

Grafisches Werkzeug zur Erstellung von Optimierungsalgorithmen für komplexe Montagesysteme

***Graphical tool to create optimisation algorithm for complex assembly
lines***

Daniel Bohn, Evangelos Angelidis, Oliver Rose, Universität der Bundeswehr
München, Neubiberg (Germany), daniel.bohn@unibw.de,
evangelos.angelidis@unibw.de, oliver.rose@unibw.de

Abstract: In this paper we focus on solution strategies for scheduling problems of complex assembly lines with workforce constraints. Typical products are airplanes, industrial machines and turbines which are produced by multi-skilled resources in small series or are even unique items. This production is defined as a Multi-Mode Resource-constrained Multi-Project Scheduling Problem with activity splitting. It is a combinatorial order problem which is NP-hard. Simulation-Based Optimization offers an auspicious manner of dealing with those domain specific problems. In our paper we present a graphical tool to create optimization algorithm for such problems. Additionally we give an example of the main features and advantages of this modelling approach.

1 Einführung

In den letzten Jahrzehnten gewann die Produktionsplanung und -steuerung immer mehr an Bedeutung. Unternehmen sind gegenwärtig mehr denn je gezwungen, Produktionskosten zu minimieren, um sich im internationalen Markt zu etablieren. Schwerpunkt unserer Forschung bildet daher die Optimierung der Produktion von industriellen Maschinen in Kleinserien und von Unikaten. Da das Problem als NP-schwer klassifiziert ist, wie in Pappert et al. (2010) beschrieben, kann es nicht in kurzer Zeit mit klassischen Methoden gelöst werden. Unser Ziel ist es daher, unter Einsatz der simulationsbasierten Optimierung (SBO) und mit Hilfe verschiedener Heuristiken in kurzer Zeit ein passendes Szenario zu erstellen. Dafür wurde eine modellgetriebene SBO-Plattform (SBOP) entwickelt, die mit minimalen Entwicklungs- und Zeitkosten entsprechende Lösungsverfahren für Unternehmen bietet. Die vorliegende Publikation widmet sich der Analyse von SBOP und präsentiert einen grafischen Modellierungsansatz, um Fachanwender in die Lage zu versetzen, selbst Optimierungsalgorithmen zu entwickeln.

2 Problembeschreibung

Diese Publikation beschäftigt sich mit der Optimierung von komplexen Montagelinien zur Herstellung von bspw. Turbinen, Flugzeugen, industriellen Maschinen in Kleinserien und Unikaten. Hierbei besteht das Fabrikmodell aus unterschiedlichen Aufträgen. Die Aufträge wiederum bestehen aus verschiedenen Produkten mit unterschiedlichen Produktionsplänen, Rangfolgebeschränkungen, Montagecharakteristiken von Maschinen und Tausenden von Arbeitsvorgängen. Die unterschiedlichen Ausführungen der Arbeitsvorgänge werden mit Hilfe von sogenannten Modi beschrieben. Diese definieren die Menge der Qualifikationen der zugeordneten Ressourcen und die Bearbeitungszeit für die einzelnen Arbeitsvorgänge. Die Arbeitsvorgänge und Produkte haben viele zeitgebundene Anforderungen wie bspw. muss ein Teil des Produktionsplans in einer bestimmten Zeit abgearbeitet sein, damit der Klebstoff nicht austrocknet bevor die entsprechenden Werkstücke zusammengeführt sind. Ein weiteres Merkmal der Produktion ist die beschränkte Verfügbarkeit der Ressourcen (Schichten, Pausen und Ausfälle) und die endlichen Puffer. Ein vereinfachtes Fabrikmodell wird in Abbildung 1 aufgezeigt. Die Arbeitsvorgänge sind hier durch Verbindungslinien zwischen 2 Knoten (dargestellt als Kreis) abgebildet. Durch den Knoten wird bestimmt, welcher Arbeitsvorgang als nächstes abgearbeitet wird. Handelt es sich um einen Knoten mit paralleler Abarbeitung, werden alle nachfolgenden Arbeitsvorgänge bearbeitet. Bei einem alternativen Knoten wird nur ein bestimmter Arbeitsvorgang durchgeführt. Dem Arbeitsvorgang zugeordnete Ressourcen sind durch Quadrate unterhalb des Vorgangs abgebildet, hierbei wird auf die Unterscheidung nach Qualifikation verzichtet.

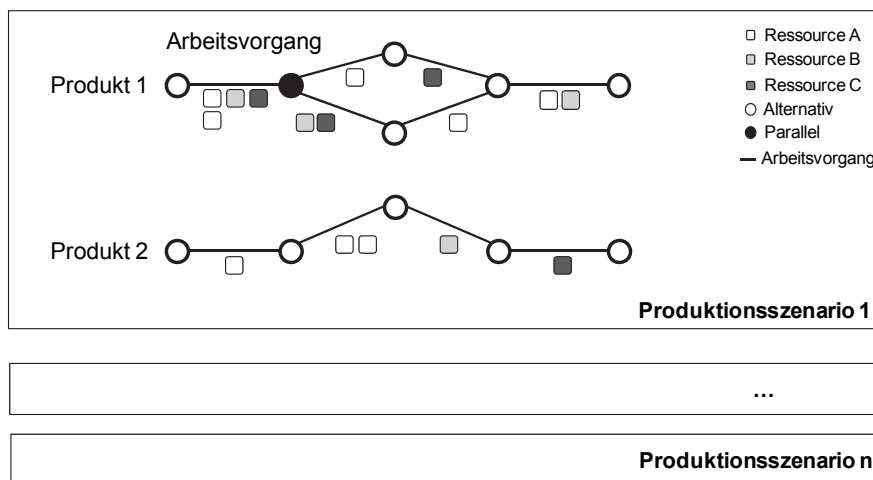


Abbildung 1: Fabrikmodell mit unterschiedlichen Produktionsszenarien

Die Optimierung solcher Montagelinien ist ein komplexes kombinatorisches Problem und wird in der Literatur häufig als Multi-Mode Resource-Constrained Multi-Project Scheduling Problem (MMRCMPSP) with activity splitting bezeichnet. Es handelt sich dabei um ein NP-schweres Problem, das in realen Dimensionen nicht

mit klassischen Ansätzen in kurzer Zeit gelöst werden kann. Die Forschung in diesem Bereich basiert vor allem auf kleinen oder mittelgroßen Problemen, die häufig durch exakte Ansätze oder genetische Algorithmen gelöst werden. Ein alternativer Weg, um eine Lösung in kurzer Laufzeit zu finden, ist der simulationsbasierte Optimierungsansatz wie er in (April et al. 2003; Law und McComas 2002; Pappert et al. 2010; Shapiro 1996) beschrieben ist. Die grundlegende Idee dieses Ansatzes ist es, aus dem Fabrikmodell ein Simulationsszenario zu definieren, es anschließend zu simulieren, zu analysieren und ein neues Szenario unter Berücksichtigung der vorliegenden abgeschlossenen Szenarien und des Fabrikmodells zu erstellen. Ziel dieser Iterationen ist es, innerhalb weniger Schritte eine zulässige Lösung zu finden. Dabei werden für die Erstellung des Simulationsszenarios verschiedene Techniken verwendet. Diese können zum Einen genetische Algorithmen und zum Anderen iterative Algorithmen sein (Pappert et al. 2010; Angelidis et al. 2012a; Angelidis et al. 2012b). Die Integration der Algorithmen in die Plattform (SBOP) geschah bisher auf Code-Ebene, was sich aufgrund der Komplexität des Algorithmus sehr schwierig und zeitaufwändig gestaltete. Des Weiteren wurden die ersten Entwürfe des Algorithmus auf Papier angefertigt, was wiederum eine Arbeit im Team sehr schwierig und eine Versionierung nahezu unmöglich machte.

3 Simulationsbasierte Optimierungsplattform

In der Optimierung von Montagelinien bedarf es neuer Lösungsstrategien, die mit riesigen Suchräumen umgehen können und innerhalb kurzer Laufzeiten ein passendes Szenario finden. Der erste Ansatz eines Frameworks, das sich mit dieser Problemstellung beschäftigt, wird in Pappert et al. (2010) beschrieben. Das nachfolgend beschriebene Konzept stellt eine Weiterentwicklung dar.

Die erste Herausforderung besteht darin, das reale System in ein sogenanntes Fabrikmodell zu transformieren. Dieses Modell beschreibt die Produktionsanlage und enthält alle notwendigen Informationen, um alle möglichen Produktionsszenarien abzubilden. Der Optimierer stellt basierend auf der Lösungsstrategie und der bereits simulierten Szenarien Simulationsmodelle zur Verfügung. Durch die Einbeziehung der Ergebnisse von vorangegangenen Szenarien wird versucht, eine Erhöhung der aktuellen Lösungsbewertung zu erzielen. Die Ergebnisse einer Simulation werden als Simulationslog bezeichnet.

Das Simulationsmodell enthält alle Produktionspläne mit angepasstem frühesten Start- und spätestem Endtermin. Diese werden als activity-on-arc (AOA) Netzwerke mit alternativen Produktionspfaden definiert. Die zu den Produktionsplänen zugehörigen Arbeitsvorgänge haben ebenfalls angepasste früheste Start- und späteste Endtermine sowie einen gewählten Modus, mit dem die Menge und die Qualifikation der zugeordneten Ressourcen bestimmt werden. Die Ressourcen wiederum haben definierte Verfügbarkeitspläne. Weiterhin besitzt das Modell eine Liste von ausgewählten Prioritätsregeln und anderen Eigenschaften, um das Modell realitätsnah abzubilden.

Der Simulationslog enthält alle notwendigen Informationen, um das Szenario zu beschreiben und zu bewerten, wie bspw. die Zustände der Ressourcen und den Zeitplan der Arbeitsvorgänge. Aufbauend auf diesen werden die unterschiedlichen Key Performance Indicators (KPI) berechnet, damit der Optimierer den kompletten

Simulationszyklus effizienter auswerten kann. Bspw. könnte das die Restzeit (Differenz zwischen Liefertermin und Fertigstellungszeitpunkt) von Produkten, Vorgängen oder Aufträgen sein. Weitere Beispiele sind die Verletzung von Fertigstellungszeiten, Produktionskosten, aber auch Ressourcenauslastung. SEMI (2004) gibt einen Überblick aller verfügbaren KPIs. Der gesamte Ablauf ist in Abbildung 2 dargestellt.

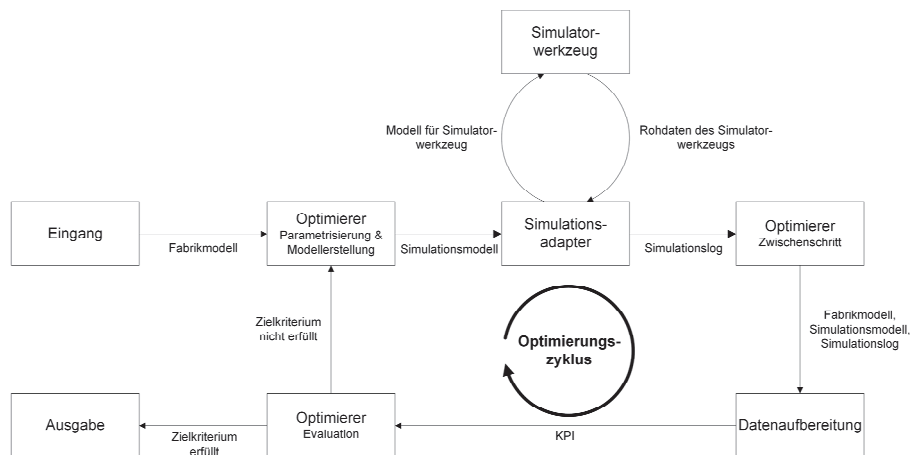


Abbildung 2: Darstellung eines klassischen Optimierungszyklus

Mit der Integration des Datenmetamodells in die Plattform wurde die Grundlage für die modellgetriebene Entwicklung gelegt (Angelidis et al. 2011). In einem nächsten Schritt wird der Fachanwender durch den Einsatz eines grafischen Werkzeuges zur Modellierung von Optimierungsalgorithmen in die Lage versetzt, beliebig komplexe Algorithmen selbst zu erstellen und zu validieren. Diese erstellten Algorithmen werden anschließend in der Plattform ausgeführt. Somit wird eine vollständige Integration mit einem einfachen Softwareverteilungsmodell gewährleistet.

4 Entwicklung von Algorithmen durch Visualisierungsmethoden

Um eine modellgetriebene Entwicklung von Optimierungsalgorithmen zu erreichen, wurde die Anwendung Visual Rules (Bosch SI 2013) in die simulationsbasierte Optimierungsplattform integriert. Dies ermöglicht es dem Fachanwender, komplexe Regelsysteme mittels eines Editors zu modellieren. Dabei stehen folgende Elemente zur Verfügung:

- Zuweisung (z.B. um einer Variablen einen Wert zu geben)
- Verzweigung (z.B. um in Abhängigkeit von einem bestimmten Wert einen anderen Teilbaum zu durchlaufen)
- Möglichkeit der Anbindung von Datenbanken (z.B. um auf Daten in einer externen Datenbank zuzugreifen)
- Aktionen
- Serviceaufrufe (z.B. Wiederverwendung von fachlichen Services der Plattform)

- Zustandsautomaten und Ablaufregeln
- Dokumentation
- Etc.

Darüber hinaus bietet es die Möglichkeit, ein vorhandenes Datenmodell während der Erstellung wiederzuverwenden, eine Ablaufregel hierarchisch aufzubauen und einen Zustandsautomaten zu definieren. Der Zustandsablauf kann jederzeit neu gestartet werden und bedarf nicht zwingend eines Endzustands. Des Weiteren wurde die Integration so gestaltet, dass der Zustandsautomat völlig automatisiert abläuft sobald dieser einmal angestoßen wurde. Während der Entwurfsphase stehen sowohl das Datenmetamodell als auch die von der Plattform bereitgestellten Services zur Verfügung, so dass eine vollständige Integration erreicht wird. Das Algorithmusmodell kann zur Laufzeit über einen Web-Service oder eine Java-Schnittstelle ausgeführt werden. Das Zusammenspiel aller Komponenten ist in Abbildung 3 dargestellt.

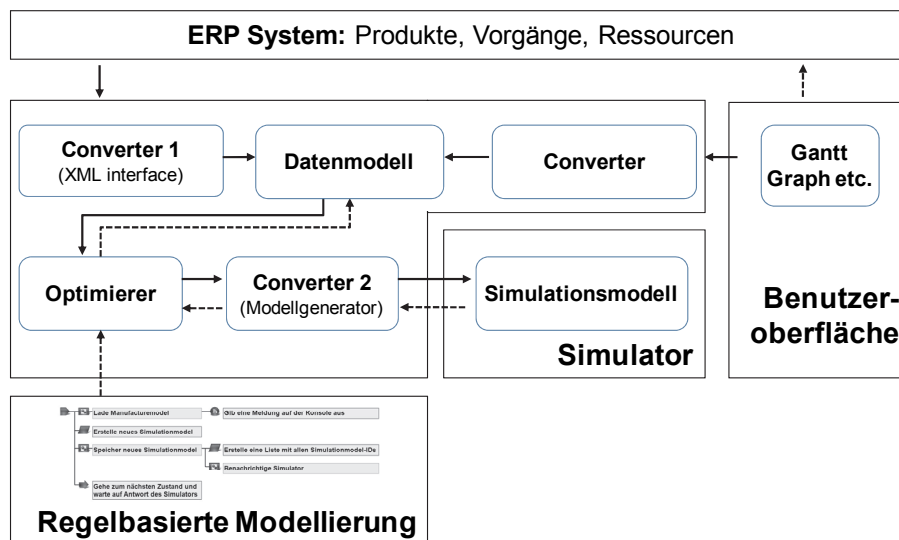


Abbildung 3: Integration von Visual Rules in SBOP

5 Beispiel eines iterativen Algorithmus mit Visual Rules

Anhand des folgenden Beispiels soll die Verwendung von Visual Rules für die Erstellung eines Optimierungsalgorithmus gezeigt werden. Dabei steht die Variabilität und einfache Verwendbarkeit des grafischen Werkzeugs im Fokus und nicht die Ausgabe eines repräsentativen praxisrelevanten Ergebnisses. Als Beispiel wird ein dreistufiger Algorithmus verwendet, wie Abbildung 4 zeigt. Im Folgenden werden die einzelnen Schritte im Detail erläutert.

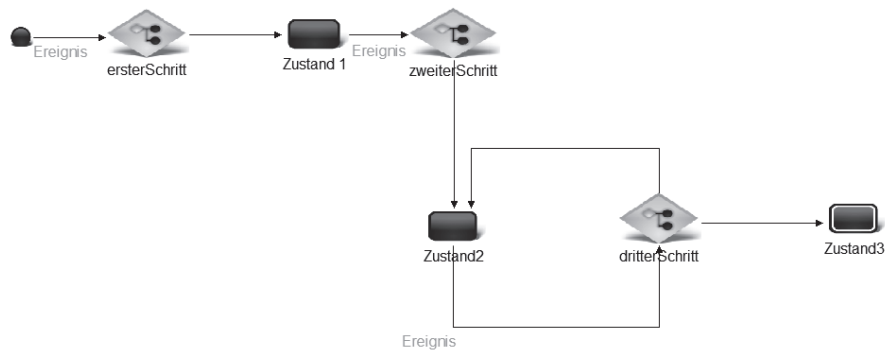


Abbildung 4: Beispiel eines Optimierungsalgorithmus in Visual Rules

Im ersten Optimierungsschritt wird das theoretische Maximum für die Optimierung ermittelt. Dies wird mit den folgenden Schritten erreicht und ist in Abbildung 5 dargestellt:

1. Wähle für jede Ressource zu jedem Zeitpunkt die maximal verfügbare Anzahl mit der Folge, dass in diesem Schritt die Schichten der Ressourcen ignoriert werden.
2. Wähle den frühesten Starttermin, spätesten Endtermin für Aufträge, Produkte und Arbeitsvorgänge
3. Wähle eine zufällige Montagecharakteristik (Modus).

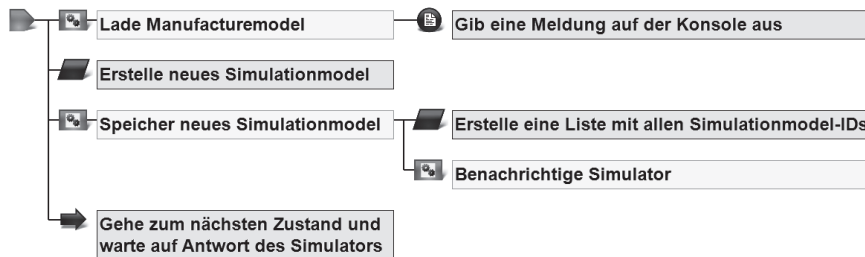


Abbildung 5: Erster Optimierungsschritt

Im zweiten Optimierungsschritt wird versucht, die Auslastung der Ressourcen ohne eine Verletzung der Fertigstellungszeitpunkte von Aufträgen, Produkten oder Arbeitsvorgängen zu minimieren. In Abbildung 6 sind die dazu notwendigen Schritte grafisch dargestellt.

1. Wähle den Schichtplan mit der höchsten Verfügbarkeit.
2. Wähle die Montagecharakteristik (Modus) mit der kürzesten Bearbeitungszeit.
3. Setze die folgenden Prioritätsregeln für die Abarbeitungsreihenfolge bei Knotenpunkten:
 - a. frühester Endtermin (EDD)
 - b. frühester Rahmenstarttermin ERD)

- c. kürzeste Bearbeitungszeit (SPT)
4. Falls das gewählte Szenario kein gültiges Ergebnis liefert, erzeuge ein neues Szenario beginnend von Punkt 1. Jedoch werden bei Punkt 3 andere Prioritätsregeln ausgewählt.

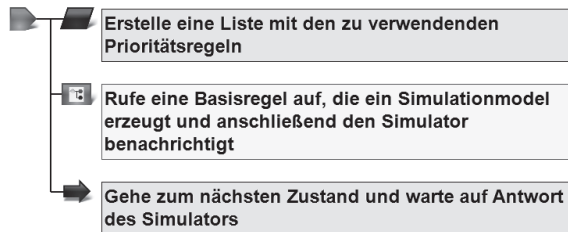


Abbildung 6: Zweiter Optimierungsschritt

In Abbildung 7 ist der letzte Optimierungsschritt dargestellt. Er besteht aus den folgenden Schritten:

1. Wähle den Schichtplan mit der höchsten Verfügbarkeit.
2. Wähle eine zufällige Montagecharakteristik (Modus).
3. Verwende dieselben Prioritätsregeln aus Optimierungsschritt 2, die ein gültiges Ergebnis geliefert haben.

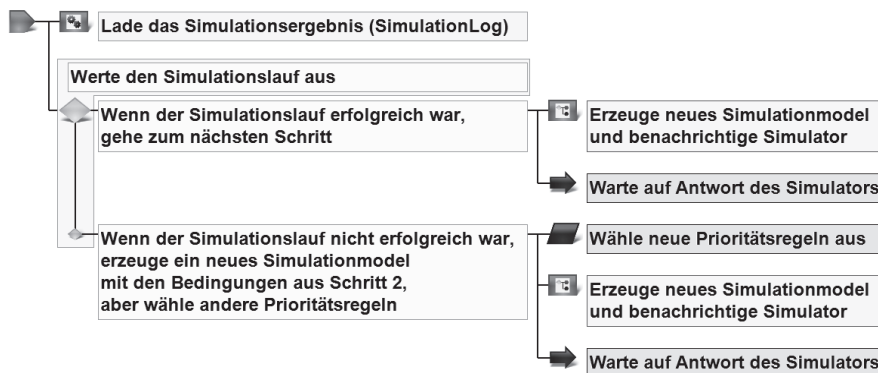


Abbildung 7: Dritter Optimierungsschritt

6 Ergebnisse und Zusammenfassung

Durch die Integration und Bereitstellung von Visual Rules wurde eine Möglichkeit geschaffen, bereits in der Entwurfsphase von Optimierungsalgorithmen auf ein grafisches Werkzeug zurückzugreifen. Der Anwender muss also nicht zwingend über Programmierkenntnisse verfügen. Erste Ideen können so modelliert, mit Erläuterungen versehen und erst im weiteren Verlauf mit der eigentlichen technischen Funktionalität hinterlegt werden. Das einfache Verteilungsmodell und der Modellierungsansatz erlauben ein sehr variables und schnelles Ändern und Ausführen von Optimierungsalgorithmen innerhalb der Plattform. Damit wird eine Möglichkeit geschaffen, auf einfache Art und Weise sehr viele Algorithmen zu testen und dessen Ergebnisse gegenüberzustellen. Durch den generischen Ansatz, den Visual Rules bereitstellt, ist es zudem möglich, dieses Werkzeug in anderen Bereichen, wie z.B. bei der Online-Optimierung oder der Berechnung von KPIs einzusetzen. Durch den Einsatz der neuen Methode soll nicht zwingend ein Performancegewinn bei der Ausführung der Algorithmen erreicht werden. Jedoch wird durch die visuelle Darstellung eine Erhöhung der Übersichtlichkeit und damit verbunden eine eventuelle Ideengenerierung für neue Algorithmen gefördert. Desweiteren wird ein zeitlicher Gewinn bei der Erstellung von Algorithmen erreicht.

Literatur

- Angelidis, E.; Pappert, F. M.; Rose O.: A Prototype Simulation Tool for a Framework For Simulation-based Optimization of Assembly Lines. In: Jain, S.; Creasey, R. R.; Himmelspach, J.; White, K. P.; Fu, M. (Hrsg.): Proceedings of the 2011 Winter Simulation Conference (WSC). Institute of Electrical and Electronics Engineers, Inc. Piscataway, New Jersey (USA), 2011, S. 2383–2394.
- Angelidis, E.; Naumann, A.; Rose O. (2012a): An Extended Critical Path Method for Complex Assembly Lines. In: Lim, G.; Herrmann, J. W. (Hrsg.): Proceedings of the 2012 Industrial and Systems Engineering Research Conference, Norcross, Georgia (USA), 2012
- Angelidis, E.; Bohn, D.; Rose, O. (2012b): A Simulation-Based Optimization Heuristic Using Self-Organization for Complex Assembly Lines. In: Laroque, C.; Himmelspach, J.; Pasupathy, R.; Rose, O.; Uhrmacher A. M. (Hrsg.): Proceedings of the 2012 Winter Simulation Conference (WSC). Institute of Electrical and Electronics Engineers, Inc. Piscataway, New Jersey (USA), 2012
- April, J.; Glover, F.; Kelly, J. P.; Laguna, M.: Practical Introduction to Simulation-Optimization. In: Chick, S.; Sánchez, P. J.; Ferrin, D.; Morrice, D. J. (Hrsg.): Proceedings of the 2003 Winter Simulation Conference. Institute of Electrical and Electronics Engineers, Inc. Piscataway, New Jersey (USA), 2003, S. 71-78,
- BOSCH SI. Visual Rules. Aufruf am 24.02.2013. <http://www.bosch-si.com/de/technologie/business-rules-management-brm/brm-komponenten/visual-rules-modeler.html>, 2013.
- Law, A. W.; McComas, M. G.: Simulation-Based Optimization. In: Yücesan, E.; Chen, C.-H.; Snowdon, J. L.; Charnes, J. M. (Hrsg.): Proceedings of the 2002 Winter Simulation Conference (WSC) . Institute of Electrical and Electronics Engineers, Inc. Piscataway, New Jersey (USA), 2002, S. 41-44
- Pappert, F. M.; Angelidis, E.; Rose, O.: Framework For Simulation-Based Scheduling of Assembly Lines. In: Johansson, B.; Jain, S.; Montoya-Torres, J.;

- Hugan, J.; Yücesan, E. (Hrsg.): Proceedings of the 2010 Winter Simulation Conference (WSC). Institute of Electrical and Electronics Engineers, Inc. Piscataway, New Jersey (USA), 2010, S 1690-1968
- SEMI, EMI E10-0304E: Specification for Definition and Measurement of Equipment Reliability, Availability, and Maintainability (RAM), 2004
- Shapiro, A.: Simulation Based Optimization. In: Charnes, J. M.; Morrice D. J.; Brunner, D. T.; Swain, J. J. (Hrsg.): Proceedings of the 1996 Winter Simulation Conference (WSC) . Institute of Electrical and Electronics Engineers, Inc. Piscataway, New Jersey (USA), 1996, S. 332-336