7. Workshop des ASIM-Arbeitskreises "Simulation und Künstliche Intelligenz", Braunschweig, April 1994, ASIM-Mitteilungen, Heft 44.

Ein regelbasiertes System zur automatischen Planung und Durchführung von Simulationsläufen

Dipl.-Inf. Sven Hader
Fakultät für Informatik
TU Chemnitz-Zwickau

Kurzfassung

Vorgestellt wird das regelbasierte System DIM_EXPERTE, das in der Lage ist, automatisch Simulationsläufe zu planen und durchzuführen. Die Planung erfolgt auf der Grundlage regelbasierten Wissens über das Simulationsmodell und der Ergebnisse vorangegangener Simulationsläufe. Damit ist es möglich, eine automatische Optimierung von Simulationsmodellen vorzunehmen. Das System liefert eine (semi)optimale Lösung, die je nach Anforderung sofort übernommen oder vom menschlichen Experten weiter optimiert werden kann. Dadurch kann der Zeitaufwand des Experten für die Beschäftigung mit dem Simulationsmodell verringert werden.

1 Einleitung

Die Simulation hat sich in den letzten Jahren zu einer der vielversprechendsten Schlüsseltechnologien entwickelt. Die wesentliche Triebkraft dafür ist die Notwendigkeit, in der Praxis mit immer komplexeren Systemen umzugehen. Für die Analyse dieser Systeme existieren meist keine mathematischen Verfahren oder sind wegen der Dimension der Probleme (quasi) nicht anwendbar. Deshalb ist die Simulation in vielen Fällen die einzig brauchbare Methode, um Informationen über das Verhalten der betrachteten Systeme zu erhalten. Diese Informationen werden meist benutzt, um die Systeme bezüglich bestimmter Zielvorgaben zu optimieren.

Die konventionelle Vorgehensweise bei der Optimierung von Simulationsmodellen[1)] besteht in der Durchführung mehrerer Simulationsläufe mit jeweils geänderten Eingangsparametern. Die Wahl der Eingangsparameter für den nächsten Simulationslauf erfolgt durch den Nutzer unter Berücksichtigung der Ergebnisse der letzten Simulationsläufe und des gewünschten Zieles. Dabei muß der Nutzer sein Wissen über das System einsetzen, um Faktoren zu erkennen, die die Systemleistung begrenzen und versuchen, durch zielgerichtete Wahl neuer Eingangsparameter die Systemleistung zu verbessern. Dieser Ansatz kann als multidimensionales Hillclimbing interpretiert werden. Das vom Nutzer benötigte fachspezifische Wissen umfaßt sowohl Wissen über das System (z.B. Beziehung Eingangsparameter $\rightarrow$ Systemleistung) als auch Wissen über die Methodik der Simulation (z.B. statistische Auswertung von Simulationsläufen).

Die geschilderte Vorgehensweise besitzt eine Reihe von Nachteilen. Der Nutzer darf i.allg. kein Laie sein, d.h. er muß über das oben beschriebene Wissen verfügen. Außerdem bringt es der große Zeitaufwand für einen Simulationslauf (insbesondere bei komplexen Systemen) und die große Anzahl durchzuführender Simulationsläufe mit sich, daß der Nutzer eine lange Zeit mit der Optimierung beschäftigt ist. Das ist insbesondere bei hochspezialisierten Fachleuten nicht wünschenswert. Deshalb wurden sowohl auf dem Gebiet der mathematischen Optimierung (z.B. evolutionäre Algorithmen, siehe [Sch93]) als auch auf dem Gebiet der Künstlichen Intelligenz (u.a. regelbasierte Systeme) Verfahren entwickelt, die eine Automatisierung der

[1)] Wenn im weiteren Verlauf der Arbeit von Simulationsmodellen oder kurz von Modellen gesprochen wird, dann sind damit ausführbare Programme gemeint, die das Verhalten von realen Systemen nachbilden.

Optimierung von Simulationsmodellen erlauben sollen (automatic analysis). Dabei sollte sich das Augenmerk jedoch aus heutiger Sicht nicht ausschließlich auf eine automatische Optimierung (gerade komplexer Systeme) richten, sondern vielmehr auf eine sinnvolle Verbindung der automatischen mit der manuellen Optimierung. Ein automatisches Optimierungssystem kann dem Experten eine (semi)optimale Lösung liefern, die dieser analysiert und ggf. weiter optimiert. Auf diese Weise ergibt sich eine wesentliche Entlastung des Experten.

In dieser Arbeit wollen wir uns mit der Verwendung von regelbasierten Systeme zur automatischen Optimierung von Simulationsmodellen befassen. Dieser Ansatz liegt nahe, wenn man die oben beschriebene Vorgehensweise bei der manuellen Optimierung betrachtet. An die Stelle des menschlichen Experten, der nach jedem Simulationslauf eine Auswertung durchführt und neue Eingangsparameter bestimmt, tritt das regelbasierte System. Es besitzt Wissen in Form von Regeln über das simulierte System und kann somit in ähnlicher Weise wie ein Experte vorgehen.

Der geschilderte Ansatz wird zumindest seit Beginn der 80er Jahre verstärkt propagiert. Es existiert eine Vielzahl von Artikeln, die diesen Ansatz befürworten und allgemein beschreiben, jedoch nur wenige Simulationssysteme, die eine Komponente zur automatischen Optimierung beinhalten. Das wohl bekannteste und umfassendste dieser Systeme ist KBS (Knowledge-Based Simulation System), das an der Carnegie-Mellon University entwickelt wurde (siehe [Red82], [Fox89]). Dieses System besitzt Komponenten zur Generierung von Simulationsmodellen, zur automatischen Ausführung von Simulationsläufen unter Verwendung einer Regelbasis, zur Datensammlung während der Simulation, zum Tracing und zur Animation der Simulation und zur Auswertung der Daten nach der Simulation. Die Regeln der Regelbasis können über ein spezielles Lernverfahren verfeinert und quantifiziert werden.

2 Das regelbasierte System DIM_EXPERTE

Bei dem in dieser Arbeit beschriebenen regelbasierten System DIM_EXPERTE handelt es sich um den Prototyp eines Systems zur automatischen Durchführung von Simulationsläufen. Anhand dieses Systems sollen Erfahrungen über die Formalisierung von fachspezifischem Expertenwissen in Form von Regeln und über die automatische Anwendung dieser Regeln bei der Optimierung von Simulationsmodellen gesammelt werden. Um sich auf das Wesentliche zu konzentrieren, enthält das System keinerlei Komponenten, die sich mit der eigentlichen Simulation beschäftigen. Die Generierung des Simulationsmodells erfolgt wie bisher mittels höherer Programmier- bzw. Simulationssprachen oder spezieller Simulationssysteme, so daß dem Nutzer die Einarbeitung in ein neues System erspart bleibt und auch bereits existierende Simulationsmodelle genutzt werden können. Einzige Einschränkung ist, daß das Simulationsmodell in Form eines selbständig lauffähigen Programmes vorliegt, das über Textdateien mit dem regelbasierten System kommunizieren kann.

Im Zusammenhang mit der Optimierung von Simulationssystemen wird in der Literatur der Begriff des **Szenario** (engl. scenario, siehe z.B. KBS) verwendet. Darunter versteht man die Menge aus dem Simulationsmodell und einem Satz konkreter Eingangsparametern, d.h. ein vollständig quantifiziertes Abbild des zu modellierenden Systems. Um die Leistung bzw. Güte eines Szenarios zu ermitteln, wird ein Simulationslauf mit den entsprechenden Eingangsparametern durchgeführt (auch als Experiment bezeichnet). Die Optimierung von Simulationsmodellen bedeutet also die Suche nach einem optimalen Szenario.

2.1 Aufbau des Systems

Die wesentlichen Komponenten des regelbasierten Systems und ihre Beziehungen untereinander sind in Bild 1 dargestellt.

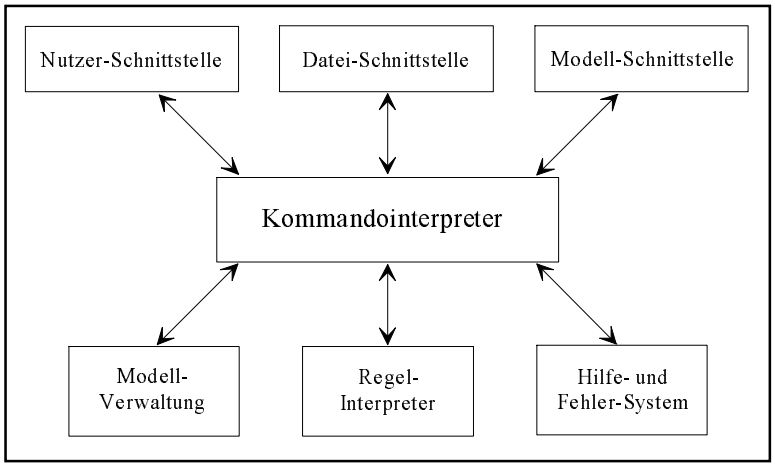

Bild 1. Der Aufbau des regelbasierten Systems DIM_EXPERTE

Das Steuerzentrum von DIM_EXPERTE bildet ein **Kommandointerpreter**, der Nutzerkommandos entgegennimmt, sie interpretiert und ausführt und eventuelle Statusmeldungen an den Nutzer zurückliefert. Die Kommunikation mit dem Nutzer erfolgt über eine textorientierte Schnittstelle, wobei die Eingaben des Nutzers mittels einer stark vereinfachten 'natürlichsprachlichen' Analyse in die konkreten Befehle des Systems transformiert werden.

Die **Datei-Schnittstelle** dient dem Lesen und Schreiben von Textdateien. Dabei wird zwischen Modellbeschreibungsdateien, Regeldateien und Ergebnisdateien unterschieden. Jede dieser Dateiarten besitzt eine eigene Syntax.
Modellbeschreibungsdateien enthalten formale Beschreibungen der zu optimierenden Simulationsmodelle. Dazu gehören Angaben über die wesentlichen Eingangsparameter, Konstanten, Leistungskenngrößen und Restriktionen der Modelle sowie die bei der Optimierung zu erfüllenden Ziele. Des weiteren wird die Schnittstelle zu den konkreten Simulationsmodellen beschrieben. Anhand der gelesenen Informationen werden in der **Modell-Verwaltung** Speicherstrukturen angelegt, die im Verlaufe der Arbeit mit diesem Modell benötigt werden. Von besonderer Bedeutung sind dabei die sogenannten Datenobjekte, die numerische Objekte des Simulationsmodells beschreiben (z.B. Eingangsparameter, wichtige Konstanten, Leistungskenngrößen). Modellbeschreibungsdateien können nur gelesen werden.
Regeldateien enthalten explizite und nutzerlesbare Formulierungen von fachspezifischem Wissen, das zur Optimierung der Modelle benötigt wird. Dieses Wissen in Form von Regeln wird beim Einlesen einer systeminternen Regelbasis hinzugefügt. Regeldateien können sowohl gelesen als auch geschrieben werden.
Ergebnisdateien enthalten Informationen über das gefundene optimale Szenario (d.h. eine Beschreibung des Modells sowie die gefundenen optimalen Werte der Eingangsparameter und Leistungskenngrößen) und Angaben über den Optimierungsprozeß. Sie können in gewisser Weise als Protokoll verwendet und ausgedruckt werden. Ergebnisdateien können nur geschrieben werden.

Der **Regel-Interpreter** ist der eigentliche Kern von DIM_EXPERTE. Er übernimmt die automatische Planung, Durchführung und Auswertung von Simulationsläufen des verwendeten

Modells. Dabei bedient er sich der systeminternen Regelbasis, um das aktuelle Szenario zu analysieren und anhand der gewonnenen Erkenntnisse ein neues Szenario zu generieren (d.h. eine neue Belegung der Eingangsparameter zu ermitteln), das die Zielkriterien wahrscheinlich besser erfüllt (eine ausführlichere Beschreibung des Regelinterpreters findet sich in Abschnitt 2.2). Über die **Modell-Schnittstelle** erfolgt die Abarbeitung des Simulationsmodells und der Datenaustausch zwischen dem System und dem Modellprogramm.

Falls während der Arbeit mit DIM_EXPERTE Fragen bzw. Fehler auftreten, dann kann das **Hilfe- und Fehler-System** verwendet werden. Es ist in der Lage, die dem Kommandointerpreter bekannten Befehle zu erläutern und die (wahrscheinliche) Ursache aufgetretener Fehler zu erklären.

Neben den in Bild 1 gezeigten und bis hierher beschriebenen Komponenten existieren weitere, deren Kenntnis für das allgemeine Verständnis des Systems jedoch nicht notwendig ist. Es handelt sich dabei u.a. um eine spezielle Arithmetikeinheit mit erweitertem Funktionsumfang, ein einfaches Modul zur Gleichungsmanipulation, ein Modul zur Durchführung einer numerischen Optimierung (auf der Basis der Evolutionsstrategie) sowie spezielle Ein- und Ausgabefunktionen.

Über die beschriebenen Schnittstellen erfolgt die Kommunikation von DIM_EXPERTE mit der 'Außenwelt'. Dazu gehören vor allem das Lesen von Modellbeschreibungs- und Regeldateien, das Schreiben von Ergebnisdateien, die Ausführung des Simulationsmodells inkl. Datenaustausch und die Kommunikation mit dem Nutzer. Eine Darstellung des Systems im Verhältnis zu seiner Umgebung zeigt Bild 2.

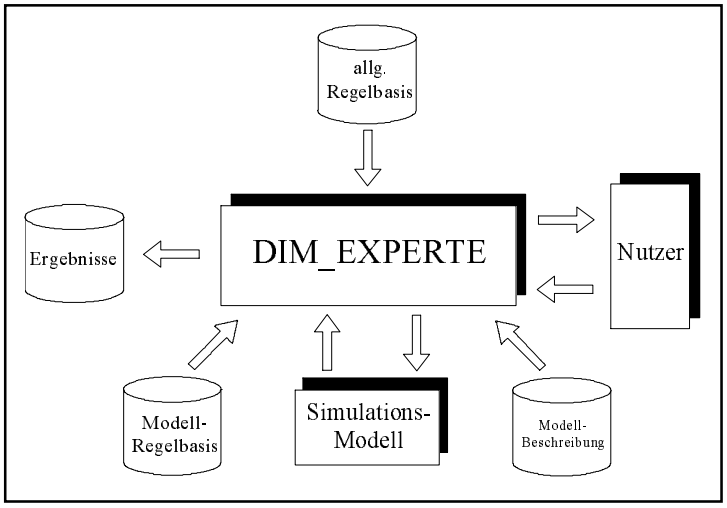

Bild 2. Kommunikationsverbindungen von DIM_EXPERTE mit der Systemumgebung

2.2 Der Regelinterpreter

Die in der systeminternen Regelbasis gespeicherten und vom Regelinterpreter verwendeten Regeln besitzen eine einfache IF-THEN-Struktur.

IF < Bedingungen > THEN < Schlußfolgerungen > bzw.

IF < Situation > THEN < Aktionen >

Der **Situationsteil** enthält eine mehr oder weniger vollständige Beschreibung eines Szenarios, das während der Optimierung auftreten kann. Er besteht aus einer Liste konjunktiv verknüpfter 'Bedingungen'. Dazu gehören sowohl echte Bedingungen (ergeben bei Auswertung den Wahrheitswert TRUE oder FALSE; vor allem arithmetische Vergleiche) als auch Wertermittlungsfunktionen (ergeben bei Auswertung immer TRUE und liefern zusätzlich die Werte von Datenobjekten). Jede Bedingung kann negiert oder nichtnegiert auftreten.

Der **Aktionsteil** enthält eine Liste von Aktionen, um das im Situationsteil beschriebene Szenario in ein anderes, wahrscheinlich besseres Szenario zu überführen und dieses zu testen. Die wesentlichen Aktionen sind die Veränderung des Wertes von Eingangsparametern und der Test eines Szenarios mittels automatischer Durchführung eines Simulationslaufes.

Es existieren sowohl eine allgemeine Regelbasis (wird vom System automatisch geladen) als auch modellabhängige Regelbasen (werden je nach Bedarf vom Nutzer geladen). Die allgemeine Regelbasis enthält Regeln zur Arbeit des Regelinterpreters und allgemeine Regeln über die Simulationsmethodik. Die modellabhängigen Regelbasen enthalten Optimierungsregeln für konkrete Modelle. Bei der Optimierung eines Modells setzt sich die verwendete systeminterne Regelbasis aus der allgemeinen Regelbasis und einer oder mehrerer modellabhängiger Regelbasen zusammen. Die Regeln aus den verschiedenen Regelbasen werden vom Regelinterpreter gleich behandelt.

Beim verwendeten Regelinterpreter handelt es sich um einen **vorwärtsverkettenden Interpreter mit eingeschränktem Backtracking**. Vorwärtsverkettend bedeutet, daß der Interpreter anhand der Daten des aktuellen Szenarios eine Regel sucht, deren IF-Teil durch diese Daten vollständig erfüllt ist. Danach werden die Aktionen des THEN-Teils der Regel ausgeführt usw.. Backtracking bedeutet, daß eine Regel, die ein Szenario testet (d.h. einen entsprechenden Simulationslauf durchführt), das gegenüber bisherigen Szenarien keine Verbesserung darstellt, rückgängig gemacht und eine alternative Regel gesucht wird. Das Backtracking ist eingeschränkt, da die Regelrücknahme nur bis zu der Regel erfolgen kann, die das bisher beste Szenario lieferte (ein einmal entstandener Optimierungsfortschritt kann nicht rückgängig gemacht werden). Der Prozeß der Optimierung erzeugt also eine Folge von Szenarien, die die vom Nutzer gestellten Ziele immer besser erfüllen.

Der Regelinterpreter kann auf zwei Arten beendet werden: entweder existieren für das aktuelle Szenario keine (weiteren) passenden Regeln mehr oder eine der Regeln beendet den Regelinterpreter explizit durch ein Kommando.

Sowohl während der Optimierung als auch nachträglich ist es möglich, den Inferenzprozeß (die Folge der Regelanwendungen) am Bildschirm zu verfolgen. Dadurch kann die Vollständigkeit der Regelbasis überprüft und diese ggf. durch Hinzufügen bzw. Modifizieren einzelner Regeln verändert werden. Außerdem kann diese Funktion als einfache Erklärungskomponente benutzt werden, um die Plausibilität eines ermittelten Szenario kontrollieren zu können.

2.3 Implementation

DIM_EXPERTE wurde auf einer Sun-Workstation in SNI-Prolog v3.0A (Siemens-Nixdorf) entwickelt. Ausschlaggebend für die Wahl von Prolog als Implementierungssprache waren vor allem das leistungsfähige Prinzip der Unifikation, der eingebaute Backtracking-Mechanismus und die Möglichkeit der einfachen Definition neuer Operatoren. Gerade der letzte Punkt erleichtert stark die schnelle Entwicklung von Syntaxprüfern und -übersetzern, wie sie für die Modellbeschreibungs- und Regeldateien notwendig sind. Zusätzlich besitzt Prolog bei Beachtung grundlegender Programmiertechniken eine sehr gute Lesbarkeit und Verständlichkeit des Quelltextes, insbesondere bei konsequenter Anwendung der Modul-Technik.

Der Quelltext des Systems umfaßt im Augenblick etwa 5000 Programmzeilen, von denen ca. 1000 auf die Regelverwaltung und den Regelinterpreter und 2000 auf die verschiedenen Schnittstellenfunktionen entfallen.

3 Erfahrungen bei der Nutzung des Systems

DIM_EXPERTE wurde bisher vorrangig für die optimale Gestaltung von speziellen Produktionsstrukturen verwendet. Dabei handelt es sich um Flexible Fertigungssysteme (FMS), die sich mit Hilfe von Central-Server-Modellen nachbilden lassen (siehe hierzu [Had93], [Had94]). Die entsprechenden Simulationsmodelle wurden in C implementiert und verfügen über eine Schnittstelle, die den Datenaustausch mit DIM_EXPERTE ermöglicht. Als Eingangsparameter wurden verschiedene Kombinationen aus folgender Menge verwendet: Anzahl Paletten im FMS, Anzahl Kanäle der einzelnen Arbeitsstationen, Routingwahrscheinlichkeiten, (s,Q)-Lagerhaltungsstrategie eines vorgelagerten Eingangslagers. Als Zielfunktionen wurden einfache Kosten- bzw. Gewinnfunktionen verwendet, die zu minimieren bzw. zu maximieren waren. Die Ergebnisse der regelbasierten Optimierung wurden mit den Ergebnissen einer einfachen numerischen Optimierung (zweigliedrige Evolutionsstrategie) verglichen.

Der Vergleich ergab, daß die regelbasierte Optimierung in den meisten Fällen bessere Ergebnisse mit weniger Simulationsläufen erzielte. Das liegt daran, daß 'normale' numerische Optimierungsverfahren, die nach der direkten Methode[1)] arbeiten, meist nach dem Trial-and-Error-Verfahren vorgehen, wobei die Testpunkte (Parameterbelegungen) durch mehr oder weniger zufällige Veränderungen der bisher besten Punkte erzeugt werden. Dieser Prozeß der Generierung neuer Varianten (Szenarien) arbeitet zu einem großen Teil blind, während bei der regelbasierten Optimierung meist nur 'erfolgversprechende' Testpunkte generiert und untersucht werden. Ein Vergleich der regelbasierten Methode mit dem Vorgehen eines menschlichen Experten konnte nicht vorgenommen werden, da es sich bei den betrachteten Problemen meist um stark vereinfachte, nicht unbedingt praxisrelevante Aufgaben handelte und kein entsprechender Experte zur Verfügung stand.

Als das wesentliche Problem des Einsatzes von DIM_EXPERTE wurde die Bereitstellung und Formalisierung des Regelwissens erkannt. Dieses Wissen liegt i.allg. nicht als Lehrbuchwissen, sondern als Erfahrungswissen vor. Selbst den Personen, die täglich diese Erfahrungen nutzen, fällt es zumeist schwer, dieses Wissen explizit zu formulieren. Darüber hinaus erfordert die Formalisierung des Wissens ein gewisses Maß an Kenntnissen über die Darstellung und Abarbeitung von Regeln, so daß der Fachexperte sicher in den meisten Fällen die Hilfe eines KI-Experten (in Form eines Knowledge Engineer) benötigt.

[1)] Diese Optimierungsmethoden verwenden nur Informationen über Zielfunktions- und Restriktionswerte für konkrete Parameterbelegungen (keine Kenntnis über Ableitungen u.ä.).

Anhand der während der Arbeit mit DIM_EXPERTE gewonnenen Erfahrungen erscheint mir die Verwendung des folgenden Regelschemas als empfehlenswert:

Benutzt wird eine modellabhängige Regelbasis mit vier Gruppen von Regeln, die die folgenden Funktionen realisieren:

- Erkennen des allgemeinen Problems
  (z.B. Optimierung eines Lagers, Optimierung eines FMS)
- Erkennen von leistungsmindernden Engpässen des aktuellen Szenarios
  (z.B. zu wenig Material, zu geringe Maschinenkapazität)
- Wertänderung von Eingangsparametern, um die erkannten Engpässe zu beseitigen
  (z.B. erhöhe Kanalanzahl einer Station, erhöhe Palettenanzahl des FMS)
- Test des so generierten neuen Szenarios durch einen Simulationslauf
  (Regel-Backtracking, falls das neue Szenario keine Verbesserung bringt)

Durch das Zusammenwirken dieser vier Gruppen von Regeln kann eine Optimierungsstrategie realisiert werden, die der des menschlichen Experten nahe kommt und die die sukzessive Weiterentwicklung und Verbesserung der Regelbasis erleichtert.

Als weitere potentielle Probleme automatischer Optimierungssysteme wurden folgende erkannt (siehe dazu [Yam92]):

1. Die Beachtung von gegenseitigen Abhängigkeiten zwischen Eingabeparametern.
   Die Wertzuweisung an einen Eingabeparameter kann Änderungen der Werte anderer Eingangsparameter nach sich ziehen. Werden diese Abhängigkeiten nicht beachtet, kann es zu Inkonsistenzen innerhalb der Eingabemenge kommen.

2. Die Verwendung strukturbestimmender Eingabeparameter.
   Die Wertzuweisung an einen Eingabeparameter kann umfangreiche strukturelle Änderungen im Modell nach sich ziehen. Wenn sich die Eigenschaften von Modell-Datenobjekten ändern (z.B. deren Dimension), die das Optimierungssystem benutzt, muß während der Optimierung eine Anpassung der zugehörigen Strukturen erfolgen.

Die Lösung dieser Probleme führt zu einer stark zunehmenden Komplexität der Optimierungssysteme, so daß abzuwägen ist, ob nicht statt dessen als erster Schritt die Art verwendbarer Eingangsparameter eingegrenzt werden kann. Dadurch kann es natürlich dazu kommen, daß sich bestimmte Simulationsmodelle mit diesen Systemen nicht mehr optimieren lassen.

4 Zusammenfassung und Ausblick

In dieser Arbeit wurde das regelbasierte System DIM_EXPERTE vorgestellt. Es dient zur Optimierung von Simulationsmodellen, wobei die sukzessive Planung, Durchführung und Auswertung der Simulationsläufe automatisch auf der Grundlage des dem System zugänglichen fachspezifischen Wissens erfolgt. Der menschliche Einfluß bei der Optimierung beschränkt sich vorrangig auf die Formalisierung des fachspezifischen Wissens über das Modell in Form von IF-THEN-Regeln. Das System eignet sich in erster Linie als Prototyp für die Erforschung der Formalisierung und Anwendung von modellspezifischem Wissen bei der Beurteilung und Optimierung von Systemen. Da bezüglich praktisch relevanter Aufgabengebiete die Regelbasis

meist nicht vollständig ist, sollte ein Experte das vom System gefundene (semi)optimale Szenario analysieren und evtl. weiter manuell optimieren. Wichtig für die Verbesserung der verwendeten Regelbasis ist dabei die Untersuchung, warum das System eventuell vorhandene bessere Lösungen nicht gefunden hat. Dies sollte dann zur Ergänzung bzw. Modifizierung der Regelbasis führen.

Für die weitere Entwicklung des Systems ergeben sich folgende Ansatzpunkte:

- Einordnung in ein umfassenderes System zur wissensbasierten Simulation, das auch Elemente zur Generierung von Simulationsmodellen besitzt
- Verwendung von Lernverfahren, um das während der Simulationsläufe gesammelte Wissen über das Modellverhalten formalisieren zu können
- Erstellung von Modellbibliotheken, die den Typ des zu optimierenden Modells ermitteln und automatisch die zu verwendende Regelmenge zur Verfügung stellen

Literatur

[Fox89] Fox, Mark S.; Husain, Nizwer; McRoberts, Malcolm; Reddy, Y.V.: Knowledge-Based Simulation: An Artificial Intelligence Approach to System Modeling and Automating the Simulation Life Cycle. In: [Wid89], pp. 447-486.

[Had93] Hader, Sven: Anwendung von Methoden der KI bei der simulativen und analytischen Untersuchung speziell strukturierter Systeme. Diplomarbeit, TU Chemnitz-Zwickau, Fachbereich Informatik, 1993.

[Had94] Hader, Sven; Köchel, Peter: A Model for Manufacturing Systems with Input Storage. 8th International Working Seminar on Production Economics, Innsbruck, Austria, February 21-25, 1994.

[Pin89] Pinto, Ivan Campos: Wissensbasierte Unterstützung bei der Lösung von Optimierungsaufgaben. Dissertation, Uni Dortmund, Fachbereich Informatik, 1989.

[Red82] Reddy, Y.V.; Fox, Mark S.: KBS: An Artificial Intelligence Approach to Flexible Simulation. Technical Report CMU-RI-TR-82-1, Robotics Institute, Carnegie-Mellon University, Pittsburgh, PA, 1982.

[Sch93] Schwefel, Hans-Paul; Bäck, Thomas; Hammel, Ulrich: Modelloptimierung mit evolutionären Algorithmen. In: Sydow, A. (Hrsg.): Simulationstechnik. 8. Symposium in Berlin, September 1993, Vieweg, Braunschweig/Wiesbaden, 1993, S. 49-57.

[Wid89] Widman, Lawrence E.; Loparo, Kenneth A.; Nielsen, Norman R. (eds.): Artificial Intelligence, Simulation, and Modeling. John Wiley & Sons, New York, 1989.

[Yam92] Yampolsky, L.S.; Yampolsky, S.L.; Lavrov, A.A.: Elements of Intelligent Control of Simulation Experiments. Proceedings of the 1992 European Simulation Symposium, Dresden, Germany, November 5-8, 1992, pp. 600-604.