

Simulationsumgebung zur Analyse von Algorithmen für dynamische Scheduling-Probleme im Produktionsumfeld

Todor Dimitrov, Dr. Michael Baumann

Fraunhofer Institut für Informations- und Datenverarbeitung IITB
Fraunhoferstr. 1
76131 Karlsruhe
todor.dimitrov@iitb.fraunhofer.de
michael.baumann@iitb.fraunhofer.de

Abstract: Eine der Aufgaben eines ME-Systems (Manufacturing Execution Systems) stellt die Feinplanung und –steuerung der Produktion dar. ME-Systeme werden in der IT-Infrastruktur eines Unternehmens zwischen dem ERP-System und dem Shop-Floor angesiedelt und bilden hier einen geschlossenen Regelkreis. Die Feinplanungsaufgabe besteht darin, den Produktionsplan permanent an die Veränderungen aus dem ERP-System und dem Shop-Floor anzupassen. Diese Abhandlung beschreibt eine dynamische Testumgebung, die alle relevanten Datenflüsse vom ERP-System und dem Shop-Floor nachbildet. Damit werden ein realistischer Test der Algorithmen zur Feinplanung und deren Parametrierung im Vorfeld der eigentlichen Inbetriebnahme eines ME-Systems ermöglicht.

1 Einführung

Die Feinplanung der Produktionsabläufe und deren Optimierung haben in den vergangenen Jahrzehnten viele Forschungseinrichtungen und Unternehmen beschäftigt, da sich dadurch drastische Kosteneinsparungen erzielen lassen. In diesem Feld wird eine Reihe von Software-Systemen zur Unterstützung der gesamten Wertschöpfungskette eingesetzt.

ERP-Systeme (Enterprise Resource Planning) sind für das Unternehmensmanagement für die Bereiche Finanz- und Rechnungswesen, Controlling, Personalwirtschaft, Einkauf, Verkauf, Produktionsplanung und Marketing verantwortlich. Da der Erfolg der Produktionsplanung mit der Durchsetzung des Plans im Produktionsbetrieb steht und fällt, haben sich neben den ERP-Systemen sog. ME-Systeme (Manufacturing Execution Systeme) etabliert [MES97]. ME-Systeme haben das ganzheitliche Fertigungsmanagement zum Ziel (s. [K106, TMF08]). Eine wesentliche Aufgabe stellt dabei die Feinplanung der Produktionsabläufe dar. Da sich die Situation in der Produktion bedingt durch Störungen, Verspätungen etc. dauernd ändert, ist es zwingend, die Planung permanent zu überarbeiten.

Bereits Mitte der neunziger Jahre des vergangenen Jahrhunderts wurden verschiedene Arbeiten in Richtung „Reactive Scheduling“ durchgeführt (vgl. [Sm94, DKT95]). Voraussetzung für eine zielgerichtete, reaktive Umplanung ist die konsequente Erfassung der Produktionsdaten mit Hilfe von BDE/MDE (Betriebs-/Maschinendatenerfassung). Diese sorgt dafür, dass immer ein realitätsnahes Produktionsabbild vorliegt (s. Abbildung 1).

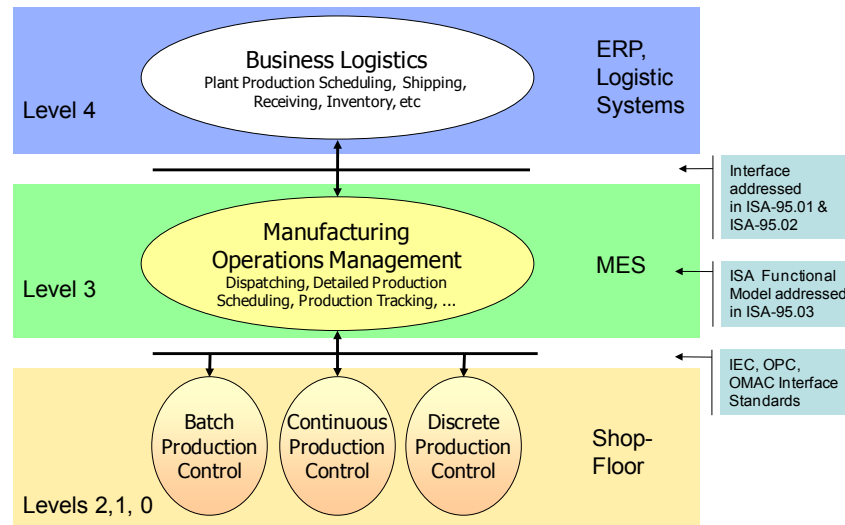


Abbildung 1: Ebenenmodell nach ISA-95 [ISA00]

Bei der Entwicklung und Bewertung von Algorithmen für diese Aufgabenstellung reicht es nicht aus, diese mit einer statischen Menge von Aufträgen zu belasten. Stattdessen ist es zwingend, die Algorithmen in einer dynamischen Testumgebung zu untersuchen und zu parametrieren. Denn im realen Produktionsbetrieb ändert sich die Planungsgrundlage permanent. Es kommen neue Aufträge hinzu, Aufträge werden geändert oder gelöscht, einzelne Arbeitsvorgänge werden gestartet, beendet, Maschinen fallen aus etc.

Diese Abhandlung skizziert eine Simulationsumgebung mit der Scheduling-Systeme im Produktionsumfeld in einer Laborsituation realistischen Tests unterzogen werden können, d.h. Tests mit dynamischem Auftragsengang sowie Zustands- und Fortschrittmeldungen. Hierzu werden ERP- und BDE-/MDE-Systeme nachgebildet, die den Scheduler zeitnah mit diesen Meldungen versorgen.

2 Simulationsumgebung zum Test von Scheduling-Algorithmen

Die hier vorgestellte Simulationsumgebung verfolgt das primäre Ziel Scheduling-Algorithmen zur Lösung der Feinplanungsaufgabe von ME-Systemen zu analysieren und einzustellen. Zu diesem Zweck muss die Simulationsumgebung den realen Einsatz der Algorithmen nachbilden und mit dem Scheduler wie in einer realen Anwendung kommunizieren.

2.1 Stand der Technik

Obgleich es eine Reihe von Simulationswerkzeugen gibt, konnte von den Autoren keines gefunden werden, das die Anforderungen zum Test von Scheduling-Lösungen in der Produktion exakt trifft. Simulationstechnik wird zur Entscheidungsunterstützung bei einer Vielzahl von betrieblichen Fragestellungen eingesetzt. Es geht hierbei von der Produktentwicklung, den zugehörigen Produktionssystemen über die eigentliche Produktion bis hin zur Vermarktung. Im gesamten Product Lifecycle Management (PLM) gibt es eine Vielzahl von Anwendungsfällen für Simulationstechniken [Sc06]. Häufig wird im Rahmen der Auslegung und Prozessplanung für Produktions- und Logistiksysteme auf die Simulation zurückgegriffen. Im Bereich der „Digitalen Fabrik“ gibt es mittlerweile einige Anbieter von Simulationssystemen mit denen interaktiv, teilweise graphisch, Modelle aufgebaut und Simulationsexperimente durchgeführt werden können [Kü06, Ba08].

In der „Digitalen Fabrik“ werden meist mit Hilfe von 3D Modellierung komplette Produktionshallen abgebildet. Die realitätsnahen Modelle dienen meist der Verifizierung und Optimierung des Hallenlayouts sowie der Organisation von Produktion und Logistik. Hier werden auch Prozessabläufe, Transporte sowie die Bewegungsfolgen von Robotern simuliert. Dies kann bis hin zu den Werkzeugeingriffen oder der Verifikation von SPS-Programmen gehen [Fe97].

Ein anders gearteter Anwendungsfall stellt die simulationsgestützte Produktionssteuerung dar [Ze92]. Hier dient das Simulationsmodell der Entscheidungsunterstützung bei der dispositiven Steuerung, um verschiedene Strategien zu bewerten und eine für den Produktionsprozess auszuwählen. Diese Art der „online“ Simulation stellt hohe Anforderungen an die Integration in bestehende Informationssysteme, denn die Entscheidungsfindung muss auf der Basis des aktuellen Produktionszustands erfolgen. Die Lösungen im Bereich simulationsgestützter Produktionssteuerung prognostizieren den voraussichtlichen Produktionsablauf unter Einsatz einer bestimmten Steuerungsstrategie unter der Annahme, dass keine Störungen auftreten. Hierbei wird nicht an der Optimierung und Parametrierung von Algorithmen gearbeitet, sondern daran, ausgehend von einem konkreten Produktionszustand Entscheidungsunterstützung für die Einlastung von Produktionsprozessen zu geben.

Der Ansatz, der in dieser Abhandlung verfolgt wird, versucht vielmehr, Algorithmik in einer Laborsituation einzustellen, um sie dann im Produktionsbetrieb so zu verwenden und eine automatische Optimierung der Produktionsabläufe zu ermöglichen. Die klassischen Simulationssysteme bieten für diesen Einsatzzweck keine Unterstützung an. Und spezifische Simulationsumgebungen zum dynamischen Test von Scheduling Lösungen für die Produktion werden derzeit nicht angeboten.

2.2 Architektur der Simulationsumgebung

Die Simulationsumgebung muss wie eingangs bereits erwähnt, die Integration in die betriebliche IT-Landschaft nachbilden. Hierzu gehört einerseits das ERP-System, das den Scheduler mit Aufträgen und deren Änderungen versorgt. Andererseits der Produktionsprozess (Shop-Floor), von wo über BDE oder MDE der aktuelle Zustand/Fortschritt an den Scheduler gemeldet wird. Daher sind zum realitätsnahen Test eines Schedulers ERP- und BDE-System nachzubilden. Und zwar möglichst so, dass der Scheduler optimal auf seine spätere Aufgabe hin, die Produktion zu optimieren, voreingestellt werden kann. Zu diesem Zweck wird eine Simulationsumgebung bestehend aus ERP- und BDE-Substitut aufgebaut, die den Scheduler mit allen relevanten Ereignissen beschickt. Das ERP-Substitut generiert Aufträge und deren Änderungen. Das BDE-Substitut bildet auf der Basis des vom Scheduler ermittelten Produktionsplans den Fertigungsablauf nach und generiert daraus Meldungen.

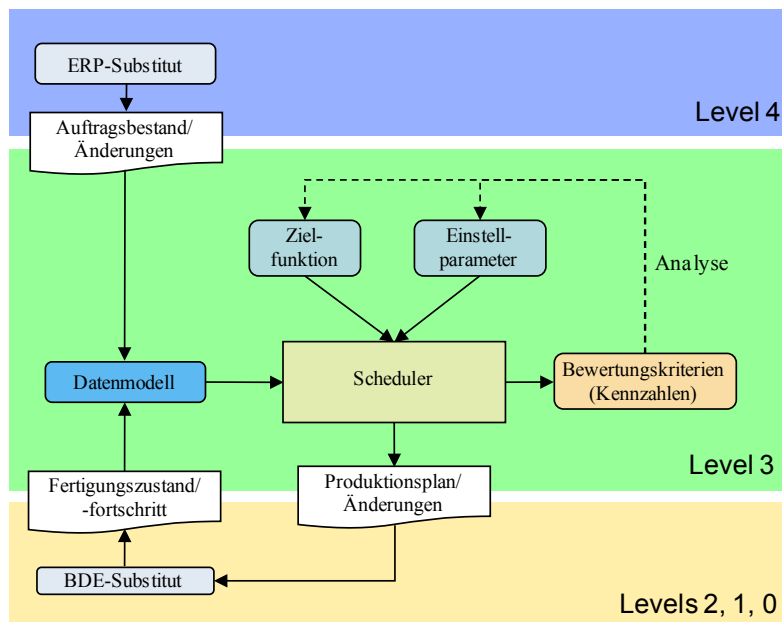


Abbildung 2: Simulationsumgebung für den Test von Scheduling-Algorithmen

Um einen realistischen Fertigungsablauf abzubilden, werden Störungen und Abweichungen in den Produktionsplan eingebracht (vgl. [CF09]). Dabei können auch Probleme der Datenübertragung oder der Datenqualität simuliert werden. Denn gerade aus manuell bedienten BDE-Systemen werden die Ereignisse häufig fehlerhaft, verspätet oder nicht korrekt abgesetzt.

Die Simulationsumgebung setzt sich aus dem ERP-Substitut, dem BDE-Substitut, dem zugehörigen Datenmodell, einer Parametrierkomponente für die Algorithmen sowie einer Bewertungskomponente zusammen (vgl. Abbildung 2).

Ein vom Scheduler erzeugter Produktionsplan wird über die Bewertungskomponente nach verschiedenen Kennzahlen wie Termintreue, Minimale Lagerbestände, Minimale Durchlaufzeiten etc, evaluiert. Hieran schließt sich ggf. ein weiterer Simulationszyklus an, für den der Benutzer durch die Parametrierkomponente die Zielfunktion einstellt, einen Algorithmus auswählt und parametriert.

2.3 Datenmodell von ERP- und BDE-Substitut

Das Datenmodell orientiert sich am ISA-95 Standard [ISA01], das alle Objekte in der Produktion und Logistik wie Maschinen, Werkzeuge, Transporteinheiten und Lager beschreibt. Alle Informationen über die zu produzierenden Aufträge werden vom ERP-Substitut generiert und an den Scheduler über eine Schnittstelle, in diesem Fall B2MML, übertragen. B2MML steht für *Business To Manufacturing Markup Language* und stellt eine XML-Implementierung des ISA-95 Standards bereit [B2M08].

Ein Auftrag o wird definiert über das Produkt $q_o^{product}$, die Menge q_o^{qty} sowie einem frühesten Starttermin $q_o^{earliest}$ und einem spätesten Sollendtermin q_o^{latest} . $q_o^{release}$ stellt den Freigabezeitpunkt an den Scheduler dar. Ein Auftrag beschreibt außerdem die Prozesse zur Herstellung des Produkts. Die Prozesse j_k sind die Hauptartefakte eines Feinplanungssystems. Sie sind über Anordnungsbeziehungen mit einander verknüpft. Durch die Anordnungsbeziehungen bekommt ein Prozess j_k eine Menge von Vorgängerprozessen V_k bzw. Nachfolgerprozessen N_k . Dabei können V_k und N_k auch Prozesse von anderen Aufträgen beinhalten, um Auftragsverkettungen abzubilden. Anordnungsbeziehungen können zeitbehaftet sein.

Des Weiteren haben die Prozesse Ressourcenbedarfe. Ressourcen stehen hier allgemein für Maschinen, Werkzeuge, Personal oder Material. Ein Prozess j_k verarbeitet eine Menge eines Eingangsmaterials M_k^{in} zu einer Menge eines Ausgangsmaterials M_k^{out} und benötigt hierzu die Bearbeitungsdauer $p_k^{duration}$, die von der Auftragsmenge (Losgröße) abhängen kann (vgl. Abbildung 3). Transportprozesse können über eigene Prozesse oder lediglich durch Übergangszeiten zwischen zwei Prozessen modelliert werden.

Neben diesen Soll-Vorgaben aus dem ERP-Substitut umfasst das Datenmodell auch Ist-Start-/Endezeiten der Prozesse, verwendete Ressourcen und Einsatzdauern sowie die verbrauchten und produzierten Materialien (Ein-/Ausgangsmaterialien), die vom BDE-Substitut gemeldet werden.

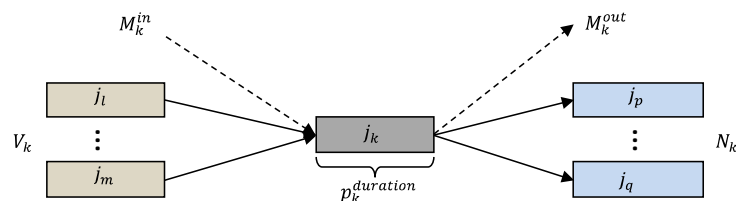


Abbildung 3: Prozessmodell (vgl. [BD08])

3 ERP-Substitut

Das ERP-Substitut ersetzt das ERP-System in der Simulationsumgebung und generiert Meldungen für die relevanten Ereignisse auf der Verwaltungsebene. Das ERP-Substitut besteht aus zwei unabhängigen Komponenten – dem Ereignisplaner und dem Meldungsgenerator.

Der Ereignisplaner erstellt eine vollständige Ereignisliste zu Beginn der Simulation. Diese Liste wird vom Meldungsgenerator zum Ereigniszeitpunkt sukzessiv in einzelne Meldungen verpackt und an den Scheduler geschickt. Dieser Ansatz wurde im Gegensatz zu einer Real-Time Ereignis-Generierung gewählt, um wiederholte Tests auf demselben Datenbestand durchführen zu können. So können unterschiedliche Algorithmen und deren Parametrierungen verglichen werden.

Im Ereignisplaner können verschiedene Ereignisse definiert werden:

- Änderung der Stammdaten: z.B. neue Maschine oder zusätzliches Personal.
- Änderung in den Bewegungsdaten: z.B. neue Aufträge, zusätzliche Schichten oder ein Materialeingang zu einem bestimmten Zeitpunkt.

Das Hauptaugenmerk des Ereignisplaners liegt aber bei den Änderungen des Auftragsbestands. Dabei werden Ereignisse für das Einfügen, Ändern oder Löschen von Aufträgen im Auftragsbestand vorgesehen. Aufgrund des Umfangs des Auftragsbestands in einem Produktionsunternehmen ist es nicht möglich den kompletten Änderungsablauf manuell nachzubilden. Die Parametrierung erfolgt über Auftragsschablonen, welche für jeden Auftragsstyp separat konfiguriert werden können. Pro Auftragsschablone a sind die folgenden Parameter einzustellen:

- $e_a^{product}$ zu produzierendes Produkt (daraus werden die Prozesse abgeleitet)
- $e_a^{totalqty}$ Gesamtmenge der Auftragsschablone
- e_a^{qty} Menge pro Auftrag (Losgröße)
- $e_a^{leadtime}$ Zeitspanne zwischen frühestem Startzeitpunkt und spätestem Endzeitpunkt. Ist die Zeitspanne unendlich, handelt es sich um einen Grundlastauftrag.
- $e_a^{releasetime}$ Zeitspanne zwischen dem Freigabezeitpunkt des Auftrags an den Scheduler und dem frühesten Startzeitpunkt. Ist diese Zeitspanne nicht definiert, so ist der Auftrag sofort nach seiner Bekanntgabe durchführbar.
- $e_a^{modified}$, $e_a^{deleted}$ Anteil geänderter und gelöschter Aufträge (in Prozent)

Zur einfacheren Bedienung müssen diese Parameter nicht für jede Auftragsschablone eingegeben werden. Stattdessen werden sie einer Hauptschablone entnommen und ggf. variiert. Um eine gewisse Streuung einzubringen kann für jeden Parameter seine Varianz $v_a^{<param>}$ definiert werden.

Der Algorithmus für das Generieren der Aufträge erstellt pro Schablone a einen neuen Auftrag o mit der zugehörigen Menge $q_o^{qty} = f(e_a^{qty}, v_a^{qty})$, solange die Summe aller erstellten Aufträge pro Schablone $e_a^{totalqty} / (1 - e_a^{deleted})$ nicht überschreitet. Die Generierung für eine Schablone bricht ab, wenn diese Summe überschritten wird. Die Funktion $f(e, v)$ generiert mit Hilfe eines Erwartungswerts e und der Varianz v Zufallszahlen, so dass sie der Normalverteilung entsprechen. Anschließend werden die Startzeiten $q_o^{earliest}$ von allen generierten Aufträgen o pro Schablone, einer Gleichverteilung entsprechend, generiert. Daraus werden die Solltermine für jeden Auftrag o errechnet: $q_o^{latest} = q_o^{earliest} + f(e_a^{leadtime}, v_a^{leadtime})$. Für den Freigabezeitpunkt gilt: $q_o^{release} = q_o^{earliest} - f(e_a^{releasetime}, v_a^{releasetime})$.

Für den Anteil der geänderten Aufträge $e_a^{modified}$ wird ein neuer $q_o^{release}$ erzeugt und die Parameter $q_o^{earliest}$, q_o^{latest} und q_o^{qty} erneut berechnet. Beim Anteil gelöschter Aufträge $e_a^{deleted}$ wird $q_o^{release}$ neu ermittelt und $q_o^{qty} = 0$ gesetzt.

Eine Verifizierung des auf diesem Weg generierten Auftragsbestands auf Machbarkeit wird nicht durchgeführt. Dieses Verfahren wurde jedoch gewählt, um extrahierte Daten aus einer realen Produktionsumgebung leicht nachzubilden. Es kann allerdings passieren, dass Verspätungen insbesondere bei hohem Termindruck unvermeidbar sind.

Ein alternativer Weg ist die Generierung der Aufträge anhand von Kapazitätsgrenzen. Der Konfigurationsbedarf ist damit größer. Die zusätzlich zu definierenden Parameter sind:

- e_r^{load} Belastungsgrenze der Maschine r (in Prozent)
- e_{pr}^{load} Belastung der Maschine r durch Produkt p

Hier wird der Simulationszeitraum in gleich große Zeitbereiche z.B. Schichten zerlegt und pro Bereich d und Maschine r ein Belastungskonto k_{dr}^{load} geführt (vgl. [Mo93]). Der Generator erstellt Aufträge von zufällig gewählten Schablonen a und aktualisiert die Belastungskonten der Maschinen mit $e_{pr}^{load} * (1 - e_a^{deleted})$. Er schließt eine Schablone aus, wenn die daraus generierten Aufträge mindestens eine Maschine r belasten, deren Belastungsgrenze e_r^{load} überschritten ist. Der Generator startet mit dem nächsten Zeitbereich, wenn alle Schablonen für den aktuellen Zeitbereich ausgeschlossen sind. Nachdem alle Zeitbereiche verarbeitet wurden, markiert der Generator wie in der ersten Strategie $e_a^{modified}$ und $e_a^{deleted}$ der Aufträge als modifiziert bzw. gelöscht.

Diese Generierungsart wird verwendet, um theoretische Probleme zu generieren und zu analysieren. Es wird mit unterschiedlichen Belastungsgrenzen z.B. 75% oder 120% experimentiert. Ein Ziel dabei ist es beispielsweise, einen Zusammenhang zwischen der Belastungsgrenze und der erzielten Terminverspätung der verschiedenen Scheduling-Algorithmen zu erkennen.

OrderID	ProductionID	Publish Date - FT	Start Date	End Time - SE	Units
2463	B	2008-07-25T20:13:46	2008-07-29T11:59:17	2008-08-07T01:52:46	1061
2464	D	2008-07-27T18:04:44	2008-07-31T04:15:08	2008-08-09T16:41:43	26
2465	B	2008-07-23T15:18:30	2008-08-06T10:34:05	2008-08-11T14:29:40	1135
2466	F	2008-07-25T02:30:54	2008-08-02T03:00:47	2008-08-05T17:24:35	11000
2467	D	2008-08-01T14:17:13	2008-08-03T05:06:21	2008-08-04T18:27:48	25
2468	B	2008-07-30T17:28:32	2008-08-03T15:37:26	2008-08-10T16:25:36	1026
2469	D	2008-07-29T22:35:27	2008-08-13T03:30:48	2008-08-16T14:36:12	26
2470	D	2008-07-27T13:50:35	2008-08-06T04:23:13	2008-08-06T09:12:33	47
2472	B	2008-07-27T21:20:37	2008-08-10T17:35:19	2008-08-15T14:03:11	986
2475	B	2008-08-02T19:55:13	2008-08-10T19:39:04	2008-08-13T10:51:42	1018
2465	B	2008-08-02T19:55:13	2008-08-10T19:39:04	2008-08-13T10:51:42	1218
2476	B	2008-08-04T13:55:26	2008-08-12T07:05:35	2008-08-20T02:06:14	1005
2469	D	2008-08-14T20:52:29	2008-08-17T09:42:46	2008-08-22T20:12:01	90
2482	B	2008-08-15T15:18:14	2008-08-18T14:55:52	2008-08-27T19:37:38	1006
2483	B	2008-08-10T19:05:07	2008-08-19T12:19:02	2008-08-20T19:51:11	1024
2472	B	2008-08-12T17:20:07			0
2484	B	2008-08-18T00:23:04	2008-08-21T23:07:23	2008-08-31T00:39:41	1014

Abbildung 4: Änderungsverlauf des Auftragsbestands

Abbildung 4 zeigt beispielhaft das Ergebnis des Ereignisplaners. Die Liste enthält Ereignisse für das Einfügen neuer Aufträge, ggf. Änderungen sowie das Löschen von Aufträgen. Diese Liste ist der Ausgangspunkt für den Meldungsgenerator, der jeden Eintrag in eine Meldung umsetzt und zum Ereigniszeitpunkt an den Scheduler überträgt.

4 BDE-Substitut

Das BDE-Substitut muss den kompletten Produktionsprozess ausgehend vom Produktionsplan nachbilden. Durch Maschinenausfälle, Materialverspätung, etc. wird der simulierte Produktionsablauf gegenüber dem Produktionsplan variiert. Dabei werden die einzelnen Ereignisse nicht wie beim ERP-Substitut zu Beginn der Simulation in einer Liste abgelegt, sondern während des Simulationsablaufs erstellt, da der Produktionsplan in Abhängigkeit von Fertigungszustand und –fortschritt permanent von der eingesetzten Scheduling-Algorithmen angepasst wird.

Die Hauptaufgabe des BDE-Substituts liegt darin, anhand des Produktionsplans Fortschrittmeldungen zu erzeugen und Materialab- und -zugänge zu melden. Wenn keine Störungsparameter konfiguriert sind, wird ein Ist-Ablauf simuliert, der mit dem Produktionsplan übereinstimmt, falls dieser zulässig ist. Durch Parametrierung können jedoch verschiedene Störungen modelliert werden, wie sie im realen Produktionsablauf an der Tagesordnung sind. Prozesse dauern länger oder kürzer als geplant, die Abläufe weichen vom geplanten Ablauf ab, der Materialverbrauch und die tatsächlich produzierten Zwischen-/Endprodukte weichen von der Spezifikation ab. Weitere Störungen sind Maschinenausfall, Verspätung/Krankheit von Personal oder Materialverlust durch Qualitätsprobleme. Alle Arten von Störungen haben Auswirkungen auf die Prozessereignisse Prozessstart, Prozessfortschritt und Prozessabschluss. Diese werden anhand der folgenden Parameter gestört:

- $b_r^{duration}$ Durchschnittliche Abweichung von der Soll-Dauer auf Maschine r (in Prozent)
- $b_r^{breakdown}$ Störungsanteil der verfügbaren Kapazität von Maschine r (in Prozent)
- $b_r^{breakdownduration}$ Durchschnittliche Störungsdauer auf Maschine r
- b_m^{qty} Anteil der Materialmenge mit Qualitätsstörung von Material m (in Prozent)
- b_m^{qtydev} Durchschnittliche Abweichung der Menge bei Qualitätsstörung von Material m (in Prozent)
- $b_r^{progress}$ Zeitintervall zwischen Fortschrittmeldungen auf Maschine r

Diese Parameter werden für alle Maschinen bzw. Materialien gemeinsam eingegeben und können für einzelne Maschinen/Materialien separat geändert werden.

Neben den Störungen im Produktionsablauf treten auch Störungen bei der Datenerfassung oder –übertragung auf. Die manuelle Eingabe von Betriebsdaten durch das Werkstattpersonal ist häufig sehr fehlerbehaftet, falsche Prozesse werden als gestartet oder beendet markiert oder mit der falschen Menge gemeldet. Allgemein kommt es zu Meldungsverlust, Meldungsverspätung oder Meldung mit falschen Daten. Über folgende Parameter kommt dies zum Ausdruck:

- $b_{lostmsg}$ Anteil an verlorenen Meldungen (in Prozent)
- $b_{msgdelay}$ Durchschnittliche Abweichung bei der Meldungszustellung
- $b_{falsemsg}$ Anteil fehlerhafter Meldungen (in Prozent)

Im BDE-Substitut sind die Simulationsereignisse hinsichtlich ihrer Herkunft zu differenzieren: Externe Ereignisse werden über die Kommunikationsschnittstelle empfangen z.B. neuer Produktionsplan $z^{schedule}$ vom Scheduler, neue Maschine $z^{newmachine}$ etc. Interne Ereignisse werden vom BDE-Substitut selbst generiert. Während des Simulationslaufs erstellt eine Funktion $h(z_n) = \{z_i\}$ aus einem externen oder internen Ereignis z_n eine neue Menge von internen Ereignissen.

Die Ereigniszeitpunkte t_i werden durch die oben definierten Störungsparameter variiert. Ein Beispiel dafür ist die Festlegung des neuen Prozessfortschritts $z_{i+1}^{progress}$: $t_{i+1}^{progress} = t_i^{progress} + b_r^{progress} * f(1, b_m^{duration}) + s(r)$. Die Störfunktion $s(r)$ addiert ggf. eine Störung anhand der zwei Störungsparameter $b_r^{breakdown}$ und $b_r^{breakdownduration}$. Dabei sorgt ein vom BDE-Substitut gepflegtes Störungskonto pro Maschine dafür, dass der Gesamtstörungsanteil im Simulationsablauf zu keinem Zeitpunkt größer ist als $b_r^{breakdown}$.

Neben der Belegungsdauer der Kapazitäten pflegt das BDE-Substitut auch Materialbestände. Die Menge eines von Prozess k verbrauchten Materials (Eingangsmaterial) $m \in M_k^{in}$ wird bei simulierten Qualitätsproblemen um b_m^{qtydev} erhöht. Bei von Prozess k produziertem Material (Ausgangsmaterial) $m \in M_k^{out}$ werden diese Mengen reduziert. Diese Berechnung läuft analog zur Berechnung bei den Maschinenstörungen. Es wird sichergestellt, dass die Materialverluste die durch Qualitätsstörungen entstehen, zu keinem Zeitpunkt die Grenze b_m^{qty} verletzen.

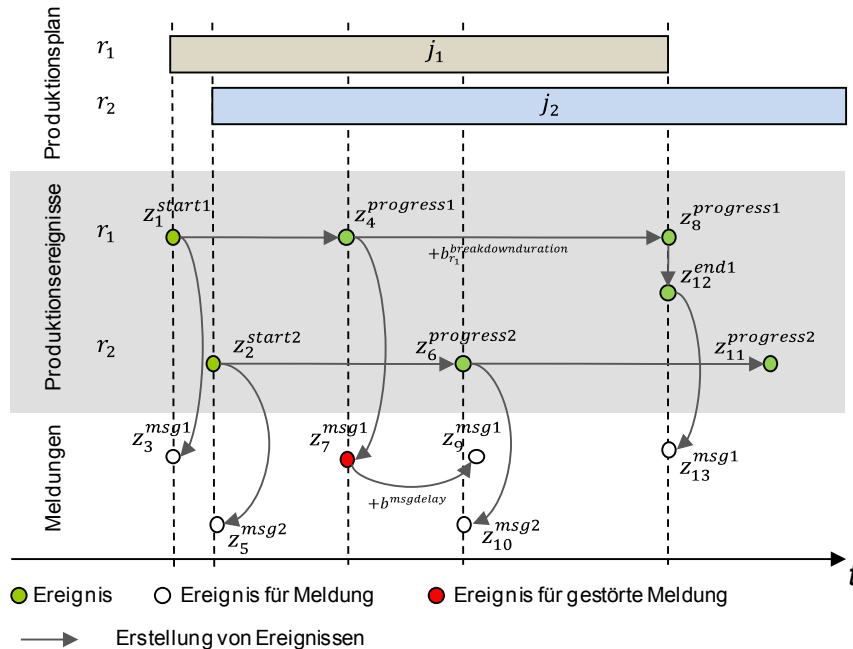


Abbildung 5: Produktionsplan, Ereignis- und Meldungsverlauf

Ein beispielhafter Simulationsablauf kann Abbildung 5 entnommen werden. Die Abbildung stellt die einzelnen Ereignisse im Zeitablauf dar. Die Indizes der Ereignisse sind fortlaufend. Aus dem Produktionsplan werden die Ereigniszeitpunkte ermittelt, daraus Meldungen abgeleitet und ggf. durch Meldungsstörungen verzögert, bevor der Transfer zum Scheduler erfolgt. Die Simulation fängt mit dem externen Ereignis $z_0^{schedule}$ an. Daraus werden die Ereignisse z_1^{start1} und z_2^{start2} für alle startbaren Prozesse erstellt. Bei der Verarbeitung von z_1^{start1} belegt der Simulator die Maschine r_1 und generiert das Fortschrittsereignis $z_4^{progress1}$ sowie die Meldung z_3^{msg1} . Bei der Erstellung des Ereignisses $z_8^{progress1}$ wird der Ereigniszeitpunkt durch eine Maschinenstörung um $b_{r_1}^{breakdownduration}$ verzögert. Bei der Bearbeitung des Fortschrittsereignisses $z_8^{progress1}$ prüft der Simulator, ob der Prozess abgeschlossen ist und generiert das Fertigereignis z_{12}^{end1} . Nach dem Fertigereignis wird die belegte Maschine r_1 freigegeben, ein neuer Prozess kann gestartet werden. Ein Beispiel für eine Meldungsverzögerung stellt das Ereignis z_7^{msg1} dar, das die Meldung um $b^{msgdelay}$ verzögert.

5 Anwendungen der Simulationsumgebung

Die vorgestellte Simulationsumgebung kann in unterschiedlichen Szenarien ihren Einsatz finden. Sie ist auf keine spezielle Branche beschränkt, es können unterschiedliche Produktionsarten und Prozesse inkl. Logistik abgebildet werden.

5.1 Anwendung bei einem mittelständischen Elektronikfertiger

Die Simulationsumgebung wurde mit realen Produktionsdaten eines mittelständischen Unternehmens aus der Elektronikfertigung evaluiert. Das Produktionsspektrum dieses Unternehmens reicht von Platinen bis zu kompletten Steuergeräten incl. Gehäuse. Der Maschinenpark umfasst insgesamt 200 Maschinen und manuelle Arbeitsplätze. Ziel war es eine Langfristanalyse von Planungsalgorithmen durchzuführen und deren Nutzen für das Unternehmen nachzuweisen. Als Zielfunktion der Bewertung sollte die Gesamtverspätung aller Aufträge verwendet werden, da hier nach Meinung des Unternehmens die größten Probleme liegen. Der Simulationshorizont wurde auf ein Jahr festgelegt. Zur Parametrierung der Simulationsumgebung konnten repräsentative Produktionsdaten über einen Zeithorizont von vier Wochen aus dem eingesetzten ERP- und BDE-System entnommen werden. Mittels statistischer Analysen wurden zunächst aus den Produktionsdaten die Parameter für das ERP- und BDE-Substitut abgeleitet.

Parameter für das ERP- und BDE-Substitut

Für das ERP-Substitut wurden aus dem Auftragsbestand der vier Wochen Auftragsschablonen extrahiert. Da keine Änderungshistorie verfügbar war, wurde der Anteil nachträglich geänderter oder gelöschter Aufträge jeweils mit 5% angenommen.

Für das BDE-Substitut wurde der Anteil an Maschinenstörungen (maschinenabhängig zwischen 5% und 25%) aus den Produktionsdaten errechnet. Außerdem konnten Informationen über Materialverluste und Fehler in den Prozessspezifikationen gewonnen werden. Der Anteil fehlerhafter Meldungen wurde mit 20% hoch angesetzt, da das Werkstattpersonal der Meldungspflicht bisher nicht sehr konsequent nachgeht. Teilweise liegt dies daran, dass häufig ungelerntes Personal für einfache Tätigkeiten eingesetzt wird. Die Parameter für Meldungsverluste oder Meldungsverspätung wurden auf 5% geschätzt.

Statistische Signifikanz

Um sicherzustellen, dass die ermittelten Simulationsergebnisse auch aussagekräftig sind und keine zufälligen Ausreißer darstellen, ist zunächst zu ermitteln, wie viele Wiederholungen der Simulationsexperimente durchgeführt werden müssen. Dies ist einerseits notwendig, da auch Algorithmen untersucht werden sollten, die stochastische Komponenten wie z.B. Genetische Algorithmen enthalten. Andererseits aber auch, da das BDE-Substitut den Produktionsplan, der vom Scheduler erstellt wurde, durch die verschiedenen Parameter zufällig variiert.

Es muss ermittelt werden, aus wie vielen Wiederholungen oder Durchläufen ein repräsentativer Mittelwert gebildet werden kann. Zur Ermittlung der Anzahl an Wiederholungen wurden Pilotsimulationen mit verschiedenen Scheduling-Algorithmen durchgeführt. Abbildung 6 zeigt beispielhaft den Verlauf der Entwicklung des Mittelwerts der Gesamtverspätung in Abhängigkeit von der Anzahl der Wiederholungen.

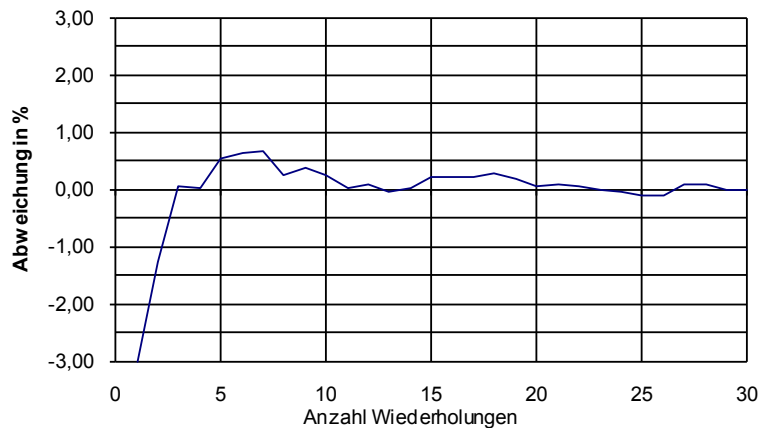


Abbildung 6: Entwicklung der Abweichung des Mittelwerts der Gesamtverspätung

Der Verlauf zeigt, dass der Mittelwert zwischen 10 und 30 Wiederholungen nie mehr als 0,5% vom Gesamtmittelwert nach 30 Wiederholungen abweicht. Als Kompromiss zwischen der Exaktheit der Ergebnisse und der notwendigen Rechenzeit wurde der Mittelwert für die Simulationsexperimente aus 10 Wiederholungen gebildet.

Simulationsergebnis

In der Abbildung 7 ist beispielhaft der Vergleich zwischen drei unterschiedlich parametrisierten genetischen Algorithmen visualisiert. Zum Zeitpunkt 1800 oder 3600 hat sich die Aufgabenstellung geändert. D.h. dass beispielweise vom EPR-Substitut ein neuer Auftrag freigegeben wurde. Bei der Parametrierung „Neuinitialisierung“ wird die komplette Population vollständig neu zufällig initialisiert. Die Lösungssuche beginnt damit von vorne. Bisherige Optimierungsergebnisse werden dabei nicht weiter verwendet. Der Graphik ist zu entnehmen, dass die erreichte Verbesserung deutlich schlechter ist als die der anderen Algorithmen. Bei der „Zufallsgesteuerten Übernahme“ werden 50% der Population durch Mutation des bisher besten Plans unter Berücksichtigung der geänderten Aufgabenstellung erstellt, der Rest zufällig. Bei der „Vollständigen Aktualisierung“ wird die gesamte Population aktualisiert. Dabei werden alle Individuen der Population um die neuen Informationen aus dem ERP- oder BDE-Substitut angereichert und bewertet. Somit geht der bisherige Fortschritt der Optimierung nicht verloren. Wie aus der Abbildung zu ersehen ist, optimiert dieser Algorithmus den Plan schneller und findet dabei eine bessere Lösung.

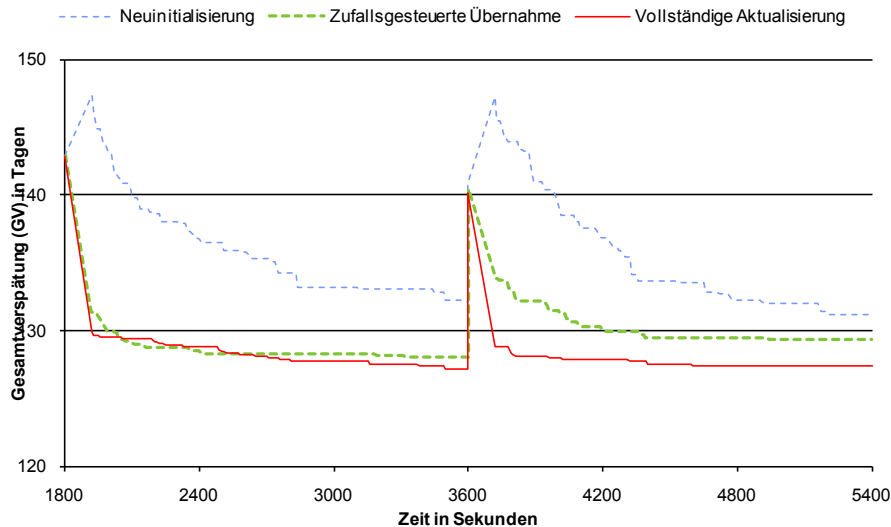


Abbildung 7: Vergleich von drei unterschiedlich parametrisierten Genetischen Algorithmen

Insgesamt konnten dem Unternehmen verschiedene Varianten von Such-Algorithmen vorgeschlagen werden, durch deren Einsatz sich die Termineinhaltung gegenüber der derzeitigen Situation (ohne Scheduler) deutlich verbessern lässt. Im Vergleich zu einer Planung mit der Prioritätsregel „Kürzeste Pufferzeit“ (KPZ) kann die Gesamtverspätung noch um ca. 25% verbessert werden. Andere einfache Prioritätsregeln z.B. Earliest Due Date (EDD) (vgl. [Ha89]) schnitten bei der Aufgabenstellung noch schlechter ab.

5.2 Weitere Einsatzszenarien

Die Simulationsumgebung ist unmittelbar zum Test der Scheduling-Komponente von ME-Systemen einsetzbar, die über eine B2MML Schnittstelle verfügen. Da die Simulationsumgebung aus zwei eigenständigen Komponenten besteht, ist es möglich, nur das BDE-Substitut oder ERP-Substitut zu verwenden. Das BDE-Substitut kann eingesetzt werden, wenn ein Systemhaus beispielsweise sein ME-System an ein weiteres ERP-System anbinden möchte. Mithilfe des BDE-Substituts kann dann die Schnittstelle unter Berücksichtigung des gesamten Arbeitslaufs im Labor getestet werden.

Ein anderes Anwendungsszenario ergibt sich, wenn ein Unternehmen ein neues Werk mithilfe eines digitalen Modells als Digitale Fabrik aufbaut, noch bevor die realen Gebäude und Anlagen existieren. Dann kann auf der Basis des Fabrikmodells und des ERP-Substituts eine geeignete Steuerungsstrategie und hierfür geeignetes ME-System ausgewählt werden.

Bei ME-Systemen handelt es sich um IT-Systeme mit Echtzeitanforderungen. Erschwerend kommt hinzu, dass Fluten von Meldungen aus der Produktion verarbeitet werden müssen, die die Basis der Feinplanung verändern können. Hier unterstützt die Simulationsumgebung auch, um weit vor der Inbetriebnahme die Konzepte für einen hochdynamischen Datenaustausch und die Verarbeitung der Daten in einer frühen Phase der Anwendungsentwicklung zu entwerfen und zu testen.

6 Fazit

Diese Abhandlung beschreibt eine Simulationsumgebung, die zur Analyse und zum Test von Scheduling-Lösungen in der Produktion eingesetzt werden kann. Diese Umgebung generiert Ereignisse, die in einem ISA-95 konformen Schnittstellenformat abgelegt und vom Scheduler verarbeitet werden. Dadurch ist es möglich den Scheduler unter realen Bedingungen dynamischen Tests zu unterziehen. D.h. dass Aufträge hinzukommen oder wegfallen, Arbeitsvorgänge gestartet, beendet, verzögert werden. Maschinen fallen aus oder Material ist nicht verfügbar. Alle diese Ereignisse verändern die Planungsgrundlage des Schedulers, wodurch der reale Einsatz eines Schedulers in der Produktion nachgebildet wird. Damit wird es möglich, den Scheduler zu parametrieren und Einstellungen zu testen, ohne dass der reale Produktionsbetrieb davon in Mitleidenschaft gezogen wird. Die Optimierung der Parametrierung erfolgt über die gezielte Variation von Suchparametern, Zielfunktionseinstellungen sowie Rechenkapazitäten. Voraussetzung für die Güte der Parametrierung des Schedulers ist jedoch, dass die Nachbildung der Produktionsereignisse möglichst gut den realen Produktionsbetrieb widerspiegelt.

Die vorgestellte Simulationsumgebung besteht aus den beiden Komponenten ERP-Substitut und BDE-Substitut. Das ERP-Substitut vertritt im Labortest das ERP-System, das den Scheduler mit Aufträgen und deren Änderungen versorgt. Das BDE-Substitut steht stellvertretend für die eigentliche Produktion, von der der Produktionsfortschritt und -zustand gemeldet wird. Der Produktionsprozess wird aus statistischen Erhebungen simuliert. Es werden Start-, Fortschritt- und Fertigmeldungen für Arbeitsvorgänge generiert. Dabei kommen Störmodelle zum Einsatz, die über statistische Verteilungen Maschinenausfälle oder Lieferverspätungen von Materialien einbringen. Daneben können auch Wahrscheinlichkeiten für Kommunikationsausfälle, Meldungsverluste oder -verspätungen vorgegeben werden.

Die Simulationsumgebung kann eingesetzt werden, um die Kosten und Risiken der Einführung eines Schedulers in der Produktion zu reduzieren. Systemanbieter können damit ihre Systeme im Vorfeld der Inbetriebnahme demonstrieren. Dies kann den Nutzen einer Scheduling-Lösung bei potentiellen Kunden transparenter machen. Insbesondere wenn über eine geeignete Bewertungsfunktionalität das Produktionsergebnis mit und ohne Scheduler verglichen werden kann. Damit kann der Kundennutzen besser kommuniziert werden und gleichzeitig die Voraussetzung für die erfolgreiche Einführung der Lösung geschaffen werden.

Literaturverzeichnis

- [B2M08] Business To Manufacturing Markup Language, B2MML, Version 0401. World Batch Forum, 2008.
- [Ba08] Bangsow, S.: Fertigungssimulation mit Plant Simulation und SimTalk: Anwendung und Programmierung mit Beispielen und Lösungen. Hanser, München; Wien, 2008.
- [BD08] Baumann, M.; Dimitrov, T.: Hierarchisierte Modelle in Advanced Planning and Scheduling (APS). In: at:Automatisierungstechnik, Oldenbourg Verlag, 02/2008, S. 90-97.
- [CF09] Curry, G.L.; Feldman R.M.: Manufacturing Systems Modeling and Analysis. Berlin, Heidelberg, Springer-Verlag, 2009.
- [DKT95] Dorn, J.; Kerr, R.; Thalhammer, G.: Reactive Scheduling - improving the robustness of schedules and restricting the effects of shop floor disturbances by fuzzy reasoning. In: International Journal of Human-Computer Studies, 42, 1995, S. 687 - 704.
- [Fe97] Feindt, E.G.: Entwurf und Simulation industrieller Steuerungen für den PC und die SPS: Eine Einführung für Informatiker und Automatisierungstechniker. Oldenbourg, München; Wien, 1997.
- [Ha89] Haupt, R.: A survey of priority rule-based scheduling. In: OR-Spektrum, Vol. 11, 1989, S. 3-16.
- [ISA00] ANSI/ISA-95.00.01-2000, Enterprise-Control System Integration, Part 1: Models and Terminology. ISA, 2000.
- [ISA01] ANSI/ISA-95.00.02-2001, Enterprise-Control System Integration, Part 2: Object Model Attributes. ISA, 2001.
- [Kl06] Kletti, J.: MES Manufacturing Execution System. Springer-Verlag, 2006.
- [Kü06] Kühn, W.: Digitale Fabrik: Fabriksimulation für Produktionsplaner. Hanser, München; Wien, 2006.
- [MES97] MESA: MES functionalities & MRP to MES data flow possibilities. MESA International, White Paper 2, 1997.
- [Mo93] Moser, M.: Regelung der Maschinenbelegung in der flexiblen Fertigung. VDI-Verlag, Düsseldorf, 1993.
- [Sc06] Scheer, A.W.: Prozessorientiertes Product Lifecycle Management. Springer, Berlin; Heidelberg, 2006.
- [Sm94] Smith, S.F.: OPIS: A Methodology and Architecture for Reactive Scheduling. In: Zweben, M.; Fox, M.S.: Intelligent Scheduling. San Francisco, Morgan Kaufman, 1994, S.29-66.
- [TMF08] Thiel, K.; Meyer, H.; Fuchs, F.: MES – Grundlage der Produktion von morgen. Oldenbourg Industrieverlag, 2008.
- [Ze92] Zell, M.: Simulationsgestützte Fertigungssteuerung. Oldenbourg, München, 1992.