

DISKUSSIONSARBEITEN
DER FAKULTÄT FÜR WIRTSCHAFTSWISSENSCHAFTEN
DER UNIVERSITÄT BIELEFELD

**Simulation von Reentrant Lines mit ARENA:
Ergebnisse eines Projektes zur Betriebsinformatik**

Martin Feldmann und Stephanie Müller

Auf Basis des Abschlussberichts der Gruppe II:
Jan Herrmann / Valeska Kathöfer / Andre Peterwerth / Wilke Pfirmann

mfeldmann@wiwi.uni-bielefeld.de

smueller@wiwi.uni-bielefeld.de

Fakultät für Wirtschaftswissenschaften
Lehrstuhl für Betriebswirtschaftslehre und Unternehmensforschung
Universität Bielefeld
Postfach 10 01 31
33501 Bielefeld

Discussion Paper No. 511

Januar 2004

Gliederung

1. Einleitung.....	1
1.1 Motivation.....	1
1.2 Projekt zur Betriebsinformatik.....	3
2. Modell Fab 1.24	5
2.1 Darstellung des Modells.....	5
2.2 Implementierung in ARENA	11
2.3 Modellierung von Störungen in ARENA.....	21
2.4 Auswertung und Vergleich der Ergebnisse.....	23
3. Mehrprodukt-Reentrant-Line: System I.....	26
3.1 Darstellung des Modells.....	26
3.2 Implementierung in ARENA	29
3.3 Erweiterung I „Batching“	32
3.3.1 Berücksichtigung einer Batch-Station.....	32
3.3.2 Auswertung und Vergleich der Ergebnisse.....	35
3.4 Erweiterung II „kapazitierte Lager“	39
3.4.1 Begrenzte Lagerkapazität an einer Station.....	39
3.4.1.1 Implementierung	39
3.4.1.2 Variation der Lagerkapazität und deren Auswirkungen auf die Mittlere Durchlaufzeit	47
3.4.2 Begrenzte Lagerkapazität an zwei Stationen	53
4. Fazit.....	65

1. Einleitung

1.1 Motivation

Die Organisation einer Fertigung kann sich generell an der Verrichtung oder an dem Objekt orientieren und entsprechend als Werkstattfertigung oder als Fließfertigung vorgenommen werden (KISTNER/STEVEN, 2001). Sind große Stückzahlen eines weitgehend homogenen Produktes zu bearbeiten, bietet es sich an, eine Fließfertigung in Betracht zu ziehen. Für eine wiederholte, funktionsgleiche Bearbeitung sind gegebenenfalls identische Maschinen mehrfach zu installieren, wenn entsprechend den klassischen Arbeiten zum Flow Shop Scheduling vorausgesetzt wird, dass jeder Auftrag nur einmal auf jeder Maschine einzulasten ist (BAKER, 1974). Ein Wiedereintritt in die Fertigungslinie wird damit ausgeschlossen und die Konkurrenz eines Produktes in verschiedenen Bearbeitungsstadien um eine Maschine vermieden. Erfordert der Produktionsprozess allerdings, dass identische Bearbeitungsschritte häufig durchlaufen werden, kann es – insbesondere bei kostenintensiven Betriebsmitteln – sinnvoll sein, den Wiedereintritt zuzulassen. Muss zudem der Produktionsprozess unter Reinraumbedingungen ablaufen, kann die Montage weiterer Maschinen zu einer ökonomisch nicht mehr zu rechtfertigenden Raumgröße führen.

In der Praxis sind diese Bedingungen bei der Fertigung von Halbleitern, hoch integrierten Schaltkreisen oder optischen Präzisionsinstrumenten regelmäßig gegeben, so dass dort mit dem Problem der Planung und Steuerung von Wiedereintrittslinien, den sogenannten Reentrant Lines umzugehen ist. Da in den genannten Fällen typischer Weise sehr hohe Investitionen erforderlich werden und außerdem die erzeugten, hochwertigen Produkte entsprechende Absatzpreise ermöglichen, kommt einer optimierten Maschinenauslastung eine hohe Bedeutung zu.

Der Begriff Reentrant Line wurde durch KUMAR (1993) eingeführt, um eine Struktur zu beschreiben, die im Rahmen der Fertigung von Halbleiter-Scheiben („wafer“) auftritt. Um solche Strukturen für Fragen der Produktionsplanung und -steuerung abzubilden, bieten sich prinzipiell offene Warteschlangennetzwerke an (CHEN/YAO 2001), die sich dadurch auszeichnen, dass Stationen mehrfach in der Stationsfolge der Produkte auftreten. Entsprechend konkurriert ein Produkt mit unterschiedlichem Fertigstellungsgrad in mehreren Pufferlagern vor jeder Station mit sich selbst um die weitere Bearbeitung. Die Abbildung 1.1 (vgl. NARAHARI/KHAN, 1998, S. 587) verdeutlicht diesen Zusammenhang für eine

Reentrant Line mit drei Stationen. Die Stationenfolge des Produktes ist: 1, 2, 2, 2, 3, 3, 2, 1, 1, 2, 3. Entsprechend werden elf Pufferlager eingerichtet, die in der Abbildung mit b_{ij} gekennzeichnet sind. Der Index i gibt die Station an, auf deren Bearbeitung die Produkte im Pufferlager b_{ij} warten. Der Index j gibt an, zum wievielten Mal das Produkt die Station i durchläuft.

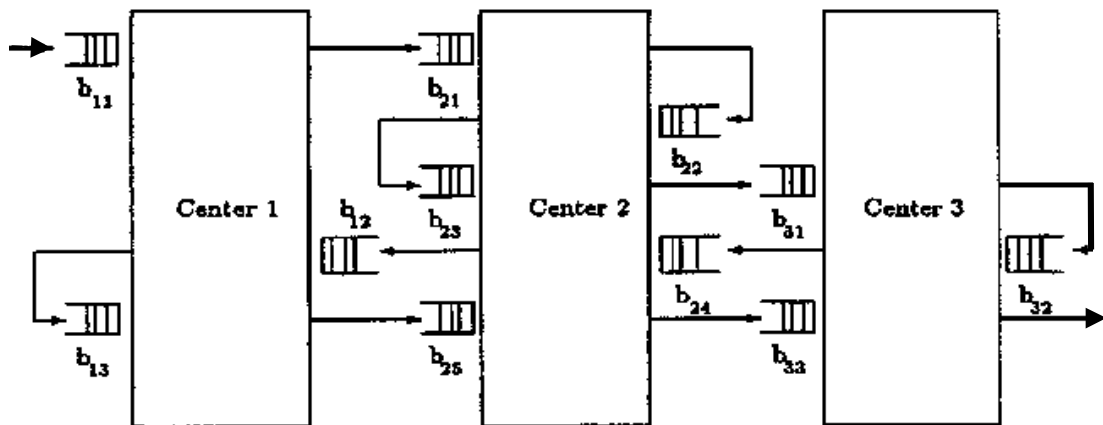


Abb. 1.1: Reentrant Line mit 3 Stationen und 11 Lagern

In der vorliegenden Arbeit sollen Reentrant Lines mit deterministischer Maschinenfolge zunächst für den Einprodukt-Fall untersucht werden. Folgt man den mittlerweile publizierten Ergebnissen (KUMAR/MEYN, 1995; KUMAR/KUMAR, 2001; HSIEH et al., 2001), entziehen sich Reentrant Lines i.d.R. dem analytischen Zugang. Insbesondere erschwert die mit dem Fertigstellungsgrad variierende Dauer der Bearbeitung auf der gleichen Station eine Modellierung.

Um in derartigen Strukturen dennoch Aussagen über die Dimensionierung und die anschließende Steuerung einer Linie treffen zu können, werden in der Literatur Simulationsstudien vorgeschlagen (WEIN, 1988; DAI et al., 1997; HSIEH et al., 2001; PARK et al. 2002). Gesucht werden beispielsweise:

- ◆ Einlastungspolitiken für die Aufträge in das System.
- ◆ Lagerabgangspolitiken für die vor den Stationen wartenden Aufträge.
- ◆ Empfehlungen für die Dimensionierung der maschinellen Kapazität bzw. der Größe der Pufferlager.
- ◆ Möglichkeiten zur Berücksichtigung von Rüstzeiten, z.B. durch Gruppierung von Produkten (Batching).
- ◆ Berücksichtigung des Mehr-Produktfalls.

1.2 Projekt zur Betriebsinformatik

Im Rahmen des Diplomstudienganges Betriebswirtschaftslehre der Fakultät für Wirtschaftswissenschaften an der Universität Bielefeld ist für alle Studierenden des Hauptstudiums ein *Projekt zur Betriebsinformatik* obligatorisch. Nach erfolgreicher Teilnahme an den Vorlesungen *Grundlagen der Betriebsinformatik* und mindestens einer der drei folgenden Vertiefungsveranstaltungen *Optimierungsmethoden*, *Lagerhaltungstheorie* oder *Netzplantechnik* können Studierende an einem *Projekt zur Betriebsinformatik* (mit 3 SWS) am Lehrstuhl für Allgemeine Betriebswirtschaftslehre, insbesondere Unternehmensforschung (Prof. Dr. K.-P. Kistner) teilnehmen.

Im Sommersemester 2003 fand unter unserer Leitung das Betriebsinformatik-Projekt *Planung und Steuerung von Reentrant Lines* mit 12 Teilnehmer/innen (eingeteilt in drei Gruppen) statt. Von der Vorbesprechung mit Bekanntgabe des Themas im Februar 2003 bis zur Abgabe des Projektberichts im Oktober 2003 wurden folgende Inhalte behandelt:

- ◆ Der Umgang mit der Simulations-Software ARENA 5.0 in der Professional Edition unter Windows 2000,
- ◆ Grundlagen der Reentrant Line,
- ◆ Fragestellungen der statistischen Auswertung einer Simulation resp. des Experimental Design.

Zur Mitte des Semesters hatte jede der drei Gruppen ein Pflichtenheft vorzulegen, in dem die bis Oktober 2003 zu untersuchende Fragestellung konkretisiert wurde.

- ◆ Die erste Gruppe befasste sich mit dem Einfluss diverser Einlastungspolitiken und den Auswirkungen einer zwischenzeitlichen Qualitätskontrolle auf die Performance der Linie.
- ◆ Die zweite Gruppe untersuchte, inwiefern sich kapazitierte Lager und Batching in Mehrprodukt-Reentrant Lines auswirken.
- ◆ Als Untersuchungsgegenstand wählte die dritte Gruppe den Vergleich der Auswirkungen von verschiedenen Lagerabgangspolitiken auf die Performance der Linie.

Die folgenden Ausführungen basieren auf dem Projektabschlussbericht der zweiten Gruppe und fassen diesen mit dem Ziel zusammen, die Vorgehensweise bei der Simulation einer Reentrant Line mit ARENA exemplarisch vorzustellen. Die erarbeiteten Ergebnisse sollen einem breiteren Publikum zugänglich gemacht werden und Studierenden in späteren Projekten als Muster dienen. Den Teilnehmern dieser Gruppe:

Jan Herrmann

Valeska Kathöfer

Andre Peterwerth

Wilke Pfirrmann

möchten wir an dieser Stelle ganz herzlich für die sehr engagierte Mitarbeit im Projekt und die freundliche Unterstützung bei der Herausgabe dieser Zusammenfassung danken.

Beim Ausdrucken dieser pdf-Datei ist zu beachten, dass das Seitenformat des Anhangs Papier der Größe DIN A3 erfordert.

Bielefeld, im Januar 2004

Dr. M. Feldmann

Dipl.-Kffr. S. Müller

2. Modell Fab 1.24

2.1 Darstellung des Modells

In diesem Kapitel wird auf die Arbeit von WEIN (1988) eingegangen, der sich mit der Produktionsstruktur der Halbleiterfertigung bei Hewlett-Packard befasst. Die von WEIN (1988) untersuchte Produktionsstruktur mit 24 Stationen für den Ein-Produkt-Fall wird als Fab 1.24 bezeichnet. Die Zielsetzung besteht darin, zu zeigen, wie diese Struktur in ARENA simuliert werden kann. Der Vergleich der Simulationsergebnisse mit den von WEIN (1988) präsentierten Werten dient der Validierung der Vorgehensweise und schließt dieses Kapitel ab.

Im Folgenden werden die wesentlichen Modelldaten zur Implementierung des Fab 1.24 vorgestellt. Die Daten der von WEIN (1988) veröffentlichten Problemstellung sind zunächst zu interpretieren und derart anzupassen, dass sie mit den in ARENA vorgegebenen Datenstrukturen vergleichbar und für die Implementierung verwendbar aufbereitet werden. Es ist sicherzustellen, dass die verwendeten Daten in der ARENA-Simulation im gleichen Kontext wie bei WEIN (1988) genutzt werden. An den entsprechenden Stellen wird hierauf noch genauer Bezug genommen.

Die Fertigung der Halbleiter durchläuft 172 Fertigungsschritte auf 24 Stationen. Um die Struktur dieser Reentrant-Line zu verdeutlichen, wird zunächst die folgende Abbildung von KUMAR/KUMAR (2001, S. 4) verwendet (Abb. 2.1).

Sie zeigt eine vergleichsweise kleine Reentrant-Line mit 12 Stationen auf denen 60 Bearbeitungsschritte ausgeführt werden. Da eine graphische Darstellung der Produktionsstruktur mit 172 Fertigungsschritten nicht sinnvoll ist, wird in Anlehnung an die Darstellung bei WEIN (1988) die Stationenfolge angegeben. Sie gibt die Reihenfolge der Bearbeitungen wieder, die der zu fertigende Halbleiter auf den Stationen erfährt (Abb. 2.2).

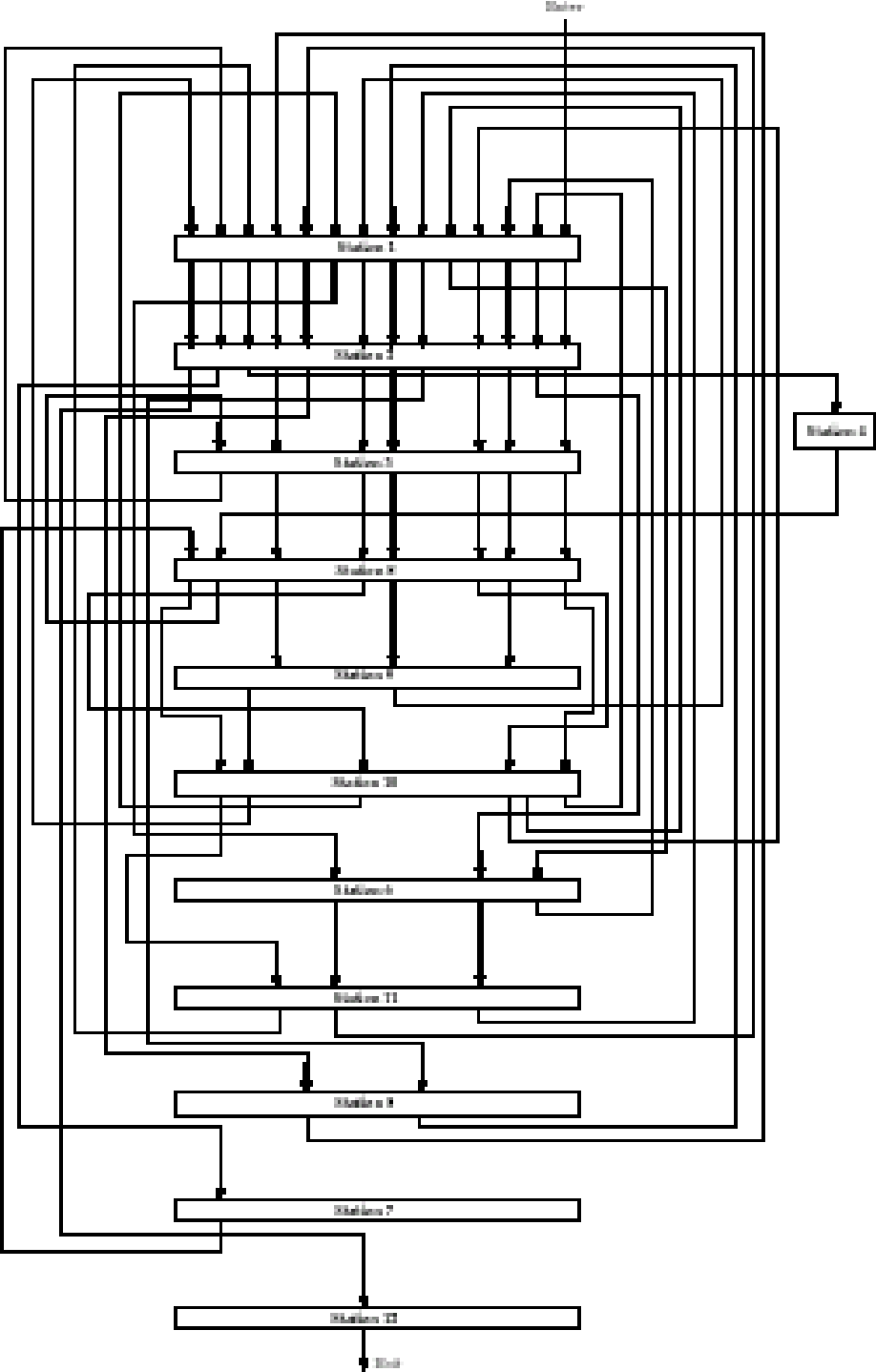


Abb. 2.1: Halbleiterfertigung mit 12 Stationen nach KUMAR/KUMAR (2001)

ENTER – 1 – 2 – 13 – 14 – 23 – 15 – 20 – 22 – 23 – 22 – 17 – 13 –
 14 – 15 – 23 – 16 – 24 – 23 – 22 – 17 – 1 – 8 – 4 – 22 – 22 – 1 – 2 –
 8 – 13 – 14 – 18 – 23 – 15 – 16 – 23 – 18 – 22 – 1 – 1 – 13 – 14 –
 23 – 15 – 16 – 24 – 23 – 22 – 17 – 1 – 2 – 8 – 9 – 21 – 22 – 1 – 4 –
 22 – 22 – 1 – 2 – 13 – 14 – 23 – 15 – 16 – 24 – 24 – 23 – 22 – 17 –
 24 – 1 – 2 – 7 – 1 – 3 – 22 – 13 – 15 – 23 – 22 – 22 – 22 – 17 – 13 –
 14 – 18 – 23 – 15 – 16 – 20 – 23 – 1 – 17 – 1 – 1 – 3 – 13 – 14 – 16 –
 24 – 23 – 22 – 17 – 9 – 21 – 1 – 3 – 13 – 14 – 15 – 23 – 15 – 16 –
 24 – 23 – 22 – 17 – 1 – 3 – 13 – 14 – 23 – 15 – 16 – 23 – 15 – 16 –
 24 – 23 – 22 – 17 – 1 – 3 – 10 – 22 – 12 – 6 – 22 – 6 – 1 – 1 – 4 –
 10 – 19 – 23 – 1 – 10 – 13 – 14 – 16 – 21 – 12 – 13 – 14 – 18 – 23 –
 15 – 15 – 15 – 16 – 19 – 23 – 22 – 17 – 11 – 13 – 14 – 15 – 21 – 23 –
 5 – EXIT

Abb. 2.2: Stationenfolge im Fab 1.24

Hinsichtlich der maschinellen Ausstattung der Stationen betrachtet WEIN (1988) drei verschiedene Varianten. Diese unterscheiden sich dadurch, dass an manchen Stationen die Anzahl der eingesetzten Maschinen variiert wird. So hat z.B. das hier untersuchte Fab 1.24 zwei Maschinen an der dritten Station, statt einer Maschine wie das Fab 3.24 (vgl. WEIN 1988, S. 117).

WEIN (1988) hat diese Modelle in SIMAN programmiert und mit ihnen die Auswirkungen verschiedener Einlastungspolitiken (Release-Policies) in Simulationsstudien untersucht. Als Performance-Kennzahl betrachtet er vorrangig die Mittlere Durchlaufzeit (Mean Cycle Time). Sie ergibt sich als die gemittelte Zeitspanne, die ein Stück von der Einlastung in das System bis zu seinem Austritt aus dem Modell benötigt. Der von WEIN (1988) simulierte Zeitraum umfasst jeweils 3,5 Jahre, wobei mit 20 Replikationen und ohne eine Einschwingphase gearbeitet wird. Da die Zielsetzung in der Validierung des Vorgehens liegt, genügt es, nur eine der von WEIN (1988) verwendeten Einlastungspolitiken zu berücksichtigen. Einige Politiken sind in ARENA bereits vorprogrammiert verfügbar. Zu diesen zählt die Poisson-Verteilung, die

hier als Einlastungspolitik Verwendung findet. WEIN (1988) geht von einer Ankunftsrate von 0,0236 Losen pro Stunde aus.

Weiter hat WEIN (1988) mit unterschiedlichen Lagerabgangspolitiken gearbeitet. Hier wird auf die FIFO-Regel zurückgegriffen, die ebenfalls in ARENA als Standardfunktion vorhanden ist.

Die einzelnen Bearbeitungs- und Wartezeiten auf den Stationen sind stochastisch und unterliegen einer Gammaverteilung. Gemäß ihrer Dichtefunktion (vgl. LU et al. 1994, S. 379; SCHLITGEN 2000, S. 225f.) können die unterschiedlichen Erwartungswerte und Varianzen der Inputparameter berechnet werden. Die Dichtefunktion der Gammaverteilung lautet:

$$f(x) = \frac{1}{(\alpha-1)!} \theta^\alpha x^{\alpha-1} e^{-\theta x} \quad \forall x > 0$$

Der Parameter α beträgt für die Mittlere Bearbeitungszeit (Mean Processing Time) 2,0. Für die Mittlere Reparaturzeit (Mean Time to Repair) und für die Mittlere Zeit zwischen zwei Störungen (Mean Time between Failures) beträgt $\alpha = 0,5$. Für das Fab 1.24 stehen die in Tabelle XX angegebenen Kennzahlen zur Verfügung, die durch die Simulation mit SIMAN und der Zusammenarbeit mit Hewlett Packard ermittelt wurden (vgl. WEIN 1988, S. 117, Table I):

Die Maschinenauslastung (Utilization) wird bei WEIN (1988) wie folgt berechnet:

$$\text{Maschinenauslastung} = \left[\left(\frac{\lambda * (NV / L) (MPT)}{\# \text{Machines}} \right) + \left(\frac{(MTTR)}{(MTBF + MTTR)} \right) \right] * 100\%$$

Machines : Anzahl an Maschinen

NV/L : Zahl der Bearbeitungen je Entität (Number of Visits per Lot)

MPT : Mittlere Bearbeitungszeit (Mean Processing Time)

MTBF : Mittlere Zeit zwischen zwei Störungen (Mean Time between Failures)

MTTR : Mittlere Reparaturzeit (Mean Time to Repair)

λ : Ankunftsrate

In ARENA ist diese Funktion nicht enthalten, so dass die Formel neu erstellt werden muss. In diesem Fall ist es möglich, die Maschinenauslastung über zwei in ARENA bereits implemen-

tierte Funktionen zu bestimmen. Die Maschinennutzung zuzüglich der relativen Ausfallzeiten ergibt den gesuchten Funktionswert. Die in Tabelle 2.1 zusammengefassten Werte sind für die SIMAN-Simulation verwendet worden und führen zu den in der Tabelle 2.2 gegebenen Performance-Kennzahlen (vgl. WEIN 1988, S. 121).

Tab. 2.1: Performance-Kennzahlen für Fab 1.24 nach WEIN (1988)

Stationen Nummer	Anzahl Maschinen	NV/L	MPT	MTBF	MTTR	Maschinen-auslastung
1	2	19	1,55	42,18	2,22	39,8
2	2	5	4,98	101,11	10,00	38,4
3	2	5	5,45	113,25	5,21	37,0
4	1	3	4,68	103,74	12,56	43,9
5	1	1	6,14	100,55	6,99	21,0
6	1	2	7,76	113,25	5,21	41,4
7	1	1	6,23	16,78	4,38	35,4
8	1	3	4,35	13,22	3,43	51,4
9	1	2	4,71	10,59	3,74	48,3
10	1	3	4,05	47,53	12,71	49,8
11	1	1	7,86	52,67	19,78	46,2
12	1	2	6,10	72,57	9,43	40,3
13	4	13	4,23	22,37	1,15	37,3
14	3	12	7,82	21,76	4,81	91,9
15	1	15	0,87	387,20	12,80	34,0
16	2	11	2,96	–	–	38,4
17	1	10	1,56	119,20	1,57	38,1
18	1	4	3,59	–	–	33,9
19	2	2	13,88	46,38	17,42	60,1
20	1	2	5,41	36,58	9,49	46,1
21	2	4	7,58	36,58	9,49	56,4
22	2	21	1,04	118,92	1,08	26,7
23	2	23	1,09	–	–	29,6
24	2	8	3,86	55,18	12,86	55,3

Tab. 2.2: Politiken und Performance-Kennzahlen für Fab 1.24 nach WEIN (1988)

Einlastungspolitik	Poisson
Lagerabgangspolitik	FIFO
Mittlere Durchlaufzeit (MCT)	1035
Standardabweichung der Durchlaufzeit (SDCT)	258
Mittlere Wartezeit (MQT)	486

Als erste Fragestellung dieser Arbeit ist zu prüfen, ob es mit der Implementierung des Modells in ARENA gelingt, näherungsweise dieselben Performance-Kennzahlen zu erzielen wie WEIN (1988, S. 121). Dabei werden die Mittlere Durchlaufzeit, die Mittlere Gesamtwartezeit und die Maschinenauslastung besonders hervorgehoben. Die Mittlere Gesamtwartezeit ergibt sich aus der über alle Einheiten gemittelten Summe der Wartezeiten, die jede Einheit beim Warten auf die Bearbeitung verbringt (vgl. WEIN 1988, S. 121).

2.2 Implementierung in ARENA

Zur Erstellung des Modells werden die in ARENA zur Verfügung stehenden Werkzeuge, insbesondere die in der folgenden Tabelle aufgeführten Module genutzt (vgl. KELTON et al. 2002). Die Übersicht stellt die Module mit ihrer Bezeichnung im Text dem in ARENA verwendeten Ausdruck gegenüber und gibt eine kurze Beschreibung der Funktion jedes Moduls wieder.

Tab. 2.3: Die Module, ihre Bezeichnung und ihre Funktion

Bezeichnung		Funktion
im Text	in ARENA	
Empfangs-Modul	Station	Das Empfangs-Modul stellt den Ankunftspunkt dar, zu dem Entitäten von einem Sende-Modul gelangen.
Entscheidungs-Modul	Decide	Das Entscheidungs-Modul erlaubt, Entscheidungsprozesse innerhalb des Systems abzubilden. Es beinhaltet die Möglichkeit, Entscheidungen in Abhängigkeit von Bedingungen oder Wahrscheinlichkeiten zu treffen. Bedingungen können an Attribute, Variablen-Typen, Entitätstypen oder Ausdrücke (z.B. Anzahl der Teile in einer Schlange) gebunden sein.
Greif-Modul	Remove	Das Greif-Modul entnimmt eine einzelne Entität (in der vorliegenden Arbeit die Entität an der Stelle J) aus einer spezifizierten Warteschlange und sendet dieses an ein bestimmtes Folgemodul.

Tab. 2.3: Fortsetzung

Bezeichnung		Funktion
im Text	in ARENA	
Gruppierungs-Modul	Batch	Das Gruppierungs-Modul fügt eine bestimmte Anzahl Entitäten zu einem Batch (Gruppierung) zusammen. Diese Gruppierungen können abschnittsweise oder permanent sein. Es besteht die Möglichkeit, nur Entitäten mit einem gleichen Attribut zusammenzufassen. Bei abschnittsweiser Gruppierung werden die Attribute bzw. Variablenwerte der einzelnen Entitäten gespeichert und diesen nach Auflösen des Batches wieder zugewiesen (vgl. Separierungs-Modul). Das Batch verlässt das Modul als eigenständige Entität.
Halte-Modul	Hold	Dieses Modul hält Entitäten in einer Warteschlange, bis ein bestimmtes Signal empfangen wird, ein bestimmter Systemzustand eintritt oder unbegrenzt lange. Im letzten Fall müssen die Entitäten später mit Hilfe eines Greif-Moduls entfernt werden.
Lese-/Schreibe-Modul	ReadWrite	Das Lese-/Schreibe-Modul liest Daten aus einer Datei oder der Tastatur ein und weist diese einem Attribut bzw. einer Variablen zu. Alternativ kann es auch Attribut- bzw. Variablenwerte auf den Monitor ausgeben oder in eine Datei schreiben.
Prozess-Modul	Process	Prozess-Module stellen die Hauptbearbeitungsmodule in einer Simulation in ARENA dar. Optionen zum Belegen und Freigeben von Ressourcen sind hier anzuwenden. Die Bearbeitungszeit wird der Entität zugeordnet und kann als wertschöpfend, nicht wertschöpfend, Transportzeit, Wartezeit oder Sonstige eingestellt werden.

Tab. 2.3: Fortsetzung

Bezeichnung		Funktion
im Text	in ARENA	
Quellen-Modul	Create	Dieses Modul stellt den Startpunkt für Entitäten im Simulationsmodell dar. Entitäten werden nach einem festen Plan erzeugt oder einem stochastischen Prozess folgend gemäß der Zwischenankunftszeit generiert. Der erste Ankunftszeitpunkt, die maximalen Ankünfte sowie die Zahl der Entitäten pro Ankunft werden im Quellen-Modul festgelegt.
Sende-Modul	Route	Das Sende-Modul transferiert Entitäten zu einer bestimmten Station (vgl. Empfangs-Modul) oder der nächsten Station innerhalb der Stationenfolge. Es ist an dieser Stelle möglich, eine Transferzeit zu definieren.
Senken-Modul	Dispose	Die Senke stellt den Endpunkt für Entitäten im Simulationsmodell dar. Bevor die Entitäten das Modell über dieses Modul verlassen, werden die Entitäts-Statistiken gespeichert.
Separierungs-Modul	Separate	Das Separierungs-Modul dient sowohl dem Duplizieren von Entitäten als auch dem Auflösen von Batches. Dabei werden die zuvor gruppierten Entitäten in derselben Reihenfolge und mit denselben Variablenwerten bzw. Attributen aus dem Modul entlassen, wie sie das Gruppierungs-Modul erreicht haben.
Such-Modul	Search	Das Such-Modul durchsucht bei Eintreffen einer Entität eine Schlange, um die Position der ersten gefundenen Entität, die die Suchbedingung(en) erfüllt, als globale Variable <i>J</i> an das System zurückzugeben. (vgl. Greif-Modul)

Tab. 2.3: Fortsetzung

Bezeichnung		Funktion
im Text	in ARENA	
Speicher-Modul	Record	Das Speicher-Modul sammelt Statistiken, so ist es u.a. möglich, Zwischenankunftszeiten oder Entität-Statistiken zu speichern. Weiterhin können benutzerdefinierte Ausdrücke angelegt werden, die bestimmte Werte berechnen und speichern.
Verzögerungs-Modul	Delay	Das Verzögerungs-Modul hält eine Entität für eine bestimmte Zeitspanne fest und entlässt sie danach wieder. Diese Zeitspanne kann wie im Prozess-Modul eingestellt werden.
Zuweisungs-Modul	Assign	Dieses Modul wird genutzt, um Variablen, Entitäts-Attributen, Entitäts-Typen, Entitäts-Bildern oder anderen System-Variablen neue oder geänderte Werte zuzuweisen. Es ist möglich, mehrere Zuweisungen mit einem einzigen Zuweisungs-Modul durchzuführen.

Im ersten Schritt wird ein Quellen-Modul (siehe Abb. 2.3) verwendet, durch das die Entitäten erzeugt und in das System eingelastet werden. Hierbei soll der von WEIN (1988) angegebene Poissonprozess mit der Ankunftsrate $\lambda = 0,0236$ pro Stunde als Einlastungspolitik dienen. Da in ARENA statt λ die erwartete Zeitdauer zwischen zwei aufeinander folgenden Ankünften (die Zwischenankunftszeit) $1/\lambda$ einzugeben ist, muss diese zunächst ermittelt werden (vgl. HILLIER/LIEBERMAN 1997, S. 508):

$$\frac{1}{0,0236} = 42,378836 \text{ [ZE]}$$

Die Abbildung 2.3 zeigt, wie eine derartige Eingabe im Dialogfeld des Quellen-Moduls durchgeführt wird.

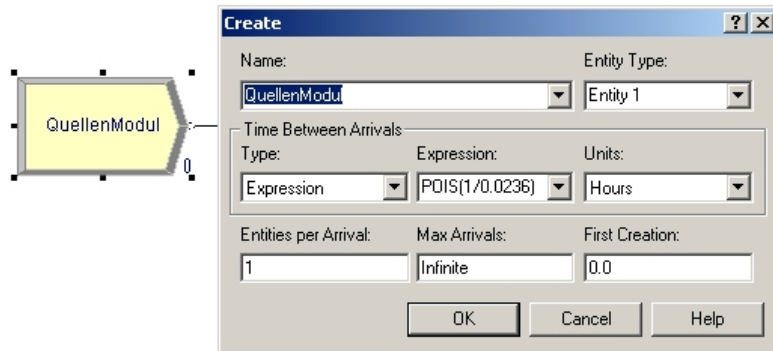


Abb. 2.3: Dialogfenster des Quellen-Moduls

Im nächsten Modul, einem Zuweisungs-Modul, wird dem Attribut „Entity.Sequence“ der gerade erzeugten Entität der Vektor *seq* zugewiesen, der die in Abbildung 2.2 dargestellte Stationenfolge wiedergibt. In jeder Entität ist somit die zu durchlaufende Folge der Stationen gespeichert. Indiziert wird sie mit *i*, wobei $i = 0, 1, \dots, 173$ gilt. Der Wert 173 setzt sich aus den 172 Operationen der in Abbildung 2.1 dargestellten Stationenfolge zuzüglich einer weiteren Operation für den Ausgang über ein Senken-Modul zusammen. Dieses entspricht dem Ende der Produktionslinie in ARENA. Über den Index *i* kann auf die Stationenfolge der Entität zugegriffen werden. Jede Entität hat zur Verwaltung des jeweiligen Fertigungsfortschritts ein Attribut mit dem Namen „Bearbeitungsstand“. Wird der Index *i* der Stationenfolge auf den Wert dieses Attributs gesetzt, ergibt sich die Station, auf der die gegenwärtige Bearbeitung der Entität stattfindet.

Im nachfolgenden Sende-Modul wird der Wert *i* des Attributs „Bearbeitungsstand“ auf $i+1$ erhöht und die Entität an die entsprechend nächste Station weitergeleitet. Zur Verdeutlichung zeigt Abbildung 2.4 noch einmal die Anfangsschritte, die zur Implementierung in Arena nötig sind: Das Quellen-Modul in Verbindung mit dem Zuweisungs-Modul und dem Sende-Modul. Zusätzlich wird ein Ausschnitt der eingegebenen Stationenfolge dargestellt. Das Fenster umfasst in voller Größe alle 173 Bearbeitungsschritte. Wird eine Entität erstmals in das System

gegeben, ist der Wert des Attributs „Bearbeitungsstand“ $i = 0$. Sobald das erste Sende-Modul erreicht wird, erhöht sich i um Eins und die erste Station in der Stationenfolge wird ermittelt.

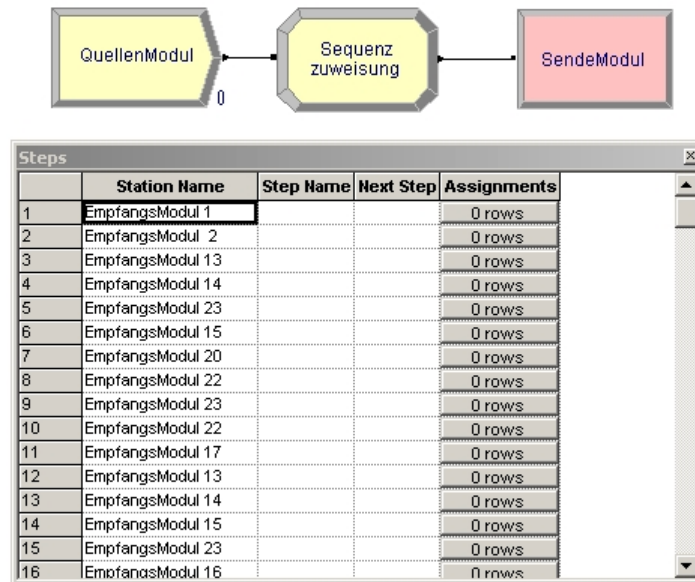


Abb. 2.4: Drei Module und das Dialogfenster zur Eingabe der Stationenfolge

Gemäß Abbildung 2.2 hat die Entität als erstes die Station 1 zu durchlaufen. Die Bearbeitung in der Station 1 wird durch ein Prozess-Modul abgebildet. Die Abbildung 2.5 zeigt, wie die Lagerabgangspolitik im Dialogfenster der Warteschlange eingegeben wird, die vor jedem Prozess-Modul die wartenden Entitäten aufnimmt.

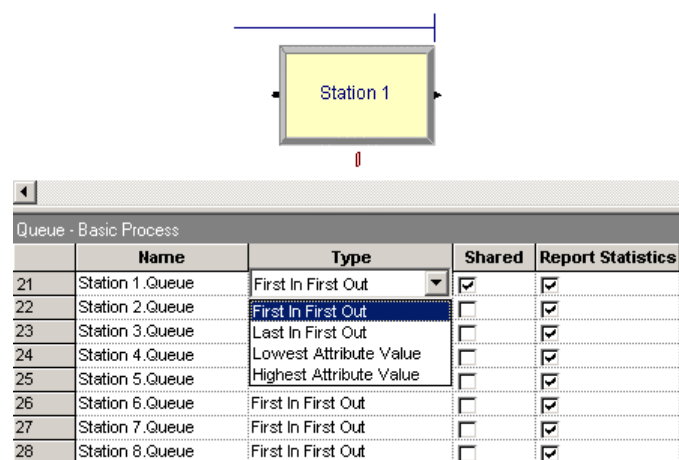


Abb. 2.5: Dialogfenster zur Eingabe der Lagerabgangspolitik

Störungen und Bearbeitungszeiten werden im Prozess-Modul nach den Angaben von WEIN (1988) gesetzt. Im Anschluss an die Bearbeitung wird die Entität wieder zu einem Sendemodul geführt. Entsprechend erhöht sich der Wert des Attributs „Bearbeitungsstand“ auf $i+1 = 2$. Anschließend wird der zweite Schritt der Bearbeitungsfolge ermittelt und analog über ein Sendemodul und ein Prozess-Modul durchgeführt. Dies wiederholt sich so lange, bis der Wert des Attributs „Bearbeitungsstand“ 173 erreicht ist. In der Position 173 der Stationenfolge ist ein Senken-Modul eingetragen, das die Bearbeitung der Entität abschließt. Die Transportzeiten zwischen den Stationen werden in dem Modell vernachlässigt.

Die Bearbeitungszeiten sind vor der Implementierung an die ARENA-Umgebung anzupassen. Die in der Tabelle 2.1 angegebenen Werte unterliegen der Gammaverteilung und sind Erwartungswerte. In ARENA werden bei einer Gammaverteilung der θ - und der α -Wert getrennt voneinander eingegeben. Der Erwartungswert einer Gammaverteilung errechnet sich in ARENA durch $\alpha \cdot \beta$. Bei WEIN (1988) gilt allerdings α/θ . Daraus folgt $\beta = 1/\theta$, so dass der angegebene Erwartungswert, z.B. für die Mittlere Reparaturzeit, mit dem Kehrwert von α multipliziert werden muss, um den β -Wert für die Eingabe in ARENA zu erhalten. Hierbei sind wieder die oben bereits erwähnten Parameterwerte zu setzen: $\alpha = 2$ für die Mittlere Bearbeitungszeit und $\alpha = 0,5$ für die Mittlere Reparaturzeit sowie für die Mittlere Zeit zwischen zwei Störungen. Im Prozess-Modul werden die Bearbeitungszeiten ebenso wie die für den Prozess zu belegenden Ressourcen angegeben. Eine Ressource wird definiert, indem ihr in einem gesonderten Spreadsheet „Ressource“ im Basic Process Panel jeweils ein Name und eine Kapazität zugewiesen werden. Das Spreadsheet „Failure“ im Advanced-Process-Panel dient der Berücksichtigung verschiedener Störungsarten. Hier ist jeweils die Störung zu benennen und es sind die Eintrittsbedingung sowie ihre Parameter anzugeben. Die Eintrittsbedingung kann sich auf die Zeit beziehen oder auf die Anzahl der bearbeiteten Produkt-Entitäten zwischen zwei Störungen. Die verschiedenen Ausfallarten können stochastisch oder deterministisch sein. In dem von WEIN (1988) betrachteten Modell unterliegen die Störungen einer Zufallsverteilung. Bei der Eingabe der Parameter sind, analog zu den Bearbeitungszeiten, die bei WEIN (1988) gegebenen Erwartungswerte mit dem Kehrwert von α zu multiplizieren, um die korrekten β -Werte zu erhalten. In der Abbildung 2.6 wird die Eingabe der Mittleren Bear-

beitungszeit im Dialog-Fenster des Prozess-Moduls für die Station 14 gezeigt. Der einzugebende β -Wert beträgt $3,91 = 7,82 \cdot 0,5$.

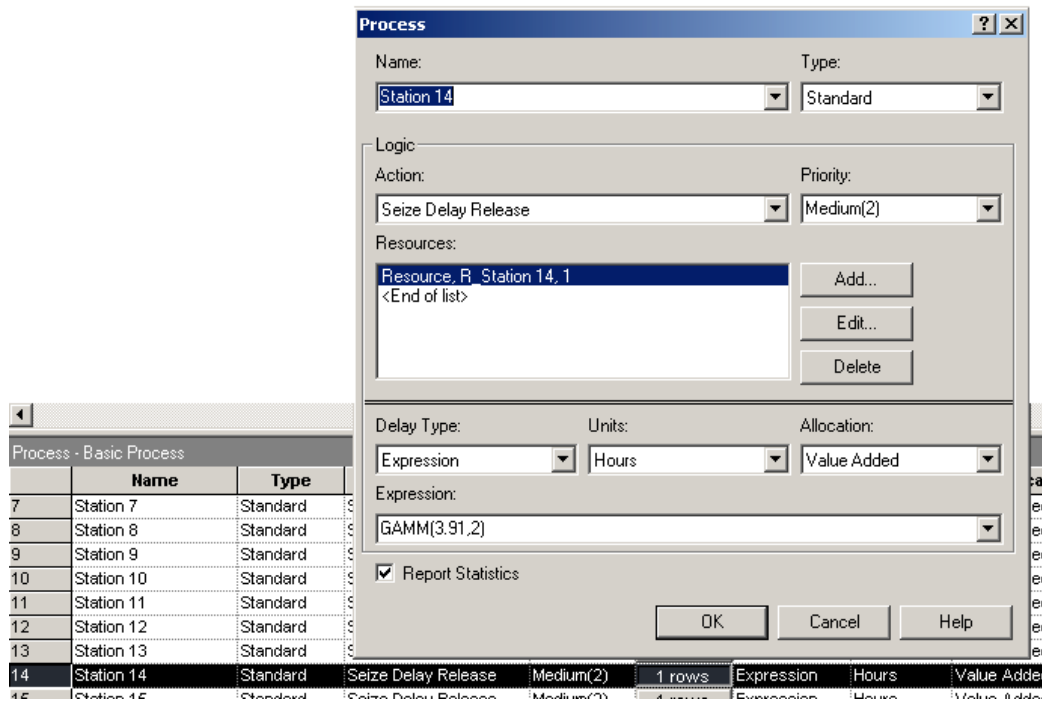


Abb. 2.6: Eingabe der Mittleren Bearbeitungszeit im Dialogfenster des Prozess-Moduls

Nach Anlegen der Störungen können diese im Spreadsheet „Resource“ im Ordner Basic Process einer (oder mehreren) Ressource(n) zugewiesen werden, die bei Eintritt der Störung mit der gesamten Kapazität ausfällt (ausfallen). In dem hier betrachteten Modell wird jeder Station nur eine Ressource zugewiesen. Im Fall einer Störung fällt die Station komplett aus. Hinsichtlich der Störungen stehen in ARENA folgende Optionen zur Verfügung (Abb. 2.7):

1. Eine Störung kann im WAIT-Modus auftreten. In diesem Fall „wartet“ die Störung solange, bis die laufende Bearbeitung der Entität abgeschlossen ist. Erst dann tritt die Reparaturzeit in voller Länge ein.
2. Im Modus IGNORE „wartet“ die Störung ebenfalls, bis die laufende Bearbeitung der Entität abgeschlossen ist – allerdings verkürzt sich die Reparaturzeit um die Wartezeit der Störung. Ist die Bearbeitungsdauer länger als die Reparaturzeit, kann eine Störung im IGNORE-Modus ohne weitere Konsequenzen bleiben.

3. Im Modus PREEMPT wird bei Auftreten der Störung die Bearbeitung der Entität sofort unterbrochen. Die Störung respektive die Reparaturzeit tritt in voller Länge ein. Die unvollständig bearbeitete Entität wird wieder aufgelegt und ihre Bearbeitung fortgesetzt.

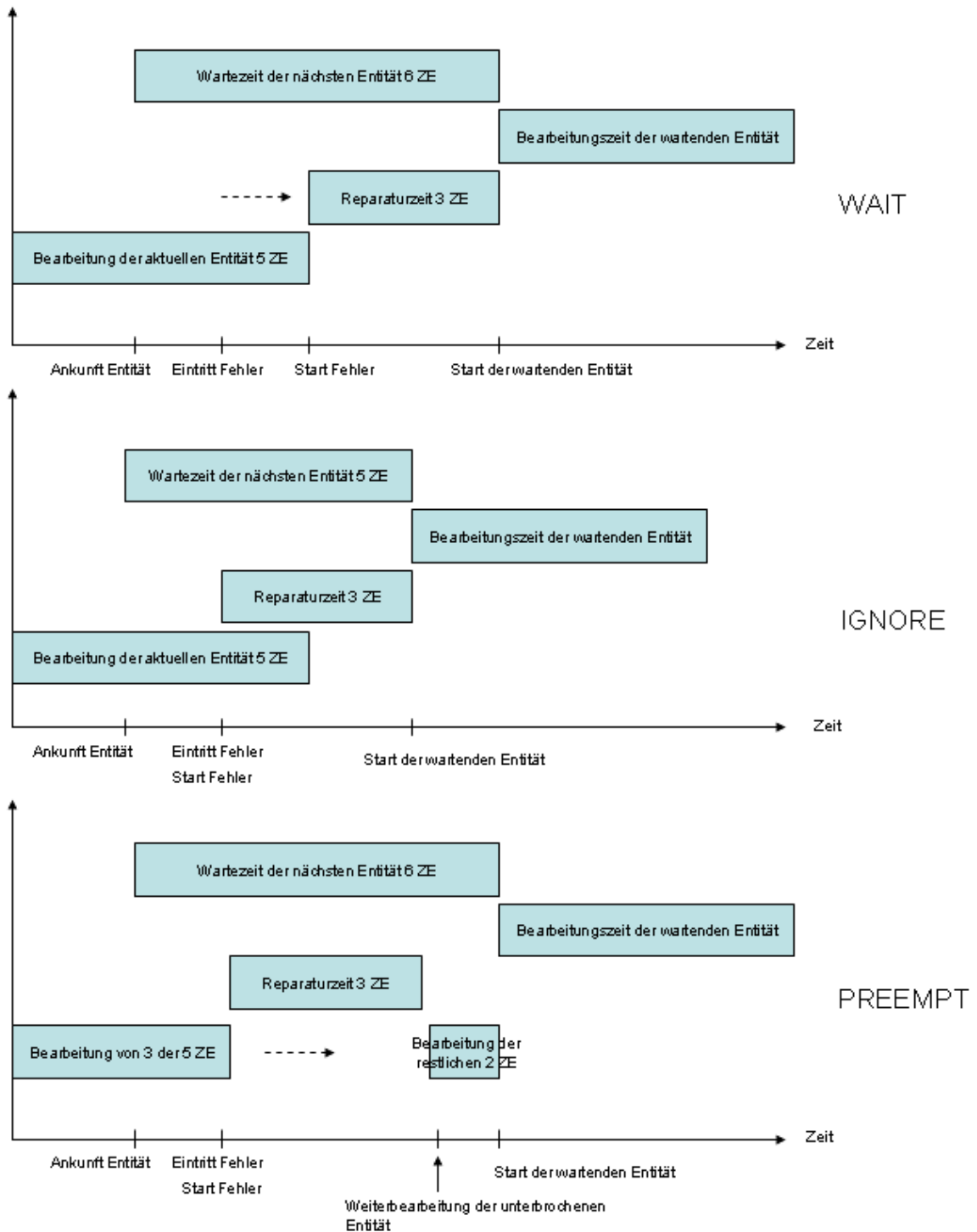


Abb. 2.7: Drei Störungen: WAIT, IGNORE und PREEMPT

Die Interpretation der Störungen bei WEIN (1988) ist mit Schwierigkeiten verbunden. Der Modus PREEMPT und der Modus IGNORE können ausgeschlossen werden: „Maschine Failures are modeled as high priority nonpreemptiv customers [...]“ (vgl. WEIN 1988, S. 118). Die Beschreibung bei WEIN (1988) lässt sich nur so interpretieren, dass Störungen im WAIT-Modus auftreten sollen. Simulationsergebnisse zeigen allerdings, dass die in ARENA vorgefertigten Standard-Routinen zur Modellierung von Fehlern im WAIT-Modus zu Werten führen, die signifikant von denen abweichen, die WEIN (1988) präsentiert (vgl. HERRMANN et al. 2003, S. 15-20).

Um dennoch Störungen im WAIT-Modus modellieren zu können, wird hier der folgende Weg gewählt: Jede Störung wird als eine zusätzliche Entität mit in die Warteschlange des jeweiligen Prozesses aufgenommen. Die Störung belastet dann die Ressource mit zusätzlicher Arbeit. Dieses Vorgehen wird im Weiteren detailliert betrachtet.

2.3 Modellierung von Störungen in ARENA

Um Störungen zu modellieren, wird pro Station eine Dummy-Station eingeführt, die die Störungs-Entität mit einer bestimmten Häufigkeit λ_D auftreten lässt, wobei der Index D für den Störungs-Dummy steht. Weiter ist die Implementierung der Reparaturzeit an der Ressource umzusetzen. Diese Einstellungen werden, wie oben beschrieben, in den Quellen- sowie den Prozess-Modulen vorgenommen. Damit existieren zwei Arten von Prozessen: Bearbeitungs- und Störprozesse. Jede Station hat zwei Arten von Entitäten (Störungs- und Produkt-Entitäten) mit unterschiedlichen Prozesszeiten zu bearbeiten. Den Störungs-Entitäten ist in den Warteschlangen der Prozesse oberste Priorität einzuräumen (Abb. 2.8), d.h. bereits wartende Produkt-Entitäten haben Störungs-Entitäten immer den Vorzug zu gewähren. Somit existiert eine Konkurrenz zwischen den zu bearbeitenden Produkten und den Störungs-Entitäten der Dummy-Stationen um die Ressource einer Station.

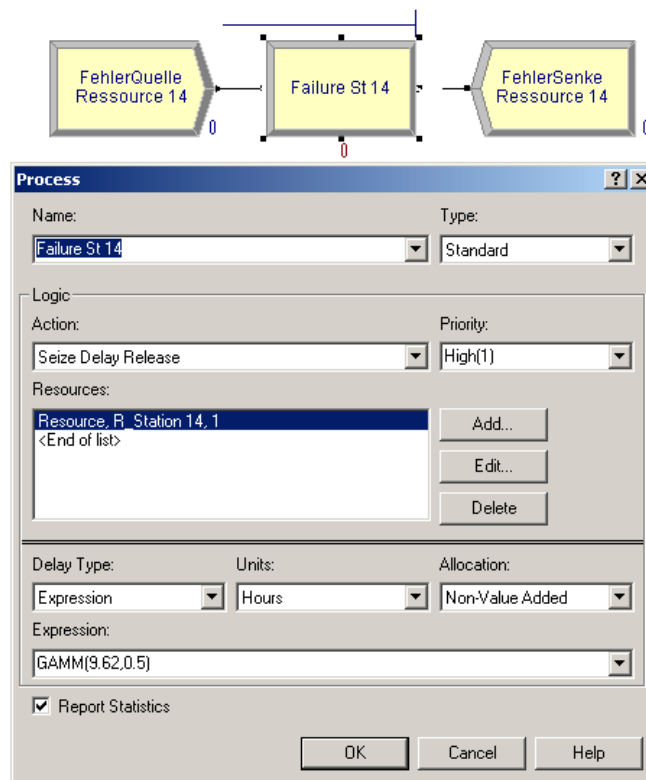


Abb. 2.8: Dialogfenster der Dummy-Station

In der „Bearbeitungszeit“ der Störung entsteht keine Wertschöpfung, sondern nur eine Verzögerung der Produktionszeit um die Mittlere Reparaturzeit. Die einzugebenden Werte Mittlere

Reparaturzeit und Mittlere Zeit zwischen zwei Störungen sind gammaverteilt und in der Tabelle 2.1 gegeben.

In Abbildung 2.9 ist ersichtlich, dass zwischen aufeinander folgenden Störungen zunächst die Mittlere Reparaturzeit verstreichen muss, bevor die Mittlere Zeit zwischen zwei Störungen (MTBF) neu beginnt. Der fehlerfreie Betrieb nach einer Störung kann erst wieder aufgenommen werden, wenn die Reparatur abgeschlossen ist. Die Berechnung der Ankunftsrate der ankommenden Störungen erfordert somit folgende Rechenschritte:

$$\lambda_D = \frac{1}{\alpha} \cdot MTTR + \frac{1}{\alpha} \cdot MTBF \quad \text{mit} \quad \alpha = 0,5$$

Bei der Bestimmung der Ankunftsrate λ_D ist neben der Mittleren Zeit zwischen zwei Störungen (MTBF) auch die Mittlere Reparaturzeit (MTTR) zu berücksichtigen.

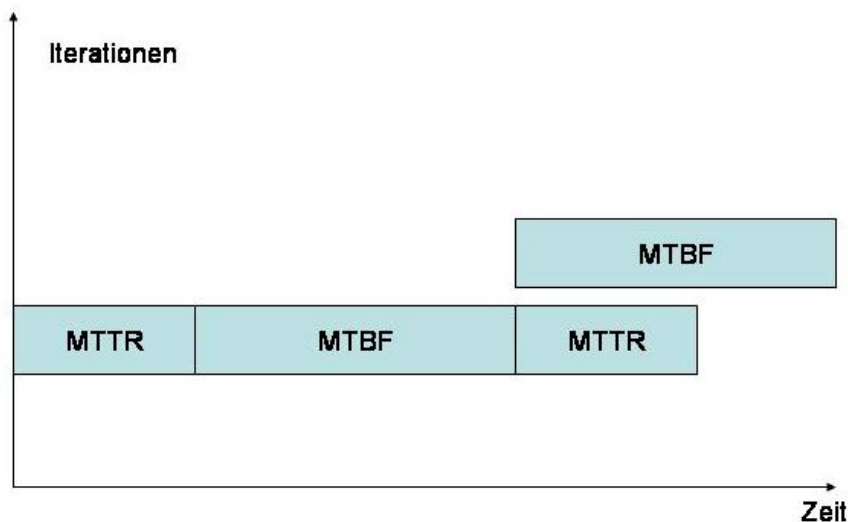


Abb. 2.9: Bestimmung der Ankunftsrate λ_D

Die Einstellung der Bearbeitungszeit erfolgt analog zur oben beschriebenen Implementierung. Hier wird allerdings statt der Mittleren Bearbeitungszeit die Mittlere Reparaturzeit eingegeben (Abb. 2.8).

Durch diese Implementierung des Fab 1.24 wird nun versucht, die von WEIN (1988) ermittelten Werte mit ARENA nachzubilden. Über den Befehl „Report“ ist es möglich, die gesamten Modelldaten abzufragen. Diese Werte werden im nächsten Abschnitt miteinander verglichen.

2.4 Auswertung und Vergleich der Ergebnisse

Nach der Simulation einer Produktionssequenz in ARENA können über den Link „Report“ die gesammelten Daten abgerufen werden, wobei in ARENA 'Usage' statt 'Utilization' ((Maschinen-) Auslastung) genutzt wird. Um die Daten für den Vergleich der Werte zu erhalten, wird die Funktion „Report“ verwendet.

In Tabelle 2.4 werden die in SIMAN von WEIN (1988) ermittelten Simulationsergebnisse den Werten gegenübergestellt, die in der ARENA-Simulation unter der Verwendung der Standardfunktion WAIT (ARENA-Standard) sowie der Modellierung von Störungen mittels Dummy-Stationen (ARENA-Dummy) erreicht wurden. Zur Vereinfachung der Notation wird im Folgenden W für WEIN (1988), S für ARENA-Standard und D für ARENA-Dummy verwendet und die ermittelten Abweichungen werden mit M für die Maschenauslastung und mit P für die Performance-Kennzahlen Mittlere Durchlaufzeit (MCT) und Mittlere Wartezeit (MQT) indiziert. Ein Vergleich der prozentualen Abweichungen zwischen Δ_D^M und Δ_S^M zeigt, dass mit der Verwendung der Dummy-Stationen die ermittelte Maschinenauslastung nahezu identisch zu den von WEIN (1988) ermittelten Werten ist. Die maximale, prozentuale Abweichung beträgt lediglich 2,98% an Station 21, während bei Verwendung der ARENA-Standard-Funktion WAIT eine maximale prozentuale Abweichung von 30% bei Station 24 auftritt. Die deutlichen Verbesserungen sind in der folgenden Tabelle fett hervorgehoben.

Tab. 2.4: Gegenüberstellung der Ergebnisse für die Maschinenauslastung

Station	W	S	D	$\Delta_S^M = \frac{ W - S }{W}$	$\Delta_D^M = \frac{ W - D }{W}$
1	39,80	39,53	39,58	0,0068	0,0055
2	38,40	38,22	39,43	0,0047	0,0268
3	37,00	36,43	37,48	0,0154	0,0130
4	43,90	43,31	43,36	0,0134	0,0123
5	21,00	20,58	20,92	0,0200	0,0038
6	41,40	41,01	40,90	0,0094	0,0121
7	35,40	34,94	34,58	0,0130	0,0232
8	51,40	49,74	51,23	0,0323	0,0033
9	48,30	47,43	48,62	0,0180	0,0066
10	49,80	48,42	49,28	0,0277	0,0104
11	46,20	43,61	44,86	0,0561	0,0290
12	40,30	40,53	40,04	0,0057	0,0065
13	37,30	36,55	37,13	0,0201	0,0046
14	91,90	80,04	91,74	0,1291	0,0017
15	34,00	33,54	33,80	0,0135	0,0059
16	38,40	38,39	38,36	0,0003	0,0010
17	38,10	38,25	38,00	0,0039	0,0026
18	33,90	33,31	33,94	0,0174	0,0012
19	60,10	56,97	60,33	0,0521	0,0038
20	46,10	45,64	45,54	0,0100	0,0121
21	56,40	54,08	54,72	0,0411	0,0298
22	26,70	26,70	26,56	0,0000	0,0052
23	29,60	29,68	29,42	0,0027	0,0061
24	55,30	38,65	56,01	0,3011	0,0128

Als zweites Kriterium zur Beurteilung der Validität der ARENA-Modelle werden die Performance-Kennzahlen mit den Werten von WEIN (1988) verglichen (Tab. 2.5).

Tab. 2.5: Gegenüberstellung der Performance Kennzahlen

	W	S	D	$\Delta_S^P = \frac{W - S}{W}$	$\Delta_D^P = \frac{W - D}{W}$
MCT	1035	782,37	1137,37	0,2441	0,0989
MQT	486	233,14	588,62	0,5203	0,2112

Zwar sind die Abweichungen Δ_D^P mit ca. 10% und 20% signifikant hoch, bei Betrachtung der absoluten Werte wird allerdings deutlich, dass diese Abweichungen systematisch erfolgen. Zwischen den Realisationen beider Kennzahlen liegen jeweils ca. 102 ZE.

Als Ursachen der Abweichungen lassen sich die Bearbeitungszeiten ausschließen. Die in der Simulationsstudie ermittelten Value-Added Time (VAT) - siehe „Report“ unter „Category Overview“ – stimmt mit 549,22 ZE weitgehend mit der theoretischen VAT 545,07 überein. Diese ergibt sich aus der Multiplikation der Zahl der Bearbeitungen je Entität (NV/L) mit der jeweiligen Mittleren Bearbeitungszeit (MPT) und der Addition über alle Stationen ($VAT = \sum_{j=1}^{24} NV / L_j * MPT_j$). Inhaltlich gibt die VAT die gesamte Zeit der Bearbeitung an, die ein Produkt mindestens benötigt. Da der simulativ ermittelte Wert mit dem theoretischen Wert nahezu übereinstimmt, ist davon auszugehen, dass die Produktionsstruktur des Fab 1.24 richtig implementiert wurde.

Die verbleibenden Abweichungen sind auf implizite Annahmen bei WEIN (1988) zurückzuführen und sollen hier vernachlässigt werden, da sie systematisch auftreten.

Zusammenfassend ist festzuhalten, dass es gelungen ist, das Modell Fab 1.24 erfolgreich in ARENA zu implementieren. Durch die Simulation sind die Ergebnisse von WEIN (1988) nahezu erreicht worden.

3. Mehrprodukt-Reentrant-Line: System I

3.1 Darstellung des Modells

In diesem Kapitel wird eine von PARK et al. (2002) als System I bezeichnete Mehrprodukt-Reentrant-Line betrachtet, die aus sieben Stationen besteht. Im Gegensatz zum Fab 1.24 werden auf dieser Produktionslinie drei verschiedene Produkt-Typen (Classes) bearbeitet. Als weitere Besonderheit setzen PARK et al. (2002) zwei unterschiedliche Maschinen-Typen ein:

- Single-Maschinen können nur einen Auftrag bzw. nur eine Entität zu einem Zeitpunkt bearbeiten.
- Batch-Maschinen können mehrere Entitäten beschleunigt bearbeiten, die zuvor zu einer Einheit zusammengefasst wurden.

Prinzipiell ist ein Batch ein aus mehreren unterschiedlichen Produkten zusammengefasstes Los, das eine gemeinsame Bearbeitung erfährt. Auf die unterschiedlichen Varianten des Batching geht z.B. JORDAN (1996) näher ein. Bei PARK et al. (2002) wird eine Sonderform unterstellt: Es können nur die Entitäten eines Produkt-Typs zusammengefasst werden, die im gleichen Bearbeitungsstand auf den nächsten Schritt im Bearbeitungsprozess warten. Durch die Zusammenfassung als Batch reduziert sich allerdings die Bearbeitungsdauer je Entität ohne Beachtung der Wartezeit erheblich. Die Bearbeitungsdauern für die Single- und Batch-Maschinen sind in Tabelle 3.1 angegeben.

Tab. 3.1: Bearbeitungszeiten und Zahl der Bearbeitungen im System I

Station	Bearbeitungszeit		Zahl der Bearbeitungen je Entität		
	Einzel	Batch (Größe)	Produkt-Typ 1	Produkt-Typ 2	Produkt-Typ 3
1	1,7		2	3	4
2	0,9	1,9 (5)	3	3	5
3	1,4	2,5 (2)	3	3	3
4	2,5	4,3 (5)	2	2	3
5	3,0	4,0 (4)	2	3	3
6	0,8		1	3	3
7	1,2		1	1	1

Die Bearbeitungszeiten sind exponentialverteilt und sowohl vom Produkt-Typ als auch vom Bearbeitungsstand unabhängig. Das bedeutet, dass z.B. die erste Bearbeitung des Produkt-Typs 3 in der Station 3 im Mittel genauso lange dauert, wie die zweite Bearbeitung des Produkt-Typs 1 auf der Station 3. Die Produkt-Typen unterscheiden sich damit lediglich durch ihre Stationenfolge voneinander. Bei den Batch-Maschinen bezieht sich die Dauer der Bearbeitung des gesamten Batches an einer Maschine immer auf die in Tabelle 3.1 in Klammern angegebene Zahl an Entitäten. Wenn z.B. eine Batch-Maschine an Station 2 genutzt wird, dann werden fünf Entitäten im Mittel in 1,9 ZE bearbeitet. Durchlaufen die 5 Entitäten hingegen die Single-Maschine, sind fünfmal 0,9 ZE zu veranschlagen. Die im Vergleich deutlich schnellere Bearbeitung eines Batches ist durch Einsparungen beim Rüsten und Einrichten der Aggregate zu erklären. Explizit werden Rüstzeiten bei PARK et al. (2002) nicht berücksichtigt.

Die Stationenfolge ist für jeden Produkt-Typ in Abbildung 3.1 dargestellt. Jede Entität eines Produkt-Typs besucht zuerst die Station 1, durchläuft dann in der vorgegebenen Reihenfolge den Produktionsprozess und verlässt diesen schließlich über die Station 7.

Wie bereits bei dem Modell von WEIN (1988) wird auch bei PARK et al. (2002) innerhalb jedes Pufferlagers als Lagerabgangspolitik FIFO angewendet. Insgesamt werden 54 Puffer benötigt, da vor jeder Station Lager für die wartenden Produkte jedes Produkt-Typs differenziert nach dem jeweiligen Bearbeitungsstand vorhanden sein müssen. Konkurrieren Entitäten in verschiedenen Pufferlagern um die Weiterbearbeitung einer Station, werden sie ebenfalls entsprechend der FIFO-Lagerabgangspolitik entnommen.

Hinsichtlich der Einlastungspolitik unterstellen PARK et al. (2002) einen konstanten Work-In-Process (CONWIP), d.h. die Anzahl der Teile, die sich im System befinden. Die Simulation wird von ihnen für verschiedene WIP-Einstellungen $N = 54, 108, 162, \dots, 1026$ durchgeführt, wobei sich N als Vielfaches der Gesamtanzahl aller Puffer im System ergibt.

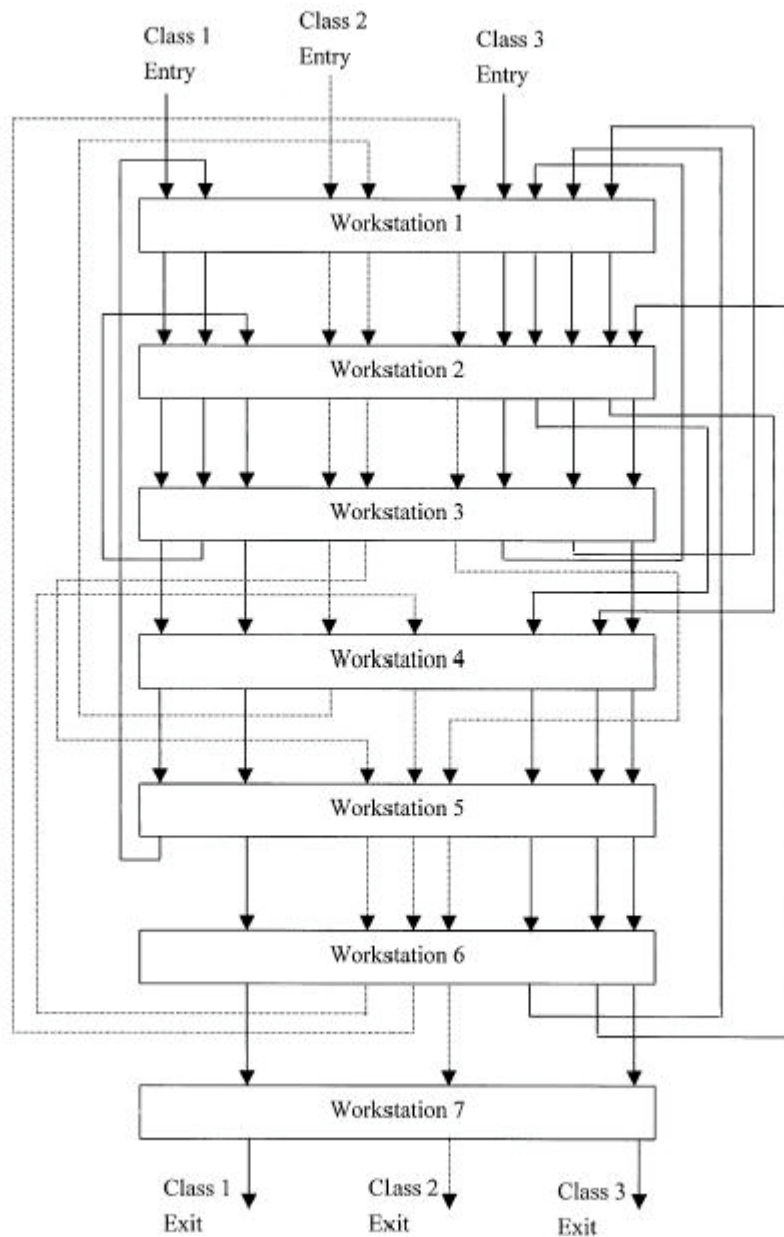


Abb. 3.1: Das System I

Die Zielsetzung dieses Kapitels besteht zunächst darin, aufzuzeigen, wie die beschriebene Struktur des System I in ARENA für $N = 108$ simuliert werden kann. Danach wird das Batch-
ing berücksichtigt und zu weiteren Untersuchungszwecken N , entsprechend den Vorgaben von PARK et al. (2002), variiert. Abschließend werden im Abschnitt 3.4 die Auswirkungen begrenzter Lagerkapazitäten simuliert und ausgewertet.

3.2 Implementierung in ARENA

Die Implementierung des System I in ARENA greift in weiten Teilen auf die des Fab 1.24 zurück. Die Bearbeitung der Entitäten und der Ablauf der Sequenz über die Stationen erfolgen analog zu der Erstellung des Fab 1.24. Die Einlastung der Produkte auf den Stationen muss so erfolgen, dass sich ein konstanter WIP einstellt. Im Gegensatz zum Fab 1.24 liegt dem Ankunftsprozess der Entitäten im System keine Verteilung zugrunde. Die erste Befüllung des Systems soll derart erfolgen, dass alle Entitäten „gleichzeitig“, d.h. zu Beginn der Simulation und im Verhältnis der Puffer der einzelnen Produkt-Typen eingelastet werden. Die 54 Puffer teilen sich wie folgt auf die einzelnen Produkt-Typen auf (Abb. 3.1):

- Produkttyp 1 = 14 Pufferlager,
- Produkttyp 2 = 18 Pufferlager,
- Produkttyp 3 = 22 Pufferlager.

Daraus ergeben sich folgende Verhältnisse der Pufferlager für die erste Befüllung:

$$r_1 = \frac{14}{54} = 0,2593 \qquad r_2 = \frac{18}{54} = 0,3333 \qquad r_3 = \frac{22}{54} = 0,4074.$$

Danach wird die Menge der Teile im System konstant gehalten. Wesentlich ist, dass nicht alle Entitäten auf einmal (zum gleichen Zeitpunkt) eingelastet werden, da ARENA das System in diesem Fall zunächst mit allen Entitäten des Produkt-Typs 1, dann mit allen Entitäten des Produkt-Typs 2 und schließlich mit allen Entitäten des Produkt-Typs 3 befüllen würde. Mit der hier gewählten Vorgehensweise wird eine Mischung der drei Produkte erreicht und gleichzeitig das System nur einmal über die Quellen-Module gefüllt. Die Umsetzung in ARENA wurde wie folgt vorgenommen: Aufeinander folgend treffen zu Beginn der Simulation die eingelasteten Mengen im Sekundentakt ein. Bezogen auf die im System sonst verwendete Zeiteinheit von Stunden sind die für diese Einlastung benötigten Sekunden vernachlässigbar klein. Das Quellen-Modul ist wie folgt zu modifizieren: Unter „Time Between Arrivals“ ist keine Verteilung wie im Fab 1.24 anzugeben, sondern die Zwischenankunftszeit ist auf Konstant (constant) 1 Sekunde einzustellen. Da pro Ankunft eine Entität in das Modell eintritt, dürfen maximal WIP $14/54$ (für Produkt-Typ 1) Ankünfte stattfinden. Dieser Ausdruck ist unter „Max Arrivals“ einzugeben. Da bei PARK et al. (2002) verschiedene WIP-Level simuliert werden, wird der WIP in ARENA als eine eigenständige Variable (*WIP*) eingerichtet. Soll z.B. ei-

ne Simulation mit einem WIP von 162 durchgeführt werden, ist unter „Variable“ im „Process-Panel“ der Wert des WIP von den vorgegebenen 108 Einheiten auf 162 zu setzen.

Auffällig an dem Systemaufbau ist das Fehlen eines Senken-Moduls, das gewöhnlich ein Modell beendet. Wegen der Bedingung des konstanten WIP und wegen des konstanten Verhältnisses der Anzahl an Produkt-Typen muss Folgendes erreicht werden: Eine Entität, die die Station 7 erreicht hat und somit vollständig bearbeitet das System verlässt, ist durch eine neue noch zu bearbeitende Entität desselben Produkt-Typs zu ersetzen. Die Entitäten gelangen aus diesem Grund nach Durchlaufen von Station 7 über die Empfangs-Module 8 bis 10 wieder in den „Quellen-Bereich“ zurück.

Hier wird mit einem Speicher-Modul die Zeit erfasst, die die Entität für den kompletten Produktionsdurchlauf benötigt hat. Diese wird ermittelt, indem von der aktuellen Simulationszeit, die der Entität bei Eintritt in das System zugewiesene (damals) aktuelle Simulationszeit abgezogen wird (vgl. Abb. 3.4). Die erfassten Werte können nach Simulationsende im Report eingesehen werden.

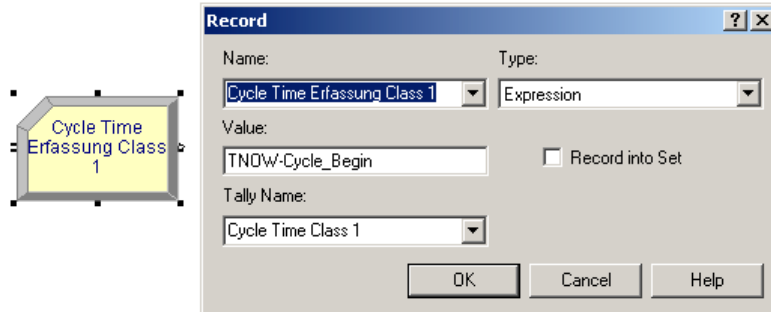


Abb. 3.2: Dialogfenster des Speicher-Moduls

Darauf folgend gelangen die Entitäten in ein Zuweisungsmodul, das sie bereits zu Beginn der Produktion passiert haben. In diesem ersten Durchlauf zu Produktionsbeginn wurden ihnen hier folgende Werte zugewiesen:

- Die aktuelle Simulationszeit.
- Die Anzahl der bisher durchlaufenen Arbeitsschritte wird auf Null gesetzt.
- Die jeweilige Stationenfolge.
- Ein farbiges Symbol (blau, grün, gelb) zur Unterscheidung der Produkt-Typen.

Gelangt eine Entität nach Durchlaufen des Produktionsprozesses abermals in das Zuweisungs-Modul, werden die Werte erneut zugewiesen. Durch das Zurücksetzen der Arbeitsschritte nimmt die Entität wieder den Status einer neu in das System eingelassenen Entität an und durchläuft die Produktion erneut. Die wiederholte Zuweisung der Stationenfolge und des Farbsymbols sind nicht zwingend notwendig. Dennoch wurde diese Vorgehensweise gewählt, um das Einfügen eines weiteren Zuweisungs-Moduls zu vermeiden. Die Zuweisung der aktuellen Simulationszeit darf nicht unterlassen werden, da ansonsten die Durchlaufzeit der Entität nicht korrekt berechnet werden könnte.

Für eine korrekte Datenanalyse muss die Einschwingphase (die Zeitspanne, die ein System benötigt, um in einen auswertungsfähigen Zustand zu kommen) berücksichtigt werden. Deshalb wird das Modell in einem Zwischenschritt mit einem zusätzlichem Lese-/Schreib-Modul ausgestattet. Dieses Modul wird innerhalb des „Quellen-Bereichs“ zwischen das Empfangs-Modul und das Speicher-Modul geschaltet.

Das Lese-/Schreib-Modul reagiert selbständig auf die Ankunft einer Entität. Sobald sie das Modul erreicht, wird die dazugehörige Datei darauf überprüft, ob diese aktiv bzw. geöffnet ist. Ist dies nicht der Fall, wird die Datei automatisch geöffnet. Im Fall des Lese-/Schreib-Moduls „ReadWrite1“ wird die dazugehörige Datei „CTC1“ automatisch geöffnet. Die Daten „TNOW“ (aktuelle Simulationszeit) und „TNOW-Cycle-Time“ (aktuelle Simulationszeit abzüglich dem Startzeitpunkt des Durchlaufs der Entität = Durchlaufzeit) werden in die Datei geschrieben. Anhand der Daten in den Dateien „CTC1“, „CTC2“ und „CTC3“ kann festgestellt werden, dass eine Einschwingphase von 20.000 Stunden ausreicht, um eine Datenauswertung vorzunehmen, die nicht durch Startunregelmäßigkeiten verfälscht ist. Von den insgesamt 40.000 Stunden, die simuliert werden, werden die ersten 20.000 Stunden aus der Auswertung herausgenommen, so dass sich die Analyse auf die restliche Zeit bezieht. Um weitere Störgrößen auszuschließen, werden jeweils 20 Replikationen durchgeführt. Nach der (einmaligen) Bestimmung der Einschwingphase kann das Lese-/Schreib-Modul aus dem Modell entfernt werden, da es für den restlichen Ablauf der Simulation nicht mehr benötigt wird.

3.3 Erweiterung I „Batching“

3.3.1 Berücksichtigung einer Batch-Station

In Tabelle 3.2 sind die Anzahl und die Position der Batch-Stationen für fünf verschiedene Simulationsaufbauten angegeben (vgl. PARK et al. 2002, S. 1019).

Tab. 3.2: Position und Anzahl der Batch-Stationen in den 4 Szenarien

Szenario	Batch-Stationen	
	Anzahl	Position
1-0	0	–
1-1	1	3
1-2	2	3,5
1-3	3	2,3,5
1-4	4	2,3,4,5

Um eine Batch-Station einzufügen, wird das Grundmodell des System I durch eine Ergänzung im Prozessbereich, z.B. an Station 3, modifiziert. Der Quellen-Bereich, in dem die Entitäten bereitgestellt werden, bleibt unverändert (Abb. 3.3).

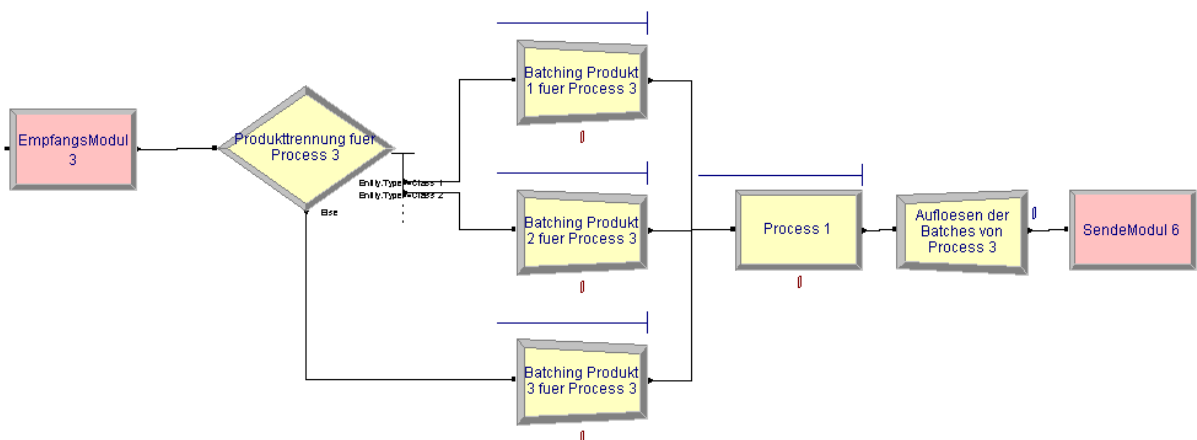


Abb. 3.3: Module im Batch-Bereich des Systems I

Zuerst wird ein Entscheidungs-Modul eingefügt, das die Produkttrennung für die Bearbeitung an Station 3 vornimmt (Abb. 3.4). Es erkennt Entitäten des Produkt-Typs 1 und des Typs 2, so dass diese – ihrem Produkt-Typ entsprechend – in das zugehörige Gruppierungs-Modul weitergeleitet werden können.

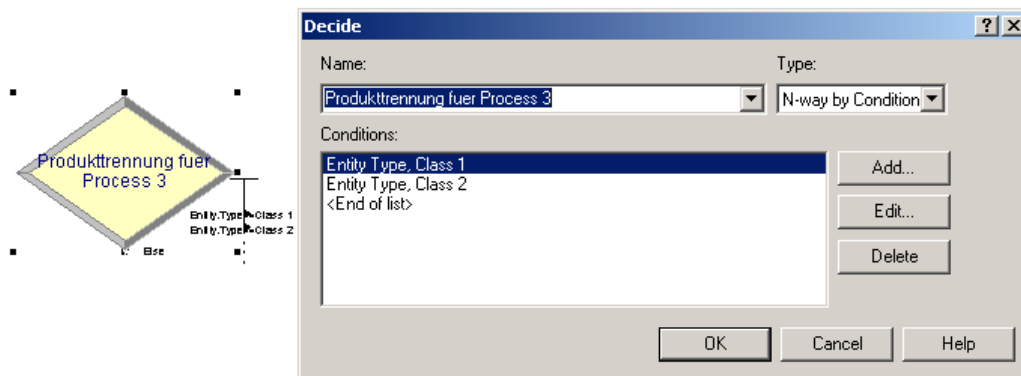


Abb. 3.4: Dialogfenster des Entscheidungs-Moduls

Alle übrigen Entitäten müssen zwangsläufig zum Produkt-Typ 3 gehören und werden dem dritten Gruppierungs-Modul zugeordnet (Abb. 3.5). Das Batching einzelner Entitäten erfolgt anhand ihres Attributes „Bearbeitungsstand“, indem nur solche mit dem gleichen Attributwert zusammengefasst werden. In der Warteschlange einer Station können daher mehr als zwei Entitäten warten, sofern diese einen unterschiedlichen Bearbeitungsstand aufweisen. Da die Zusammenfassung nur vorübergehend erfolgen soll, wird im Gruppierungs-Modul der Typ „Temporary“ zugewiesen. Die Größe eines Batches kann unter „Batch-Size“ des jeweiligen Moduls vorgegeben werden. Bei Station 3 müssen immer zwei Entitäten zu einem Batch zusammengefasst werden.

Die Größe des Batches ist in Tabelle 3.1 bereits angegeben worden. Hinter den Bearbeitungszeiten stehen in Klammern die Größen der Batches.

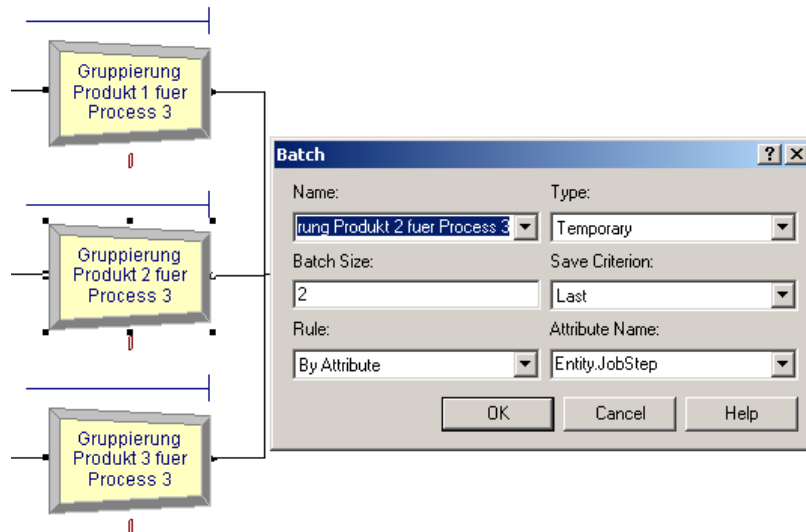


Abb. 3.5: Dialogfenster des Gruppierungs-Moduls

Vom Gruppierungs-Modul gelangen die zusammengefassten Entitäten in das Prozess-Modul, in dem sie – unter Berücksichtigung der veränderten Bearbeitungszeiten (siehe Tabelle 3.1, Spalte 2) – bearbeitet werden. Danach werden die Batches im Separierungs-Modul wieder in die ursprüngliche Anzahl von Entitäten getrennt, aus denen sie vor dem Bearbeiten zusammengesetzt wurden (Abb. 3.6). Durch die Option „Retain Original Entity Values“ unter dem Eintrag Member Attribute im Separierungs-Modul „Auflösen der Batches von Process 3“ wird jeder Entität der Zeitpunkt des Starts ihrer ursprünglichen Bearbeitung „Cycle Begin Time“ erneut zugewiesen und die Bearbeitung kann fortgesetzt werden.

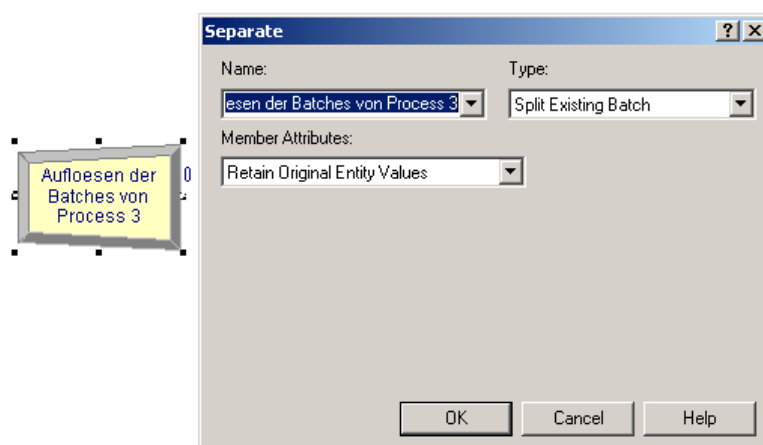


Abb. 3.6: Dialogfenster des Separierungs-Moduls

Sollen statt einer Batch-Station mehrere implementiert werden, sind – analog zum beschriebenen Vorgehen – die oben genannten Erweiterungen bspw. den Stationen 2, 4 und 5 hinzuzufügen.

3.3.2 Auswertung und Vergleich der Ergebnisse

Zur Validierung des ARENA-Modells sind die Ergebnisse von PARK et al. (2002) mit den hier ermittelten Daten zu vergleichen. Da PARK et al. (2002) ihre Ergebnisse aggregiert und in Form einer Graphik darstellen, werden auch die mit ARENA gefundenen Ergebnisse in eine Graphik eingebunden und der ursprünglichen gegenübergestellt. Für die Simulation mit variierenden WIP-Werten wird der von Rockwell Software mitgelieferte Process Analyzer angewendet. Dieser soll im Folgenden kurz dargestellt werden, bevor auf die einzelnen Graphen und ihren Vergleich näher eingegangen wird.

Der Process Analyzer (PAN) ist ein Tool, mit dem ein in ARENA implementiertes Modell einfach und effektiv mit variierenden Daten zu simulieren ist. Wie aus Abbildung 3.10 ersichtlich, unterteilt sich das sogenannte Grid-Window in drei größere Bereiche.

- In dem Bereich der Scenario Properties ist u.a. der Dateiname des zu analysierenden Modells anzugeben.
- Im Bereich des Control finden sich u.a. die Spalten der Controls, die auch Faktoren genannt werden. Sie geben die interessierenden Level der Variablen wieder. Durch die unter Kapitel 3.2 beschriebene Deklaration des WIP's als eigenständige Variable kann diese als „Control“ in den PAN geladen werden (vgl. KELTON et al. 2002, S. 242 ff.). Nach dem Duplizieren der Szenarien kann für jedes Szenario ein individueller WIP, durch separates Ändern in der Controls-WIP-Spalte, zugewiesen werden. Die Anzahl der Controls ist dabei nicht auf Eins beschränkt. Es können auch mehrere Variable als Controls eingefügt werden.
- Im Bereich Responses sind die Variablen festzulegen, die zwischen den verschiedenen Simulationen verglichen werden sollen. Im vorliegenden Fall ist es die Mittlere Durchlaufzeit für die drei Produkt-Typen.

Der Process Analyzer arbeitet jeweils mit speziellen Process-Analyzer-Dateien. Diese sind erkennbar an der Dateiendung .p und sie werden nach der Simulation des Modells in ARENA automatisch unter demselben Namen und in demselben Verzeichnis wie die ARENA-Datei erzeugt.

Nachdem die Simulation durchgeführt ist, können die Daten auch graphisch in dem als Chart Windows bezeichneten Fenster dargestellt werden (siehe hierzu auch Abbildung 3.7). Durch die schnelle und automatisierte Möglichkeit der Szenarienvariation ist der PAN bei der Modellanalyse sehr hilfreich.

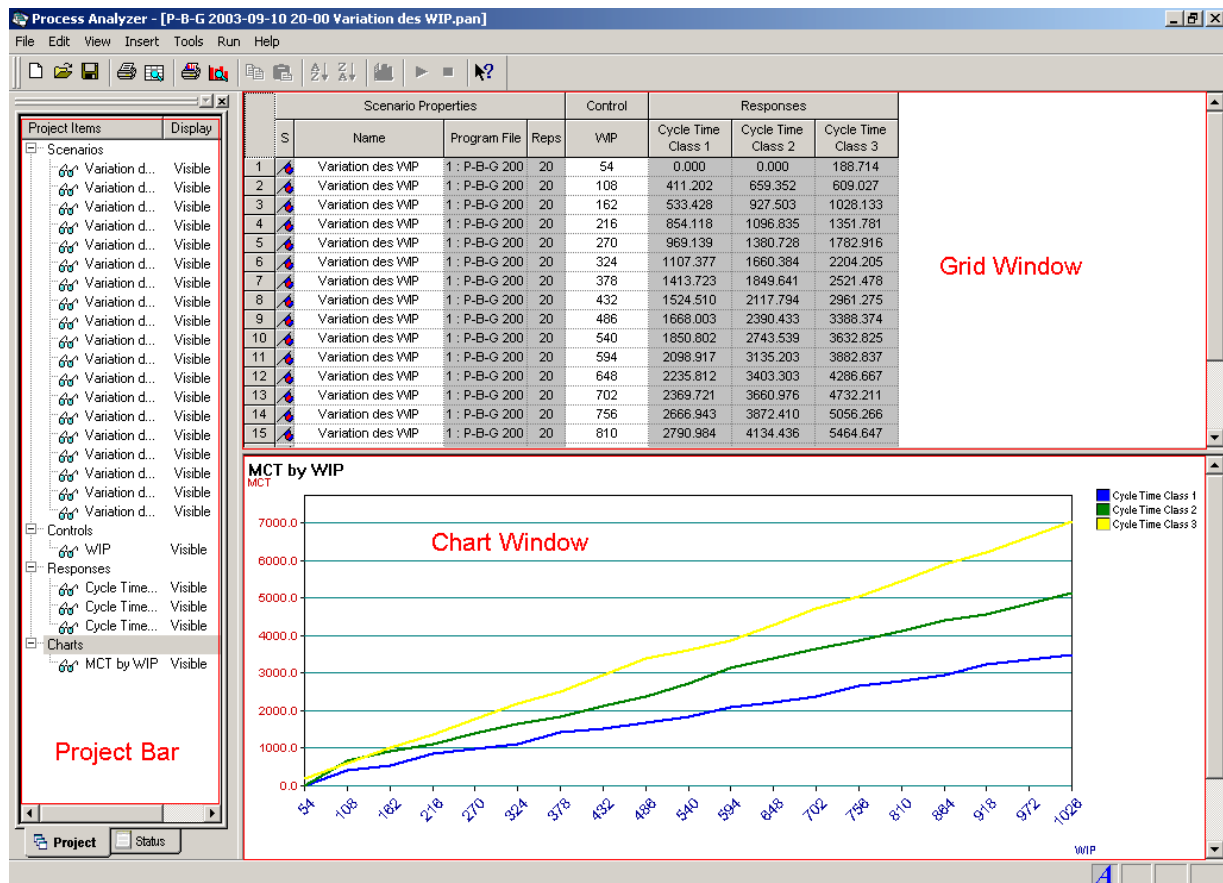


Abb. 3.7: Ansicht des Process Analyzer

In Abbildung 3.8 sind die Anzahl der Teile im System (der WIP) auf der Abszisse abgetragen und auf der Ordinate die Mittlere Durchlaufzeit. Für den Vergleich mit den Daten aus ARENA

sind nur drei der sechs Graphen relevant und zwar C1_Sim, C2_Sim und C3_Sim. Die anderen stellen (Zwischen-)Ergebnisse dar, die hier nicht benötigt werden.

Zum Vergleich mit den von PARK et al. (2002, S. 1022) ermittelten Werten werden die mit ARENA gefundenen Ergebnisse in ein identisch aufgebautes Diagramm eingezeichnet. Die drei Graphen stellen wie zuvor die Mittlere Durchlaufzeit der drei Produkt-Typen in Abhängigkeit zur Anzahl der Teile im System dar.

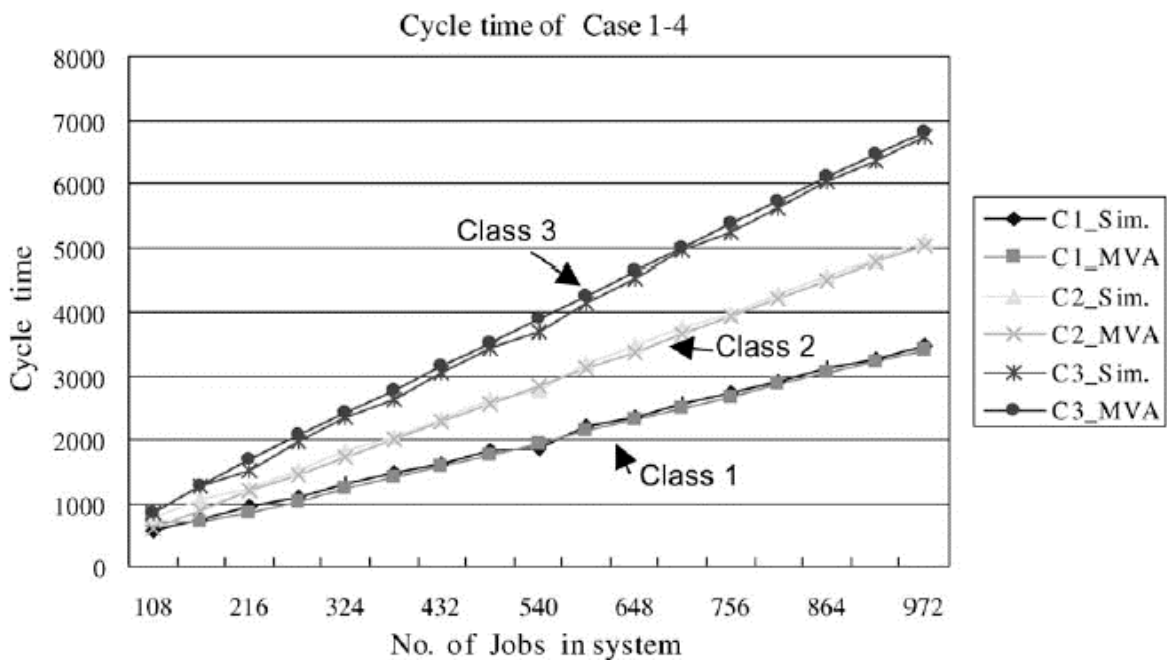


Abb. 3.8: Die Ergebnisse von PARK et al. 2002 für das System I

In beiden Graphiken sind drei unterschiedlich stark ansteigende Funktionen zu erkennen. Alle drei starten nahezu im gleichen Ordinatenschnittpunkt, um dann unterschiedlich schnell anzusteigen. Die Kurve des Produkt-Typs 3 steigt am stärksten und die Kurve für den Produkt-Typ 1 steigt am geringsten. Um die Graphiken noch genauer zu vergleichen, werden sie in einem weiteren Schritt übereinander gelegt.

Die Graphik in Abbildung 3.9 zeigt sehr deutlich, dass die Ergebnisse der ARENA Simulation mit denen der Simulation von PARK et al. (2002) nahezu identisch sind. Die Implementierung in ARENA ist daher als erfolgreich anzusehen. Auf diesem Ergebnis aufsetzend, kann die Implementierung kapazitierter Lager im System I erfolgen.

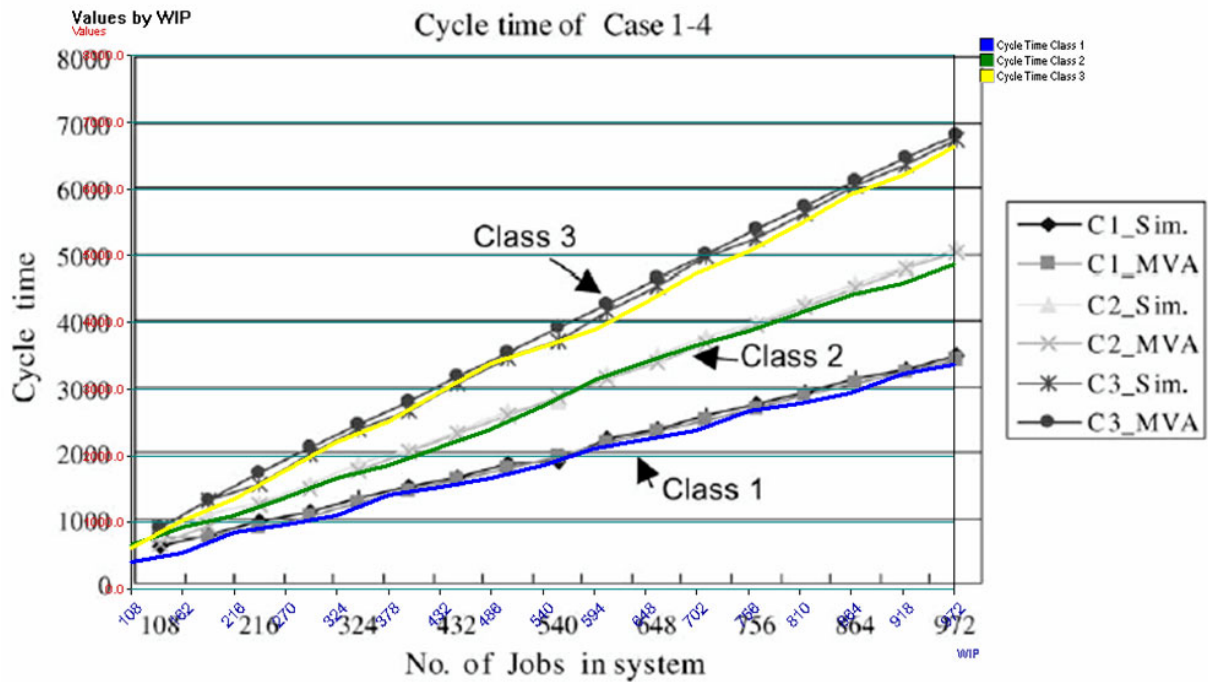


Abb. 3.9: Gegenüberstellung der Ergebnisse für das System I

3.4 Erweiterung II „kapazitierte Lager“

In diesem Abschnitt wird untersucht, wie sich die Mittlere Durchlaufzeit verändert, wenn Kapazitätsbeschränkungen für Lager vor einzelnen Stationen eingeführt werden. Dabei sind Effekte wie das Blockieren vorgelagerter Stationen (Blocking) sowie das Aushungern nachgelagerter Stationen (Starving) möglich.

3.4.1 Begrenzte Lagerkapazität an einer Station

3.4.1.1 Implementierung

Zunächst wird ein begrenztes Lager an Station 3 modelliert, indem das System I an einer Stelle in ARENA erweitert wird. Entitäten, die an Station 3 bearbeitet werden müssen, durchlaufen im vorherigen Bearbeitungsschritt Station 2 (Abb. 3.1). Eine Ausnahme bilden Entitäten des Produkt-Typs 3, die zwar in ihrem fünften Bearbeitungsschritt Station 2 durchlaufen, in Bearbeitungsschritt 6 allerdings Station 3 übergehen, um an Station 4 weiter bearbeitet zu werden. Ähnliches geschieht in den Bearbeitungsschritten 13 und 14. Die Modellierung dieses Sonderfalls wird später betrachtet. Zunächst soll dargestellt werden, wie die Entitäten bei kapazitiertem Lager von Station 2 direkt zu Station 3 gelangen. Ist die Lagerkapazität vor Station 3 ausgeschöpft, müssen die Entitäten schon vor ihrer Bearbeitung an Station 2 aufgestaut werden, da sonst die Kapazitätsgrenze überschritten wird. Zur Modellierung dieses Sachverhalts gibt es zwei Möglichkeiten:

1. Es müsste ein Halte-Modul vom Type „Scan for Condition“ implementiert werden, das die Bedingung

$$NQ(\text{Process 3.Queue}) < \text{Lagerkapazitaet3}$$

prüft, und Entitäten nur passieren lässt, wenn diese Bedingung erfüllt ist. Der Ausdruck NQ (Number in Queue) gibt die Anzahl der Entitäten in der Warteschlange „Process 3.Queue“ an. Die Größe *Lagerkapazitaet3* – die Namensgebung folgt der ARENA-Konvention, Umlaute zu vermeiden – ist eine vom Benutzer selbst definierte Variable, deren Wert je nach gewünschter Lagerkapazität im Data Modul „Variable“ im „Process Analyzer“ geändert

werden kann. Hinsichtlich der Modellierung des Sonderfalls erfordert dieses Vorgehen jedoch eine umfassende Änderung des Grundmodells, um alle Entitäten des Produkt-Typs 3 berücksichtigen zu können. Daher wird diese Vorgehensweise verworfen.

- Die Idee, der hier gefolgt wird, besteht in der Erstellung einer zusätzlichen Entität, einer so genannten Prüf-Entität (im ARENA-Modell: Pruef-Entity). Diese hat zum einen die Aufgabe, zu kontrollieren, ob in Prozess 2 gearbeitet wird und ob die Lagerkapazität an Station 3 ausgeschöpft ist. Zum anderen müssen in der Schlange vor Prozess 2 wartende Entitäten des Produkt-Typs 3, die sich im 5. oder 13. Bearbeitungsschritt befinden, vorgezogen werden.

Die Umsetzung der zweiten Vorgehensweise wird im Folgenden erläutert. Wie Abbildung 3.1 zeigt, müssen die Entitäten nicht vor der Bearbeitung an Station 2 zurückgehalten werden, wenn das Lager an Station 3 ausgefüllt ist, sondern können gegenüber wartenden Entitäten, die im nächsten Arbeitsschritt Station 3 passieren müssen, an Station 2 vorgezogen werden (vgl. auch die Abbildung im Anhang). Ein Halte-Modul „Wartemodul fuer Process 2“ wird vor Prozess 2 eingerichtet. In obiger Bedingung wird die Variable *Lagerkapazitaet3* auf "Infinite Hold" gesetzt, um zu gewährleisten, dass ankommende Entitäten in jedem Fall in diesem Modul warten müssen. Die Erweiterung des Grundmodells durch eine Prüf-Entität beinhaltet ein Quellen-Modul „Erstelle Pruef-Entity“. Die Einstellungen im Dialog-Fenster dieses Moduls müssen so getroffen werden, dass zu Beginn der Simulation nur eine Prüf-Entität für die gesamte Laufzeit des Systems erstellt wird (Abb. 3.10).

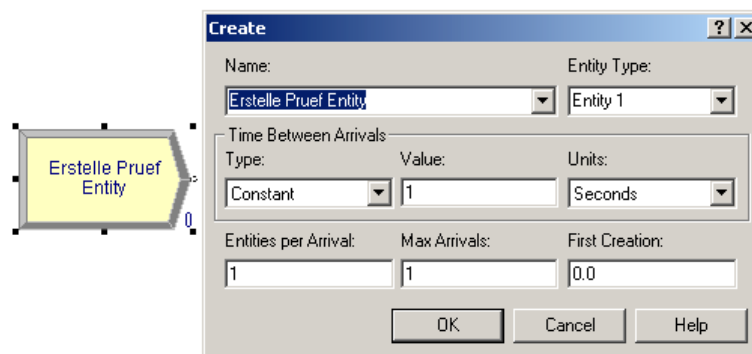


Abb. 3.10: Dialogfenster des Quellen-Moduls

Weiter wird ein Halte-Modul „Pruef Entity wartet bis Process 2 Idle“ eingefügt, das die Prüf-Entität in ihrem ersten Schritt festhält und prüft, ob sich Produkt-Entitäten in der Warteschlange vor Prozess 2 befinden und ob Ressource 2 leerläuft (Abb. 3.11).

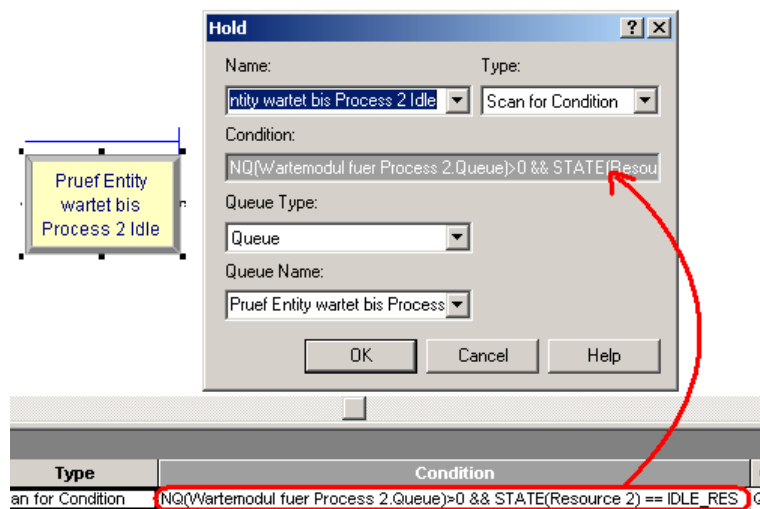


Abb. 3.11: Dialogfenster des Halte-Moduls

Die zugehörige Bedingung lautet:

"Condition: $NQ(\text{Wartemodul fuer Process 2.Queue}) > 0 \ \&\& \ \text{STATE}(\text{Resource2}) = = \text{IDLE_RES}$ "

Das doppelte Gleichheitszeichen wird benutzt, da ein logischer Operator nötig ist, um den Vergleich einzufügen. Ein einfaches Gleichheitszeichen dient der Zuweisung eines Wertes. Das doppelte &-Zeichen wird verwendet, um ein „und“ als Konjunktion einzusetzen. Die Funktion STATE ist in ARENA bereits implementiert und kann genutzt werden, um den aktuellen Zustand einer Ressource anzuzeigen. STATE liefert die Zahl zurück, die der Position des Zustands der Ressource im Stateset entspricht. Das Stateset ist als Liste vorgegeben, die die während der Simulation möglichen Zustände der Ressource umfasst. IDLE_RES kann, als eine der vier in ARENA vorhandenen „Resource State“-Konstanten benutzt werden, um eine Ressource auf einen leeren Zustand zu prüfen. Eine Ressource wird nur dann als leer angesehen, wenn alle Einheiten dieser Ressource frei sind und sie nicht defekt oder inaktiv ist. Der Zustand inaktiv liegt vor, wenn die Maschine z.B. aufgrund von Zeitplänen zu bestimmten Zeiten nicht genutzt werden kann.

Ist die Bedingung nicht erfüllt, wartet die Prüf-Entität weiter in der Warteschlange „Pruef Entity wartet bis Process 2 Idle.Queue“, bis die Bedingung erfüllt ist und die Prüf-Entität zum nächsten Entscheidungs-Modul „Lager an Station 3 nicht voll?“ vorgelassen wird. In diesem Modul vom Typ „2-way by condition“ wird ermittelt, ob die Anzahl wartender Produkte in der Warteschlange vor Prozess 3 (Process 3.Queue) kleiner ist als der Wert der Variablen *Lagerkapazitaet3* (Abb. 3.12):

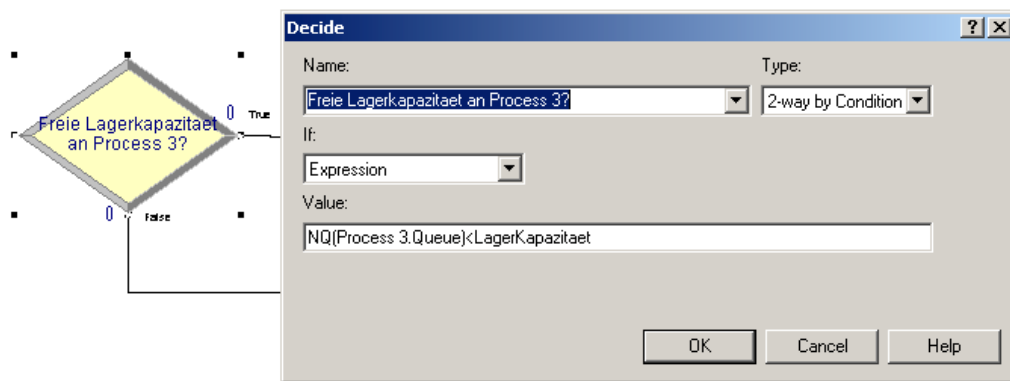


Abb. 3.12: Dialogfenster des Entscheidungs-Moduls

In der Bedingung: "If: Expression; Value: $NQ(\text{Process 3.Queue}) < \text{Lagerkapazitaet}$ " wird das strikte Ungleichheitszeichen gewählt, um zu vermeiden, dass die Produkt-Entität, die im betrachteten Zeitpunkt an Station 2 eingelastet werden soll, früher fertiggestellt wird als die an Station 3 bearbeitete Entität. Entfällt diese Bedingung, verlässt eine Entität Station 2, trifft in dem noch ausgefüllten Lager an Station 3 ein und verletzt die Kapazitätsrestriktion. Es ist daher zu prüfen, ob die Produkt-Entitäten vor Prozess 2 aufgehalten werden müssen. Ist die Ungleichung erfüllt, können die Produkte an Station 2 weiterbearbeitet werden und die Prüf-Entität wird dem „True“-Pfad folgen und dort in das nächste Modul „Suche erste Einheit“ gelangen. Andernfalls müssen die Entitäten weiter in der Warteschlange vor Prozess 2 aufgehalten werden und die Prüf-Entität wird in das nächste Modul „Suche Einheit die nach Process 2 nicht auf Process 3 geht“ auf den „False“-Pfad geschickt. Dieses Such-Modul überprüft, ob in einer vorgegebenen Warteschlange Entitäten auftreten, die eine gegebene Bedingung erfüllen. Die hier zu untersuchende Schlange heißt „Wartemodul fuer Process 2.Queue“ (Abb. 3.13).

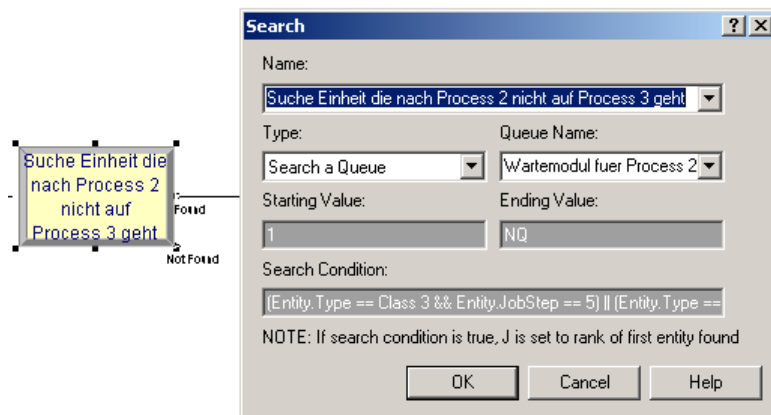


Abb. 3.13: Dialogfenster des Such-Moduls

Mit Hilfe der Eintragungen in den Feldern „Starting Value“ und „Ending Value“ des Dialogfensters kann der zu durchsuchende Teil der Warteschlange definiert werden. Die Suche beginnt bei der Entität, die sich an der J -ten Position – also an der Position, die durch den Wert des „Starting Value“ beschrieben wird – befindet, und wird der Reihenfolge nach bis zu dem Schlangenelement fortgesetzt, dessen Rang den Wert des „Ending Value“ aufweist. Hier werden die Entitäten der gesamten Warteschlange „Wartemodul fuer Process 2.Queue“ der Reihe nach auf Entitäten geprüft, die die Suchbedingung erfüllen. Sobald eine Entität gefunden wird, die die Suchbedingung erfüllt, nimmt der Index J den Wert des Ranges dieser Entität an und der Suchvorgang wird abgebrochen. Der „Starting Value“ kann kleiner oder größer sein als der „Ending Value“, je nachdem, ob die Suche von vorne oder hinten beginnen soll.

Wie dem Feld „Search Condition“ in Abbildung 3.13 zu entnehmen ist, lautet die Suchbedingung, dass die betrachteten Entitäten dem Produkt-Typ 3 angehören und sich entweder im 5. oder im 13. Bearbeitungsschritt befinden müssen (\parallel ist ein Operator für ein inklusives „Oder“). Diese Bearbeitungsschritte beschreiben die oben bereits angesprochenen Ausnahmen, da die Entitäten nach der Bearbeitung an Station 2 nicht im nächsten Schritt Station 3 erreichen, sondern den übrigen Entitäten in der Warteschlange vorgezogen werden. Erfüllt keine Entität die Suchbedingung, wird die Variable J auf Null gesetzt und die Prüf-Entität verlässt das Modul auf dem „Not Found“-Pfad. Gibt es jedoch mindestens eine Entität, die der Bedingung genügt, wird die Prüf-Entität das Modul auf den „Found“-Pfad verlassen. Auf diesem Weg gelangt es zunächst in das Greif-Modul „Entferne erste bzw nicht Process 3 Einheit aus Schlange und

gebe diese auf Process 2“. Anhand der Position in der Warteschlange entfernt das Greif-Modul eine Entität. Dem Dialogfenster in Abbildung 3.14 sind die Einstellungen zu entnehmen.

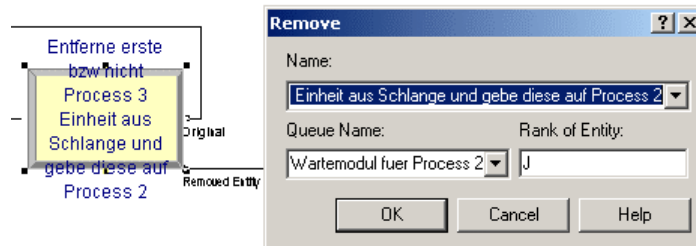


Abb. 3.14: Dialogfenster des Greif-Moduls

Bei der Warteschlange handelt es sich um „Wartemodul fuer Process 2.Queue“. Die zu entfernende Entität hat den J -ten Rang, wie im Dialog-Feld „Rank of Entity“ ersichtlich ist. Es soll die Entität aus der Schlange entfernt werden, die im vorherigen Modul ausgewählt wurde. Diese Entität wird auf dem „Removed Entity“-Pfad zu Prozess 2 geleitet. Die Prüf-Entität, die das Modul über den Pfad „Found“ des Such-Moduls betreten hat, verlässt es auf dem „Original“-Pfad (vgl. Abb. 3.14) und gelangt zu einem Verzögerungs-Modul „Pruef Entität wartet 1 Minute“, das an späterer Stelle noch erklärt wird.

Wie dem Anhang zu entnehmen ist, kann die Prüf-Entität auch über einen anderen Weg in das Greif-Modul gelangen. Hierzu muss die Ungleichungsbedingung im Entscheidungs-Modul „Lager an Station 3 nicht voll?“ erfüllt sein, d.h., dass Entitäten an Station 2 weiter bearbeitet werden können, die Prüf-Entität dem „True“-Pfad folgt und dort in das nächste Modul „Suche erste Einheit“ gelangen kann.

Dort wird die Warteschlange vor Prozess 2 nach einer Entität durchsucht, die der Bedingung $IDENT > 0$ genügt (Abb. 3.19).

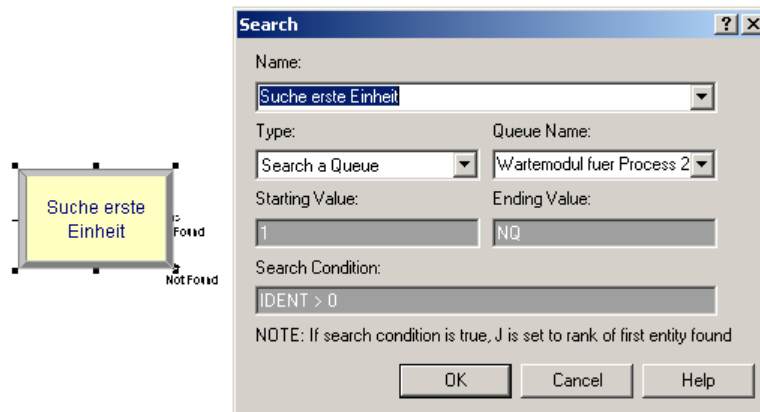


Abb. 3.15: Dialogfenster des Such-Moduls "Suche erste Einheit"

Die Funktion dieses Such-Moduls besteht in dem Ersatz eines Zuweisungs-Moduls für die Variable J . Dies ist notwendig, da im folgenden Modul „Entferne erste bzw. nicht Prozess 3 Einheit aus Schlange und gebe diese auf Process 2“ die Prüf-Entität mit dem Wert $J = 1$ benötigt wird. Diese Aufgabe kann nicht von einem Zuweisungs-Modul übernommen werden. In Zuweisungs-Modulen können zwar verschiedenen Variablen Werte zugewiesen werden, jedoch dürfen diese Veränderlichen keine vordefinierten und von ARENA reservierten Größen sein, wie z.B. die Indexvariable J .

Das beschriebene Such-Modul wird daher ‘zweckentfremdet’ verwendet. Der Pfad „Not Found“ bleibt ungenutzt, erscheint aber, da er für die primäre Funktion eines allgemeinen Such-Moduls unerlässlich ist. Im Such-Modul wird die in ARENA vorgegebene Funktion IDENT genutzt. Jeder Entität wird bei ihrer Erstellung eine Identitätsnummer (IDENT) zugewiesen. Da Entitäten nur mit natürlichen Zahlen, ausgenommen Null, versehen werden, ist die Bedingung $IDENT > 0$ für alle Entitäten in der Schlange erfüllt. Auch in diesem Such-Modul startet die Suche mit dem ersten Element der Warteschlange. Die erste Entität in der Warteschlange vor Prozess 2 wird auf die Bedingung geprüft, die in jedem Fall erfüllt ist, so dass J den Wert Eins erhält.

Die Prüf-Entität gelangt auf diesem Weg zum Modul „Entferne erste bzw. nicht Process 3 Einheit aus Schlange und gebe sie auf Process 2“, das bereits oben beschrieben wurde.

Es ist noch auf die Vorgehensweise für den Fall einzugehen, dass die erste Einheit entfernt werden soll, d.h. die Prüf-Entität im letzten Schritt das Modul „Suche erste Einheit“ passiert. Dort wurde dem Index J der Wert Eins zugewiesen. Das Greif-Modul entfernt in der Warteschlange „Wartemodul fuer Process 2.Queue“ die erste Einheit und schickt diese auf dem „Removed Entität“-Pfad zu Prozess 2. Die Prüf-Entität verlässt das Greif-Modul über den „Original“-Ausgang und wird zum Verzögerungs-Modul „Pruef Entität wartet 1 Minute“ geschickt (Abb. 3.16).

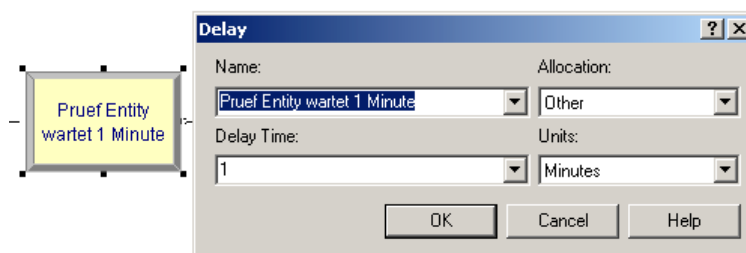


Abb. 3.16: Dialogfenster des Verzögerungs-Moduls

Dieses Verzögerungs-Modul wird benötigt, um die Prüf-Entität für eine kurze Zeitspanne aufzuhalten. Da die Prüf-Entität den Weg durch die bereits beschriebenen Module in Null Zeiteinheiten zurücklegt, gelangt sie im selben Zeitpunkt zu dem Verzögerungs-Modul, wie sie im Halte-Modul „Pruef Entität wartet bis Process 2 Idle“ startet. Wird auf das Verzögerungs-Modul verzichtet, gerät das System in eine Endlosschleife. Sie muss daher um ε Zeiteinheiten verzögert werden. Da in den Prozessen die Zeitdimension Stunden gewählt ist, genügt hier eine Verzögerung um eine Minute; kürzere Verzögerungen führen zu unnötig vielen Zyklen der Prüf-Entitäten.

Zum Ende dieses Kapitels soll noch einmal auf die geänderte Bedingung in dem zuerst beschriebenen Halte-Modul „Wartemodul fuer Process 2“ eingegangen werden. Die eingehenden Produkt-Entitäten müssen an dieser Stelle in jedem Fall festgehalten werden, um das geplante Vorgehen des Greif-Moduls „Entferne erste bzw. nicht Process 3 Einheit aus Schlange und gebe diese auf Process 2“ zu ermöglichen. Es muss gewährleistet sein, dass alle Produkt-Entitäten den Prozess 2 über das Greif-Modul erreichen, um eine vollständige Überprüfung der genannten Abfragen zu garantieren.

3.4.1.2 Variation der Lagerkapazität und deren Auswirkungen auf die Mittlere Durchlaufzeit

Mit Hilfe des Process Analyzer werden im Anschluss an die Implementierung des Modells verschiedene Szenarien simuliert und die Ergebnisse der Simulationsläufe ausgewertet. Die Einstellungen für die einzelnen Szenarien finden sich im Grid-Window (Abb. 3.17), die über die Kategorie „Scenario“ in der Project-Bar des Process Analyzers zu erreichen sind.

	Scenario Properties				Controls		Responses		
	S	Name	Program File	Reps	Lagerkapazität	WIP	Cycle Time Class 1	Cycle Time Class 2	Cycle Time Class 3
1		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	1	108	653.280	964.175	981.896
2		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	2	108	650.504	960.896	978.650
3		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	3	108	649.928	960.169	979.111
4		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	4	108	650.793	961.391	980.268
5		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	5	108	651.292	961.754	980.879
6		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	6	108	648.236	957.707	976.367
7		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	7	108	651.957	963.836	982.978
8		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	8	108	652.862	965.121	983.375
9		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	9	108	651.016	962.502	980.488
10		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	10	108	652.986	964.773	983.562
11		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	108	108	650.697	961.002	980.618
12		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	1	162	973.951	1446.331	1463.498
13		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	2	162	973.717	1446.606	1464.647
14		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	3	162	975.942	1452.199	1468.864
15		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	4	162	975.187	1449.336	1467.463
16		Scenario 1	2 : P-L-K 2003-09-11 20-34.p	20	5	162	972.407	1446.483	1464.937

Abb. 3.17: Ausschnitt eines Grid-Window

Es werden 198 Szenarien mit jeweils 20 Replikationen gewählt. Die Szenarien ergeben sich aus der Kombination von 11 verschiedenen Lagerkapazitäten und 18 unterschiedlichen WIP-Werten. Die Level des Faktors *Lagerkapazität* variieren von 1 bis 10. Ergänzend wird *Lagerkapazität* auf den WIP-Level gesetzt und damit die Situation unbegrenzter Lager erfasst. Hinsichtlich des Faktors *WIP* werden Vielfache des Wertes 54 beginnend mit 108 (108, 162, 216, 270, 324, 378, 432, 486, 540, 594, 648, 702, 756, 810, 864, 918, 972, 1026) betrachtet. Es handelt sich hier um ein [11 x 18]-Design mit quantitativen Faktoren (vgl. FISHMAN 1978; MYERS/MONTGOMERY 1995; MONTGOMERY 1997). Die hohe Anzahl an Replikationen wird gewählt, um mögliche Störgrößen zu kontrollieren (vgl. KLEIJNEN 1987). Durchgeführt wird ein Full Factorial Design, d.h. alle möglichen Faktorlevel-Kombinationen werden simuliert.

Da in diesem Abschnitt nur die Auswirkung der variierenden Lagerkapazitäten auf die Mittlere Durchlaufzeit betrachtet werden soll, ist der Einfluss zufälliger Schwankungen in den Bearbeitungszeiten auszuschließen. Diese sollen zwar weiterhin einer Exponentialverteilung unterliegen, jedoch zwischen den einzelnen Szenarien nicht variieren. Durch die Vorgabe eines Seeds für den Zufallszahlengenerator kann erreicht werden, dass in jedem Szenario für die Bearbeitungszeiten der gleiche Strom von Zufallszahlen genutzt wird (ausführlich dazu HERRMANN et al. 2003, S. 61f.).

Die Simulation der Szenarien und Replikationen wird durch den Process Analyzer automatisch nach entsprechender Programmierung durchgeführt, so dass dem Benutzer lediglich die Aufgabe der Auswertung bleibt. Diese erfolgt anhand von „Charts“, die aus der Datentabelle des Grid-Windows erzeugt werden können.

In der vorliegenden Simulation interessieren besonders die Auswirkungen der Faktorlevel von *Lagerkapazität* und von *WIP* auf die Mittlere Durchlaufzeit. Die Graphen in Abbildung 3.18 suggerieren, dass die Mittlere Durchlaufzeit mit steigender Kapazität des Lagers vor Station 3, aber konstantem *WIP* für alle Produkt-Typen im Mittel nahezu konstant bleibt.

In Abbildung 3.18 ist weiter zu erkennen, dass sich die Mittlere Durchlaufzeit mit steigendem *WIP* um einen annähernd konstanten Wert erhöht. Dieser beträgt bei Produkt-Typ 1 durchschnittlich 323,318 Stunden, bei Produkt-Typ 2 484,893 Stunden und bei Produkt-Typ 3 485,757 Stunden. Die über die einzelnen Replikationen gemittelte Mittlere Durchlaufzeit der Produkt-Typen 2 und 3 verläuft nahezu identisch, während die Mittlere Durchlaufzeit für Produkt-Typ 1 in jedem Szenario deutlich geringer ist. Daher werden im Weiteren die Produkt-Typen 2 und 3 zu einer Produktgruppe zusammengefasst. Inhaltlich lässt sich dies auf Ähnlichkeiten in den Produktionsprozessen zurückführen (Abb. 3.1). Mit zunehmendem *WIP* wird die Differenz der Mittleren Durchlaufzeit zwischen den beiden Produktgruppen größer. Eine weitere wichtige Beobachtung ist, dass keine Interaktionseffekte auftreten. Unter einem Interaktionseffekt wird die Auswirkung eines Faktors in Abhängigkeit von einem oder mehreren anderen Faktoren verstanden (vgl. KLEIJNEN 1987, S. 199).

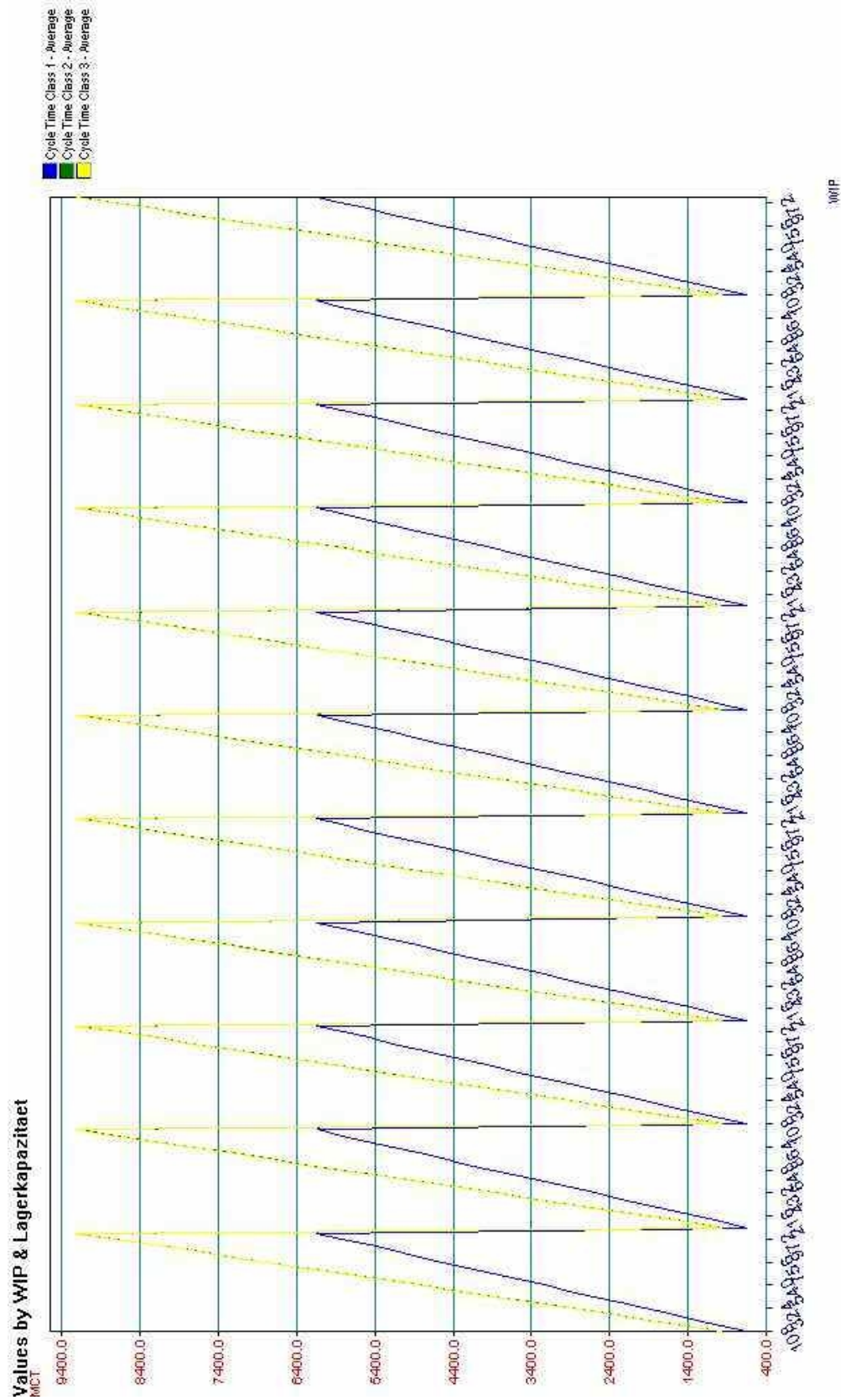


Abb. 3.18: Auswirkungen von Lagerkapazitaet3 und von WIP auf die Mittlere Durchlaufzeit

Abbildung 3.18 zeigt, dass die Graphen nahezu parallel verlaufen. Gleiches gilt in abgeschwächter Form für die Graphen in Abbildung 3.19, bei denen keine Schnittpunkte vorliegen, die auf Interaktion hindeuten. Da die vorliegenden Graphiken eindeutige Ergebnisse liefern, wird an dieser Stelle auf statistische Tests verzichtet.

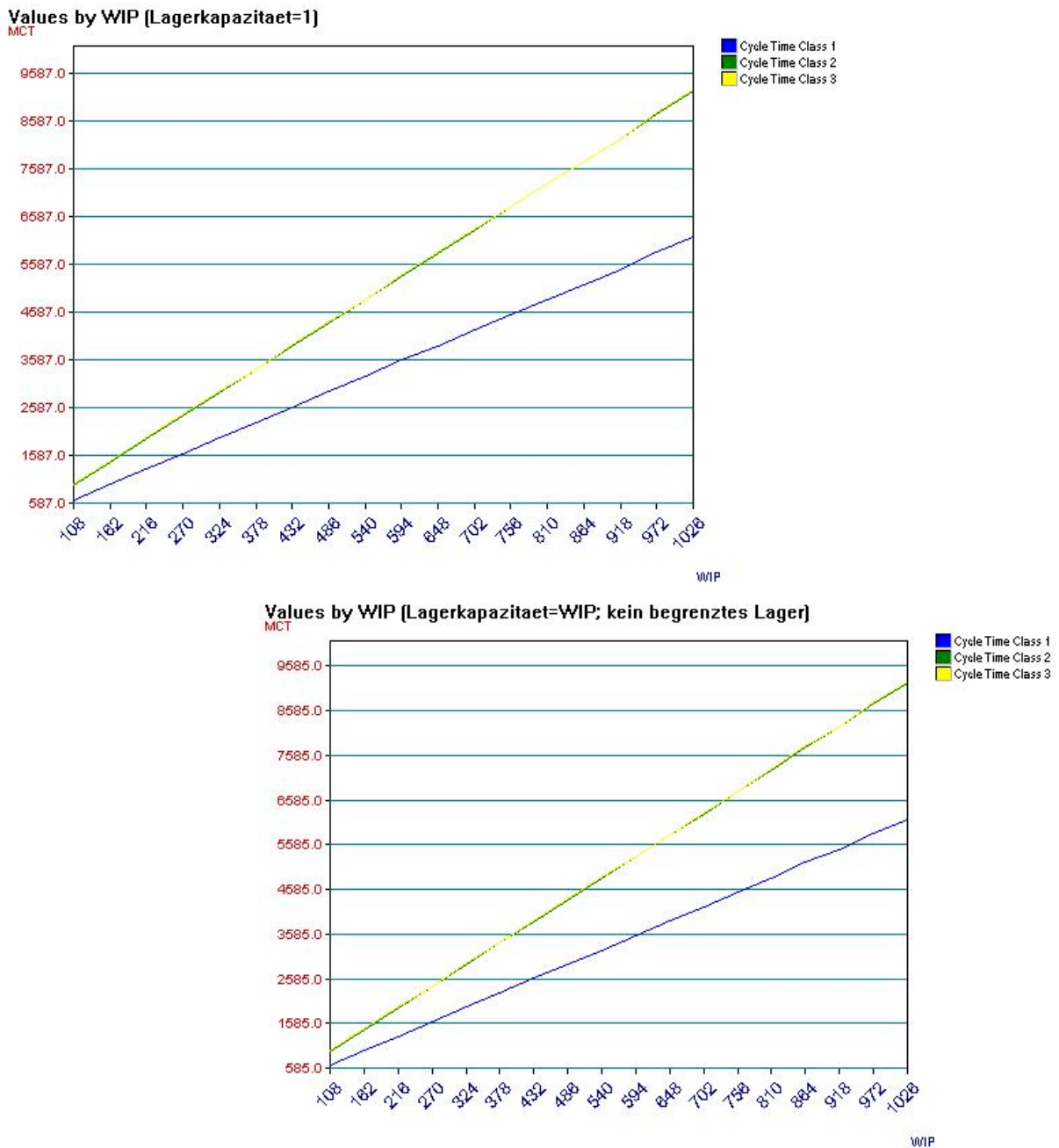


Abb. 3.19: Keine Interaktion zwischen den Produkt-Typen bei Variation von *Lagerkapazitaet3* und *WIP*

Ein Vergleich der verschiedenen *WIP*-Level bei konstanter *Lagerkapazitaet3* (Abb. 3.19) zeigt, dass die Mittlere Durchlaufzeit linear zunimmt. Es ergeben sich folgende Regressionsgraden für die einzelnen Produkt-Typen, bei denen erneut zu erkennen ist, dass die Geraden der Produkt-Typen 2 und 3 bzgl. ihrer Steigung und ihrer Ordinatenschnittpunkte ähnlich verlaufen:

Produkt-Typ 1: Mittlere Durchlaufzeit 1 = 3, 232 + 6 * *WIP*

Produkt-Typ 2: Mittlere Durchlaufzeit 2 = -9, 880 + 9 * *WIP*

Produkt-Typ 3: Mittlere Durchlaufzeit 3 = 8, 745 + 9 * *WIP*

Die von Null abweichenden Ordinatenschnittpunkte lassen sich durch die Simulation und die Approximation der Geraden erklären und sind aufgrund der großen Werte der Mittleren Durchlaufzeit als nicht signifikant einzuschätzen.

Die Werte, die dieser Geradenschätzungen zugrunde liegen, weisen bei wechselnder *Lagerkapazitaet3* nur eine geringe Streuung auf (Abb. 3.20).

Die geringfügig zunehmende Streuung bei steigendem *WIP* lässt sich durch die wachsende Anzahl der Produkt-Entitäten begründen. Die Änderung der *Lagerkapazitaet3* hat offensichtlich keinen signifikanten Einfluss auf die Mittlere Durchlaufzeit der einzelnen Produkt-Typen, wie die nahezu identischen Boxplots in der Abbildung 3.20 vermuten lassen.

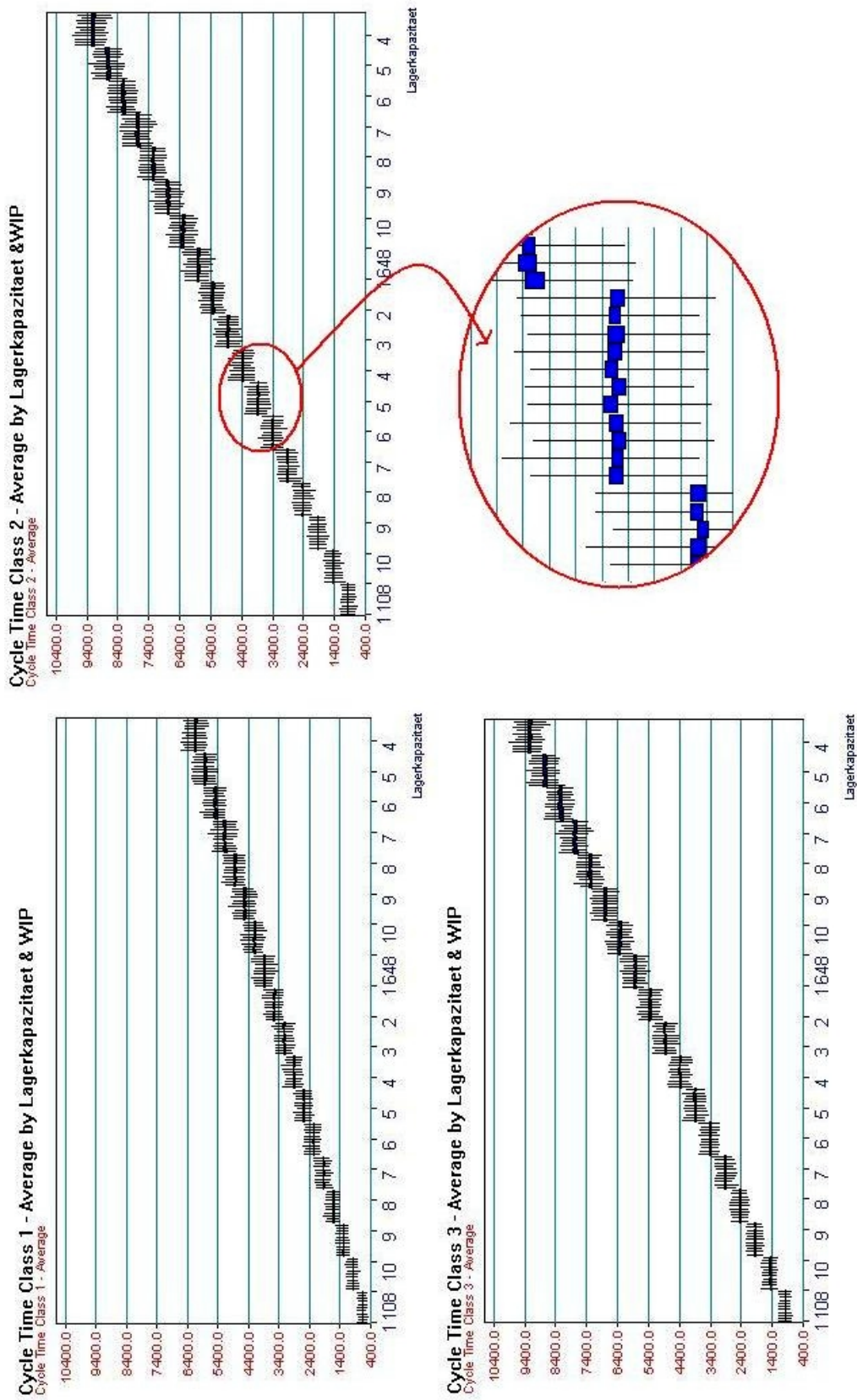


Abb. 3.20: Streuung der Mittleren Durchlaufzeiten für alle Produkt-Typen in allen Szenarien

3.4.2 Begrenzte Lagerkapazität an zwei Stationen

Im Folgenden wird das System I mit einem kapazitierten Lager um ein zusätzliches Lager an Station 5 erweitert. Wie bereits in Abschnitt 3.4.1.1 beschrieben, ist auch hier zu berücksichtigen, dass ein Rückstau vor der Station 5 nicht alle Entitäten betrifft (Abb. 3.1). Analog dem bereits dargestellten Vorgehen werden Prüf-Entitäts-Zyklen eingefügt. Zur detaillierten Beschreibung des Vorgehens kann auf HERRMANN et al. (2003, S. 68-78) verwiesen werden.

Durch den geänderten Aufbau des Modells kann kein Full Factorial Design durchgeführt werden. Es werden mit 216 und 432 zwei unterschiedliche *WIP*-Level sowie ausgewählte Lagerkapazitäten für beide Stationen betrachtet. Die Lagerkapazität an Station 3 wird weiterhin mit *Lagerkapazitaet3*, die an Station 5 mit *Lagerkapazitaet5* bezeichnet. Die Zusammensetzung der Szenarien kann Abbildung 3.21 entnommen werden.

S	Scenario Properties				Controls			Responses		
	Name	Program File	Reps	WIP	Lagerkapazitaet3	Lagerkapazitaet5	Cycle Time Class 1	Cycle Time Class 2	Cycle Time Class 3	
1	WIP=216, Lag=5, Lag5=11	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	5	11	2039.683	1127.552	3067.858	
2	WIP=216, Lag=5, Lag5=51	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	5	51	1795.689	1249.916	2694.093	
3	WIP=216, Lag=5, Lag5=101	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	5	101	1567.177	1458.720	2357.557	
4	WIP=216, Lag=5, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	5	151	1404.271	1684.046	2111.010	
5	WIP=216, Lag=5, Lag5=201	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	5	201	1311.395	1906.829	1976.266	
6	WIP=216, Lag=5, Lag5=216	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	5	216	1300.254	1937.999	1956.079	
7	WIP=216, Lag=10, Lag5=101	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	10	101	1553.248	1460.662	2337.848	
8	WIP=216, Lag=10, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	10	151	1406.506	1682.667	2118.415	
9	WIP=216, Lag=10, Lag5=201	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	10	201	1306.611	1903.306	1970.668	
10	WIP=216, Lag=10, Lag5=216	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	10	216	1295.572	1934.151	1948.994	
11	WIP=216, Lag=50, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	50	151	1390.012	1680.219	2094.622	
12	WIP=216, Lag=216, Lag5=216	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	216	216	216	1303.114	1940.644	1961.183	
13	WIP=432, Lag=5, Lag5=101	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	5	101	3623.792	2467.989	5439.955	
14	WIP=432, Lag=5, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	5	151	3370.909	2651.981	5040.254	
15	WIP=432, Lag=5, Lag5=201	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	5	201	3148.552	2862.138	4716.563	
16	WIP=432, Lag=5, Lag5=251	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	5	251	2966.847	3077.612	4460.383	
17	WIP=432, Lag=5, Lag5=301	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	5	301	2854.585	3327.468	4285.849	
18	WIP=432, Lag=5, Lag5=351	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	5	351	2682.828	3522.169	4043.517	
19	WIP=432, Lag=5, Lag5=401	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	5	401	2622.448	3797.618	3956.565	
20	WIP=432, Lag=5, Lag5=432	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	5	432	2591.984	3874.673	3893.993	
21	WIP=432, Lag=10, Lag5=101	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	10	101	3645.162	2465.850	5470.933	
22	WIP=432, Lag=10, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	10	151	3373.285	2655.179	5057.118	
23	WIP=432, Lag=10, Lag5=201	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	10	201	3158.921	2872.341	4752.788	
24	WIP=432, Lag=10, Lag5=251	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	10	251	2960.212	3094.568	4458.451	
25	WIP=432, Lag=10, Lag5=301	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	10	301	2828.104	3327.645	4252.732	
26	WIP=432, Lag=10, Lag5=351	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	10	351	2700.227	3548.977	4065.317	
27	WIP=432, Lag=10, Lag5=401	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	10	401	2625.811	3796.770	3951.768	
28	WIP=432, Lag=10, Lag5=432	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	10	432	2581.117	3858.934	3876.814	
29	WIP=432, Lag=50, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	432	50	151	3381.917	2653.484	5096.368	

Abb. 3.21: Grid-Window bei variierender Lagerkapazität

Die Szenarien werden so gewählt, dass ein Vergleich der Mittleren Durchlaufzeit die Auswirkungen der Variation der Lagerkapazitäten und des WIP widerspiegelt. Die verschiedenen Lagergrößen werden in unterschiedlichen Konstellationen mit den Levels der anderen Faktoren kombiniert. Die Anzahl der Replikationen wird von 20 auf 10 verringert. Da ein konstanter Seed gewählt wird, bleiben die Ergebnisse unverzerrt.

Wie Abbildung 3.22 zeigt, verläuft die Funktion in Abhängigkeit von dem Faktor *Lagerkapazität* für alle Produkt-Typen nahezu parallel zur Abszisse.

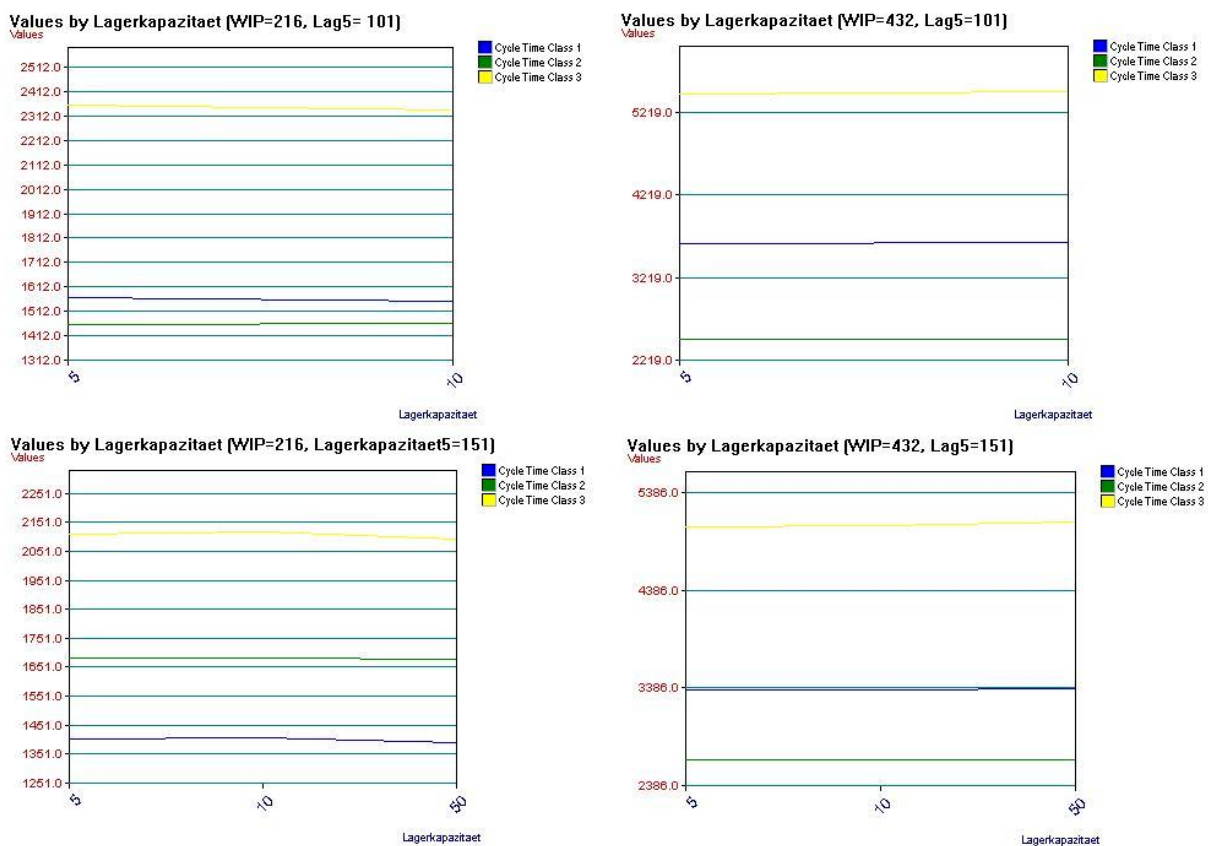


Abb. 3.22: Unabhängigkeit der Mittleren Durchlaufzeit der Produkt-Typen von der gewählten Lagerkapazität

Es bestätigt sich, dass die Mittlere Durchlaufzeit der Produkt-Typen unabhängig von der gewählten Lagerkapazität ist (vgl. Kap. 3.4.1.2). Die Ursache dafür kann in dem Verhältnis der Bearbeitungszeiten gesehen werden (Tab. 3.1). Station 1 hat eine längere Bearbeitungszeit (1,7 Std.) als Station 3 (1,4 Std.) und Station 2 (0,9 Std.) eine kürzere. Wie Abbildung 3.1

zeigt, existieren an Station 1 und 3 gleich viele Inputkanäle und alle Entitäten mit Ausnahme von zwei Produkten passieren vor Station 3 auch Station 1. Der beobachtete Verlauf der Graphen kann auf diesen „Bremseffekt“ der Station 1 zurückgeführt werden.

Ein Vergleich der Graphen für unterschiedliche *WIP*-Level zeigt, dass die Höhe der Mittleren Durchlaufzeit für alle Produkt-Typen vom *WIP* abhängt (Abb. 3.23). Je mehr Teile sich im System befinden, desto länger wird die Wartezeit des einzelnen Teils vor den Stationen.

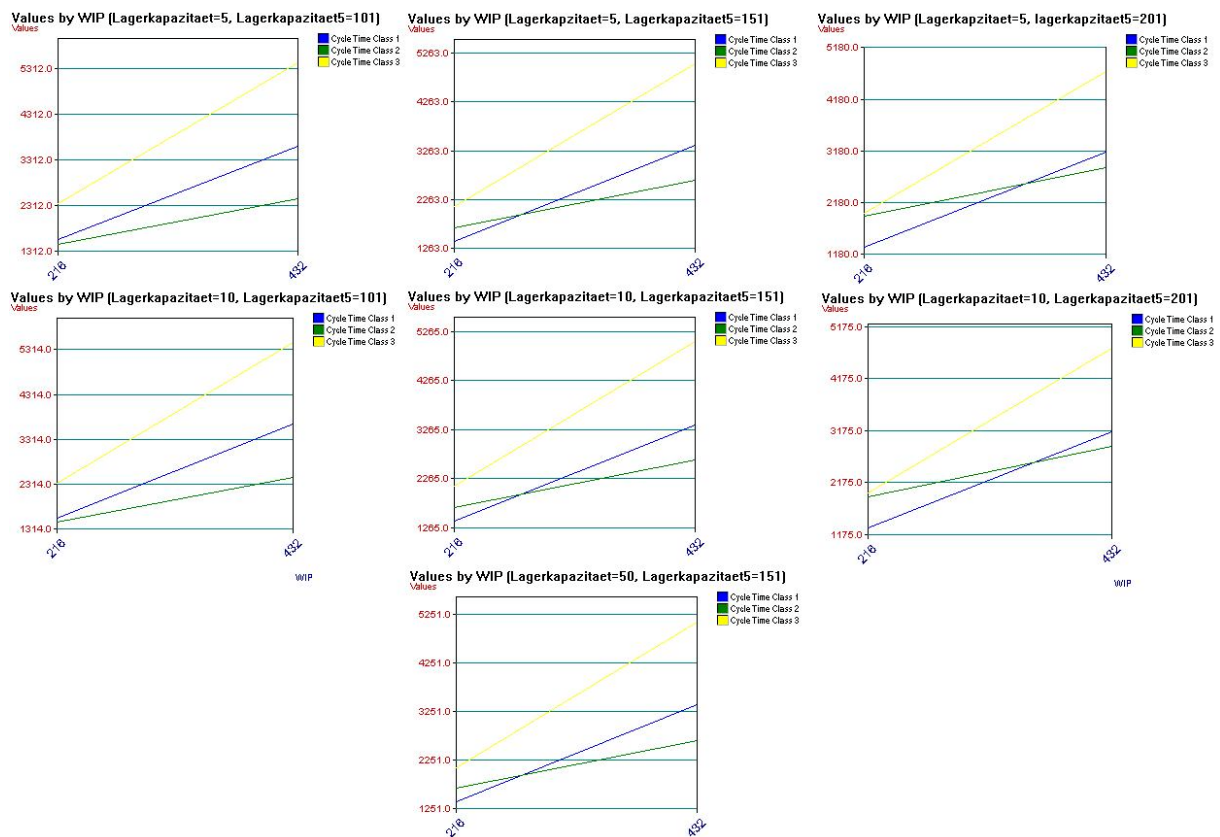


Abb. 3.23: Veränderung der Mittleren Durchlaufzeit bei Variation des *WIP*

Um Aussagen über die Veränderung der Mittleren Durchlaufzeit in Abhängigkeit von *WIP* zu erhalten, wird zunächst die Steigung zwischen den beiden Punkten der getesteten *WIP*-Level für alle Produkt-Typen bestimmt. Daraus ergeben sich die in Abbildung 3.23 dargestellten Geraden, die lediglich der Feststellung des Unterschieds der Mittleren Durchlaufzeit zwischen den Leveln dienen. In der Tabelle 3.3 sind die Steigungen aller Geraden zwischen den Punkten

für verschiedene Kombinationen von Leveln der Faktoren *Lagerkapazitaet3* und *Lagerkapazitaet5* gegenübergestellt.

Tab. 3.3: Steigungen der Graphen aus Abbildung 3.23 für ausgewählte Szenarien

<i>Lagerkapazitaet 5</i>	101	151	201
Produkt-Typ 1	9,5214	9,1048	8,5054
Produkt-Typ 2	4,6725	4,4812	4,4227
Produkt-Typ 3	14,2704	13,5613	12,6866
Produkt-Typ 1	9,6848	9,2218	8,5755
Produkt-Typ 2	4,6536	4,5024	4,4863
Produkt-Typ 3	14,505	13,6051	12,8802
Produkt-Typ 1		9,2218	
Produkt-Typ 2		4,5059	
Produkt-Typ 3		13,897	

Die Steigungen der Geraden für einzelne Produkt-Typen sind im Vergleich zwischen den Szenarien mit $WIP = 216$ und $WIP = 432$ nahezu identisch. Bei einem Vergleich der Graphiken in Abbildung 3.23 fällt weiter auf, dass sich die Werte der Mittleren Durchlaufzeit bei variierender *Lagerkapazitaet3* und konstanter *Lagerkapazitaet5* kaum verändern. Die Geraden der Produkt-Typen 1 und 2 schneiden sich unabhängig von der Lagerkapazität bei einem nahezu gleichen WIP , d.h. es existiert ein WIP , bei dem sich für Produkt-Typ 1 und für Produkt-Typ 2 die gleiche Mittlere Durchlaufzeit realisiert.

Weiter verdeutlichen die in Abbildung 3.23 gezeigten Verläufe, dass eine Erhöhung der *Lagerkapazitaet5* bei konstanter *Lagerkapazitaet3* zu einem Absinken der Mittleren Durchlaufzeit der Produkt-Typen 1 und 3 führt, wohingegen die Mittlere Durchlaufzeit des Produkt-Typs 2 ansteigt. Die sich schneidenden Graphen deuten auf Interaktionseffekte zwischen den Produkt-Typen 1 und 2 sowie zwischen den Produkt-Typen 3 und 2 hin. Bei Faktorinteraktionen kann das Incomplete Factorial Design, das hier verwendet wurde, allerdings zu falschen Ergebnissen führen (vgl. KLEIJNEN 1987, S. 270). Die hier gefundenen Interaktionseffekte sollten daher genauer überprüft werden.

Bei einer geringen Kapazität ist das Lager an Station 5 häufiger gefüllt. Erhält die zugehörige Prüf-Entität (Prüf-Entität II) die Information, dass nur noch ein Lagerplatz frei ist, wird die erste Produkt-Entität in der Schlange vor Prozess 3 zu diesem durchgelassen. Im folgenden Schritt wird die entsprechende Prüf-Entität (Prüf-Entität III) feststellen, dass die Kapazität des Lagers vor Station 5 ausgeschöpft ist. Daher werden keine Produkte zur Bearbeitung vorgelassen, die im Anschluss auf freie Plätze in diesem Lager angewiesen sind. Dies kann wiederholt in kürzester Zeit erfolgen, so dass Produkte, die von Station 3 direkt zu 5 gelangen, schneller fertig gestellt werden als andere. Diese Produkte gehören ausschließlich dem Produkt-Typ 2 an. Bei steigender Kapazität des Lagers vor Station 5 und konstantem WIP füllt sich das Lager dagegen langsamer. Die Wahrscheinlichkeit, dass die Prüf-Entität III ein gefülltes Lager an Station 5 vorfindet, sinkt.

In Abbildung 3.24 und 3.25 werden die Werte für die Mittlere Durchlaufzeit bei einem WIP-Level von 216 und 432 und konstanter Lagerkapazität 3 von 5 bzw. 10 in Abhängigkeit von einer variierenden Lagerkapazität 5 abgetragen.

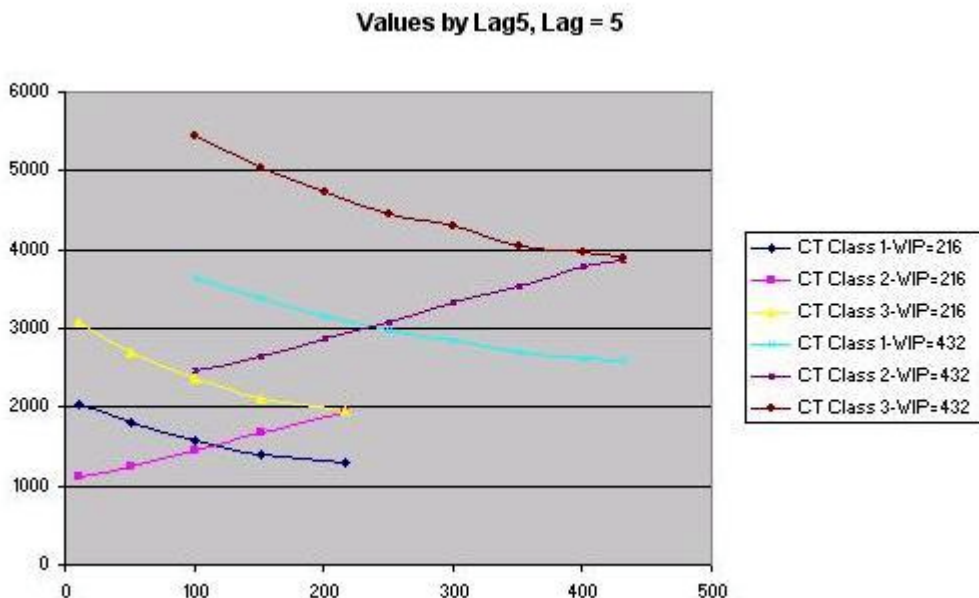


Abb. 3.24: Mittlere Durchlaufzeit bei variierender Lagerkapazität 5 für zwei konstante WIP-Level und Lagerkapazität 3 = 5

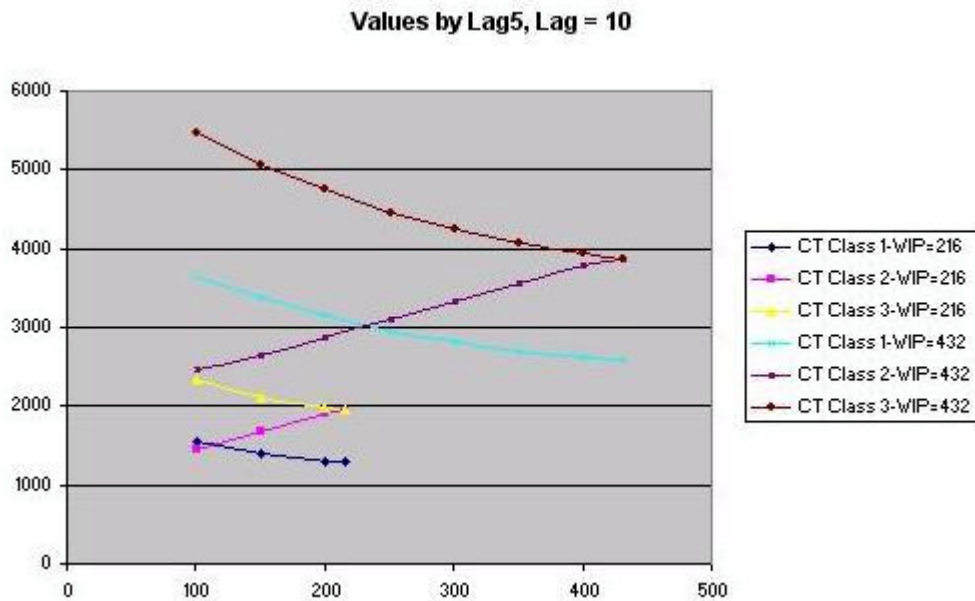


Abb. 3.25: Mittlere Durchlaufzeit bei variierender *Lagerkapazität 5* für zwei konstante *WIP-Level* und *Lagerkapazität 3 = 10*

Die sich bei einem WIP von 216 einstellenden Verläufe liegen unterhalb derer, die sich bei einem WIP von 432 ergeben. Dies lässt sich durch die längere Wartezeit der einzelnen Teile vor den Stationen erklären. Weiter ist festzustellen, dass die Werte für Produkt-Typ 2 und 3 bei unbegrenzter Lagerkapazität an Station 5 für beide WIP-Level nur geringfügig voneinander abweichen.

Werden die auf den Achsen abgetragenen Werte relativiert, indem *Lagerkapazität 3 = WIP* auf 100% und die Mittlere Durchlaufzeit des Produkt-Typs 3 bei unbegrenzter *Lagerkapazität 5* auf 100 gesetzt wird, ergeben sich die Abbildung 3.26 und 3.27.

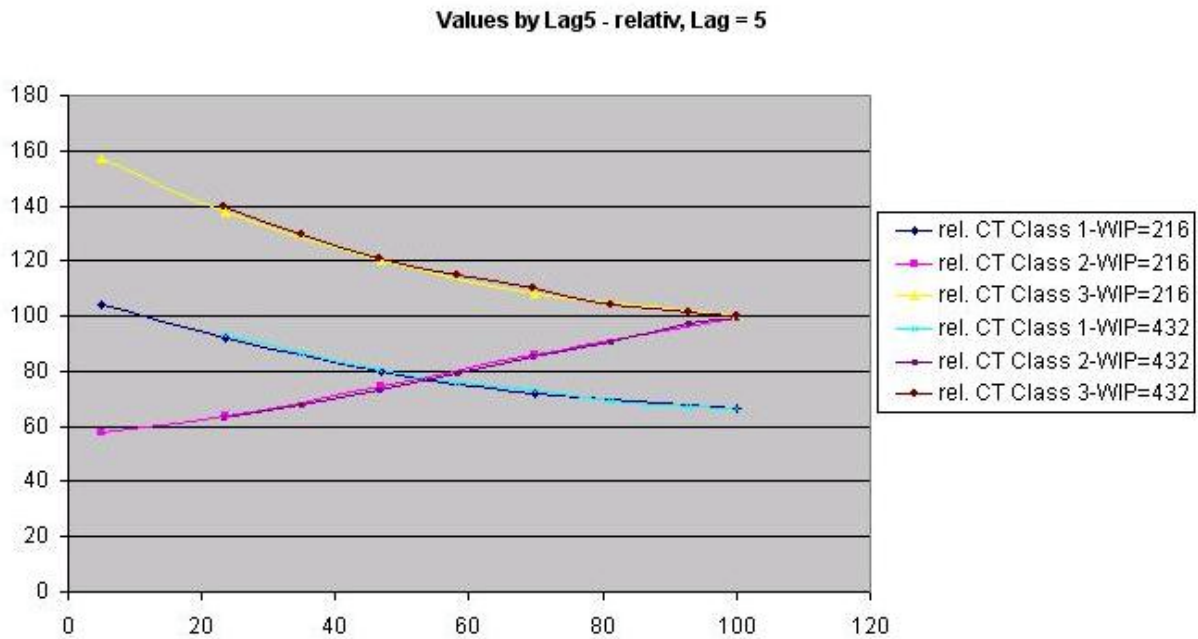


Abb. 3.26: Relativierte Mittlere Durchlaufzeit bei variierender *Lagerkapazitaet* 5 für zwei konstante *WIP*-Level und *Lagerkapazitaet* 3 = 5

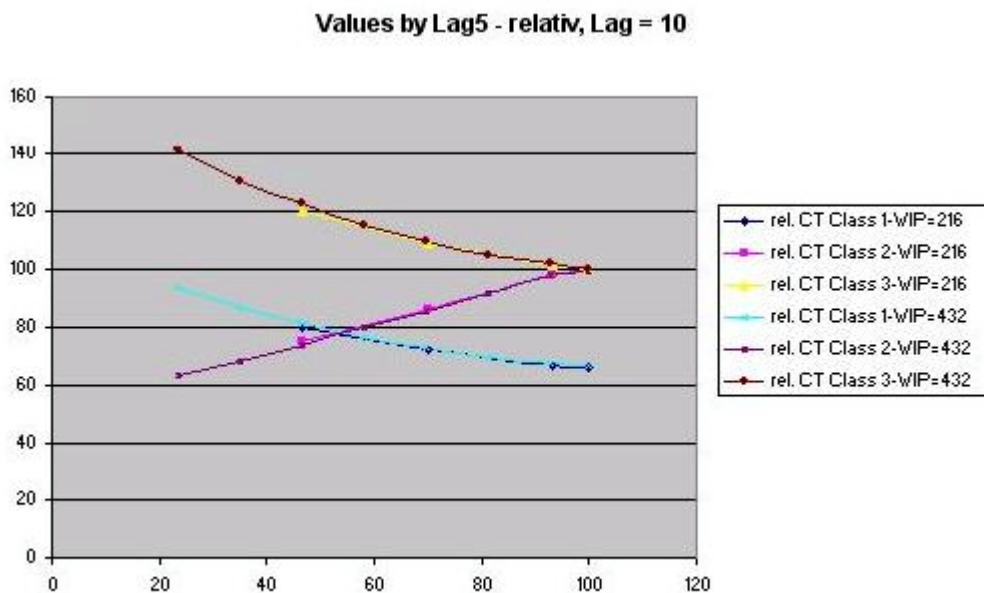


Abb. 3.27: Relativierte Mittlere Durchlaufzeit bei variierender *Lagerkapazitaet* 5 für zwei konstante *WIP*-Level und *Lagerkapazitaet* 3 = 10

Es ist deutlich zu erkennen, dass die Steigungen der relativierten Kurven für beide *WIP*-Level fast identisch sind. Dies bedeutet, dass die relative Zunahme der Mittleren Durchlaufzeit für alle *WIP*-Level gleich ist und damit die Wartezeit an Station 5 in einem konstanten Verhältnis zunimmt. Bei gleicher Bearbeitungszeit an den Maschinen steigt die Mittlere Durchlaufzeit proportional. Dieser Zusammenhang scheint für alle Level des Faktors *Lagerkapazitaet3* gegeben zu sein. Diese Vermutung wird durch die bereits interpretierte Abbildung 3.22 bestärkt, die den kaum vorhandenen Einfluss der Lagerkapazität auf die Mittlere Durchlaufzeit widerspiegelt.

Es ist allerdings fraglich, ob dieser Zusammenhang generell gilt. Um dies zu untersuchen, werden weitere ausgewählte Szenarien zu einem *WIP*-Level von 324 simuliert (Abb. 3.28).

	Scenario Properties				Controls			Responses		
	S	Name	Program File	Reps	WIP	Lagerkapazität	Lagerkapazität5	Cycle Time Class 1	Cycle Time Class 2	Cycle Time Class 3
30	🔥	WIP=324, Lag=5, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	324	5	101	2569.338	1941.055	3857.791
31	🔥	WIP=324, Lag=5, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	324	5	151	2377.598	2184.336	3572.765
32	🔥	WIP=324, Lag=5, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	324	5	201	2184.069	2388.041	3286.396
33	🔥	WIP=324, Lag=5, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	324	5	251	2057.540	2610.951	3089.968
34	🔥	WIP=324, Lag=5, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	324	5	324	1944.166	2897.670	2922.928
35	🔥	WIP=324, Lag=10, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	324	10	151	2354.483	2154.011	3536.591
36	🔥	WIP=324, Lag=50, Lag5=151	1 : P-L-G 2003-09-17 21-33 nur 10 Rep.p	10	324	50	151	2367.627	2166.012	3555.333

Abb. 3.28: Grid-Window für *WIP* = 324

Die Ergebnisse sind in Abbildung 3.29 dargestellt. Werden die Werte wie oben beschrieben relativiert, zeigt sich, dass die Graphen für verschiedene *WIP*-Level übereinander liegen.

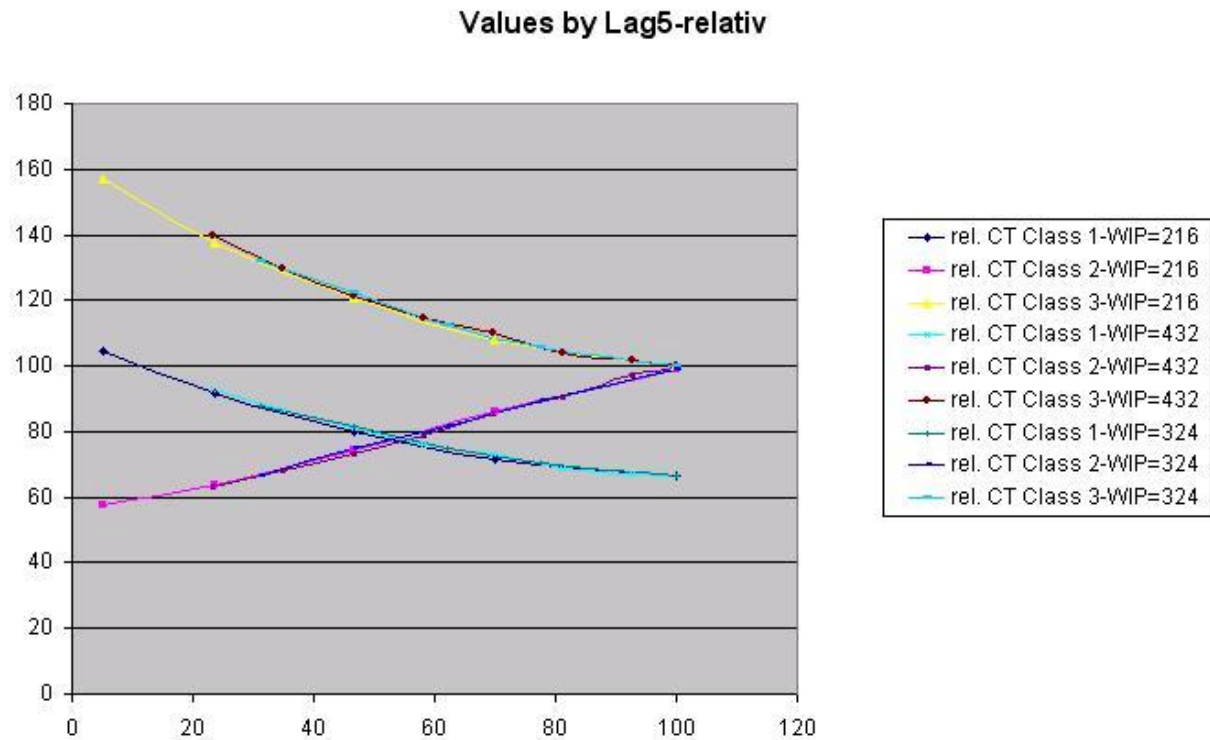


Abb. 3.29: Gegenüberstellung der relativierten Verläufe

Der Zusammenhang zwischen der Mittleren Durchlaufzeit und einem steigendem WIP bei unbegrenztem Lager an Station 5 ist linear. Interessant wäre die Ermittlung einer Formel oder eines Faktors, um die Verschiebung in Abhängigkeit des *WIP*-Levels wiederzugeben. Es wird daher eine Regressionsgerade mit Hilfe der Kleinste-Quadrate-Methode berechnet, die die Ermittlung der Mittleren Durchlaufzeit für verschiedene *WIP*-Level bei unbegrenzter Kapazität des Lagers vor Station 5 ermöglicht.

Regressionsgerade: $Y = \beta X + \alpha$

$$\text{Produkt-Typ 3: } \beta = \frac{\sum_{i=1}^n (x_i - \bar{x}) \cdot (y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} = 8,9718$$

$$\bar{y} = \beta x + \alpha$$

$$\alpha = 2924,3333 - (8,9718 * 324) = 17,4701$$

$$Y = 17,4701 + 8,9718 * WIP$$

$$\begin{aligned} \text{Produkt-Typ 2: } & \beta = 8,9661; \alpha = -1,5691 \\ & Y = -1,5691 + 8,9661 * WIP \end{aligned}$$

$$\begin{aligned} \text{Produkt-Typ 1: } & \beta = 5,9802; \alpha = 7,8832 \\ & Y = 7,8832 + 5,9802 * WIP \end{aligned}$$

Weiterhin werden Regressionsgeraden zur Approximation der Funktionsverläufe in Abbildung 3.29 bestimmt, so dass für verschiedene *WIP*-Level und unterschiedliche Kapazitäten des Lagers vor Station 5 für die einzelnen Produkt-Typen die Mittlere Durchlaufzeit geschätzt werden kann.

Wenn es möglich ist, die Graphen der Abbildung 3.29 so zu verschieben, dass sie für verschiedene *WIP*-Level aufeinander liegen, besteht eine Systematik in der Steigung und der Zunahme der absoluten Mittleren Durchlaufzeit in Abhängigkeit vom *WIP*-Level. Im Umkehrschluss muss es möglich sein, die Ergebnisse zu schätzen, womit eine Simulation weiterer Szenarien überflüssig wird. Diese Überlegung liegt der Entwicklung eines Excel-Spreadsheets zugrunde. Zur Schätzung der Mittleren Durchlaufzeiten aller Produkt-Typen ist lediglich die Eingabe des gewünschten *WIP*-Levels und der geplanten Kapazität des Lagers vor Station 5 erforderlich.

Durch Regressionsanalyse der Punktwolken kann für jeden Produkt-Typ die Steigung der Regressionsgeraden β und der Ordinatenschnittpunkt α bestimmt werden. Dabei ergeben sich folgende Gleichungen, die sich auf die relativierten Werte beziehen:

$$\text{Produkt-Typ 1: } \text{rel.MCT}(WIP, \text{Lag}5, \text{Lag} = 5) = 100,668 - 0,374 * (\text{Lag}5 / WIP * 100)$$

$$\text{Produkt-Typ 2: } \text{rel.MCT}(WIP, \text{Lag}5, \text{Lag} = 5) = 53,112 + 0,462 * (\text{Lag}5 / WIP * 100)$$

$$\text{Produkt-Typ 3: } \text{rel.MCT}(WIP, \text{Lag}5, \text{Lag} = 5) = 151,015 - 0,558 * (\text{Lag}5 / WIP * 100)$$

wobei *Lag*5 die *Lagerkapazitaet*5 und *Lag*=5 die *Lagerkapazitaet*3=5 bezeichnet.

Die absolute Mittlere Durchlaufzeit ergibt sich für alle Produkt-Typen aus dem Basiswert und der relativierten Mittleren Durchlaufzeit:

$$MCT(WIP, Lag5, Lag = 5) = rel.MCT * Basiswert / 100$$

Dabei gibt der Basiswert den Punkt (100,100) in Abbildung 3.29 an, der der relativen Mittleren Bearbeitungszeit des Produkt-Typs 3 bei unbegrenzter Lagerkapazität an Station 5 entspricht.

Wie bereits festgestellt wurde, übt die Kapazität des Lagers vor Station 3 keinen Einfluss auf die Höhe der Mittleren Durchlaufzeit aus. Es wird daher davon ausgegangen, dass die beschriebenen Berechnungen unabhängig vom Level des Faktors *Lagerkapazitaet3* Gültigkeit besitzen. Anhand der bekannten Werte für $WIP = 324$ können der Funktionsverlauf in Abbildung 3.30 und die berechneten Steigungswerte in Tabelle 3.4 geprüft werden. Abbildung 3.30 zeigt einen linearen Verlauf der Funktionen.

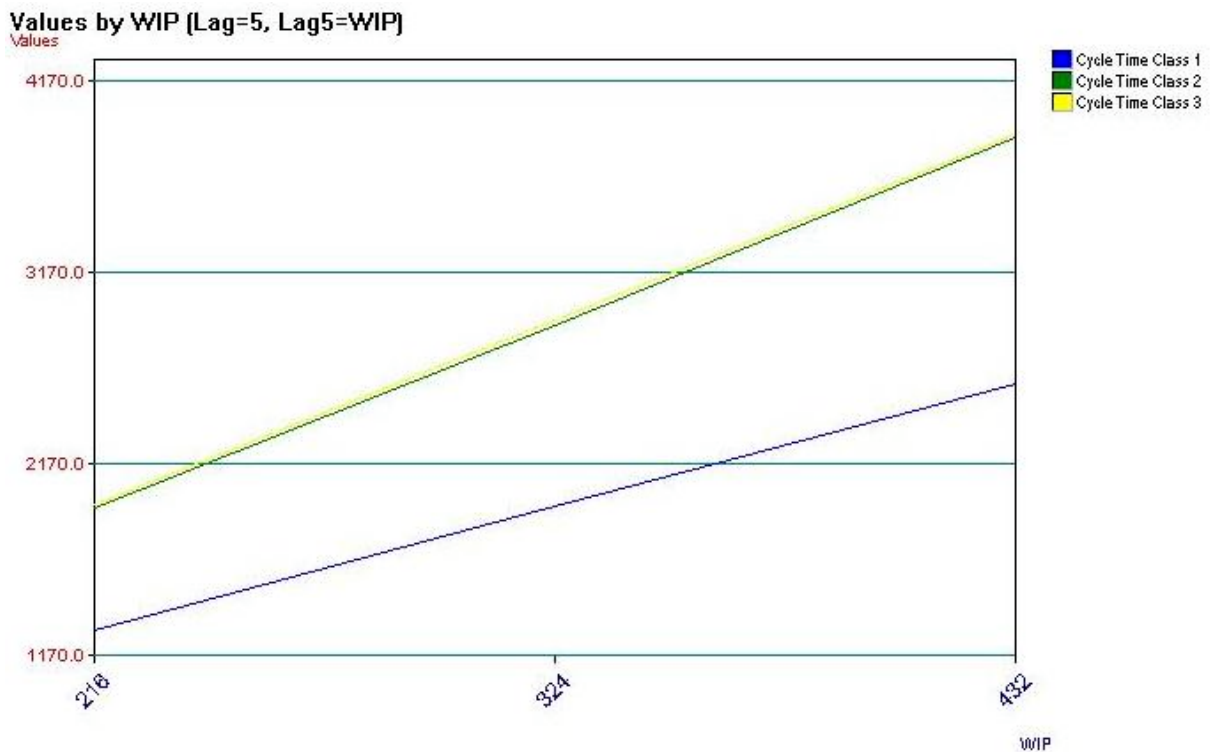


Abb. 3.30: Mittlere Durchlaufzeit bei Variation des *WIP*

Tabelle 3.4 liefert einen Vergleich der Werte für die Mittlere Durchlaufzeit, die mittels der geschätzten Steigung für einen *WIP*-Level von 324 berechnet werden (Spalte 1) mit den Ergebnissen der Simulation (Spalte 2).

Tab. 3.4: Gegenüberstellung der errechneten mit den simulativ ermittelten Mittleren Durchlaufzeiten

	Mittlere Durchlaufzeit		Abweichungen	
	errechnet	simuliert	absolut	relativ
Produkt-Typ 1	2595,4882	2569,338	26,1502	1,0075
Produkt-Typ 2	1963,35	1941,055	22,295	1,1355
Produkt-Typ 3	3898,7602	3857,791	40,9692	1,0508
Produkt-Typ 1	2387,58946	2377,598	9,9914	0,4185
Produkt-Typ 2	2168,0156	2184,336	-16,3204	0,7472
Produkt-Typ 3	3575,6304	3572,765	2,8654	0,0801
Produkt-Typ 1	2229,9782	2184,069	45,9092	2,0587
Produkt-Typ 2	2384,4806	2388,041	-3,5604	0,1491
Produkt-Typ 3	3346,4188	3286,396	60,0228	1,7936

In der letzten Spalte sind die relativen Abweichungen angegeben. Die maximale Abweichung liegt bei 2,059%, die mittlere bei 0,938% und die minimale Abweichung bei 0,149%, womit bestätigt wird, dass die für den Zusammenhang verwendete lineare Approximation gut geeignet ist.

4. Fazit

Zur Validierung der Vorgehensweise wird zunächst das Fab 1.24 von WEIN (1988) in ARENA nachgebildet. Der Vergleich der Simulationsdaten mit den Ergebnissen aus der Literatur zeigt, dass sich bei Verwendung der standardmäßig verfügbaren Störungsart WAIT erhebliche Differenzen zwischen den Modellen ergeben. Diese können durch explizites Modellieren der Störung insoweit behoben werden, dass die Werte für die Maschinenauslastung den von WEIN (1988) ermittelten Werten weitestgehend entsprechen.

Im Anschluss wird die Mehrprodukt-Reentrant-Line von PARK et al. (2002) untersucht und zunächst das System I ohne Batch-Stationen implementiert. In einem nächsten Schritt wird Station 3 zu einer Batch-Station umgewandelt. Durch Simulation dieses Modells gelingt es, die vorgegebenen Werte von PARK et al. (2002) zu bestätigen.

Als neuen Untersuchungsgegenstand greift diese Arbeit erstmals die Frage auf, ob sich strukturelle Zusammenhänge aufdecken lassen, wenn in einer Mehrprodukt-Reentrant-Line die Lager kapazitiert sind. In einer Simulationsstudie werden die Auswirkungen der Variation der Lagergrößen ermittelt und auf mögliche Ursachen zurückgeführt. Dabei kann ein linearer Zusammenhang aufgedeckt werden, der es ermöglicht, die Mittlere Durchlaufzeit für verschiedene WIP-Level und Lagergrößen an den Stationen 3 und 5 zu berechnen. Ohne weitere Simulationsläufe durchführen zu müssen, können Werte für andere Einstellungen ermittelt werden. Eine Gegenüberstellung der durch Simulationsläufe ermittelten Werte mit den errechneten Werten belegt abschließend die hohe Güte dieser Approximation.

Literaturverzeichnis

- Arzi, Y., Raviv, D., Dispatching in a Workstation Belonging to a ReEntrant Production Line under Sequence-Dependent Set-Up Times, *Production Planning and Control* 9, 1998, S. 690-699
- Baker, K.R., *Introduction to Sequencing and Scheduling*, John Wiley & Sons, New York, 1974
- Banks, J., Dai, J.G., *Simulation Studies of Multiclass Networks*, Georgia Institute of Technology, *IIE Transactions* 29, 1997, S. 213-219
- Chen, H, Yao, D.D., *Fundamentals of Queueing Networks*, New York, Springer, 2001
- Dai, J.G., Yeh, D.-H., Zhou, C., The QNET method for re-entrant queueing networks with Priority disciplines, *Operations Research* 45, 1997, S. 610-623
- Fahrmeir, L., Künstler, R., Pigeot, I., Tutz, G., *Statistik*, Springer-Verlag, Berlin/Heidelberg/ New York, 3. Aufl., 2001
- Fishman, George S., *Principles of Discrete Event Simulation*, John Wiley & Sons, New York, 1978
- Glover, F., A Template for Scatter Search and Path Relinking, *Lecture notes in Computer Science* Nr. 1363, 1998, S. 13-27
- Hammer, P.L., Glover, F., Laguna, M., *Anal. of Operations Research* 41, 1993
- Herrmann, J., Kathöfer, V., Peterwerth, A., Pfirrmann, W., *Abschlussbericht der Gruppe II*, Universität Bielefeld, Fakultät für Wirtschaftswissenschaften, 2003
- Hsieh, B-W., Chen, C.-H., Chang, S.-C., Scheduling Semiconductor Wafer Fabrication by Using Ordinal Optimization-based Simulation, *IEEE Transactions on Robotics and Automation* 17, 2001, S. 599-608
- Jahnke, H., Biskup D., *Planung und Steuerung der Produktion*, Verlag moderne industrie AG, Landsberg, 1999
- Kelton, W.D., Sadowski, R.P., Sadowski, D.A., *Simulation with ARENA*, McGraw-Hill, Boston u.a., 2. Aufl., 2002
- Kistner, K.-P., Steven, M., Warteschlangen-Netzwerke in der hierarchischen Produktionsplanung, *OR Spektrum* 12, 1990, S. 89-101

Kistner, K.-P., Steven, M., Produktionsplanung, Physika Verlag, Heidelberg, 3. Aufl., 2001

Kleijnen, J.P.C., Statistical Tools for Simulation Practitioners, Catholic University Brabant, Tilburg, Dekker, New York, 1987

Kumar, P.R., Re-entrant Lines, Queueing Systems: Theory and Applications 13, 1993, S. 87-110

Kumar, P.R., Scheduling Queueing Networks: Stability, Performance Analysis and Design, University of Illinois Department of Electrical and Computer Engineering and the Coordinated Science Lab, Urbana, 1994

Kumar, P.R., Meyn, S.P., Stability of Queueing Networks and Scheduling Policies, IEEE Transactions on Automatic Control 40 (1995), S. 251-260

Kumar, S., Kumar, P.R., Queueing Network Models in the Design and Analysis of Semiconductor Wafer Fabs, IEEE Transactions on Robotics and Automation 17, 2001, S. 548-561

Lee, L.H., Tang, L.C., Chang, S.C., Dispatching Heuristics for Wafer Fabrication, in: Peters, B.A., Smith, J.S., Medeiros, D.J., Rohrer, M.W. (eds.), Winter Simulation Conference 2001, Arlington, VA, USA, ACM, S. 1215-1219

Lu, S.C.H., Ramaswamy, D., Kumar, P.R., Efficient Scheduling Policies to Reduce Mean and Variance of Cycle-Time in Semiconductor Manufacturing Plants, IEEE Transactions on Semiconductor Manufacturing 7, 1994, S. 374-388

Montgomery, D. C., Design and analysis of experiments, John Wiley & Sons, New York, 5. Aufl., 1997

Myers, R.H., Montgomery, D.C., Response Surface Methodology – Process and Product Optimization Using Designed Experiments, John Wiley & Sons, New York, 1995

Narahari, Y., Khan, L.M., Asymptotic loss of priority scheduling policies in closed re-entrant lines: A computational study. European Journal of Operational Research 110 (1998), S. 585-596

Park, Y., Kim, S., Jun, C., Mean value analysis of re-entrant line with batch machines and multi-class jobs, Computers and Operations Research 29, 2002, S. 1009-1024

Schlittgen, R., Einführung in die Statistik, Oldenbourg Verlag, München/Wien, 9. Aufl., 2000

Wein, L.M., Scheduling semiconductor wafer fabrication, IEEE Transactions on Semiconductor Manufacturing 1, 1988, S. 115-130

Abbildungsverzeichnis

Abb. 1.1: Reentrant Line mit 3 Stationen und 11 Lagern.....	2
Abb. 2.1: Halbleiterfertigung mit 12 Stationen nach KUMAR/KUMAR (2001)	6
Abb. 2.2: Stationenfolge im Fab 1.24.....	7
Abb. 2.3: Dialogfenster des Quellen-Moduls	15
Abb. 2.4: Drei Module und das Dialogfenster zur Eingabe der Stationenfolge	16
Abb. 2.5: Dialogfenster zur Eingabe der Lagerabgangspolitik	16
Abb. 2.6: Eingabe der Mittleren Bearbeitungszeit im Dialogfenster des Prozess-Moduls ..	18
Abb. 2.7: Drei Störungen: WAIT, IGNORE und PREEMPT	19
Abb. 2.8: Dialogfenster der Dummy-Station.....	21
Abb. 2.9: Bestimmung der Ankunftsrate λ_D	22
Abb. 3.1: Das System I.....	27
Abb. 3.2: Dialogfenster des Speicher-Moduls.....	30
Abb. 3.3: Module im Batch-Bereich des Systems I.....	32
Abb. 3.4: Dialogfenster des Entscheidungs-Moduls	33
Abb. 3.5: Dialogfenster des Gruppierungs-Moduls.....	34
Abb. 3.6: Dialogfenster des Separierungs-Moduls.....	34
Abb. 3.7: Ansicht des Process Analyzer.....	36
Abb. 3.8: Die Ergebnisse von Park et al. 2002 für das System I.....	37
Abb. 3.9: Gegenüberstellung der Ergebnisse für das System I	38
Abb. 3.10: Dialogfenster des Quellen-Moduls	40
Abb. 3.11: Dialogfenster des Halte-Moduls	41
Abb. 3.12: Dialogfenster des Entscheidungs-Moduls	42
Abb. 3.13: Dialogfenster des Such-Moduls.....	43
Abb. 3.14: Dialogfenster des Greif-Moduls	44
Abb. 3.15: Dialogfenster des Such-Moduls "Suche erste Einheit".....	45
Abb. 3.16: Dialogfenster des Verzögerungs-Moduls	46
Abb. 3.17: Ausschnitt eines Grid-Windows	47
Abb. 3.18: Auswirkungen von <i>Lagerkapazitaet3</i> und von <i>WIP</i> auf die Mittlere Durchlaufzeit.....	49
Abb. 3.19: Keine Interaktion zwischen den Produkt-Typen bei Variation von <i>Lagerkapazitaet3</i> und <i>WIP</i>	50
Abb. 3.20: Streuung der Mittleren Durchlaufzeiten für alle Produkt-Typen in allen Szenarien.....	52
Abb. 3.21: Grid-Window bei variierender Lagerkapazität.....	53
Abb. 3.22: Unabhängigkeit der Mittleren Durchlaufzeit der Produkt-Typen von der gewählten Lagerkapazität	54
Abb. 3.23: Veränderung der Mittleren Durchlaufzeit bei Variation des <i>WIP</i>	55
Abb. 3.24: Mittlere Durchlaufzeit bei variierender <i>Lagerkapazitaet 5</i> für zwei konstante <i>WIP</i> -Level und <i>Lagerkapazitaet 3 = 5</i>	57

Abb. 3.25: Mittlere Durchlaufzeit bei variierender Lagerkapazität 5 für zwei konstante <i>WIP</i> -Level und <i>Lagerkapazität</i> 3 = 10	58
Abb. 3.26: Relativierte Mittlere Durchlaufzeit bei variierender <i>Lagerkapazität</i> 5 für zwei konstante <i>WIP</i> -Level und <i>Lagerkapazität</i> 3 = 5	59
Abb. 3.27: Relativierte Mittlere Durchlaufzeit bei variierender <i>Lagerkapazität</i> 5 für zwei konstante <i>WIP</i> -Level und <i>Lagerkapazität</i> 3 = 10	59
Abb. 3.28: Grid-Window für <i>WIP</i> = 324.....	60
Abb. 3.29: Gegenüberstellung der relativierten Verläufe.....	61
Abb. 3.30: Mittlere Durchlaufzeit bei Variation des <i>WIP</i>	63

Tabellenverzeichnis

Tab. 2.1: Performance-Kennzahlen für Fab 1.24 nach WEIN (1988).....	9
Tab. 2.2: Politiken und Performance-Kennzahlen für Fab 1.24 nach WEIN (1988)	10
Tab. 2.3: Die Module, ihre Bezeichnung und ihre Funktion	11-14
Tab. 2.4: Gegenüberstellung der Ergebnisse für die Maschinenauslastung.....	24
Tab. 2.5: Gegenüberstellung der Performance Kennzahlen.....	24
Tab. 3.1: Bearbeitungszeiten und Zahl der Bearbeitungen im System I.....	26
Tab. 3.2: Position und Anzahl der Batch-Stationen in den 4 Szenarien	32
Tab. 3.3: Steigungen der Graphen aus Abbildung 3.23 für ausgewählte Szenarien.....	56
Tab. 3.4: Gegenüberstellung der errechneten mit den simulativ ermittelten Mittleren Durchlaufzeiten.....	64

Anhang

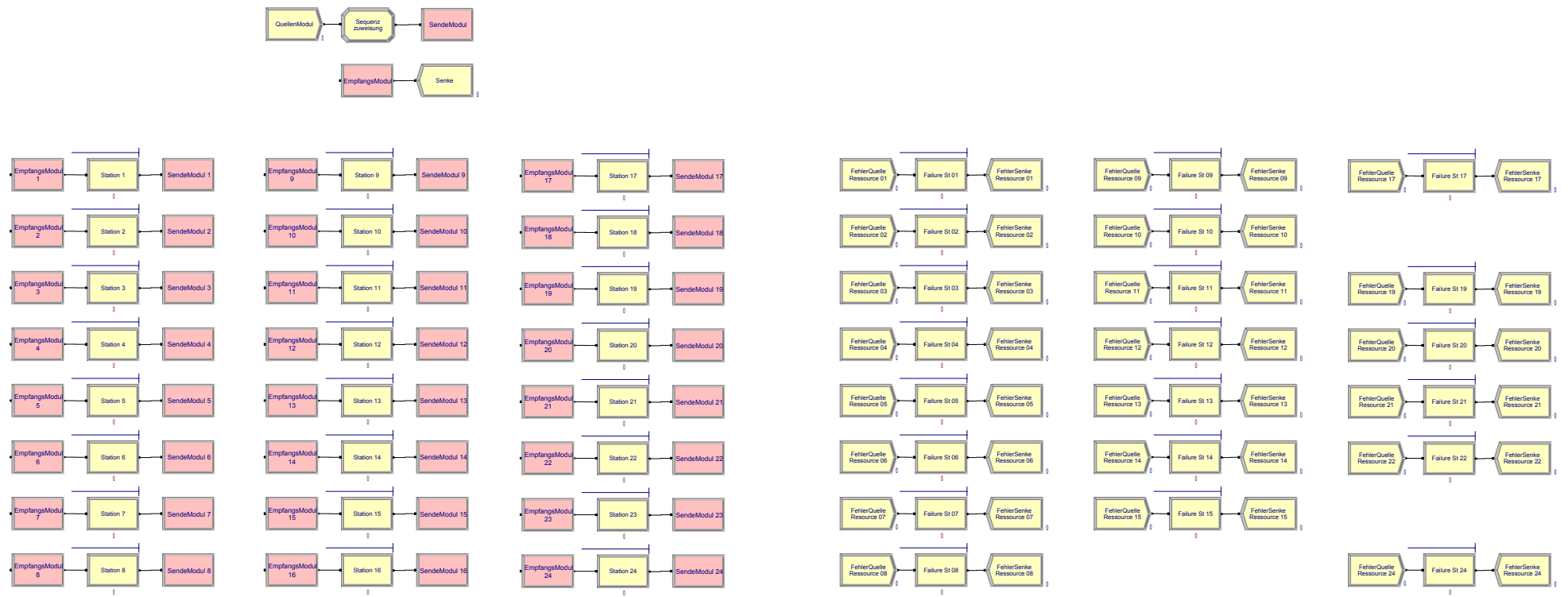


Abbildung 1: Fab1.24-Dummy

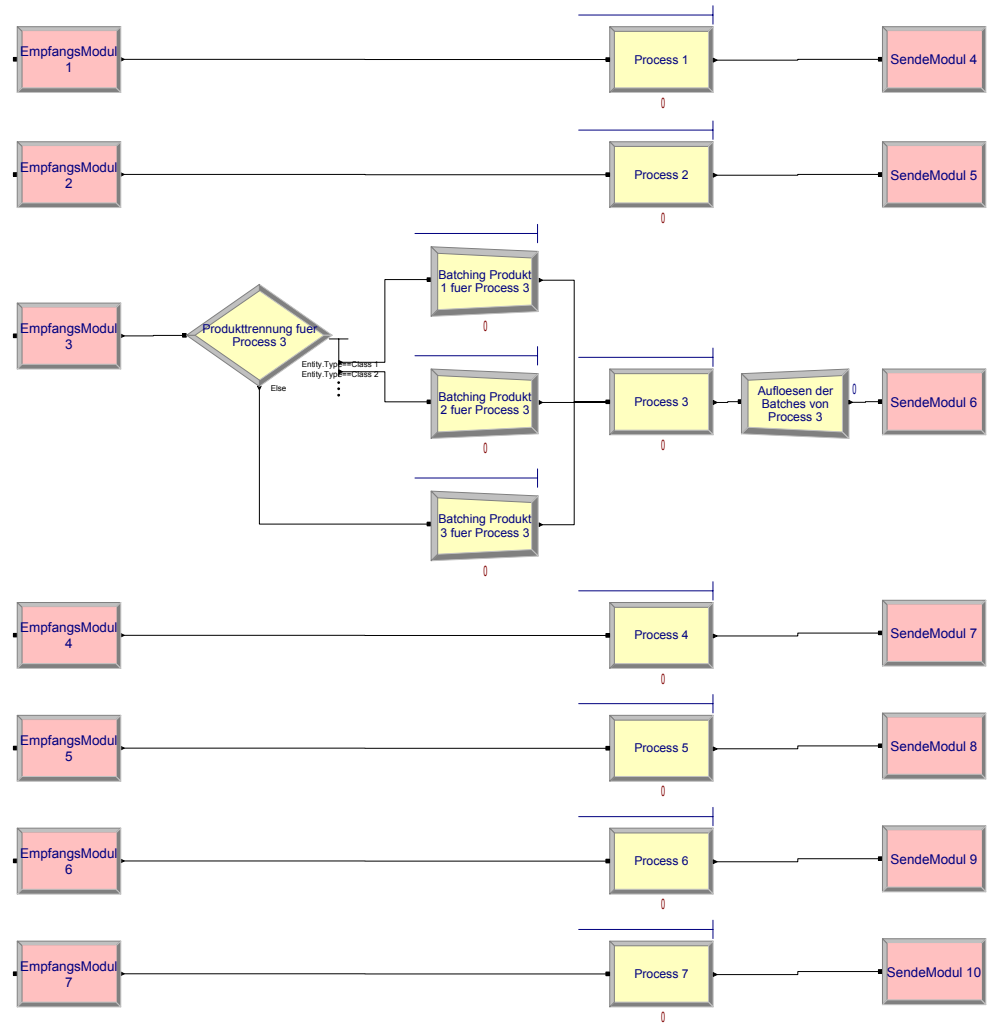
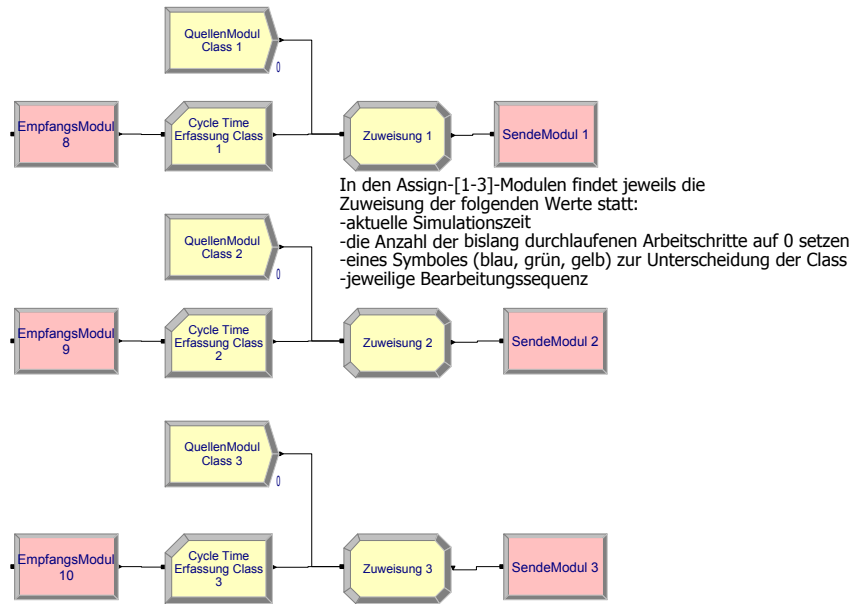


Abbildung 2: Eine Batch-Workstation

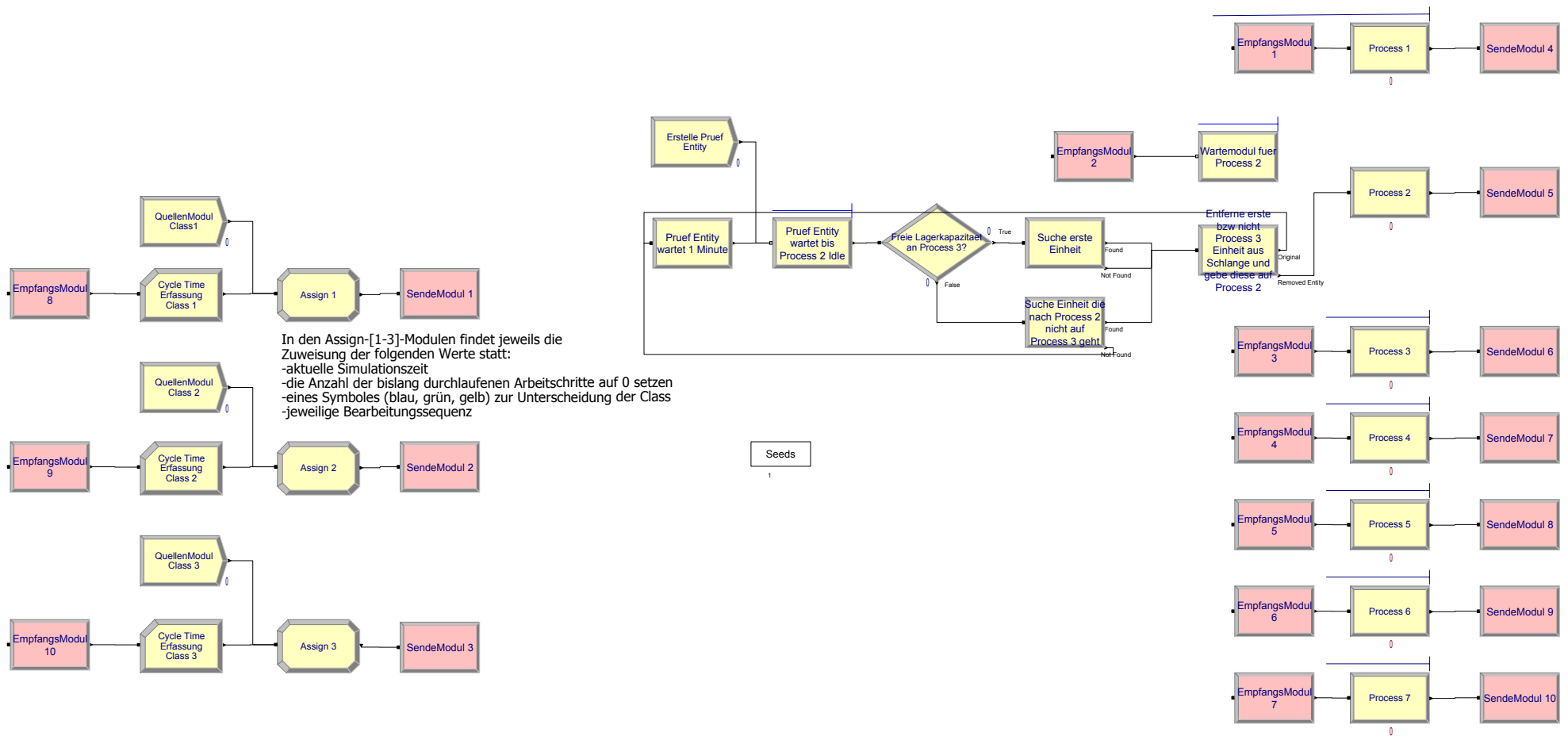


Abbildung 3: Kapazitiertes Lager an einer Workstation

Diskussionspapiere der Fakultät für Wirtschaftswissenschaften ab Januar 2000

- | | | | | | |
|------|--|--|------|--|--|
| 441. | Anne Chwolka,
Dirk Simons | Impacts of Revenue Sharing, Profit Sharing, and Transfer Pricing on Quality-Improving Investments
Januar 2000 | 455. | Alexander Krüger,
Ralf-Michael
Marquardt | Der Euro - eine schwache Währung?
Mai 2000 |
| 442. | Carsten Köper,
Peter Flaschel | Real-Financial Interaction:
A Keynes-Metzler-Goodwin
Portfolio Approach
Januar 2000 | 456. | Veith Tiemann | Symmetrische/klassische Krypto-
graphie - Ein interaktiver Überblick,
Mai 2000 |
| 443. | Th. Spitta, R. Decker,
A. Sigge, P. Wolf,
V. Tiemann | Erste Bilanz des Kreditpunkte-
systems der Fakultät für Wirt-
schaftswissenschaften
Januar 2000 | 457. | Imre Dobos | Umweltbewusste Produktionspla-
nung auf Grundlage einer dynami-
schen umweltorientierten Produk-
tionstheorie: Eine Projektbeschrei-
bung
Juni 2000 |
| 444. | Imre Dobos | A dynamic theory of production:
flow or stock-flow production
Functions
Februar 2000 | 458. | Imre Dobos | Optimal production-inventory
strategies for a HMMS-type reverse
logistics system
Juli 2000 |
| 445. | Carl Chiarella,
Peter Flaschel | Applying Disequilibrium Growth
Theory: I. Investment, Debt and
Debt Deflation
January 2000 | 459. | Joachim Frohn | Ein Marktmodell zur Erfassung von
Wanderungen (revidierte Fassung)
Juli 2000 |
| 446. | Imre Dobos | A Dynamic Environmental Theory
of Production
Maerz 2000 | 460. | Klaus-Peter Kistner
Imre Dobos | Ansaetze einer umweltorientierten
Produktionsplanung:
Ergebnisse eines Seminars
Juli 2000 |
| 447. | Anne Chwolka | "Marktorientierte Zielkostenvor-
gaben als Instrument der Verhaltens-
steuerung im Kostenmanagement",
März 2000 | 461. | Reinhold Decker | Instrumentelle Entscheidungsunter-
stützung im Marketing am Beispiel
der Verbundproblematik,
September 2000 |
| 448. | Volker Böhm,
Carl Chiarella | Mean Variance Preferences,
Expectations Formation, and the
Dynamics of Random Asset Prices
April 2000 | 462. | Caren Sureth | The influence of taxation on
partially irreversible investment
decisions - A real option approach,
April 2000 |
| 449. | Beate Pilgrim | Non-equivalence of uniqueness of
equilibria in complete and in
incomplete market models,
March 2000 | 463. | Veith Tiemann | Asymmetrische/moderne Krypto-
graphie - Ein interaktiver Überblick
Oktober 2000 |
| 450. | Beate Pilgrim | A Brief Note on Mas-Colell's First
Observation on Sunspots,
March 2000 | 464. | Carsten Köper | Stability Analysis of an Extended
KMG Growth Dynamics
December 2000 |
| 451. | Thorsten Temme | An Integrated Approach for the Use
of CHAID in Applied Marketing
Research,
May 2000 | 465. | Stefan Kardekewitz | Analyse der unilateralen
Maßnahmen zur Vermeidung der
Doppelbesteuerung im deutschen
Erbschaftsteuerrecht
Februar 2001 |
| 452. | Reinhold Decker,
Claudia Bornemeyer | Ausgewählte Ansätze zur Entschei-
dungsunterstützung im Rahmen der
Produktliniengestaltung,
Mai 2000 | 466. | Werner Glastetter | Zur Kontroverse über das angemese-
ne wirtschafts- und konjunktur-
politische Paradigma –
Einige Akzente der gesamtwirt-
schaftlichen Entwicklung West-
deutschlands von 1950 bis 1993
März 2001 |
| 453. | Martin Feldmann | Threshold Accepting with a Back
Step. Excellent results with a hybrid
variant of Threshold Accepting,
Mai 2000 | 467. | Thomas Braun,
Ariane Reiss | Benchmarkorientierte Portfolio-
Strategien
Mai 2001 |
| 454. | Willi Semmler,
Malte Sieveking | Credit Risk and Sustainable Debt:
A Model and Estimations for
Euroland
November 1999 | 468. | Martin Feldmann,
Stephanie Müller | An incentive scheme for true
information providing in SUPPLY
CHAINS,
Juni 2001 |

469. Wolf-Jürgen Beyn, Thorsten Pampel, Willi Semmler Dynamic optimization and Skiba sets in economic examples, August 2001
470. Werner Glastetter Zur Kontroverse über das angemessene wirtschafts- und konjunkturpolitische Paradigma (II) – Einige Akzente der gesamtwirtschaftlichen Entwicklung der Bundesrepublik Deutschland von 1991-1999, September 2001
471. Hermann Jahnke, Anne Chwolka, Dirk Simons Coordinating demand and capacity by adaptive decision making September 2001
472. Thorsten Pampel Approximation of generalized connecting orbits with asymptotic rate, September 2001
473. Reinhold Decker Heiko Schimmelpfennig Assoziationskoeffizienten und Assoziationsregeln als Instrumente der Verbundmessung - Eine vergleichende Betrachtung, September 2001
474. Peter Naeve Virtuelle Tabellensammlung, September 2001
475. Heinz-J. Bontrup Ralf-Michael Marquardt Germany's Reform of the Pension System: Choice between „Scylla and Charybdis“ Oktober 2001
476. Alexander M. Krüger Wechselkurszielzonen zwischen Euro, Dollar und Yen -- nur eine Illusion? Oktober 2001
477. Jan Wenzelburger Learning to predict rationally when beliefs are heterogeneous. October 2001
478. Jan Wenzelburger Learning in linear models with expectational leads October 2001
479. Claudia Bornemeyer, Reinhold Decker Key Success Factors in City Marketing – Some Empirical Evidence - October 2001
- 479 b Fred Becker Michael Tölle Personalentwicklung für Nachwuchs-wissenschaftler der Universität Bielefeld: Eine explorative Studie zur Erhebung des Ist-Zustands und zur Begründung von Gestaltungsvorschlägen Oktober 2001
480. Dirk Biskup, Martin Feldmann On scheduling around large restrictive common due windows December 2001
481. Dirk Biskup A mixed-integer programming formulation for the ELSP with sequence-dependent setup-costs and setup-times December 2001
482. Lars Grüne, Willi Semmler, Malte Sieveking Thresholds in a Credit Market Model with Multiple Equilibria August 2001
483. Toichiro Asada Price Flexibility and Instability in a Macrodynamical Model with Debt Effect, February 2002
484. Rolf König, Caren Sureth Die ökonomische Analyse der Auswirkungen der Unternehmenssteuerreform auf Sachinvestitionsentscheidungen vor dem Hintergrund von Vorteilhaftigkeits- und Neutralitätsüberlegungen - diskreter und stetiger Fall - März 2002
485. Fred G. Becker, Helge Probst Personaleinführung für Universitätsprofessoren: Eine explorative Studie an den Universitäten in Nordrhein-Westfalen zum Angebot und an der Universität Bielefeld zum Bedarf März 2002
486. Volker Böhm, Tomoo Kikuchi Dynamics of Endogenous Business Cycles and Exchange Rate Volatility April 2002
487. Caren Sureth Die Besteuerung von Beteiligungsveräußerungen - eine ökonomische Analyse der Interdependenzen von laufender und einmaliger Besteuerung vor dem Hintergrund der Forderung nach Rechtsformneutralität - Juli 2002
488. Reinhold Decker Data Mining und Datenexploration in der Betriebswirtschaft Juli 2002
489. Ralf Wagner, Kai-Stefan Beinke, Michael Wendling Good Odd Prices and Better Odd Prices – An Empirical Investigation September 2002
490. Hans Gersbach, Jan Wenzelburger The Workout of Banking Crises: A Macroeconomic Perspective September 2002
491. Dirk Biskup, Dirk Simons Common due date scheduling with autonomous and induced learning September 2002
492. Martin Feldmann, Ralf Wagner Navigation in Hypermedia: Neue Wege für Kunden und Mitarbeiter September 2002
493. Volker Böhm, Jan Wenzelburger On the Performance of Efficient Portfolios November 2002
494. J. Frohn, P. Chen, W. Lemke, Th. Archontakis, Th. Domeratzki, C. Flöttmann, M. Hillebrand, J. Kitanovic, R. Rucha, M. Pullen Empirische Analysen von Finanzmarktdaten November 2002
495. Volker Böhm CAPM Basics December 2002

496. Susanne Kalinowski , Stefan Kardekewitz Betriebsstätte vs. Kapitalgesellschaft im Ausland - eine ökonomische Analyse
März 2003
497. Jochen Jungeilges On Chaotic Consistent Expectations Equilibria
March 2003
498. Volker Böhm MACRODYN - The Handbook -
March 2003
499. Jochen A. Jungeilges Sequential Computation of Sample Moments and Sample Quantiles – A Tool for MACRODYN -
April 2003
500. Fred G. Becker, Vera Brenner Personalfreisetzung in Familienunternehmen: Eine explorative Studie zur Problematik
Juni 2003
501. Michael J. Fallgatter, Dirk Simons. "Zum Überwachungsgefüge deutscher Kapitalgesellschaften - Eine anreiz-theoretische Analyse der Vergütung, Haftung und Selbstverpflichtung des Aufsichtsrates"
Juni 2003
502. Pu Chen Weak exogeneity in simultaneous equations models
Juli 2003
503. Pu Chen Testing weak exogeneity in VECM
Juli 2003
504. Fred G. Becker, Carmen Schröder Personalentwicklung von Nachwuchs-wissenschaftlern: Eine empirische Studie bei Habilitanden des Fachs "Betriebswirtschaftslehre"
Juli 2003
505. Caren Sureth Die Wirkungen gesetzlicher und theoretischer Übergangsregelungen bei Steuerreformen – eine ökonomische Analyse steuerinduzierter Verzerrungen am Beispiel der Reform der Besteuerung von Beteiligungserträgen -
August 2003
506. Jan Wenzelburger Learning to play best response in duopoly games"
August 2003
507. Dirk Simons Quasirentenansätze und Lerneffekte
September 2003
508. Dirk Simons
Dirk Biskup Besteht ein Bedarf nach Dritthaftung des gesetzlichen Jahresabschlussprüfers?
September 2003
509. Tomoo Kikuchi A Note on Symmetry Breaking in a World Economy with an International Financial Market.
October 2003
510. Fred G. Becker
Oliver Krahn Explorative Studie zur Personaleinführung bei Unternehmen in OWL: Ergebnisübersicht
Oktober 2003
511. Martin Feldmann
Stephanie Müller Simulation von Reentrant Lines mit ARENA: Ergebnisse eines Projektes zur Betriebsinformatik
Januar 2004