

# INTERPRETATION DER KOPPELUNGSMÖGLICHKEITEN ZWISCHEN OPTIMIERUNGSAUFGABEN UND SIMULATIONSMODELLEN AM BEISPIEL VON GESCHÄFTSPROZESSEN

Christian Müller  
TH Wildau, FB Wirtschaft, Informatik, Recht  
Hochschulring 1, D-15745 Wildau, Deutschland  
E-mail: [christian.mueller@th-wildau.de](mailto:christian.mueller@th-wildau.de)

**Keywords:**

Simulation, Optimierung, Geschäftsprozesse

[DesmoJ (2013)], [Page (2005)].

**Abstrakt:**

Geschäftsprozesse in modernen Unternehmen zeichnen sich durch eine hohe Komplexität aus, die sich z.B. aus sich schnell verändernden Marktsituationen, verkürzten Produktlebenszyklen bzw. dynamischen Wechselwirkungen zwischen einzelnen Subsystemen ergeben. Das Geschäftsprozessmanagement hat das Ziel möglichst effiziente und kundenorientierte Prozesse zu implementieren. Klassischer Weise werden dazu analytische kennzahlenbasierte Verfahren verwendet. Im Gegensatz dazu sollen hier die Geschäftsprozesse mit Simulationsverfahren untersucht werden. Auf diese Weise können die Prozesse auch schon vor ihrer Implementation beurteilt werden.

Da reale Geschäftsprozesse in der Regel spezifische Entscheidungsprobleme beinhalten, die durch Optimierungsansätze gelöst werden können, erscheint es sinnvoll, eine Simulation von Geschäftsprozessen mit einer Optimierungskomponente zu verbinden [Steglich, Müller (2013)]. Andererseits liegen Geschäftsprozessmodelle oft in parametrisierter Form vor, so dass sich die Frage nach einer optimalen Wahl der Parameter stellt. [März et.al. (2010), p 3ff]. Somit stellt sich die Frage nach den möglichen Konstellationen zwischen Optimierungs- und Simulationsaufgabe und deren Interpretation.

Dazu entwickelten März und Krug [März et al. (2010), Kap. 5] ein Klassifikationsschema, welches die grundlegende Kopplungsarten zwischen Simulation und Optimierung beschreibt. Dieser Artikel diskutiert anhand einfacher Beispiele, welche sich aus Geschäftsprozessmodellen ableiten lassen, diese Koppelungsmöglichkeiten. Er baut auf den Überlegungen aus [Müller, Chr (2013)] auf, ergänzt diese durch sinngebende Interpretationen und weist auf Schwachstellen hin.

Zur Illustration der Beispiele wurde der EPC Simulator [EPC Simulator (2013)], [Müller, Chr (2012a), (2012b)] verwendet. Dieser ist ein Plugin des Modellierung-Werkzeugs Bflow [Bflow (2013)], [Kern et al. (2010)] und nutzt das DesmoJ Simulations- Framework

**Das Klassifikationsschema von März**

In [März et al. (2010)] wird das Zusammenspiel von Simulation und Optimierung anhand von industriellen Fallbeispielen untersucht. Sie klassifizieren das Zusammenspiel auf vier grundlegende Kopplungsarten.

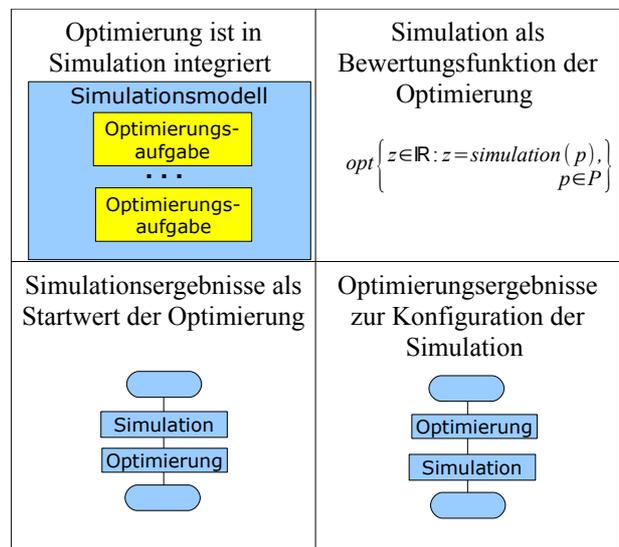


Abbildung 1: Klassifikationsschema von März und Krug

Dabei unterscheiden sie im wesentlichen zwischen sequentieller und hierarchischer Verknüpfung, der zeitlichen Abfolge der Berechnungen und deren gegenseitigen Abhängigkeiten. Ebenfalls sind Kombinationen zwischen diesen Grundtypen denkbar.

**Beispiel: Optimierung eines Bedienprozesses**

Ein Supermarkt ist von 8:00 bis 20:00 geöffnet. Die Verteilung der Zeitabstände, in denen die Kunden den Supermarkt betreten, ist vorgegeben. Jeder Kunde schaut nach dem Betreten des Geschäfts in den Kassenbereich. Wenn dort an jeder Kasse mehr als drei Kunden warten, dann verlässt er das Geschäft wieder. Es wird angenommen, dass jeder Kunde einen durchschnittlichen Erlös von € 5,- generiert und jeder Kassierer € 20,- pro Stunde kostet. Es soll die optimale Anzahl von Kassierern bestimmt werden, so dass der

Gesamterlös des Geschäfts maximiert wird.

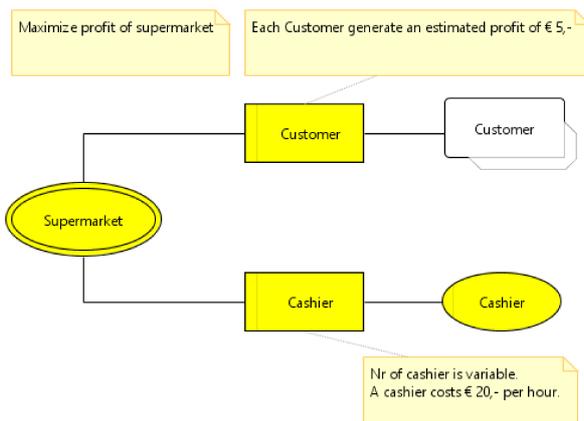


Abbildung 2: Modelldokument Bedienprozess

Das Simulationsmodell wird durch das Modelldokument (Abb. 2) und das Prozessdokument (Abb. 3) beschrieben.

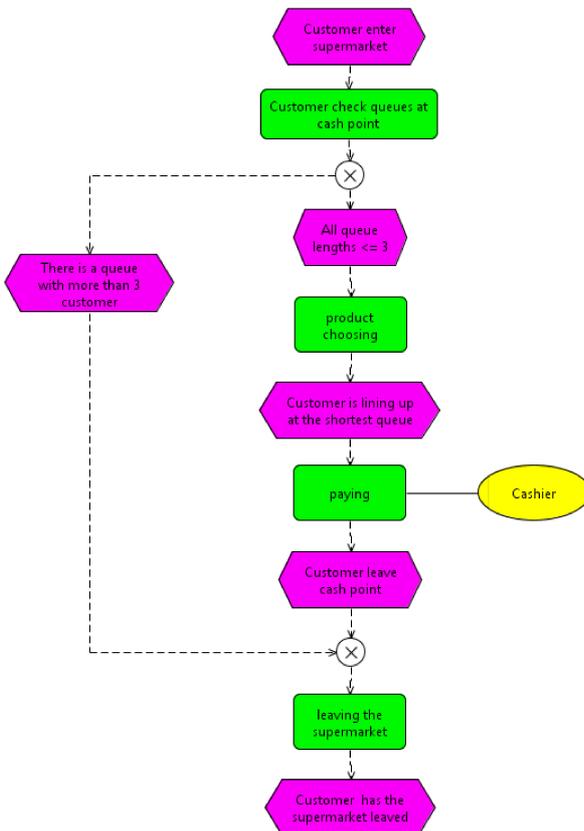


Abbildung 3: Prozessdokument Bedienprozess

Der erwartete Erlös kann mit dem Simulationsmodell, nach mehrmaliger Ausführung zu einer vorgegebenen Anzahl von Kassieren, bestimmt werden.

Mit einem externen Optimierungssystem kann nun die optimale Anzahl von Kassieren bestimmt werden. Damit ist das Simulationsmodell der Optimierungsaufgabe hierarchisch untergeordnet. Das Beispiel fällt damit in die Klasse „Simulation als Bewertungsfunktion

für die Optimierung“.

**Beispiel: Ein einfaches Fahrzeunavigationsmodell:** In einem ungerichteten Straßennetzwerk  $G = (V, A)$ , wobei  $V$  die Knoten und  $A$  die Kantenmenge ist, fahren Fahrzeuge von zufälligen Start zu zufälligen Zielknoten (Abbildung 4). Die Passierdauer auf einer Kante  $(i,j)$  ist normal verteilt, wobei die mittlere Passierdauer  $dist_{i,j} = ft_{i,j} + vt_{i,j} \cdot q_{i,j}$  abhängig von der aktuellen Anzahl der Fahrzeuge  $q_{i,j}$  auf dieser Kante  $(i,j)$  ist.

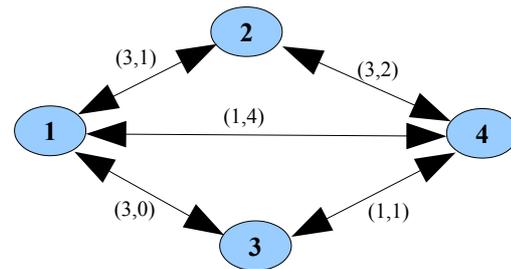


Abbildung 4: Strassennetzwerk  $G = (V,A)$  mit Kantenbewertungen  $(ft_{i,j}, vt_{i,j})$

Bei der Fahrt eines Autos wird zur Wahl der nächsten Kante, basierend auf der mittleren Passierdauer, in jedem Knoten ein kürzestes Wege Problem gelöst. Mit diesem Modell kann der Einfluss der lokalen Navigation in jedem Fahrzeug auf den globalen Verkehrsfluss studiert werden.

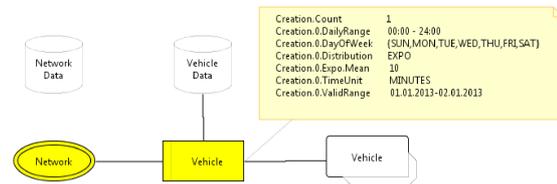


Abbildung 5: Modelldokument Fahrzeugnavigation

Um mit dem EPC Simulator ein Simulationsmodell zu generieren wird ein Modelldokument (Abb. 5) und ein Prozessdokument (Abb. 6) erstellt.

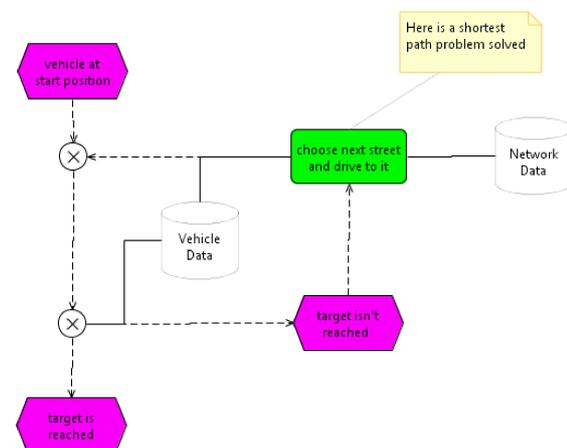


Abbildung 6: Prozessdokument Fahrzeugnavigation

In dem Modelldokument ist eine globale Datenstruktur „NetworkData“ deklariert. Insbesondere sind dort das Netzwerk, die Kantenbewertungen und die Anzahl der Fahrzeuge auf den einzelnen Kanten gespeichert.

Für jedes Fahrzeug werden die individuellen Fahrzeugdaten, dies sind insbesondere die aktuelle Position, das Ziel und eine Methode zur Berechnung des kürzesten Weges zwischen 2 Knoten des Netzwerks, in der Vehikel Datenstruktur gespeichert. Das Verhalten der Fahrzeuge ist dem Prozessdokument (Abbildung 4) beschrieben. Das Fahrzeug fährt von Knoten zu Knoten bis sein Ziel erreicht ist. In jedem Knoten wird ein Kürzestes Wege Problem gelöst, aus dessen Lösung die nächste zu befahrende Kante abgeleitet wird.

Das einzubindende Kürzeste Wege Problem [Hiller et al. (2010)] lässt sich mathematisch als:

$$\sum_{(i,j) \in A} c_{i,j} \cdot x_{i,j} \rightarrow \min!$$

s. t.

$$\sum_{(i,j) \in A} x_{i,j} - \sum_{(j,i) \in A} x_{j,i} = \begin{cases} 1 & \text{if } i=s \\ -1 & \text{if } i=t \\ 0 & \text{otherwise} \end{cases} \quad \text{für alle } i \in V$$

$$x_{i,j} \geq 0; \text{ für alle } (i,j) \in A$$

formulieren.

Bei diesem Modell ist die „Optimierung in das Simulationsmodell integriert“. Jedes Fahrzeug bestimmt in jedem Knoten individuell, basierend auf der globalen Verkehrssituation, seinen optimalen Weg. Damit beeinflusst es selbst die globale Verkehrssituation. Somit sind die Optimierungsaufgaben dem Simulationsmodell hierarchisch untergeordnet und es besteht eine direkte Rückkoppelung der Lösung der Optimierungsaufgabe in das Simulationsmodell.

### Beispiel: Ein einfaches Job Shop Problem

In einer Werkstatt liegen die Aufträge eines Tages vor. Die Werkstatt hat 5 Mitarbeiter, die 8 Std. pro Tag, also zusammen 2400 Minuten pro Tag arbeiten. Es gibt fünf Auftragsstypen wobei jeder Typ eine Planzeit hat. Die Aufträge des Tages überschreiten, gemessen an ihren Planzeiten, die Tagesarbeitszeit von 2400 Minuten nicht. Die Mitarbeiter sind spezialisiert. Ihre Bearbeitungszeiten sind normal verteilt mit individuellen Verteilungsparametern  $(\mu, \sigma)$ .

Auftrags- typ	Bearbeitungs- zeit [min] $N(\mu, \sigma)$	Mitarbeiter					Plan- zeit [min]
		1	2	3	4	5	
A	(15,5)	(20,7)	(30,10)	(40,15)	(45,15)	30	
B	(60,10)	(45,10)	(70,10)	(80,15)	(90,20)	60	
C	(80,15)	(80,20)	(60,10)	(90,10)	(100,10)	90	
D	(130,10)	(140,10)	(120,10)	(100,15)	(90,10)	120	
E	(170,40)	(160,30)	(150,10)	(120,30)	(130,20)	150	

Tabelle 1: Job Shop Problem

Morgens werden jedem Mitarbeiter von einem Manager die von ihm zu bearbeitenden Aufträge zugewiesen.

Dies ist eine stochastische Optimierungsaufgabe [Kall et al. (1994)], [Birge et al. (1997)]. Diese wird hier mit dem „expected-value“ Ansatz [Madanski (1960)] bearbeitet. Dabei wird die stochastische Optimierungsaufgabe in eine deterministische überführt, indem die Verteilungen durch ihre Erwartungswerte ersetzt werden. Anschließend werden durch eine nachgelagerte Simulation die Auswirkungen dieser Vereinfachung überprüft.

Das Modelldokument für diese Aufgabe ist in Abbildung 7 gegeben.

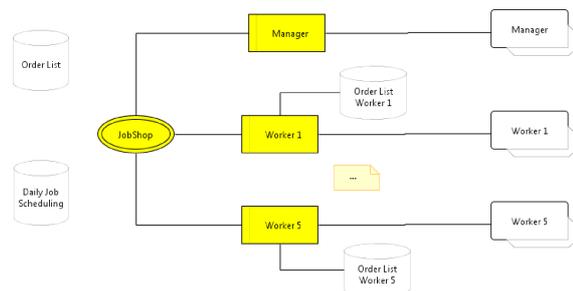


Abbildung 7: Modelldokument Job Shop Problem

In der Datenstruktur Order List stehen die Aufträge, in Daily Job Scheduling die von dem Manager vorgenommene Zuordnung der Aufträge auf die Mitarbeiter. In den Order List Worker <i> befinden sich die individuellen Arbeitspläne der Mitarbeiter. Die Planungstätigkeit des Managers ist in Abbildung 8 beschrieben.

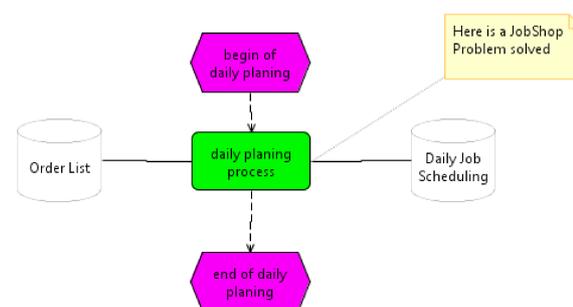


Abbildung 8: Prozessdokument Manager im Job Shop Problem

Dabei muss er ein Job Shop Problem lösen. Das Planungsergebnis, die Zuordnung der Aufträge auf die Mitarbeiter wird in der Daily Job Scheduling Datenstruktur gespeichert.

Das deterministische Job Shop Problem [Domschke et al. (1997)] hat folgende Form:

Indexmengen:

- $a \in A$  Menge der Aufträge eines Tages
- $m \in M$  Menge der Mitarbeitern
- $t \in T$  Menge der Auftragsstypen

Daten:

$$at(a, t) = \begin{cases} 1 & \text{wenn Auftrag } a \in A \text{ vom Typ } t \in T \text{ ist.} \\ 0 & \text{sonst} \end{cases} \quad \text{für alle } a \in A \text{ und } t \in T$$

$$\mu(t, m) = \text{mittlere Bearbeitungszeit eines Auftrages vom Typ } t \in T \text{ bei Mitarbeiter } m \in M$$

Entscheidungsvariablen:

- freie *arbeitszeit*

$$- \text{ SOS } \quad x(a, m) = \begin{cases} 1 & \text{Auftrag } a \in A \text{ wird von Mitarbeiter } m \in M \text{ ausgeführt} \\ 0 & \text{sonst} \end{cases} \quad \text{für alle } a \in A \text{ und } m \in M$$

$$(\text{SOS Bedingung}) \quad \sum_{m \in M} x(a, m) = 1 \quad \text{für alle } a \in A$$

Ziel: *min arbeitszeit*

Restriktionen:

$$\text{Arbeitszeit} \quad \sum_{\substack{a \in A \\ t \in T}} at(a, t) \mu(t, m) x(a, m) \leq \text{arbeitszeit} \quad \text{f.a. } m \in M$$

In dieser Formulierung werden die Aufträge zeitlich gleichmäßig auf alle Mitarbeiter verteilt, so dass die von einem beliebigen Mitarbeiter maximal zu leistende Arbeitszeit minimal wird.

Nach Abschluss der Planung arbeiten die Mitarbeiter ihre Pensen (Abbildung 9) ab.

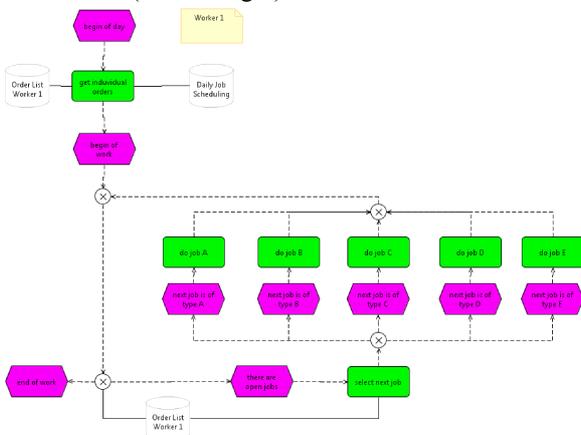


Abbildung 9: Prozessdokument Worker in Job Shop Problem

In diesem Beispiel wird auf der Basis der erwarteten Bearbeitungszeiten ein deterministisches Optimierungsmodell gelöst, da von der Zufälligkeit der Bearbeitungszeiten abstrahiert wird. Anschließend wird die Lösung des Optimierungsmodells mit einer Simulation überprüft. Hier besteht keine Wechselwirkung zwischen Optimierungsaufgabe und Simulationsmodell. Beide Schritte sind sequentiell angeordnet.

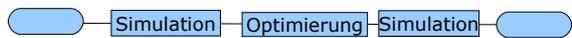


Wird die Aufgabe derart modifiziert, dass die Aufträge mittels Simulation erzeugt werden und anschließend die

Planungsaufgabe gelöst wird, dann erhalten wir bei Vernachlässigung der nachgelagerten Simulation die sequentielle Struktur:



Ohne Vernachlässigung der nachgelagerten Simulation erhalten wir:



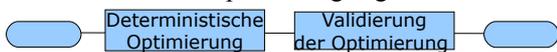
Auch diese Struktur ist in dem Klassifikationsschema als eine Kombination der Grundtypen vorgesehen.

### Interpretation der Beispiele in Bezug auf das Klassifikationsschema

Die oben dargestellten Beispiele decken alle Grundtypen des Klassifikationsschemas ab. Grundsätzlich sind die beiden hierarchischen Fälle

- *Simulation als Bewertungsfunktion der Optimierung* und
- *Optimierung ist in die Simulation integriert* zu unterscheiden. Die sequenziellen Fälle
- *Simulationsergebnis als Startwert der Optimierung* und
- *Optimierungsergebnisse zur Konfiguration der Simulation*

sind einerseits Spezialfälle des hierarchischen Falls, „Optimierung ist in die Simulation integriert“ und andererseits legen die beiden Job-Shop Beispiele auch eine Interpretation als stochastische Validierung eines deterministischen Optimierungsergebnisses



bzw. als Modellierungssystem mit zufälligen Auftragsdaten für das Job Shop Problem nahe.



März und Krug beschreiben die stochastische Validierung als „Überprüfung der Machbarkeit der vorgeschlagenen Lösung“ [März et al. (2010) p44]. Bei dem Begriff Modellierungssystem folgen wir Kallrath [Kallrath (2004)], der eine Übersicht über kommerziell verfügbare Systeme gibt. Für Open Source System sei auf Koch [Koch (2004)] und Steglich, Schleiff [Steglich, Schleiff (2010)] verwiesen. Das Modellierungssystem enthält einen stochastischen Aspekt, wenn einige Daten des Modells durch Zufallsvariablen beschrieben werden und das Modellierungssystem zum Erstellen eines deterministischen Modells konkrete Zufallsvariablen zieht.

In beiden Fällen liegt eine sequentielle Abfolge, eine charakterisierende Eigenschaft dieser Spezialfälle, vor. Somit bieten sich folgende Umbenennungen im Klassifikationsschema an:

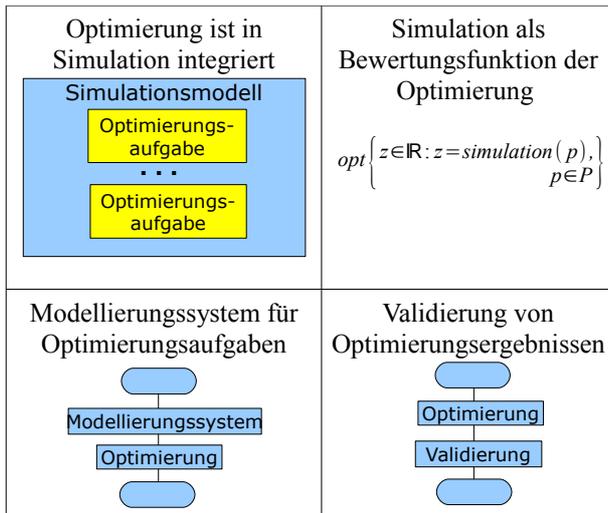


Abbildung 10: Modifiziertes Klassifikationsschema

Um die Grenzen dieser Klassifikation aufzuzeigen soll abschließend eine geringfügige Modifikation des einfachen Job Shop Problems (Abb. 11) diskutiert werden. Dabei wird als Simulationszeitraum nicht nur ein Tag, sondern eine Folge von Tagen betrachtet. Falls in dem Verifikationsschritt eines Tages festgestellt wird, das nicht alle für diesen Tag geplante Aufträge in der zulässigen Arbeitszeit erledigt werden können, dann werden sie auf den nächsten Tag verschoben. Durch diese naheliegende Regel entsteht eine Rückkoppelung von dem letzten in den ersten Simulationsschritt, die auch Konsequenzen auf die noch zu vergebenden Auftragstermine haben kann. Somit liegt nun der Fall „Optimierung ist in Simulation integriert“ vor.

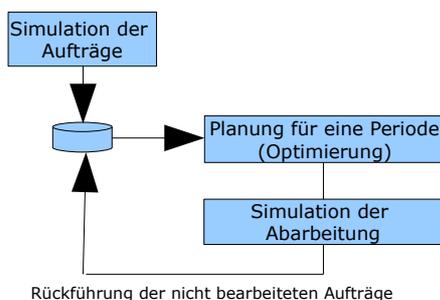


Abbildung 11: Modellvariation mit Klassenwechsel

Um solche Klassifikationswechsel durch geringfügige Modellvariationen zu vermeiden, bietet es sich an, nicht nur formalen Kriterien, sondern auch den Anwendungskontext zu betrachten. Bei den Job-Shop Beispielen dominiert der Anwendungskontext und deshalb sollte sie der Gruppe „Optimierung ist in die Simulation integriert“ zugeordnet werden. Falls allerdings der Validierungsaspekt bzw. der Modellierungsaspekt einer Optimierungsaufgabe überwiegen, dann sollte die entsprechende sequentielle Klasse betrachtet werden. In diesem Fall sind dann auch keine Modellvariationen zu erwarten.

März et. al. klassifizierten in [März et al.(2010)] 12 industrielle Praxisbeispiele aus dem Umfeld der Produktionsplanung, wobei neun Beispiele aus der Klasse „Simulation als Bewertungsfunktion“ und zwei in die Gruppe „Optimierung ist in die Simulation integriert“ kamen. Damit zeigt sich die große Bedeutung der optimalen Parameterwahl in der Praxis. Ferner gab es ein sequentielles Beispiel aus der Klasse „Simulationsergebnisse als Startwert der Optimierung“. In diesem Produktionsplanungsbeispiel wurden mittels Simulation Kapazitätsanforderungen ermittelt, um im Nachgang diese Kapazitätsbedarfe mit konkreten Betriebsmitteln zu verknüpfen [März et al.(2010) p 44]. Auch dieses Beispiel lässt sich durch täglich variierende Kapazitätsanforderungen und teilweise nicht ausreichende Betriebsmittel leicht zu einem nicht sequentiellen Modell erweitern. Bei Einordnung entsprechend des Modellkontextes in die Gruppe „Optimierung ist in Simulation integriert“ kann auch hier ein Klassifikationssprung vermieden werden.

### Zusammenfassung:

Ausgehend von dem Klassifikationsschema von März und Krug wurden einfache Beispiele vorgestellt, die die entsprechenden Koppelungsmöglichkeiten illustrieren. Dabei wurde festgestellt, dass die Klasse „Optimierung ist in Simulation integriert“ eine Oberklasse der beiden sequentiellen Klassen ist. Diese Klassen können allerdings als spezielle Modellierungs- bzw. Validierungs- Aufgabe interpretiert werden und erhalten damit eine eigene methodische Bedeutung. Praxisbeispiele die diese sequentielle Struktur aufweisen, verlieren allerdings bei naheliegenden Modifikationen diese Struktur. Deshalb ist es sinnvoll Praxisaufgaben bei denen die sequentielle Struktur nicht charakteristisch ist, gleich in die größere Klasse einzuordnen.

Aus Sicht der industriellen Praxis gehören die überwiegende Anzahl der auftretenden Kombinationen in die Klasse „Simulation als Bewertungsfunktion der Optimierung“. Diesen Schluss legen die von März und Krug [März et al. (2010)] vorgestellten Praxisbeispielen nahe.

### Literatur:

- Birge, Louveaux 1997 „Introduction to stochastic programming“ Springer Berlin Heidelberg
- Bflow 2013 „bflow\*-toolbox“ Projekt Homepage <http://www.bflow.org/>
- DesmoJ 2013 “A Framework for Discrete-Event Modelling and Simulation” <http://desmoj.sourceforge.net/home.html>
- EPC Simulator 2013 „EPC Simulator Projekt Homepage“ <http://www.tfh-wildau.de/cmuller/EpcSimulator/>
- Domschke, W.; Scholl, A.; Voß, S 1997 “Produktionsplanung”. Springer Berlin Heidelberg
- Hiller, Liebermann 2010 “Introduction to Operations Research” McGraw-Hill Higher Education
- Kall, Wallace 1994 “Stochastic Programming” John Wiley and Sons
- Kallrath, Josef 2004 “Modeling Languages in Mathematical

Optimization" Kluwer Academic Publishers  
 Kern, Kühne, Laue, Nüttgens, Rump, Storch 2010 "bflow\*  
 Toolbox - an Open-Source Business Process Modelling  
 Tool" Proc. of BPM Demonstration Track, Business  
 Process Management Conference 2010 (BPM'10),  
 Hoboken, USA  
 Koch, Thorsten 2013 "Zimpl" <http://zimpl.zib.de/>  
 Madanski 1960 „Inequalities for stochastic linear  
 programming problems“ Management Science 6:197-204,  
 1960  
 März et.al. 2010 „Simulation und Optimierung in Produktion  
 und Logistik“ Springer Berlin Heidelberg  
 Müller, Chr 2012a „Generation of EPC based simulation  
 models“ Proceedings 26th European Conference on  
 Modelling and Simulation 2012 [http://www.scs-  
 europe.net/dlib/2012/2012-0301.htm](http://www.scs-europe.net/dlib/2012/2012-0301.htm)  
 Müller, Chr 2012b „Generierung von Simulationsmodellen  
 aus ereignisgesteuerten Prozessketten“ Management und  
 IT, AKWI Jahrestagung 2012 [http://nbn-  
 resolving.de/urn/resolver.pl?urn=urn:nbn:de:kobv:526-  
 opus-1709](http://nbn-resolving.de/urn/resolver.pl?urn=urn:nbn:de:kobv:526-opus-1709)  
 Müller, Chr 2013 "Anmerkungen zur Klassifikation der  
 Koppelungsarten zwischen Optimierungsaufgaben und  
 Simulationsmodellen am Beispiel von Geschäfts-  
 prozessen" Integration und Konnektion, AKWI  
 Jahrestagung 2013 [http://nbn-resolving.de/urn/resolver.pl?  
 urn=urn:nbn:de:kobv:526-opus-1761](http://nbn-resolving.de/urn/resolver.pl?urn=urn:nbn:de:kobv:526-opus-1761)

Page, B et. al. 2005 "The Java Simulation Handbook" Shaker  
 Steglich M, Schleiff T 2010 „CMPL: Coliop Mathematical  
 Programming Language“ Wildauer Schriftenreihe -  
 Entscheidungsunterstützung und Operations Research,  
 Beitrag 1, Technische Hochschule Wildau [FH]  
 Steglich M, Müller, Chr 2013 „An approach to combine  
 simulation and optimization of business process“  
 Proceedings 27th European Conference on Modelling and  
 Simulation [http://www.scs-europe.net/dlib/2013/2013-  
 0808.htm](http://www.scs-europe.net/dlib/2013/2013-0808.htm)



**CHRISTIAN MÜLLER** studierte  
 Mathematik an der Freien Universität Berlin  
 und promovierte 1989 über Netzwerkflüsse  
 mit Nebenbedingungen. Von 1990 bis 1992  
 arbeitete er bei der Schering AG und von  
 1992 bis 1994 bei den Berliner Verkehrs-  
 betrieben (BVG) im Bereich Fahr- und Dienstplan  
 Optimierung. 1994 erhielt er einen Ruf an die  
 Technische Hochschule Wildau, Deutschland. Seine  
 Forschungsschwerpunkte sind die Konzeption von  
 Informationssystemen, Mathematische Optimierung und  
 die Simulation von Geschäftsprozessen. Seine Email  
 Adresse ist [christian.mueller@th-wildau.de](mailto:christian.mueller@th-wildau.de) und seine  
 Web Seite ist <http://www.th-wildau.de/cmuellet/> .