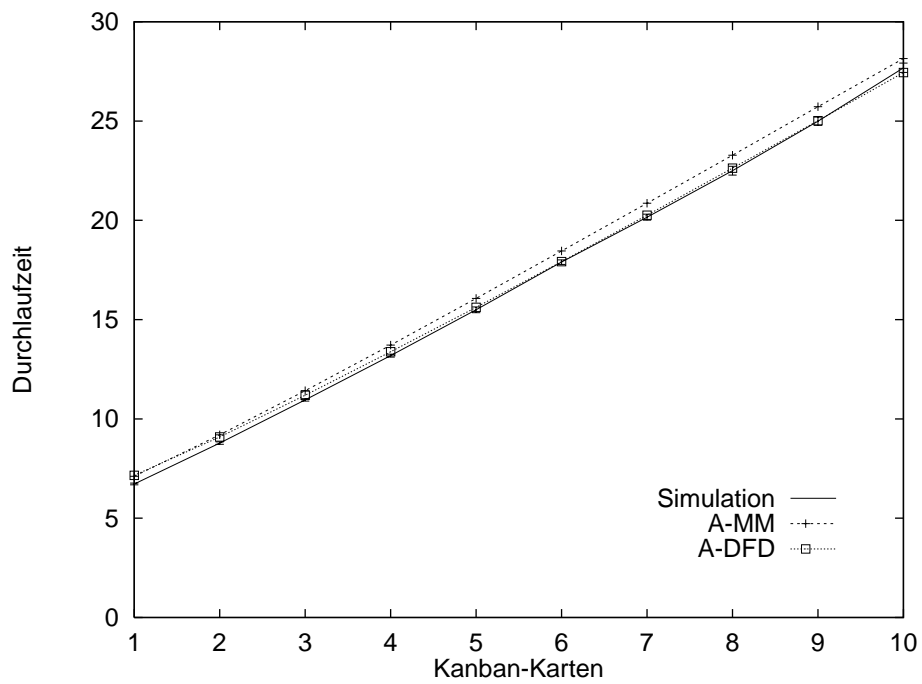
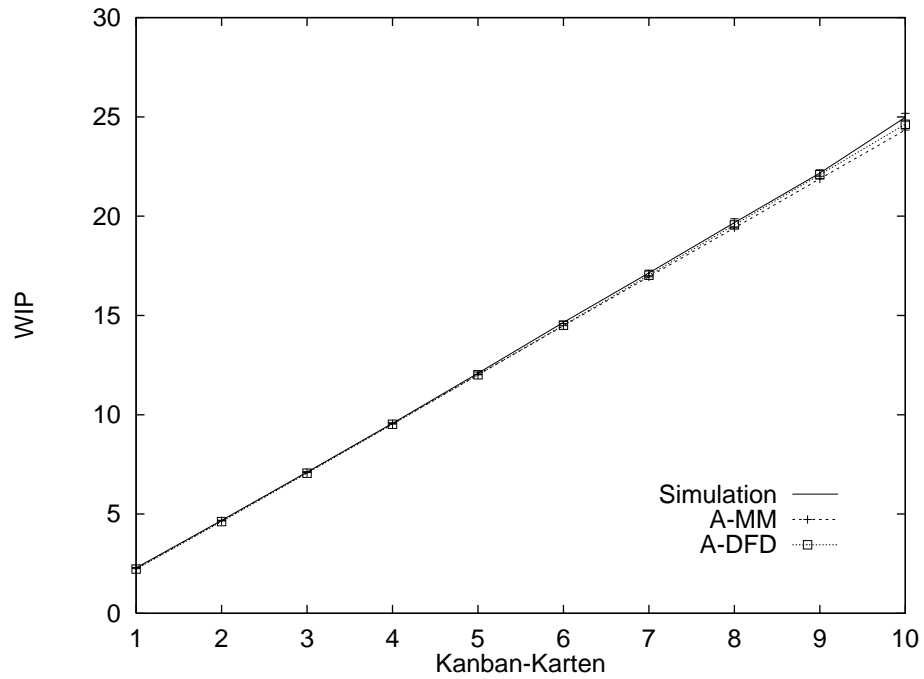


Abbildung 15: Ebenfalls lineares Ansteigen von WIP und Durchlaufzeit bei Variation der Anzahl der Kanban-Karten im Sektor Kanban-System

Literatur

- [1] Berkley, Blair J.: *A review of the kanban production control research literature*. Production and Operations Management, 1(4):393–411, Fall 1992.
- [2] Buzen, Jeffrey P.: *Queueing Network Models of Multiprogramming*. , Div. of Engineering and Applied Physics, Harvard University, 1971.
- [3] Dallery, Yves: *Approximate analysis of general open queueing networks*. Performance Evaluation, 11:209–222, 1990.
- [4] DiMascolo, Maria, Frein, Yannick, Dallery, Yves: *Queueing network modeling and analysis kanban systems*. *Third International Conference On Computer Intergrated Manufacturing*. IEEE / Rensselaer Polytechnic Institute, IEEE Computer Society Press, May 20–22 1992.
- [5] Esparrago Jr., Romeo A.: *Kanban*. Production and Inventory Management Journal, 6–10, First Quarter 1988.
- [6] Hernandez, Arnaldo: *Just-In-Time Manufacturing. A practical approach*. Prentice Hall, Englewood Cliffs, NJ, 1989.
- [7] Marie, Raymond A.: *An approximate analytical method for general queueing networks*. IEEE Transactions on Software Engineering, SE-5(5):530–538, September 1979.
- [8] Marie, Raymond A.: *Calculating equilibrium probabilities for $\lambda(n)/c_k/1/n$ queues*. ACM Sigmetrics Performance Evaluation Review, 9(2):117–125, 1980.
- [9] Mitra, Debasis Mitrani, Isi: *Analysis of a kanban discipline for cell coordination in production lines*. Management Science, 36(12):1548–1566, December 1990.
- [10] Puppe, Frank und Hestermann, Christian: *Skriptum zur Vorlesung Produktionsplanungs- und -steuerungssysteme*. Lehrstuhl für Informatik VI, Universität Würzburg, 1993/94.