

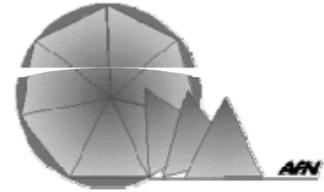
Volker Nissen, Mathias Petsch (Hrsg.)

Softwareagenten und Soft Computing im Geschäftsprozessmanagement

Innovative Methoden und Werkzeuge zur Gestaltung, Steuerung
und Kontrolle von Geschäftsprozessen in Dienstleistung,
Verwaltung und Industrie

Tagungsband zum 9. Symposium Soft Computing am 30.11.2006 an der TU Ilmenau





Volker Nissen, Mathias Petsch (Hrsg.)

Softwareagenten und Soft Computing im Geschäftsprozessmanagement

**Innovative Methoden und Werkzeuge zur Gestaltung, Steuerung
und Kontrolle von Geschäftsprozessen in Dienstleistung,
Verwaltung und Industrie**

Tagungsband zum 9. Symposium Soft Computing am 30.11.2006 an der TU Ilmenau

Bibliografische Information Der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.ddb.de> abrufbar.

1. Aufl. - Göttingen : Cuvillier, 2006

ISBN 10: 3-86727-062-7

ISBN 13: 978-3-86727-062-5

© CUVILLIER VERLAG, Göttingen 2006

Nonnenstieg 8, 37075 Göttingen

Telefon: 0551-54724-0

Telefax: 0551-54724-21

www.cuvillier.de

Alle Rechte vorbehalten. Ohne ausdrückliche Genehmigung des Verlages ist es nicht gestattet, das Buch oder Teile daraus auf fotomechanischem Weg (Fotokopie, Mikrokopie) zu vervielfältigen.

1. Auflage, 2006

Gedruckt auf säurefreiem Papier

ISBN 10: 3-86727-062-7

ISBN 13: 978-3-86727-062-5

Dieser Tagungsband erscheint mit freundlicher Unterstützung der
Arbeitsgemeinschaft Fuzzy-Logik und Soft Computing in Norddeutschland (AFN).

Vorwort

Die Agententechnologie hat in den vergangenen Jahren eine zunehmende Popularität erreicht. Die wesentlichen technischen Herausforderungen (z.B. Interaktion, Kooperation, Interoperabilität) im Zusammenhang mit Intelligenten Agenten und Multiagentensystemen scheinen gelöst bzw. es liegen Ansätze vor, die eine Lösung in den nächsten Jahren wahrscheinlich machen. Bislang ist es der Agententechnologie jedoch noch nicht gelungen, einen wirklichen Durchbruch beim Einsatz in betriebswirtschaftlichen Anwendungen zu erreichen.

Soft Computing (unter anderem Fuzzy Systeme, Neuronale Netze) ist seit geraumer Zeit ebenfalls Thema wissenschaftlicher Untersuchungen und hat teilweise auch Einzug in die betriebswirtschaftliche Praxis gefunden. Mit ihren spezifischen Eigenschaften erscheinen die beiden methodischen Bereiche der Softwareagenten sowie des Soft Computing gut geeignet, in IT-Systemen des Geschäftsprozessmanagements eingesetzt zu werden. Hierzu existieren bisher nur erste Konzepte.

Die nachfolgenden Beiträge liefern Hinweise, welche Potenziale und Voraussetzungen für die Einführung agentenbasierter Werkzeuge oder Methoden des Soft Computing im Kontext des Geschäftsprozessmanagement relevant sind, und sie bieten auch erste praktische Gestaltungsempfehlungen.

Ilmenau, im November 2006

Univ.-Prof. Dr. Volker Nissen

Dr. Mathias Petsch

Inhaltsverzeichnis

Fuzzy Business Process Management: Geschäftsprozessmanagement unter Berücksichtigung unscharfer Daten

Oliver Thomas, Thorsten Dollmann

1	Unschärfe im Geschäftsprozessmanagement.....	1
2	Von scharfen zu unscharfen Mengen	2
3	Prozessmodellierung mit der EPK.....	5
3.1	Grundlegende Sprachkonstrukte der EPK	5
3.2	Formalisierung der EPK.....	6
3.3	ARIS-Erweiterung der EPK.....	9
4	Fuzzy-Ereignisgesteuerte Prozesskette	11
4.1	Erweiterung der EPK um Attribute	11
4.2	Fuzzy-Erweiterung der EPK.....	13
5	Anwendungsszenario „Fuzzy-Customizing“	14
6	Verwandte Arbeiten.....	17
7	Zusammenfassung und zukünftige Forschungsfragen.....	18

Koordination integrierter Logistikprozesse im Hafen

Leif Meier, Helge Fischer

1	Einführung in den Problembereich	23
2	Modelle und Lösungsansätze der Literatur	24
3	Untersuchung am Beispiel sequentieller Koordination	27
4	Softwareagenten zur Koordination integrierter Logistikprozesse	31
5	Ausblick.....	35
6	Anhang.....	37

Integration von Software-Agenten und Soft-Computing-Methoden für die Transportplanung

Hagen Langer, Ingo J. Timm, Jörn Schönberger, Herbert Kopfer

1	Einleitung	39
2	Grenzen zentraler Transportprozess-Planungsansätze	40
2.1	Aktuelle Herausforderungen an die Transportplanung.....	40
2.2	Grenzen zentraler Planungsansätze.....	42
2.3	Grenzen eines verteilten Transportprozess-Managements	42

3	Vergleich der konzeptionellen Grundlagen.....	43
3.1	Paradigmen des Soft-Computing	43
3.2	Konzeption der Software-Agenten-Methodik	44
3.3	Synopsis	45
4	Integration der Informationsstrukturen.....	46
5	Ereignisgesteuerte integrierte globale und lokale Planung.....	46
5.1	Integration lokaler und globaler Planungsziele	47
5.2	Koordinations-Rolle des Wissensmanagements.....	49
6	Zusammenfassung und Ausblick	50

Softwareagenten im Krankenhaus – Ubiquitous Computing zwischen Ordnung und Chaos

Christoph Niemann

1	Einleitung	53
2	Ubiquitous Computing	54
2.1	Allgegenwärtige Sensorik	55
2.2	Verarbeitung	56
2.3	Effektoren	57
3	Softwareagenten im UC	58
4	EMIKA – Entscheidungsunterstützung im Krankenhaus	60
4.1	Sensoren	62
4.1.1	Position und Identität	62
4.1.2	Behandlungs- und Terminpläne.....	63
4.2	Verhandlungen zwischen den Agenten auf Basis eines künstlichen Marktes	63
4.3	Effektoren	65
5	Zusammenfassung und Ausblick	65

Softwareagenten im Wissensmanagement

Volker Nissen; Mathias Petsch

1	Einleitung	69
1.1	Begriff des Wissens	69
1.2	Begriff und Strategien des Wissensmanagements	71
1.3	Wissensnetzwerke	72
2	Intelligente Agenten.....	72
3	Anwendungsmöglichkeiten von Softwareagenten im Wissensmanagement.....	73
3.1	Aufgaben und Anforderungen des Wissensmanagements	73

3.2	Nachteile heutiger zentralisierter Wissensmanagementsysteme	76
3.3	Anwendung von Agenten im Wissensmanagement.....	77
3.3.1	Informationsagenten.....	77
3.3.2	Agentenarchitekturen im Wissensmanagement	79
3.3.3	Bestehende Agentenanwendungen im Wissensmanagement.....	80
3.3.4	Generelle Agentenanwendungen und Anwendungsumgebungen...	83
3.3.5	Rückschlüsse auf Anwendungsmöglichkeiten im Wissensmanagement.....	86
3.4	Anwendungsmöglichkeiten von „Schwarmintelligenz“ im Wissensmanagement	89

Robustheit von Reputationssystemen: Ein evolutionärer Bewertungsansatz auf Basis eines Multiagentensystems

Ivo Reitzenstein

1	Einleitung	97
1.1	Problemstellung	97
1.2	Ziel des Beitrags	99
1.3	Vorgehensweise	99
2	Manipulation von Reputationssystemen.....	99
3	Spieltheoretische Grundlagen	100
3.1	Das Konzept der evolutionären Spieltheorie.....	100
3.2	Modellierung evolutionärer Spiele.....	101
4	Bewertung der Robustheit von Reputationssystemen.....	102
4.1	Basismodell.....	103
4.2	Strategie der Käufer.....	105
4.3	Strategien der Verkäufer.....	107
4.3.1	kooperieren.....	107
4.3.2	manipulieren	107
4.3.3	Imitation	108
5	Simulationskonzept	109
6	Zusammenfassung und Ausblick	110

Economic Coordination in Service-Oriented Architectures

Falk Kretzschmar

1	Introduction	115
2	Economic coordination in SOA.....	116
2.1	Definition and classification.....	116
2.2	Negotiations in SOA.....	118
2.3	Existing standards for implementing negotiations in SOA.....	119

2.4	Extended service broker architecture for automated negotiations in SOA	120
3	An agent-based approach for simulating the economic coordination in SOA.....	122
3.1	Simulating the economic coordination in SOA	122
3.2	Model overview	123
3.3	Model parameters and agent interaction.....	124
4	Discussion and future work	126

SIMJADE – Ein Simulationsdienst für die verteilte Ausführung agentenbasierter Modelle

Dirk Pawlaszczyk

1	Einleitung	129
2	Agentenbasierte Simulation	130
2.1	Begriffsbestimmung	130
3	Synchronisationsansätze	131
3.1	Der Zeitbegriff in der Simulation.....	131
3.2	Zeitfortschreibung innerhalb der Simulation.....	131
3.3	Ansätze der verteilten Simulation.....	132
4	Überblick zu verteilten Zeitsteuerungsansätzen innerhalb der Agentenbasierten Simulation	135
5	Der Simulationsdienst	136
5.1	Anwendungsbereiche von <i>SIMJADE</i>	137
5.2	Architektur.....	137
5.3	Implementierung eigener Agenten	138
5.4	Speichermanagement.....	139
5.5	Zeitsteuerung	139
6	Simulationsstudie	140
7	Fazit und Ausblick	142

Fuzzy Business Process Management: Geschäftsprozessmanagement unter Berücksichtigung unscharfer Daten

Oliver Thomas, Thorsten Dollmann

Institut für Wirtschaftsinformatik (IWi)
im Deutschen Forschungszentrum für Künstliche Intelligenz (DFKI),
Universität des Saarlandes
Stuhlsatzenhausweg 3, Geb. D3 2, 66123 Saarbrücken
{oliver.thomas|thorsten.dollmann}@iwi.dfki.de

Zusammenfassung: Das Management von Geschäftsprozessen ist in der Unternehmenspraxis häufig dadurch charakterisiert, dass Entscheidungsprämissen nicht in Form mathematischer Modelle oder numerischer Werte vorliegen. Entschlüsse sind durch Abwägung und Kreativität gekennzeichnet und werden meist aus unscharfen Bedingungen wie „geringe Durchlaufzeit“ oder „hohe Qualität“ abgeleitet. Obwohl die in diesen Prämissen verwendeten Adjektive nicht präzise sind, ist mit ihnen zur Erfassung einer konkreten Unternehmenssituation zusätzliche und bedeutsame Information verbunden. Für das Geschäftsprozessmanagement besitzen daher verbale Informationen sowie vage formulierte Aussagen, Prämissen, Zielvorstellungen und Restriktionen einen hohen Stellenwert. Der vorliegende Beitrag zeigt, wie unscharfe Bedingungen und vage formulierte Zielvorstellungen mit Hilfe der Fuzzy-Set-Theorie in Geschäftsprozessmodellen berücksichtigt werden können. Diese Erweiterung der Prozessmodellierung erfolgt am Beispiel der Ereignisgesteuerten Prozesskette.

1 Unschärfe im Geschäftsprozessmanagement

Die Ziele gegenwärtiger Business-Engineering-Projekte liegen in der Gestaltung der Geschäftsprozesse sowie in der Analyse der Anforderungen an deren IT-Unterstützung unter Berücksichtigung von Unternehmensstrategien [ÖsWi2003]. Die Prozessgestaltung muss dabei einem umfassenden Ansatz folgen, der sowohl die Planung und Kontrolle als auch die Steuerung – das Management – der betrieblichen Abläufe umfasst [BKRo2005]. Zur Unterstützung eines systematischen Vorgehens bei der Prozessgestaltung hat sich die Modellierung als hilfreich erwiesen. Modellierungssprachen, wie z.B. die Ereignisgesteuerte Prozesskette (EPK) [KNSc1992, STAd2005], dienen der Explikation der Modelle. Softwarewerkzeuge zur Geschäftsprozessmodellierung, wie z.B. das ARIS-Toolset [IDS2003], können den Business Engineer durch Systemkomponenten zur Erhebung, Analyse und Simulation von Geschäftsprozessmodellen unterstützen.

Zur Erfassung und Verbesserung von Geschäftsprozessen, deren Generalisierung in Referenzmodellen sowie zur unternehmensspezifischen Anpassung im Customizing sind zahlreiche Konzepte erarbeitet worden, die situationsspezifische Problemstellungen betrachten. Viele Ansätze legen einen Schwerpunkt auf die nutzerfreundliche und intuitive Verwendbarkeit der Methoden, indem diese an menschliche Denkweisen angenähert werden. Dabei werden jedoch für notwendige Entscheidungen die exakte Quantifizierung und Formalisierung der Entscheidungsregeln verlangt. Bei Geschäftsprozessen liegen jedoch vielfach nur unsichere, unpräzise und vage Informationen über die häufig nicht technisch determinierten Abläufe vor [Völk1998, Fort2002, Hüss2003]. Ebenso ist das der Prozessgestaltung zu Grunde liegende Zielsystem in der Regel durch ungenaue Formulierungen und implizite Interdependenzen geprägt. Dies demonstriert beispielsweise die Aussage „die Durchlaufzeit von Aufträgen mit ‚sehr hoher‘ Priorität soll unter Beibehaltung einer ‚hohen‘ Bearbeitungsqualität ‚wesentlich‘ gesenkt werden, indem die Bearbeitungsintensität ‚angemessen‘ reduziert wird“. In diesem Beispiel können weder die konkrete Ausprägung der beiden genannten Ziele bzgl. Durchlaufzeit und Bearbeitungsqualität noch die abgeleitete Maßnahme ohne Informationsverlust quantifiziert und damit unmittelbar verarbeitbar gemacht werden. Informations-, insbesondere Referenzmodelle, sowie Methoden zu deren unternehmensspezifischer Adaption berücksichtigen diese Formen der Unschärfe nach wie vor unzureichend.

Diesem Umstand soll in diesem Beitrag durch die Erweiterung von Prozessmodellierungssprachen zur Berücksichtigung und Verarbeitung von Unschärfe mit Hilfe der Fuzzy-Set-Theorie begegnet werden. Diese unscharfe Erweiterung wird am Beispiel der EPK nachvollzogen.

Die Auswahl der Prozessmodellierungssprache EPK ist maßgeblich anhand ihrer Popularität in der Modellierungspraxis begründet. Die vorgestellte Erweiterung ist jedoch nicht auf die EPK oder „verwandte“ Modellierungssprachen eingeschränkt, wobei mit letzterem Begriff auf Sprachen verwiesen sein soll, die analog zur EPK bspw. keine formale Semantik besitzen oder ebenfalls dem Paradigma der strukturierten Systementwicklung folgen. Der präsentierte Ansatz lässt sich auch auf objektorientierte Modellierungssprachen (z.B. UML-Aktivitätsdiagramm) oder Modellierungssprachen, die eine formale Semantik aufweisen (z.B. Petri-Netz), übertragen. Der Beitrag beschreibt die dazu erforderlichen Schritte und Werkzeuge und ist wie folgt gegliedert: Zunächst wird der Begriff „Unschärfe“ präzisiert sowie die Berücksichtigung unscharfer Daten mit Hilfe der Fuzzy-Set-Theorie motiviert (Abschnitt 2). Anschließend wird in Abschnitt 3 die EPK als Modellierungssprache eingeführt, formal definiert sowie um die zur Fuzzifizierung notwendigen Sprachkonstrukte erweitert. Die Einführung der Fuzzy-EPK, die auf einer Attributierung der EPK-Sprachkonstrukte aufbaut, erfolgt in Abschnitt 4. In Abschnitt 5 werden Anwendungsszenarien des entwickelten Konzepts aufgezeigt. Der Beitrag schließt mit der Analyse verwandter Arbeiten in Abschnitt 6 und der Diskussion der Ergebnisse in Abschnitt 7.

2 Von scharfen zu unscharfen Mengen

In der Literatur existiert keine einheitliche Definition des Unschärfebegriffs – es scheint fast so, als müsse das Verständnis dieses Begriffs selbst unscharf sein. Unschärfe wird meist durch eine Abgrenzung gegenüber deterministischen, stoch-

astischen und unsicheren Informationszuständen definiert [Rehf1998, 39]. Als Unschärfe wird in diesem Beitrag die Unsicherheit hinsichtlich von Daten und ihrer Interdependenzen verstanden. Auslöser der Unschärfe können die Realität selbst, die Sprache als Modellbildung über die Realität oder die Verwendung der Sprache sein [Dosc1993, 76f] (vgl. Abb. 2-1).

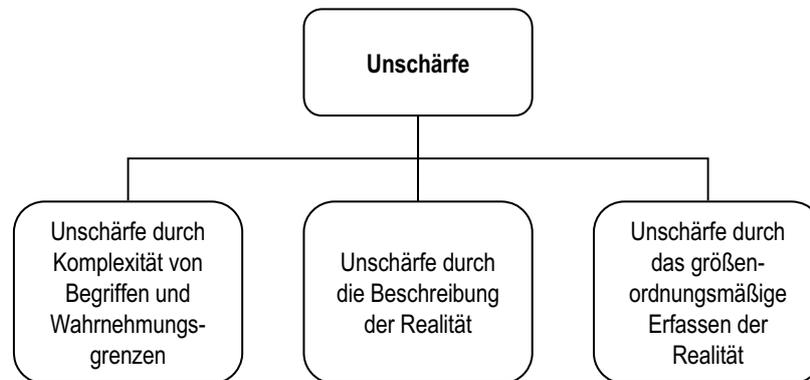


Abb. 2-1: Unschärfeaspekte

Die Komplexität des Umweltsystems und die Wahrnehmungsgrenzen des Menschen bedingen informationale Unschärfe. So enthalten beispielsweise wissensintensive Prozesse kurzlebige Informationen aus einer Vielzahl von Quellen, sodass zu einem festen Zeitpunkt nur ein Teil des Gesamtprozesses erfasst werden kann, der jedoch während der Erfassung anderer Teilaspekte bereits veraltet. Menschliche Präferenzordnungen sind in vielen Situationen nicht exakt bestimmbar, sodass es zu einer mit der informationalen Unschärfe verwandten Vagheit des Zielsystems kommt. So impliziert z. B. das Ziel „wesentliche Verminderung der Durchlaufzeit“ zwar Maßnahmen, jedoch lassen sich wegen der nicht explizierten Höhe der angestrebten Änderung und der unklaren Wertungsinterdependenzen mit anderen Zielen keine exakten Handlungen ableiten.

Die Beschreibung der Realität mit natürlicher Sprache erzeugt die intrinsische (auch: verbale oder linguistische) Unschärfe. Sowohl die Bildung eines sprachlichen Modells als auch die Kontextsensitivität von sprachlichen Aussagen tragen zur Entstehung dieser Unschärfe bei. Hiermit ist auch die Ungenauigkeit in sprachlichen Vergleichen eng verbunden. Die Aussage „der Objektwert ist viel höher als x“ ist ein Beispiel hierfür.

Das für den Menschen übliche größenordnungsmäßige Erfassen der Realität erzeugt ebenfalls Unschärfe. Die Verwendung ungenauer Daten kann jedoch vorteilhaft sein, wenn geeignete Messmethoden fehlen, der Realweltausschnitt von hoher Dynamik geprägt ist oder nicht exakt ermittelbare Abhängigkeiten bestehen.

Bei der Entwicklung von Modellen und der dazu notwendigen Quantifizierung verbaler Attribute sollte die Unschärfe qualitativer Problemstellungen als verhaltensrelevant akzeptiert werden. Die Fuzzy-Set-Theorie versucht die Trennung zwischen einer modell- und verfahrenstechnisch notwendigen Präzision einerseits sowie einer empirisch wünschenswerten Berücksichtigung qualitativer Informationen andererseits zu überwinden und einen Anteil an fehlender Präzision sowie Vagheit und Unsicherheit bei Modellierungsprozessen zu tolerieren.

Die Fuzzy-Set-Theorie als heutiges Teilgebiet des Soft Computing hat sich Mitte der 1960er-Jahre entwickelt [Zade1965]. Kernpunkt der Fuzzy-Theorie ist es, Zustände (von Objekten) nicht ausschließlich mit „wahr“ oder „falsch“ zu bewerten, sondern Zwischenstufen zuzulassen. Der ursprünglichen Idee von Zadeh folgend, wird die klassische Mengenlehre, d.h. die Theorie der scharfen Mengen, durch die Beschreibungen und Verknüpfungen unscharfer Mengen (Fuzzy-Mengen) erweitert: Für jedes Element ω einer vorgegebenen (scharfen) Grundmenge Ω wird der Grad der Zugehörigkeit zu einer Teilmenge $A \subseteq \Omega$ durch einen Wert $\mu_A(\omega)$ einer Abbildung $\mu_A : \Omega \rightarrow [0;1]$ ausgedrückt. Man wählt diese Zugehörigkeitsgrade aus dem Intervall $[0;1]$ und gibt folgende Interpretation: Je größer der Zugehörigkeitsgrad eines Elements bzgl. einer (unscharfen) Menge ist, desto mehr gehört das Element zu dieser Menge. μ_A wird Zugehörigkeitsfunktion der unscharfen Menge (Fuzzy-Menge) $\{(\omega, \mu_A(\omega)) \mid \omega \in \Omega\}$ genannt.

Mit Fuzzy-Mengen lassen sich linguistische Variablen [Zade1973] formulieren, die natürlichsprachliche Ausdrücke – so genannte linguistische Terme – als Werte annehmen. Abb. 2-2 zeigt die linguistische Variable „Auftragswert“. Sie weist die Terme „gering“, „mittel“ und „hoch“ auf. Die Zugehörigkeiten eines Objektwerts zu diesen unscharfen Mengen sind durch die Zugehörigkeitsfunktionen μ_{gering} , μ_{mittel} und μ_{hoch} ausgedrückt. Der Objektwert 70.000 € gehört z. B. zu 0.5 sowohl zur Fuzzy-Menge „mittel“ als auch zur Fuzzy-Menge „hoch“. Diese Abbildung scharfer Werte auf unscharfe Mengen heißt Fuzzifizierung. In einem scharfen Kontext wäre es nur möglich, z. B. einen Objektwert ab 70.000 € als „hohen“ Auftragswert zu charakterisieren, während 69.999 € bereits als „mittel“ gelten würde.

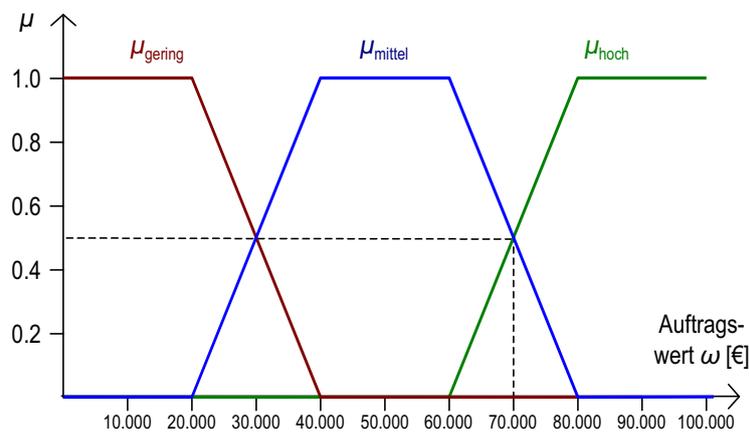


Abb. 2-2: Linguistische Variable „Auftragswert“

Ein Fuzzy-System besteht aus Ein- und Ausgangsvariablen, deren jeweilige Terme durch Regeln, bestehend aus Prämissen- und Konklusionsteil, z. B. der Form „WENN Kundeneinschätzung = mittel UND Auftragsvolumen = sehr hoch DANN Kundenauftragsbewertung = hoch“, miteinander verknüpft sind. Durch Inferenzverfahren werden die Eingangs- und Ausgangsvariablen einander zugeordnet. Für eine ausführbare Aktion, z. B. „Priorität festlegen“, wird ein scharfer Wert der Ausgangsvariablen benötigt. Ein Defuzzifizierungsschritt liefert diesen scharfen Wert.

Fuzzy-Systeme werden erfolgreich in den Bereichen Regelungstechnik, Sensorik und Datenanalyse bzw. Entscheidungsunterstützung eingesetzt. Fuzzy-Software-Werkzeuge – kurz: Fuzzy-Werkzeuge – unterstützen den Benutzer bei der Planung, Modellierung, Analyse, Simulation, und Umsetzung der Fuzzy-Systeme.

3 Prozessmodellierung mit der EPK

3.1 Grundlegende Sprachkonstrukte der EPK

Seit Beginn der 1990er-Jahre wird eine Vielzahl von Modellierungssprachen zur Beschreibung von Geschäftsprozessen eingesetzt, z.B. Flussdiagramme in Form von Struktogrammen, Programmablauf- oder Datenflussplänen, Datenflussdiagramme im Rahmen der Structured Analysis (SA)-Methode, der Netzplan, das Vorgangskettendiagramm (VKD), das Petri-Netz, die Ereignisgesteuerte Prozesskette (EPK) sowie Ansätze zur Prozessmodellierung im Rahmen objektorientierter Modellierungsmethoden. Zur Konstruktion von Geschäftsprozessmodellen auf fachlicher Ebene hat sich aufgrund ihrer Anwendungsorientierung und umfassenden Werkzeugunterstützung insbesondere im deutschsprachigen Raum die Ereignisgesteuerte Prozesskette etabliert. Sie wurde am Institut für Wirtschaftsinformatik (IWi), Saarbrücken, in Zusammenarbeit mit der SAP AG entwickelt [KNSc1992] und ist Bestandteil des ARIS-Toolset der IDS SCHEER AG sowie des Business Engineering und Customizing des SAP R/3-Systems.

In graphentheoretischer Terminologie ist ein EPK-Modell ein gerichteter und zusammenhängender Graph, dessen Knoten Ereignisse, Funktionen und Verknüpfungsoperatoren sind. Abb. 3-1 zeigt ein EPK-Modell der Kundenauftragsbearbeitung, anhand dessen die grundlegenden Sprachkonstrukte der EPK sowie ihre jeweiligen Repräsentationsformen erläutert werden.

Ereignisse sind die passiven Elemente der EPK. Sie beschreiben das Eintreten eines Zustands und werden durch Sechsecke dargestellt. Funktionen, die durch an den Ecken abgerundete Rechtecke repräsentiert werden, sind die aktiven Elemente der EPK. Der Funktionsbegriff wird in der EPK mit dem der Aufgabe gleichgesetzt. Während zur Bezeichnung der Funktionen in der Literatur [z.B. HKSc1992, 5] vorgeschlagen wird, das jeweilige Objekt der Bearbeitung und ein Verb im Infinitiv zur Kennzeichnung der zu verrichtenden Tätigkeit zu verwenden (z.B. „Kundenauftrag definieren“, vgl. Abb. 3-1), wird für Ereignisse empfohlen, das Objekt, das eine Zustandsänderung erfährt, mit einem Verb im Partizip Perfekt zu verbinden, das die Art der Änderung beschreibt (z.B. „Kundenauftrag (ist) definiert“, vgl. Abb. 3-1).

Ereignisse lösen Funktionen aus und sind deren Ergebnis. Diese beiden Beziehungen zwischen Funktionen und Ereignissen werden durch Kontrollflusskanten, die durch Pfeile repräsentiert werden, dargestellt. Um auszudrücken, dass eine Funktion durch ein oder mehrere Ereignisse gestartet werden bzw. eine Funktion ein oder mehrere Ereignisse als Ergebnis erzeugen kann, werden Verknüpfungsoperatoren (Konnektoren) eingeführt. Dabei wird in Anlehnung an die Terminologie der Aussagenlogik zwischen konjunktiven „ \wedge “, adjunktiven „ \vee “ und disjunktiven Verknüpfungen „ \otimes “ unterschieden (vgl. Abb. 3-1). Die entsprechenden Konnektoren werden vereinfacht als AND-, OR- bzw. XOR-Operatoren bezeichnet.

Mit diesen Informationen ergibt sich für das in Abb. 3-1 dargestellte Prozessmodell die folgende Interpretation: Das Modell beschreibt den Ablauf zur Definition und Durchführung von Prüffunktionen für einen Kundenauftrag. Die Entscheidung über die Annahme oder die Ablehnung des Kundenauftrags wird durch die parallele Ausführung verschiedener Teilfunktionen getroffen. Der Kundenauftrag wird auf technische Machbarkeit und aus kaufmännischer Sicht geprüft, ferner werden die Kundenbonität und die Verfügbarkeit des Produkts ermittelt. Negativergebnisse, wie z. B. „Kundenauftrag technisch nicht machbar“ oder „Kundenbonität nicht gegeben“, führen zur Ablehnung des Kundenauftrags durch die Funktion „Kundenauftrag ablehnen“.

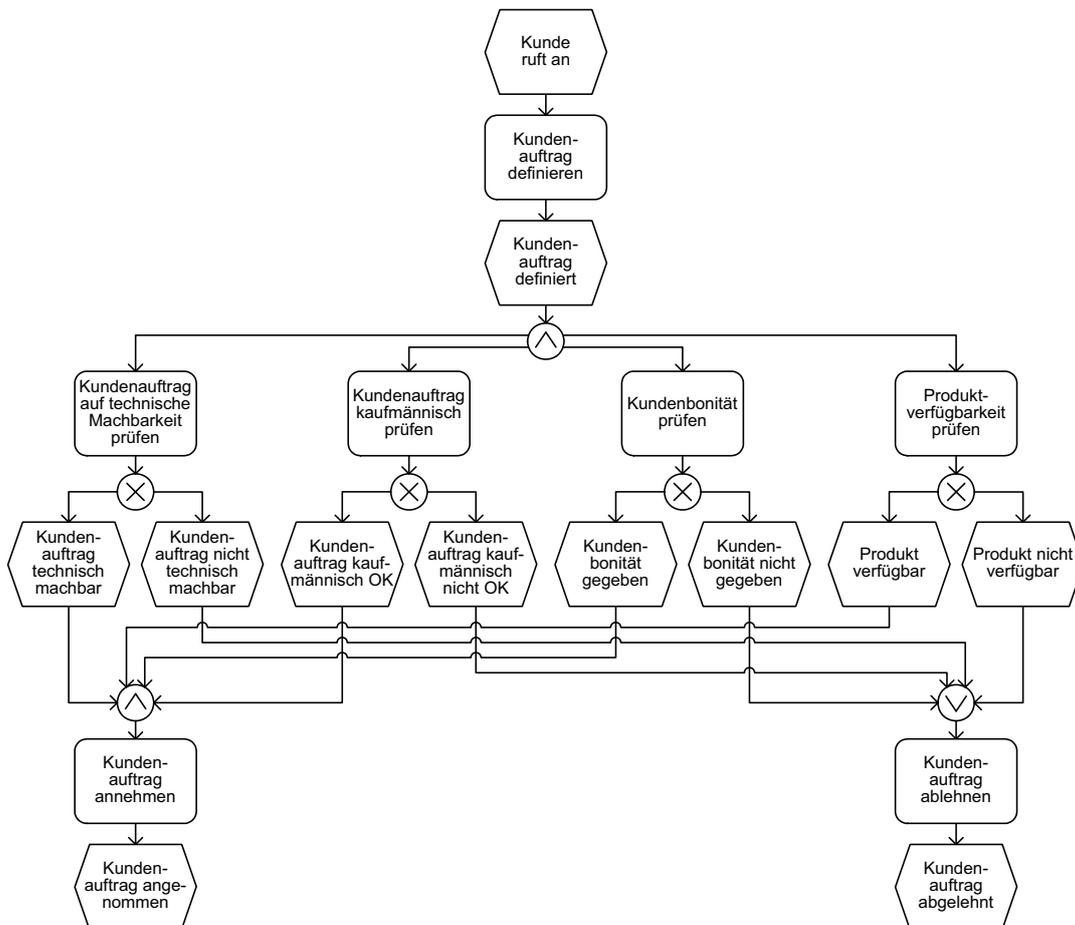


Abb. 3-1: EPK-Modell der Kundenauftragsbearbeitung

3.2 Formalisierung der EPK

Die von Keller, Nüttgens und Scheer [KNSc1992] eingeführte Notation der Ereignisgesteuerten Prozesskette wurde zunächst als eine nicht vollständig formalisierte Notation entwickelt und ohne eine feste formale Semantik benutzt. Zur Dokumentation von Prozessen und zur Verwendung der Modelle als Diskussionsgrundlage ist dies ausreichend. Für eine Konsistenzprüfung oder eine automatisierte Verarbeitung von EPK-Modellen, z. B. in Werkzeugen zur Simulation oder Verifikation, ist jedoch eine formale Definition der Syntax und Semantik der Modelle erforderlich.

Im akademischen Umfeld werden verschiedene Ansätze zur formalen Syntax- und Semantikdefinition der EPK vorgeschlagen und diskutiert [Aals1999, NüRü99, ADKi2002, Kind2004, Kind2006, RoAw06]. Im Folgenden stellen wir eine formale Definition der Syntax von EPK-Modellen in Anlehnung an [RoAa2006] vor, um darauf aufbauend eine präzise Definition einer unscharfen Erweiterung vornehmen zu können. Die resultierende mengentheoretische Spezifikation dient nicht dazu, Verhaltensaspekte von EPK-Modellen abzubilden. Ausgehend von der ursprünglichen Definition von EPK-Modellen tauchen dabei semantische Mehrdeutigkeiten insbesondere bei der Verwendung des Oder-Konnektors auf, dessen Schaltverhalten nicht immer lokal entscheidbar ist und dessen Einsatz in der Literatur diskutiert wird [LSWe1998, Ritt2000, DeRi2001]. Eine zusammenführende Oder-Verknüpfung kann dabei synchronisieren oder nicht, d.h. sie kann nach dem ersten Input, den sie erhält, schalten oder aber auf mehrere Eingangssignale warten. Kindler zeigt, dass aufgrund der Nicht-Lokalität des schließenden OR- und XOR-Konnektors die Semantik nicht immer eindeutig definiert sein kann [Kind2006]. Dieser Ambiguität muss durch eine erhöhte fachliche Abstimmung zwischen dem Modellersteller und -nutzer Rechnung getragen werden, womit eine Weiterverarbeitung der Modelle kontextabhängig erfolgen muss.

Allerdings bedeutet gerade die Verwendung der XOR- und OR-Konnektoren eine erhöhte Ausdrucksfreiheit in EPK-Modellen und ist mitverantwortlich für den Erfolg der EPK in der Praxis, da der semi-formale Charakter einen ausgewogenen Ausgleich zwischen formaler Genauigkeit und einer intuitiven Anwendbarkeit durch den Fachanwender darstellt. Gemäß der gängigen Modellierungspraxis wird hier angenommen, dass eine zusammenführende Oder-Verknüpfung aktiviert wird, wenn der erste Input eingegangen ist. Damit sei die semantische Diskussion der Kontrollflusssemantik der EPK-Methode an dieser Stelle beendet und auf die angesprochenen Beiträge verwiesen.

In formaler Schreibweise ist ein EPK-Modell ein 4-Tupel $EPC = (E, F, C, A)$. Dabei ist E eine endliche (nichtleere) Menge von Ereignissen (events), F eine endliche (nichtleere) Menge von Funktionen (functions), $C = C_{AND} \cup C_{OR} \cup C_{XOR}$ eine endliche Menge logischer Konnektoren (connectors), wobei C_{AND} , C_{OR} und C_{XOR} paarweise disjunkte Teilmengen von C sind, und

$$A \subseteq (E \times F) \cup (F \times E) \cup (E \times C) \cup (C \times E) \cup (F \times C) \cup (C \times F) \cup (C \times C)$$

eine Menge von Kanten. Die Relation A spezifiziert die Menge der gerichteten Kontrollflusskanten (arcs), welche Funktionen, Ereignisse und Konnektoren zueinander in Verbindung setzt. $V = E \cup F \cup C$ wird die Menge aller Knoten des EPK-Modells genannt.

Zur Einführung des Begriffs der syntaktischen Korrektheit von EPK-Modellen notieren wir für ein EPK-Modell die Menge der Eingangsknoten eines Knotens $v \in V$ mit $\bullet v := \{w \in V \mid (w, v) \in A\}$ und die Menge seiner entsprechenden Ausgangsknoten mit $v \bullet := \{w \in V \mid (v, w) \in A\}$. Weiterhin notieren wir einen gerichteten Pfad von einem Knoten $v_1 \in V$ zu einem Knoten $v_l \in V$ ($l \in \mathbb{N}$) als eine Folge $p = \langle v_1, \dots, v_l \rangle$ von Knoten $v_i \in V$ mit $(v_i, v_{i+1}) \in A$, wobei $1 \leq i \leq l-1$. Wir notieren für die Menge der tangierten Konnektoren auf einem Pfad von einem Knoten aus der Menge der Ereignisse zu einem Knoten aus der Menge der Funktionen

$$C_{EF} := \{c \in \{v_2, \dots, v_{l-1}\} \mid \exists \text{ Pfad } p = \langle v_1, v_2, \dots, v_{l-1}, v_l \rangle \text{ mit } v_1 \in E \wedge v_2, \dots, v_{l-1} \in C \wedge v_l \in F\}.$$

Analog lassen sich die Mengen C_{FE} , C_{FF} sowie C_{EE} definieren. Ebenso definieren wir einen gerichteten Konnektorenpfad $v_1 \xrightarrow{c} v_l$ von einem Knoten $v_1 \in V$ zu einem Knoten $v_l \in V$ als eine Folge $p_c = \langle v_1, \dots, v_l \rangle$ von Knoten mit $(v_i, v_{i+1}) \in A$ für $1 \leq i \leq l-1$ und $v_2, \dots, v_{l-1} \in C$. Auf einem Konnektorenpfad ist also der Startknoten nur über Konnektoren mit dem Endknoten verbunden.

Zur Konstruktion syntaktisch korrekter EPK-Modelle haben sich in der Wissenschaft einige Regeln etabliert [KeTe1999, 172-174, NüRü92, 68-70]. Unter deren Zuhilfenahme kann die Konsistenz eines EPK-Modells im Sinne einer Widerspruchsfreiheit und Stimmigkeit überprüft werden. Dazu muss ein EPK-Modell $EPC = (E, F, C, A)$ die folgenden Bedingungen erfüllen:

- (V, A) mit $V = E \cup F \cup C$ ist ein gerichteter, zusammenhängender Graph.¹
- Ereignisse haben höchstens eine eingehende und höchstens eine ausgehende Kante: $\forall e \in E : |\bullet e| \leq 1 \wedge |e \bullet| \leq 1$.
- Funktionen besitzen genau eine eingehende und genau eine ausgehende Kontrollflusskante: $\forall f \in F : |\bullet f| = 1 \wedge |f \bullet| = 1$.
- Es gibt mindestens ein Start- und ein Endereignis: $\exists e \in E : |\bullet e| = 0 \wedge \exists e \in E : |e \bullet| = 0$.
- Verknüpfungsoperatoren haben entweder mehrere eingehende und eine ausgehende Kontrollflusskanten (Join) oder eine ausgehende und mehrere eingehende Kontrollflusskanten (Split): $\forall c \in C : (|\bullet c| = 1 \wedge |c \bullet| > 1) \vee (|\bullet c| > 1 \wedge |c \bullet| = 1)$. Wir bezeichnen $C_J = \{c \in C \mid |\bullet c| > 1\}$ als die Menge der Join-Konnektoren und mit $C_S = \{c \in C \mid |c \bullet| > 1\}$ die Menge der Split-Konnektoren des EPK-Modells $EPC = (E, F, C, A)$ und es gilt $C_J \cap C_S = \emptyset$.
- Der vom EPK-Modell aufgespannte Graph ist „einfach“, d.h. er enthält keine Schlinge (Kante mit gleichen Anfangs- und Endknoten)² und keine Mehrfachkanten zwischen den einzelnen Knoten:

$$\forall v_1, v_2 \in V : |\{a \in A \mid a = (v_1, v_2) \vee a = (v_2, v_1)\}| \leq 1.$$

- Es gibt keinen gerichteten Kreis im aufgespannten Graphen, der nur aus Verknüpfungsoperatoren besteht: $\forall c_1, c_2 \in C : c_1 \xrightarrow{c} c_2 \Rightarrow c_1 \neq c_2$.
- Funktionen sind nur mit Ereignissen (gegebenenfalls über Verknüpfungsoperatoren) verbunden und vice versa: $C_{EE} = \emptyset \wedge C_{FF} = \emptyset$.

1 Diese Bedingung bedeutet zugleich, dass $\forall v \in V : |\bullet v| > 0 \vee |v \bullet| > 0$ gilt, d.h. dass keine isolierten Objekte im EPK-Modell existieren.

2 Diese Forderung ist für $v \in E$ und $v \in F$ durch die Einschränkung von A auf $A = V \times V \setminus \{(E \times E) \cup (F \times F)\}$ erfüllt. Für $v \in C$ wird die Forderung $(v, v) \notin A$ implizit dadurch gewährleistet, dass per definitionem gilt: $\forall v \in C : (v \in C_{EF} \vee v \in C_{FE}) \wedge (v \in C_S \vee v \in C_J)$.

- Nach Ereignissen folgt kein XOR- oder OR-Split-Konnektor im Kontrollfluss:
 $C_S \cap C_{EF} \cap C_{XOR} = \emptyset$ und $C_S \cap C_{EF} \cap C_{OR} = \emptyset$.

Die letzte Regel stellt den Ausschluss der Entscheidungsgewalt von Ereignissen in Übereinstimmung mit den erlaubten Verknüpfungsarten sicher [KNSc1992]. An diesem wichtigen Grundsatz wollen wir auch bei der Fuzzy-Erweiterung festhalten und eine entscheidungsbedingte Verzweigung von Prozessflüssen nach Ereignissen ausschließen. Wir sprechen im Folgenden weiter von EPK-Modellen und beziehen uns dabei immer auf die Menge der gemäß den vorgestellten Regeln syntaktisch korrekten EPK-Modelle.

3.3 ARIS-Erweiterung der EPK

In Forschung und Praxis existieren Erweiterungen der Modellierungssprache EPK, die unter anderem darauf abzielen, den Umfang der möglichen Sprachaussagen zu vergrößern oder die Handhabbarkeit umfangreicher Modelle zu verbessern. In der Literatur wird dementsprechend in vielen Fällen von der erweiterten EPK (kurz: eEPK) gesprochen. Dieser Sprachgebrauch wird hier jedoch nicht empfohlen, da keine einheitliche Meinung darüber existiert, welche Sprachkonstrukte zu einer Grundform und welche zu einer erweiterten Form der EPK gehören. Auch in der Modellierungspraxis wird diese Bezeichnung uneinheitlich verwendet.

Aus der Ableitung der EPK als zentrale Modellierungssprache der Architektur integrierter Informationssysteme (ARIS) [Sche2002] resultieren erweiterte Aussagen, die auf dem ARIS-Sichtenkonzept aufbauen. Diese werden durch Annotation von zusätzlichen Sprachkonstrukten an EPK-Funktionen getroffen. So werden u.a. Sprachkonstrukte vorgeschlagen, die Umfelddaten, Nachrichten, menschliche Arbeitsleistung, maschinelle Ressourcen und Computer-Hardware, Anwendungssoftware, Leistungen in Form von Sach-, Dienst- und Informationsdienstleistungen, Finanzmittel, Organisationseinheiten oder Unternehmensziele repräsentieren (vgl. Abb. 3-2).

Die Verbindung der Konstrukte, die nur mit Funktionen der EPK erfolgen kann, wird über Kanten hergestellt, die neben dem bereits eingeführten Kontrollfluss in Organisations-/Ressourcen-, Informations-, Informationsdienstleistungs- und Sachleistungs- sowie Finanzmittelfluss unterschieden werden [Sche2002, 31].

Exemplarisch werden aus den in Abb. 3-2 vorgestellten ARIS-Sprachelementen die Konstrukte der Organisations-, Daten- und Leistungssicht als zusätzliche Artefakte in die formale Repräsentation des EPK-Modells eingebettet und in einem nächsten Schritt um Attribute angereichert. Die Attributierung berücksichtigt desgleichen die Verbindungen der genannten Konstrukte mit den Funktionen der EPK, die grafisch jeweils durch Kanten des Organisations-, Daten- bzw. Leistungsflusses repräsentiert sind. Diese Erweiterung wird anschließend für die Vorstellung der beispielhaften Verarbeitung von Unschärfe in Geschäftsprozesse herangezogen.

Hierzu wird ein um ARIS-Sprachkonstrukte erweitertes EPK-Modell als ein Tupel $EPC_{ARIS} = (E, F, C, A, O, D, L, R)$ definiert. Dabei ist (E, F, C, A) ein EPK-Modell mit der Menge der Kontrollflussknoten $V = E \cup F \cup C$ und der Menge der Kontrollflusskanten A . Die Knotenmengen, welche die Artefakte der Organisations-, Daten- bzw. Leistungssicht repräsentieren, sind O für die Menge der Organisationseinheiten

(organizational units), D für die Menge der Datenobjekte (data objects) und L für die Menge der Leistungen (outputs). Für die Mengen O , D und L wird gefordert, dass sie paarweise disjunkt sind. Die Menge R enthält Mengen von Relationen, die den Funktionen die unterschiedlichen Artefakte zuordnen. Sie wird definiert als Menge $R = R^{OF} \cup R^{DF} \cup R^{FD} \cup R^{LF} \cup R^{FL}$, wobei

- $R^{OF} = \{R_1^{OF}, \dots, R_{n_{OF}}^{OF}\}$, mit R_i^{OF} ($1 \leq i \leq n_{OF}$), $n_{OF} \in \mathbb{N}$, Relationen auf $O \times F$,
- $R^{DF} = \{R_1^{DF}, \dots, R_{n_{DF}}^{DF}\}$, mit R_i^{DF} ($1 \leq i \leq n_{DF}$), $n_{DF} \in \mathbb{N}$, Relationen auf $D \times F$,
- $R^{FD} = \{R_1^{FD}, \dots, R_{n_{FD}}^{FD}\}$, mit R_i^{FD} ($1 \leq i \leq n_{FD}$), $n_{FD} \in \mathbb{N}$, Relationen auf $F \times D$,
- $R^{LF} = \{R_1^{LF}, \dots, R_{n_{LF}}^{LF}\}$, mit R_i^{LF} ($1 \leq i \leq n_{LF}$), $n_{LF} \in \mathbb{N}$, Relationen auf $L \times F$ und
- $R^{FL} = \{R_1^{FL}, \dots, R_{n_{FL}}^{FL}\}$, mit R_i^{FL} ($1 \leq i \leq n_{FL}$), $n_{FL} \in \mathbb{N}$, Relationen auf $F \times L$ sind.

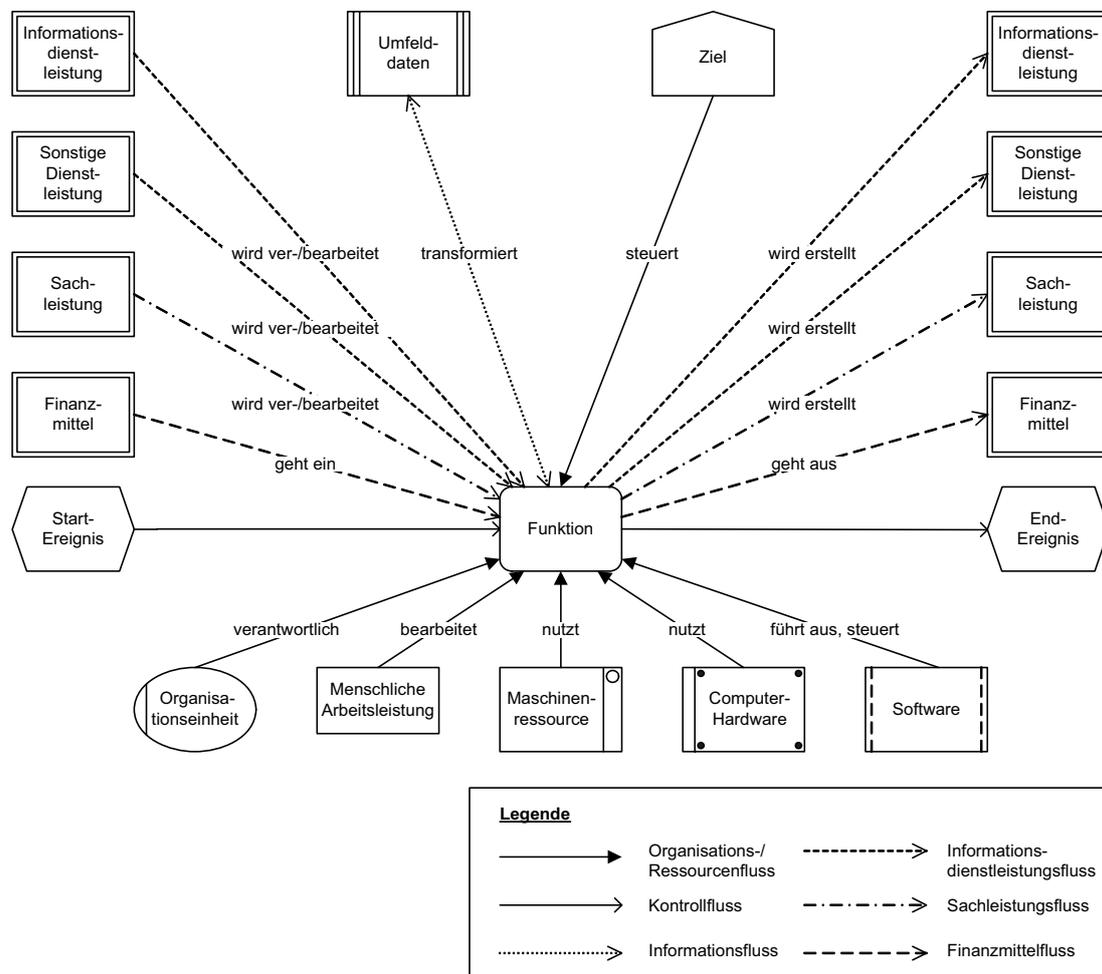


Abb. 3-2: Erweiterung der EPK um ARIS-Sprachkonstrukte [Sche2002, 31]

Die einzelnen Relationen aus den Mengen R^{OF} , R^{DF} , R^{FD} , R^{LF} und R^{FL} tragen dabei verschiedene Bedeutungen und bestimmen den Beziehungstyp zwischen den Elementen aus $O \times F, \dots, F \times L$. Eine Auswahl praxisnaher Beziehungstypen ist in Tab. 3-1 aufgelistet.

Quellobjekttyp	Zielobjekttyp	Mögliche Beziehungstypen
Organisationseinheit	Funktion	„führt aus“, „entscheidet über“, „ist verantwortlich für“, „stimmt zu“, „wirkt mit bei“, „muss informiert werden“, „muss informieren über Ergebnis von“
Datenobjekt	Funktion	„ist Input für“, „wird genehmigt von“, „wird geprüft von“
Funktion	Datenobjekt	„ändert“, „hat Output“, „erzeugt“
Leistung	Funktion	„ist Input für“, „wird verbraucht von“, „wird verwendet von“
Funktion	Leistung	„hat Output“, „produziert“

Tab. 3-1: Beziehungstypen zwischen Funktionen und ARIS-Sprachkonstrukten

Ein um ARIS-Sprachkonstrukte erweitertes EPK-Modell $EPC_{ARIS} = (E, F, C, A, O, D, L, R)$ ist genau dann syntaktisch korrekt, wenn (E, F, C, A) ein syntaktisch korrektes EPK-Modell ist und zusätzlich die folgende Bedingung erfüllt ist: Jedes Artefakt ist mit mindestens einem Knoten des EPK-Graphen (N, A) verbunden, wobei wir hier nur annotierte Artefakte an Funktionen zulassen:

- $\forall x \in O \exists f \in F \wedge \exists R \in R^{OF} : (x, f) \in R \vee (f, x) \in R,$
- $\forall x \in D \exists f \in F \wedge (\exists R \in R^{DF} \vee \exists R \in R^{FD}) : (x, f) \in R \vee (f, x) \in R,$
- $\forall x \in L \exists f \in F \wedge (\exists R \in R^{LF} \vee \exists R \in R^{FL}) : (x, f) \in R \vee (f, x) \in R.$

Auf semantische Aspekte, insbesondere das Markierungskonzept eines erweiterten EPK-Modells, wird nicht näher eingegangen. Wir nehmen per Definition an, dass das Fehlen der Realisierung eines Artefaktes zur Laufzeit den Kontrollfluss nicht blockieren, sondern lediglich zeitlich verzögern kann.

4 Fuzzy-Ereignisgesteuerte Prozesskette

4.1 Erweiterung der EPK um Attribute

Objekte der Diskurswelt können wie in der relationalen Datenbanktheorie über Tupel sie repräsentierender fachlicher Eigenschaften identifiziert werden. Die Merkmale, die allen Objekten einer Objektmenge gemeinsam sind, werden als Attribute bezeichnet. Bei der Unternehmensmodellierung referenziert man unter dem Begriff „Objekttyp“ die jeweilige Objektmenge und mit dem Begriff „Instanzen“ die einzelnen Objekte in der Objektmenge. Jedes Attribut hat einen Wertebereich, der die Menge der zugelassenen Attributwerte festlegt. Zur Identifizierung eines speziellen Objektes ist häufig nicht die Angabe aller Attributwerte erforderlich. Stattdessen kann es ausreichen, nur für bestimmte Attribute den entsprechenden Wert anzugeben, durch welche das Objekt eindeutig identifiziert wird. Ein konkreter Mitarbeiter kann beispielsweise durch Name, Alter, Adresse, Gehalt und Arbeitsstelle beschrieben werden. Beschreibende Attribute enthalten die anwendungsrelevanten Eigenschaften. Beispielsweise haben die Prozessfunktionen das Attribut „Durchlaufzeit“, das die Dauer dieses Prozessschrittes spezifiziert. Hierbei kann der Wertebereich beispielsweise als Menge von ganzen Zahlen zwischen 0 und 100 festgelegt werden. Ebenso kann die Wertemenge für das Attribut Name auf die Menge von Zeichenketten festgelegt werden, die aus alphabetischen Zeichen besteht.

Auf die Darstellung von Attributen in EPK-Modellen wird aus Gründen der Übersichtlichkeit und Komplexität in den meisten Fällen verzichtet. Allerdings ist der Detaillierungsgrad eines Modells stets so zu wählen dass der Grad der Informationen dem Anwendungszweck entspricht. Da es mit der nachfolgend vorgestellten Unschärferweiterung möglich werden soll, das bei der vereinfachenden Repräsentation im Zuge der Klassifikation bei der Abstraktion natürlichsprachlicher Aussagen durch scharfe Bildung von Klassen verlorene Prozesswissen zu explizieren, müssen die relevanten Attribute bei der Prozessmodellierung im Sinne einer problemadäquaten Präzision ausdrücklich mit einbezogen werden.

Es seien S eine Menge von Objekten der Diskurswelt, $Dom(A_i)$ ($i = 1, \dots, n$), $n \in \mathbb{N}$, Mengen von Werten und A_i ($i = 1, \dots, n$) wohldefinierte Abbildungen der Form

$$A_i : S \rightarrow Dom(A_i), s \mapsto A_i(s) \quad (i = 1, \dots, n).$$

Dann heißt $\{A_1, \dots, A_n\}$ eine Menge von Attributen (attributes) auf den Objekten der Menge S oder kurz auf S . Die Mengen $Dom(A_i)$ bezeichnet man als die Wertebereiche (domain) der Attribute A_i und die Elemente $A_i(s) \in Dom(A_i)$ werden die Attribute der Objekte s genannt. Gilt $Dom(A_i) = \{0, 1\}$, so wird A_i binäres (binary) Attribut auf S genannt. Unter den Voraussetzungen dieser Definition gibt es somit eine (interne) Darstellung der Objekte als Tupel $(A_1(s), \dots, A_n(s))$ von Attribut-Werten, d. h. als Elemente der Menge

$$Dom(A_1) \times \dots \times Dom(A_n) = \prod_{i=1}^n Dom(A_i).$$

In einem ARIS-EPK-Modell $EPC_{ARIS} = (E, F, C, A, O, D, L, R)$ notieren wir die folgenden Attribute:

- $A_1^e, \dots, A_{n_e}^e$ sind die $n_e \in \mathbb{N}$ Attribute auf dem Ereignistyp $e \in E$.
- $A_1^f, \dots, A_{n_f}^f$ sind die $n_f \in \mathbb{N}$ Attribute auf dem Funktionstyp $f \in F$.
- $A_1^o, \dots, A_{n_o}^o$ sind die $n_o \in \mathbb{N}$ Attribute auf dem Organisationseinheitstyp $o \in O$.
- $A_1^d, \dots, A_{n_d}^d$ sind die $n_d \in \mathbb{N}$ Attribute auf dem Datenobjekttyp $d \in D$.
- $A_1^l, \dots, A_{n_l}^l$ sind die $n_l \in \mathbb{N}$ Attribute auf dem Leistungstyp $l \in L$.

Umgangssprachlich gesprochen werden damit in einem fachkonzeptionellen, auf Typ-Ebene modellierten EPK-Modell jedem Knotenelement eigene Attribute zugeordnet. Verdeutlicht wird dies beispielsweise durch die Tatsache, dass ein Datenobjekt(-typ) „Kundenauftrag“ ein Attribut „Auftragssumme“ besitzt, wohingegen dieses Attribut kein Merkmal eines Datenobjekts „Artikel“ darstellt.

Wir definieren ein um Attribute erweitertes ARIS-EPK-Modell als ein Tupel $EPC_{ARIS,attr} = (E, F, C, A, O, D, L, R, M)$. Dabei sind den einzelnen Ereignissen aus E , Funktionen aus F , Organisationseinheiten aus O , Datenobjekten aus D und Leistungen aus L Attribute zugeordnet. Auf die Zuordnung von Attributen für die Menge der Kontrollflusskanten A und den Relationen aus R wird an dieser Stelle

verzichtet, da diese beschreibenden Attribute nicht zur Fuzzifizierung herangezogen werden. Die aufgezählten Attribute von Elementen aus E, F, O, D und L werden in der Menge M zusammengefasst.

Jedes Objekt weist dabei eigene identifizierende und anwendungsrelevante Attribute mit eigenen Wertemengen auf. Es sollen nur solche Attribute modelliert werden, die im jeweiligen Kontext relevant werden. Veränderungen der Attribute der Artefakte werden in der Folge nur berücksichtigt, soweit dies aus der EPK ersichtlich ist.

4.2 Fuzzy-Erweiterung der EPK

Wir definieren ein Fuzzy-EPK-Modell $FEPC = (E, F, C, A, O, D, L, R, M, FC)$ als ein um Attribute angereichertes ARIS-EPK-Modell $EPC_{ARIS,attr} = (E, F, C, A, O, D, L, R, M)$ mit den folgenden Eigenschaften:

- M ist die Menge der unscharfen Attribute des Fuzzy-EPK-Modells $FEPC$. Die Bezeichnung „unscharfes Attribut“ bezieht sich hierbei auf zwei Aspekte. Erstens wird angenommen, dass die Wertebereiche der Attribute nicht notwendigerweise scharfe Mengen sind, sondern aus Fuzzy-Mengen bestehen können. Zweitens können die Attribute als linguistische Variablen interpretiert werden. Dies impliziert, dass der Name der linguistischen Variable der Bezeichnung des Attributs entspricht und der Wertebereich des Attributs zugleich die Grundmenge der linguistischen Variablen ist.
- O , D bzw. L sind Organisationseinheits-, Datenobjekt- bzw. Leistungsmengen, die unscharfe Organisationseinheitstypen, unscharfe Datenobjekttypen bzw. unscharfe Leistungstypen enthalten. Ein unscharfer Organisationseinheitstyp, ein unscharfer Datenobjekttyp bzw. ein unscharfer Leistungstyp ist hierbei ein Organisationseinheitstyp, ein Datenobjekttyp bzw. ein Leistungstyp, der unscharfe Attribute besitzt.
- FC ist eine Menge von Fuzzy-Controllern, deren Input- und Outputgrößen als linguistische Variablen interpretierte Attribute des ARIS-EPK-Modells sind. Die Interpretation dieser Größen in Form von linguistischen Variablen ist im Controller beschrieben.
- F ist die Menge der unscharfen Funktionen des EPK-Modells. Eine unscharfe Funktion zeichnet sich dabei entweder durch ein oder mehrere unscharfe Attribute aus oder durch die Zuordnung eines Fuzzy-Controllers $fc \in FC$ zur Entscheidungsunterstützung auf der Basis unscharf formulierter Regeln bei der Ausführung. Dabei müssen alle Organisationseinheits-, Datenobjekt- bzw. Leistungstypen des EPK-Modells, deren Attribute Input- und Outputgrößen des zugeordneten Fuzzy-Controllers darstellen, über eine Kante mit dieser unscharfen Funktion verbunden sei.

- Die Menge R enthält Mengen von unscharfen Relationen³ zwischen Kontrollflussobjekten und den unterschiedlichen Artefakten: $R = \{R^{OF}, R^{DF}, R^{FD}, R^{LF}, R^{FL}\}$, mit
 - $R^{OF} = \{R_1^{OF}, \dots, R_{n_{OF}}^{OF}\}$, wobei R_i^{OF} ($1 \leq i \leq n_{OF}$), $n_{OF} \in \mathbb{N}$, unscharfe Relationen auf $O \times F$ sind.
 - $R^{DF} = \{R_1^{DF}, \dots, R_{n_{DF}}^{DF}\}$, wobei R_i^{DF} ($1 \leq i \leq n_{DF}$), $n_{DF} \in \mathbb{N}$, unscharfe Relationen auf $D \times F$ sind.
 - $R^{FD} = \{R_1^{FD}, \dots, R_{n_{FD}}^{FD}\}$, wobei R_i^{FD} ($1 \leq i \leq n_{FD}$), $n_{FD} \in \mathbb{N}$, unscharfe Relationen auf $F \times D$ sind.
 - $R^{LF} = \{R_1^{LF}, \dots, R_{n_{LF}}^{LF}\}$, wobei R_i^{LF} ($1 \leq i \leq n_{LF}$), $n_{LF} \in \mathbb{N}$, unscharfe Relationen auf $L \times F$ sind.
 - $R^{FL} = \{R_1^{FL}, \dots, R_{n_{FL}}^{FL}\}$, wobei R_i^{FL} ($1 \leq i \leq n_{FL}$), $n_{FL} \in \mathbb{N}$, unscharfe Relationen auf $F \times L$ sind.

Die Fuzzy-Erweiterung der Ereignisgesteuerten Prozesskette wird im nachfolgenden Abschnitt durch ein Beispielszenario erläutert. Dieses Szenario resultiert aus dem Forschungsprojekt „Referenzmodell-gestütztes Customizing unter Berücksichtigung unscharfer Daten“, Kennwort: Fuzzy-Customizing, Teilprojekt der Forschungskohorte „Betriebliche Referenz-Informationsmodellierung – Designtechniken und domänenbezogene Anwendung“ (BRID²), gefördert von der Deutschen Forschungsgemeinschaft (Förderkennzeichen: SCHE 185/25–1).

5 Anwendungsszenario „Fuzzy-Customizing“

Die Konstruktion von Prozessmodellen ist aus Gründen ihrer möglichen Wiederverwendung vielfach mit dem Anspruch verbunden, von unternehmensspezifischen Eigenschaften zu abstrahieren. Sie werden daher in unternehmensspezifische Prozessmodelle und Referenzprozessmodelle unterschieden. Der Begriff „unternehmensspezifisch“ charakterisiert hierbei den individuellen Charakter des entsprechenden Modells. Im Gegensatz dazu stellt ein Referenzmodell für die Entwicklung spezifischer Modelle einen Bezugspunkt dar, da es eine Klasse von Anwendungsfällen repräsentiert [Broc2003, Thom2006]. Prominente Beispiele sind im wissenschaftlichen Umfeld das Referenzmodell für industrielle Geschäftsprozesse (Y-CIM-Modell) von Scheer [Sche1997] sowie das der Unternehmenspraxis entstammende SAP R/3-Referenzmodell [KeTe1999].

Abb. 3-1 stellt einen Ausschnitt eines Referenzprozesses zur Kundenauftragsabwicklung in Form einer EPK dar. Ein Schwachpunkt des modellierten Prozesses,

3 Eine unscharfe Relation (Fuzzy-Relation) \tilde{R} über den Grundmengen Ω_1, Ω_2 ist eine Fuzzy-Menge des kartesischen Produkts $\Omega_1 \times \Omega_2$, die über eine Zugehörigkeitsfunktion $\mu_{\tilde{R}} : \Omega_1 \times \Omega_2 \rightarrow [0, 1]$ charakterisiert wird. Dabei ist jedem Element (ω_1, ω_2) , als 2-stelliges Tupel in \tilde{R} ein Zugehörigkeitsgrad $\mu_{\tilde{R}}(\omega_1, \omega_2) \in [0, 1]$ zugeordnet. Der Zugehörigkeitsgrad wird semantisch als Stärke der Fuzzy-Relation \tilde{R} zwischen den Elementen des Tupels interpretiert.

der in diesem Beitrag bislang nicht diskutiert wurde, ist erkennbar: Jedes der Negativergebnisse führt zur unmittelbaren Ablehnung des Kundenauftrags – unabhängig von den Prüfergebnissen der anderen Funktionen. Dies steht im Widerspruch zur Unternehmenspraxis, in der solche absoluten Ausschlusskriterien nur selten scharf eingehalten werden. Vielmehr werden durch menschliche Entscheidungsträger Kompensationsmechanismen angewendet, die eine Überschreitung von Grenzwerten in einem Bereich durch bessere Werte in anderen Bereichen ausgleichen. Hierbei sind die Regeln für die Wirkungszusammenhänge nicht dokumentiert, sondern beruhen auf Erfahrungswissen der Entscheidungsträger. Es handelt sich zudem meist um einfache Regeln, die nur größenordnungsmäßige Verknüpfungen herstellen und sich an Zielsystemen mit vagen Interdependenzen orientieren.

Im vorliegenden Fall könnte etwa die Entscheidung, ob das Produkt verfügbar ist, nicht nur mit einem scharfen „Ja“ oder „Nein“ beantwortet werden, sondern auch durch zusätzlichen Beschaffungsaufwand von Verhältnismäßigkeitsüberlegungen geprägt sein, sodass das Produkt z. B. aus einem anderen Lager angefordert wird, wenn alle anderen Prüfungen positiv ausfallen. Eine entsprechende Entscheidung orientiert sich an einem Trade-off zwischen den Zielen der Vermeidung von Zusatzkosten und der Ausrichtung an Kundenbedürfnissen. Hieraus ergibt sich neben dem Problemfeld der Erschließung impliziten Wissens die Herausforderung der Abbildung von Unschärfe in Referenzmodellen und Vorgehensmodellen zu deren Anpassung.

Abb. 5-1 zeigt die unscharfe Erweiterung des Referenzprozesses der Kundenauftragsabwicklung – eingebettet in die grafische Benutzeroberfläche eines Fuzzy-Modellierungswerkzeugs. Der Prozess ist im Hauptfenster in Form einer Fuzzy-EPK dargestellt. Die unscharfen Konstrukte des EPK-Modells sind durch graue Schattierung gekennzeichnet.

Nach der Definition des Kundenauftrags wird unverändert dessen Annahme geprüft. Die Prüfungen der einzelnen Funktionen des „scharfen“ Prozesses werden jedoch um Prüfungen zum Auftragsvolumen und zur Kundeneinschätzung erweitert. Die Funktionen sind dabei nicht als „untergeordnete“ Aktivitäten der Kundenauftragsprüfung, sondern als unscharfe Objektattribute der entsprechenden Datenobjekt- und Leistungstypen in Form linguistischer Variablen modelliert (vgl. Abb. 5-1, Fenster „Attribute“). Im Attribut-Explorer ist beispielsweise das Objektattribut „Auftragsvolumen“ des Datenobjekttyps „Kundenauftrag“ aktiviert. Es weist als linguistische Variable die Terme „sehr niedrig“, „niedrig“, „mittel“, „hoch“ und „sehr hoch“ auf (vgl. auch Abb. 2-2).

Im rechten Teil des Attributfensters kann der Benutzer über einen Variableneditor die Zugehörigkeitsfunktionen der linguistischen Terme verändern, z.B. durch „Ziehen“ der durch kleine Quadrate dargestellten „Eckpunkte“ der Funktionen. Ein Variablenassistent unterstützt den Benutzer durch eine automatisierte Variablendefinition. Ein Regeleditor (vgl. gleichnamiges Fenster in Abb. 5-1) zeigt die der Funktion hinterlegten Regeln an.

Im Beispiel ist der Ausschnitt einer Regelmenge mit den Eingangsvariablen „Kundeneinschätzung“ und „Auftragsvolumen“ sowie der Ausgangsvariablen „Kundenauftragsbewertung“ gegeben. Der Benutzer erzeugt die Regelmengen in der

Tabelle z. B. durch eine automatisierte Übernahme vollständiger Regelmengen aus einem mit „Konsistenzchecks“ ausgerüsteten Reglassistenten (Fuzzy-System).

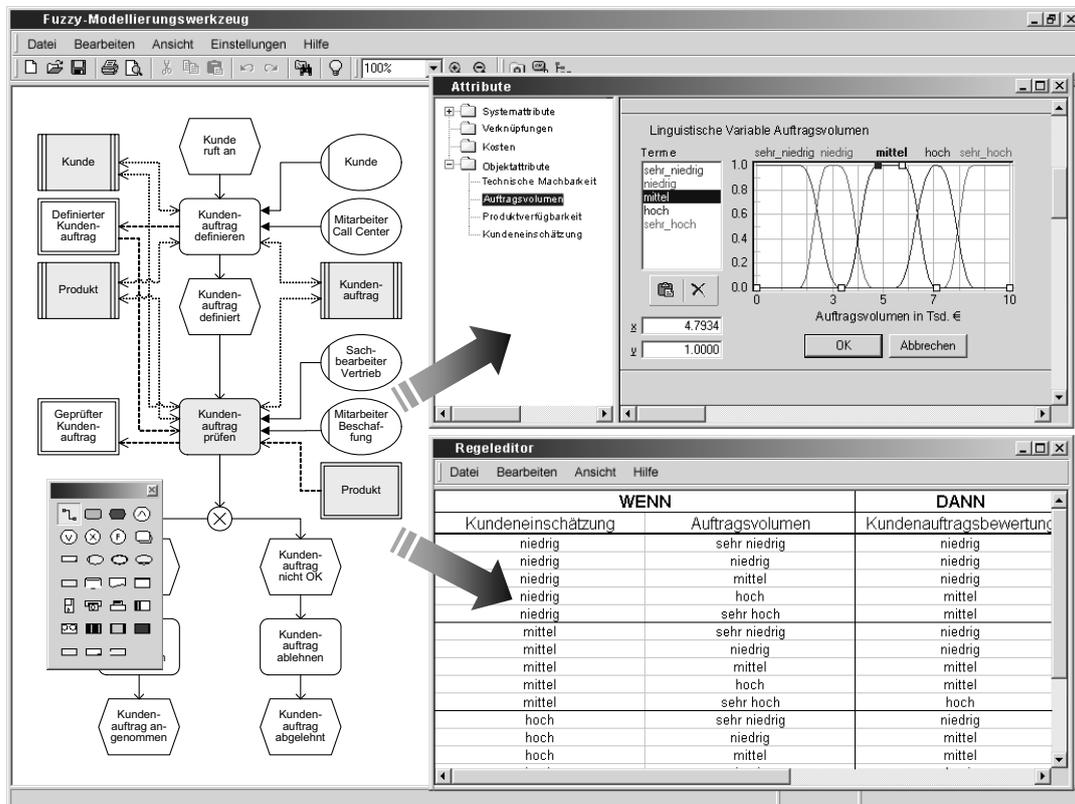


Abb. 5-1: Benutzeroberfläche des Fuzzy-Modellierungswerkzeugs

Der Referenzprozess besteht – entsprechend der in Abschnitt 4 vorgestellten Formalisierung der Fuzzy-EPK – in seiner Erweiterung aus zwei Ebenen. Die Modellierungsebene (vgl. Abb. 5-1, links) zeigt nach wie vor das Prozessmodell, im dargestellten Fall ein Fuzzy-EPK-Modell. In dieser Ebene ist die semi-formale Modellierung auf die zum Verstehen der Geschäftslogik durch den Endanwender notwendigen Inhalte begrenzt. In einer weiteren Ebene (vgl. Abb. 5-1, rechts) sind die entscheidungsunterstützenden Regeln hinterlegt, welche im Ergebnis die Annahme oder Ablehnung des Kundenauftrags bewirken. Diese Ebene greift auf Erkenntnisse der Fuzzy-Set-Theorie zurück, um die Eigenschaften abwägender Entscheidungen abzubilden.

Die Adaption eines solchen Prozesses wird nun auf das in den Entscheidungsregeln hinterlegte fachliche Wissen beschränkt und lässt die Ablauflogik des Prozesses unberührt. Durch die Berücksichtigung unscharfer Bedingungen und vage formulierter Zielvorstellungen mit Hilfe von Ansätzen der Fuzzy-Set-Theorie kann der Anwender, der über das fachliche Wissen verfügt, durch intuitive und einfache linguistische Bewertungen selbst die Adaption des Referenzprozesses vornehmen.

Dies hat ebenfalls zur Folge, dass ein bereits adaptierter Prozess prinzipiell als Referenzprozess aufgefasst werden kann – die Ablauflogik des Prozesses bleibt bei seiner Adaption unverändert und die Entscheidungsfindung muss ohnehin angepasst werden; dies obwohl der bereits adaptierte Prozess keine „Common-

Practice“-Lösung abbildet. Vielmehr ist er, wie der neu zu gestaltende Prozess, für den er als Vorlage dient, eine individuelle Ausprägung – er ist gewissermaßen die „Best-Local-Practice“-Lösung [HTBo2002, 315].

Gleichwohl ist darauf hinzuweisen, dass mit dem Anwendungsszenario „Fuzzy-Customizing“ keine Evaluierung der unscharf erweiterten EPK verbunden ist. Hierzu wurde am Institut für Wirtschaftsinformatik im DFKI, Saarbrücken, bereits eine werkzeuggestützte Simulation durchgeführt [ATLo2006]. Der vorliegende Anwendungsfall dient vielmehr dazu, aufzuzeigen, dass aus der EPK-Spracherweiterung neue Anforderungen für die fachliche Built-Time-Modellierung resultieren. Beim Design der Prozessmodelle sind Entscheidungen darüber zu treffen, welche Situationen, die bisher in der scharfen Ablauflogik des Prozessmodells selbst abgebildet werden mussten, nun mithilfe von Regeln in der Entscheidungslogik beschrieben werden können. Somit ändert sich die Vorgehensweise beim Design der fachlichen Modelle und das Konstruktionsergebnis, wie dies am Referenzprozess der Kundenauftragsabwicklung verdeutlicht wurde.

6 Verwandte Arbeiten

Es existieren nur wenige Ansätze, die Unschärfeaspekte in die Informations- bzw. Prozessmodellierung mit Hilfe der Fuzzy-Set-Theorie integrieren.

So wurde die Fuzzy-Erweiterung des Entity-Relationship-Modells (ERM) von Zvieli, Chen [ZvCh1986] beschrieben. Hierbei können Entitytypen, Beziehungstypen und Attributmengen Fuzzy-Werte annehmen. Die Berücksichtigung dieser fuzzifizierten Datenstrukturen führt konsequent zur Verarbeitung der unscharfen Daten in den entsprechenden betrieblichen Geschäftsprozessen.

Fuzzy-Theorie-basierte Erweiterungen objektorientierter Modellierungsmethoden für Geschäftsprozesse sind bei Benedicenti et al. [BSVV1998] und Cox [Cox1999, Cox2002] zu finden. Ein auf der Fuzzy-Set-Theorie basierender objektorientierter Ansatz zur Simulation von Geschäftsprozessen wird durch Völkner, Werners [Völk1998, VöWe2002] vorgestellt.

Zur Beschreibung dynamischer Aspekte betrieblicher Informationssysteme werden u. a. Petri-Netze eingesetzt. Das zweiwertige Verhalten von Stellen und Transitionen eines Petri-Netztes ist bei der Abbildung wissensintensiver und schwach strukturierter Prozesse jedoch von Nachteil. Um das Systemverhalten auch bei unscharfen Prozessbedingungen oder unvollständigen, vagen Informationen darstellen zu können, wurden Petri-Netze durch Fuzzy-Konzepte erweitert. Das Fuzzy-Petri-Netz [Lipp1988] entsteht durch die Projektion mehrerer scharfer Petri-Netze, bei der die Strukturinformationen als unscharfe Mengen abgebildet werden.

Becker, Rehfeldt, Turowski [BRTu1996, Reh1998] zeigen am Beispiel der industriellen Auftragsabwicklung die Berücksichtigung unscharfer Daten in der Geschäftsprozessmodellierung mit Ereignisgesteuerten Prozessketten exemplarisch auf. Als wesentliche, mit Unschärfe in Form von Unsicherheit behaftete exogene Eingangsdaten werden vage Vertriebsinformationen betrachtet, die in vorläufige Kundenaufträge umgewandelt werden. Diese „unscharfe Ergänzung“ der Prozesse wird durch schattierte Objekte visualisiert. Aus methodischer Sicht müssen unscharfe und scharfe Modellobjekte bei der fachkonzeptionellen Darstellung eines Geschäftsprozesses jedoch nicht unterschieden werden.

Vielmehr sind auch die das Verhalten beschreibenden Regeln sowie bekannte Parameter (z.B. Partitionierungen) im Sinne eines umfassenden Wissensmanagements unabhängig von einem Implementierungsmodell bereits beim Design der Prozesse auf der fachlichen Ebene zu erfassen.

Thomas und Adam [THAd2002, AdTh2005, ATLo2006] untersuchen mit weiteren Co-Autoren, wie unscharfe Daten zum Design wissensintensiver und schwach strukturierter Geschäftsprozesse und ihrer Implementierung in Anwendungssystemen genutzt werden können.

Zusammenfassend bleibt zu bemängeln, dass nur wenige Ansätze existieren, die auf die Aspekte und Anforderungen einer durchgängigen Integration von Unschärfe in die Unternehmens- und Geschäftsprozessmodellierung eingehen. Die Möglichkeiten der Integration von Unschärfe in die Informationsmodellierung werden vielmehr durch bestehende Ansätze, die Vollständigkeit und Präzisierung fordern, beschränkt. Die genannten Ansätze beschränken sich auf die grafische Repräsentation oder textuelle Notation „unscharfen“ Modellierens. Eine wissenschaftliche Diskussion der „Erweiterung“ des Geschäftsprozessmanagements um Unschärfeaspekte wurde bislang nicht geführt. Eine entsprechende Werkzeugunterstützung existiert ebenfalls nicht.

7 Zusammenfassung und zukünftige Forschungsfragen

In dem vorliegenden Beitrag wurde ein Ansatz zur Integration von Unschärfeaspekten in das Geschäftsprozessmanagement entwickelt. Die Integration wurde in zweierlei Hinsicht beispielhaft vollzogen. Erstens erfolgte die Berücksichtigung unscharfer Daten mit Hilfe der Fuzzy-Set-Theorie als Teilgebiet des Soft Computing. Zweitens wurde sie am Beispiel einer etablierten Modellierungssprache für Geschäftsprozesse, der Ereignisgesteuerten Prozesskette, durchgeführt. Das Konzept entspricht im übertragenen Sinne einer „Ebenenerweiterung“ der Beschreibungssprache: Während die Geschäftsprozessmodelle auf die zum Verstehen der Geschäftslogik durch den Endanwender notwendigen Inhalte begrenzt sind, wird das Fachwissen zur Entscheidungsunterstützung einzelnen Modellelementen hinterlegt.

Insbesondere in den beschriebenen Anwendungen zeigte sich, dass durch die Modellierung vagen Wissens mit Fuzzy Logic im Geschäftsprozessmanagement viele Situationen exakter als bisher beschrieben werden können. Daher eignen sich auf Fuzzy-Logik aufbauende regelbasierte Systeme in hohem Maße zur Steuerung von Prozessen. Da die Regelbasis auf Wenn-Dann-Regeln basiert, kann ihr funktionales Verhalten relativ leicht nachvollzogen und vorhandenes Wissen relativ einfach integriert werden. Bei Modifikationen des zu steuernden Prozesses können alte Regeln direkt übernommen oder müssen nur geringfügig modifiziert werden. Dies erleichtert die ständige Verbesserung der Prozessdefinitionen im Sinne eines Continuous Process Improvement [Robs1991, Sche1996].

Eine zukünftige Herausforderung für ihre Forschungstätigkeiten sehen die Autoren vor allem in der Beantwortung der Frage, ob im Fuzzy-Geschäftsprozessmanagement die Erstellung angemessener linguistischer Variablen und Regelbasen wirtschaftlich sinnvoll erfolgen kann. Als problematisch erweist sich in der Praxis insbesondere das Aufstellen der Regelbasis. Jedes ungewünschte Fehlverhalten muss vom Entwickler analysiert und entsprechend „von Hand“ korrigiert werden. Durch die

Optimierung regelbasierter Fuzzy-Systeme mittels neuronaler Netze können Fuzzy-Mengen angepasst und die Regelbasis erlernt bzw. korrigiert werden. Die Fähigkeit von künstlichen neuronalen Netzen zum Aufdecken von Geschäftslogik in Prozessen („Process Mining“) sowie zur Verbesserung von Geschäftsprozessen durch Lernen werden aktuell diskutiert [ATLo2006].

Literaturverzeichnis

- [ADKi2002] van der Aalst, W. M. P.; Desel, J.; Kindler, E.: On the semantics of EPCs: A vicious circle. In: Nüttgens, M.; Rump, F. J. (Hrsg.): EPK 2002: Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten, 21.-22. November in Trier, Proceedings. Bonn : GI, 2002, S. 71-79.
- [Aals1999] van der Aalst, W. M. P.: Formalization and verification of event-driven process chains. In: Information and Software Technology 10 (1999) 41, S. 639-650.
- [AdTh2005] Adam, O.; Thomas, O.: A Fuzzy Based Approach to the Improvement of Business Processes. In: Castellanos, M.; Weijters, T. (Hrsg.): BPI'05 : Workshop on Business Process Intelligence ; Nancy, France, September 5, 2005. Nancy, 2005, S. 25-35.
- [ATLo2006] Adam, O.; Thomas, O.; Loos, P.: Soft Business Process Intelligence – Verbesserung von Geschäftsprozessen mit Neuro-Fuzzy-Methoden. In: Lehner, F.; Nösekabel, H.; Kleinschmidt, P. (Hrsg.): Multikonferenz Wirtschaftsinformatik 2006 : Band 2. GI-TO: Berlin, 2006, S. 57-69.
- [BKRo2005] Becker, J.; Kugeler, M.; Rosemann, M. (Hrsg.): Prozessmanagement : Ein Leitfaden zur prozessorientierten Organisationsgestaltung. 5. Aufl. Springer: Berlin, 2005
- [BRTu1996] Becker, J.; Rehfeldt, M.; Turowski, K.: Auftragsabwicklung mit unscharfen Daten in der Industrie. In: Biethahn, J. et al. (Hrsg.): Betriebliche Anwendungen von Fuzzy-Technologien : Tagungsband zum 2. Göttinger Symposium Softcomputing am 29. Feb. 1996 an der Universität Göttingen, S. 51-61.
- [Bosc1993] Bosch, H.: Entscheidung und Unschärfe : Eine entscheidungstheoretische Analyse der Fuzzy-Set-Theorie. Eul: Bergisch Gladbach, 1993.
- [Broc2003] vom Brocke, J.: Referenzmodellierung: Gestaltung und Verteilung von Konstruktionsprozessen. Logos: Berlin, 2003.
- [BSVV1998] Benedicenti, L.; Succi, G.; Vernazza, T.; Valerio, A.: Object Oriented Process Modeling with Fuzzy Logic. In: Carroll, J. et al. (Hrsg.): Applied computing 1998: Proceedings of the 1998 ACM Symposium on Applied Computing; Atlanta, Georgia, February 27-March 1, 1998. ACM Press: Danvers, 1998, S. 267-271.
- [Cox1999] Cox, E.: Striving for Imprecision: Fuzzy Knowledge Bases for Business Process Modeling. In: PC AI 4 (1999) 13.
- [Cox2002] Cox, E.: Knowledge-Based Business Process Modeling: Complex Systems Design Through A Fusion of Computational Intelligence And Object-Oriented Models. In: PC AI 2 (2002) 16, S. 15-23.
- [DeRi2001] Dehnert, J.; Rittgen, P.: Relaxed Soundness of Business Processes. In: Dittrich, K. R.; Geppert, A.; Norrie, M. C. (Hrsg.): Proc. of the 13th Conference on

- Advanced Information Systems Engineering (CAISE'01). Springer: Interlaken, Berlin, 2001, S. 157-170.
- [Fort2002] Forte, M.: Unschärfen in Geschäftsprozessen. Weißensee : Berlin, 2002.
- [HTBo2002] Habermann, F.; Thomas, O.; Botta, C.: Organisational-Memory-System zur Unterstützung informationstechnisch basierter Verbesserungen von Geschäftsprozessen. In: Becker, J.; Knackstedt, R. (Hrsg.): Wissensmanagement mit Referenzmodellen. Physica: Heidelberg:, 2002, S. 291-322.
- [HKSc1992] Hoffmann, W.; Kirsch, J.; Scheer, A.-W.: Modellierung mit Ereignisgesteuerten Prozeßketten : Methodenhandbuch ; Stand: Dezember 1992. In: Scheer, A.-W. (Hrsg.): Veröffentlichungen des Instituts für Wirtschaftsinformatik, Nr. 101. Universität des Saarlandes: Saarbrücken, 1992.
- [Hüss2003] Hüsselmann, C.: Fuzzy-Geschäftsprozessmanagement. Eul: Lohmar, 2003.
- [IDS2003] IDS Scheer AG (Hrsg.): ARIS Toolset, ARIS Version 6.2.1.31203. IDS Scheer AG: Saarbrücken, 2003.
- [KNSc1992] Keller, G.; Nüttgens, M.; Scheer, A.-W.: Semantische Prozeßmodellierung auf der Grundlage "Ereignisgesteuerter Prozeßketten (EPK)". In: Scheer, A.-W. (Hrsg.): Veröffentlichungen des Instituts für Wirtschaftsinformatik, Nr. 89. Universität des Saarlandes: Saarbrücken, 1992.
- [KeTe1999] Keller, G.; Teufel, T.: SAP R/3 prozeßorientiert anwenden : Iteratives Prozess-Prototyping mit Ereignisgesteuerten Prozessketten und Knowledge Maps. 3. Aufl. Addison-Wesley: Bonn, 1999.
- [Kind2004] Kindler, E.: On the Semantics of EPCs : A Framework for Resolving the Vicious Circle. In: Desel, J.; Pernici, B.; Weske, M. (Hrsg.): Business Process Management: Second International Conference (BPM 2004), Potsdam, Germany, June 17.-18, 2004; Proceedings. Springer: 2004, S. 82-97.
- [Kind2006] Kindler, E.: On the semantics of EPCs: Resolving the vicious circle. In: Data & Knowledge Engineering 1 (2006) 56, S. 23-40.
- [LSWe1998] Langner, P.; Schneider, C.; Wehler, J.: Petri Net Based Certification of Event driven Process Chains. In: Desel, J.; Silva, M. (Hrsg.): Application and theory of Petri nets 1998: 19th international conference; proceedings. Springer: Berlin, 1998, S. 286-305.
- [Lipp1982] Lipp, H.-P.: Anwendung eines Fuzzy Petri Netzes zur Beschreibung von Koordinationssteuerungen in komplexen Produktionssystemen. In: Wissenschaftliche Zeitschrift der Technischen Universität Karl-Marx-Stadt, 5 (1982) 24, S. 633-639.
- [NüRu2002] Nüttgens, M.; Rump, F. J.: Syntax und Semantik Ereignisgesteuerter Prozessketten (EPK). In: Desel, J.; Weske, M. (Hrsg.): Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen (Promise '2002), Hasso-Plattner-Institut für Softwaresystemtechnik an der Universität Potsdam, 9.-11. Oktober 2002. Köllen: Bonn, 2002, S. 64-77.
- [ÖsWi2003] Österle, H.; Winter, R.: Business Engineering. In: Österle, H.; Winter, R. (Hrsg.): Business Engineering : Auf dem Weg zum Unternehmen des Informationszeitalters. 2. Aufl. Springer: Berlin, 2003, S. 3-19.
- [Rehf1998] Rehfeldt, M.: Koordination der Auftragsabwicklung : Verwendung von unscharfen Informationen. DUV: Wiesbaden, 1998.

- [Ritt2000] Rittgen, P.: Quo vadis EPK in ARIS? – Ansätze zu syntaktischen Erweiterungen und einer formalen Semantik. In: *Wirtschaftsinformatik*, 1 (2000) 42, S. 27-35
- [RoAa2006] Rosemann, M.; van der Aalst, W. M. P.: A configurable reference modelling language. In: *Information Systems*, 2006 (corrected proof).
- [Robs1991] Robson, G. D.: *Continuous process improvement : simplifying work flow systems*. Free Press: New York, 1991.
- [Sche2002] Scheer, A.-W.: *ARIS – Vom Geschäftsprozess zum Anwendungssystem*. 4. Aufl. Springer: Berlin, 2002.
- [Sche1996] Scheer, A.-W.: *ARIS-House of Business Engineering : Von der Geschäftsprozessmodellierung zur Workflowgesteuerten Anwendung ; vom Business Process Reengineering zum Continuous Process Improvement*. In: Scheer, A.-W. (Hrsg.): *Veröffentlichungen des Instituts für Wirtschaftsinformatik*, Nr. 133. Universität des Saarlandes: Saarbrücken, 1996.
- [Sche1997] Scheer, A.-W.: *Wirtschaftsinformatik : Referenzmodelle für industrielle Geschäftsprozesse*. 7. Aufl. Springer : Berlin, 1997.
- [STAd2005] Scheer, A.-W.; Thomas, O.; Adam, O.: *Process Modeling Using Event-driven Process Chains*. In: Dumas, M.; van der Aalst, W. M. P.; ter Hofstede, A. H. M. (Hrsg.): *Process-aware Information Systems; Bridging People and Software through Process Technology*. Wiley: Hoboken, New Jersey, 2005, S. 119-145.
- [THAd2002] Thomas, O.; Hüselmann, C.; Adam, O.: *Fuzzy-Ereignisgesteuerte Prozessketten : Geschäftsprozessmodellierung unter Berücksichtigung unscharfer Daten*. In: Nüttgens, M.; Rump, F. J. (Hrsg.): *EPK 2002 : Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten*, 21.-22. November in Trier, Proceedings. GI: Bonn, 2002, S. 7-16.
- [Thom2006] Thomas, O.: *Understanding the Term Reference Model in Information Systems Research: History, Literature Analysis and Explanation*. In: Bussler, C.; Haller, A. (Hrsg.): *Business Process Management Workshops: BPM 2005 International Workshops; Revised Selected Papers*. Springer: Berlin, 2006, S. 484-496.
- [Völk1998] Völkner, P.: *Modellbasierte Planung von Geschäftsprozessabläufen : Entwicklung eines Entscheidungsunterstützungssystems auf Grundlage objektorientierter Simulation*. Gabler: Wiesbaden, 1998
- [VöWe2002] Völkner, P.; Werners, B.: *A simulation-based decision support system for business process planning*. In: *Fuzzy Sets and Systems*, 3 (2002) 25, S. 275-288.
- [Zade1965] Zadeh, L. A.: *Fuzzy sets*. In: *Information and Control*, 3 (1965) 8, S. 338-353.
- [Zade1973] Zadeh, L. A.: *Outline of a new approach to the analysis of complex systems and decision processes*. In: *IEEE Transactions on Systems, Man and Cybernetics* 1 (1973) 3, S. 24-44.
- [ZvCh1986] Zvieli, A.; Chen, P. P.-S.: *Entity-Relationship Modeling and Fuzzy Databases*. In: *International Conference on Data Engineering*, February 5–7, 1986, Bonaventure Hotel, Los Angeles, California, USA. Washington, DC: IEEE Computer Society Press, 1986, S. 320-327.

Koordination integrierter Logistikprozesse im Hafen

Leif Meier, Helge Fischer

Georg-August-Universität Göttingen
lmeier@uni-goettingen.de
hfische1@uni-goettingen.de

Zusammenfassung: Die Komplexität von Entscheidungsproblemen in der Hafenlogistik und deren Interaktion zwingt Entscheidungsträger zu einer Zerlegung des Gesamtproblems in Teilprobleme und deren sequentiellen Wiederherstellung. In diesem Beitrag soll gezeigt werden, dass diese einfache Form der Koordination zu unerwünschten, suboptimalen Lösungen führen kann und ein agentenbasiertes Modell skizziert werden, um effizientere Koordinationsmechanismen zu untersuchen.

1 Einführung in den Problembereich

Container Terminals (CT) stellen wichtige Knotenpunkte in dem globalen Transportnetzwerk dar. Innerhalb dieser Knoten, die durch das Zusammenwirken verschiedener Verkehrsträger gekennzeichnet sind, sehen sich die Entscheidungsträger mit einer Vielzahl komplexer und wechselseitig beeinflussender Problemstellungen konfrontiert [GRZa1998]. Aufgrund der Vielzahl der Entscheidungskriterien und der Komplexität der Operationen ist es heute unmöglich, Entscheidungen zu treffen, die allumfassend die gesetzten Zielsysteme optimieren. Aus diesem Grund wird das Gesamtproblem, sowohl in der Wissenschaft als auch in der Praxis, zumeist in mehrere (sequentielle) Teilprobleme zerlegt [ZLWL+2003], bei dem der Input eines Nachfolgeproblems dem Output des Vorgängerproblems entspricht. Der sequentielle Ansatz stellt somit eine Komplexitätsreduzierung dar, bietet jedoch keinesfalls einen Ansatz, der unbedingt zu guten Lösungen führen muss.

Die Notwendigkeit der Suche nach neuen Verfahren zur Optimierung von Prozessen ist gegeben. Dem rasanten Anstieg der Nachfrage im Bereich der Containerlogistik kann nur mit entsprechender Infrastruktur, hoch entwickeltem Hafenequipment sowie optimierten Prozessen begegnet werden. Die zukünftige Konkurrenzfähigkeit von Container Terminals wird im Kern von effizienten Logistikprozessen abhängig sein, vor allem deshalb, weil die Möglichkeiten strategischer Maßnahmen, z.B. einer Erweiterung des Terminalgeländes, bereits heute oftmals erschöpft sind [SVSt2004]. Dennoch liegt der Schwerpunkt der Arbeiten zur Prozessoptimierung auf Modellen, die lediglich separate Entscheidungsprobleme innerhalb eines Prozesses aufgreifen und sich im Kern in der Betrachtung unterschiedlicher Zielfunktionen, Restriktionen und Modellannahmen unterscheiden.

Ein Ansatz, Prozesse zu optimieren, kann sich durch eine weitergehende, verbesserte Koordination von Teillösungen ergeben. Dabei gilt es zu untersuchen, inwieweit sich jeweils Vorteile gegenüber der vereinfachenden, sequentiellen Herangehensweise erwarten lassen. Hierfür ist es auch erforderlich, Zielbeziehungen ganzheitlich zu verstehen und Zielsysteme der Teilprobleme entsprechend zu gestalten, wobei das Zielsystem alle von einem Entscheidungsträger verfolgten Ziele enthält [BMRu2004]. Für die Koordination der einzelnen Planungssysteme lassen sich Multiagenten-Systeme (MAS) einsetzen, deren Eigenschaften in diesem Bereich sehr gute Einsatzmöglichkeiten erhoffen lassen.

Dieser Beitrag ist wie folgt aufgebaut. In Kapitel 2 werden Problemstellungen der unterschiedlichen Entscheidungsträger in einem Container Terminal aufgezeigt und Modelle und Lösungsansätze der Literatur vorgestellt. Kapitel 3 zeigt in einer einfachen Untersuchung mit sequentieller Koordination, dass diese Lösungen zu unerwünschten Ergebnissen führen können. In Kapitel 4 soll daraufhin ein agentenbasiertes Modell skizziert werden, mit dem sich weitere Koordinationsverfahren, aber auch unterschiedliche Lösungsansätze im Prozesszusammenhang testen lassen, die verbesserte Lösungen versprechen. Zuletzt wird in einem Ausblick auf weitere Forschungsansätze in diesem Bereich verwiesen.

2 Modelle und Lösungsansätze der Literatur

Das Kerngeschäft eines Container Terminals besteht im Laden und Löschen von Containerschiffen, der Zwischenlagerung der Container sowie ihrer landseitigen An- und Auslieferung. Somit liegt die Aufgabe des Terminals in der Transformation des Containerschiffs von einem vom Vorhafen bestimmten Start- in einem vom Reeder gegebenen Zielzustand.

Entscheidungen des Managements sollten sich an den Kundenanforderungen orientieren, die die Leistungsfähigkeit, die Prozessverlässlichkeit, die Kosten, die Qualität sowie die Anpassungsfähigkeit des Terminals umfassen [Hene2004]. Diese Merkmale beeinflussen die Wahl eines Reeders, einen bestimmten Terminal einer Region anzufahren. Noch immer, vor allem aufgrund der Prozesskomplexität, basieren Entscheidungen zumeist auf Erfahrungen von Praktikern und terminalpolitischen Vorgaben, nicht aber auf objektiven Planungen [Hene2004]. U.a. [SVSt2004, Hene2004] heben hervor, dass sich die bisherige Forschung im Bereich der Prozessoptimierung in der Hafenlogistik auf isoliert betrachtete Subsysteme konzentriert, eine integrierte oder ganzheitliche Planung findet bisher kaum statt.

[SVSt2004] fassen die derzeitige Situation wie folgt zusammen: „Until now the focus is not on optimizing the transport chain as a whole but on optimizing several separate parts of the chain.“ [LWWa2005] bestätigen zudem die Position der Vorgehensweise in der Praxis: „Such hard problems [container terminal decision problems] motivate practitioners to formulate the problem hierarchically and solve them sequentially.“

Angelehnt an [ZLWL+2003] lassen sich die Entscheidungsbereiche eines CT wie folgt, sequentiell gliedern:

Abbildung 2-1 zeigt die idealtypischen Entscheidungsbereiche im Terminal. Ausgehend von den Containerschiffen, die es vom Startzustand (Ladeplan) in den Zielzustand (Stauplan) zu überführen gilt, werden im einfachsten Fall nacheinander

die Bereiche der Liegeplatzplanung⁴, der Containerbrückenzuordnung⁵, der Yardplanung⁶ sowie der Transportplanung⁷ durchlaufen. [SVSt2004] geben einen umfassenden Überblick über die Probleme der einzelnen Teilbereiche.

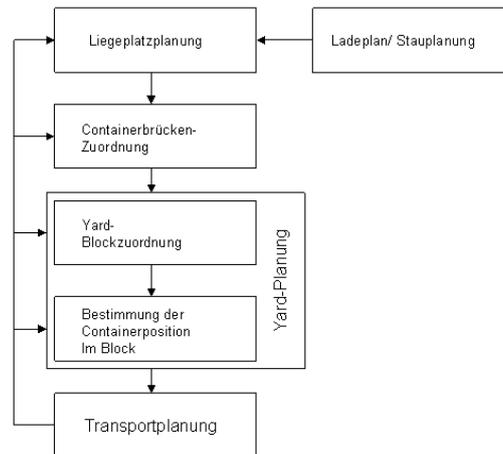


Abbildung 2-1: Entscheidungsbereiche im Terminal

Auf der Suche nach Ansätzen zur Prozessoptimierung ist es sinnvoll die Aussagen von [Pfoh2004] zu betrachten, der, mit allgemeinem Bezug, zwei wichtige Funktionen des Logistikmanagements nennt: Zum einen dient es dem Management der einzelnen Teilfunktionen, deren spezielle Bereiche zum Teil weit entwickelt sind und bei denen auf umfangreiches Potential an Methoden und Verfahren zurückgegriffen werden kann. Als Beispiele nennt [Pfoh2004] die Tourenplanung, Materialbedarfsplanung und die Stauraumplanung. Mit Bezug auf den betrachteten Bereich der Hafenlogistik lässt sich diese Aussage uneingeschränkt bestätigen. Für Teilbereiche existieren eine Vielzahl unterschiedlicher Modellierungsansätze und Lösungsvorschläge, auf die im Weiteren eingegangen werden soll. Zum anderen der zweite, von [Pfoh2004] als weitaus wichtiger betrachtete Aspekt des Logistikmanagements, „dient der Umsetzung der Logistikkonzeption durch koordinierende und integrierende Managementaktivitäten.“ Koordination und ganzheitliche Betrachtung von Logistiksystemen und -prozessen werden verlangt. „Eine wichtige Koordinationsmaßnahme sind umfassende Planungen, die mehrere technische Teilsysteme und mehrere Organisationseinheiten [...] zum Gegenstand haben. [...] Der ganzheitlichen Planung, Organisation und Führung sowie dem umfassenden Controlling kommt deshalb eine Schlüsselrolle bei der Umsetzung der Logistikkonzeption zu.“ [Pfoh2004]. Dieser Aspekt deckt sich wiederum mit den Aussagen von [SVSt2004] für den speziellen Bereich der Hafenlogistik.

4 Zuordnung von Containerschiffen, es geht um die Frage „wo“ und „wann“ ein Schiff anlegen soll.

5 Eine Containerbrücke (QC, aus dem engl. Quay crane) dient dem Be- und Entladen von Containerschiffen.

6 Der Yard dient der Zwischenlagerung der Container.

7 Die Transportplanung dient dem Transport von Containern zwischen Kai und Yard. In der Untersuchung in Kapitel 3 kommen Van Carrier (VC) zum Einsatz, die in der Lage sind, einen Container selbständig aufzunehmen, zu transportieren und abzusetzen. Andere Möglichkeiten für den Transport stellen z.B. Lkw oder AGV dar [StVS2004].

Greift man diese Punkte auf, dann stellt sich die Frage, wie und mit welchen Methoden sich die genannten Koordinations- und Integrationsmaßnahmen untersuchen lassen, um darauf aufbauend Prozessverbesserungsmöglichkeiten zu entwickeln. In Kapitel 4 soll dafür die Einsatzfähigkeit multiagentenbasierter Ansätze untersucht werden.

Bevor eine Auswahl von Modellen und Lösungsansätzen aufgezeigt werden soll, muss auf die Problematik der Modellbildung eingegangen werden. Ein Modell lässt sich als vereinfachte Darstellung der Realität verstehen [BMRu2004]. Existierende Modelle in der Hafenlogistik abstrahieren zumeist sehr stark von der Realität, die um eine Vielzahl von Details ergänzt wird. Jeder Terminal weist spezifische und entscheidungsrelevante Merkmale auf, die sich nur schwer in generischen Modellen abbilden lassen. Problemstellungen der Realität werden häufig durch eine weitere Komplexitätsreduzierung angegangen, indem beispielsweise durch Verträge bestimmte Zuordnungen vorgeschrieben werden. Beispiele hierfür sind dedizierte Liegeplätze, dedizierte Yard-Reihen oder auch eine vertraglich fixierte Anzahl operierender Containerbrücken (QC). Dieser Aspekt ist bei der Analyse von Ergebnissen einer Simulation und vor der Übertragung von Ergebnissen auf die Realität unbedingt zu berücksichtigen.

Mit Tabelle 1 soll zunächst ein Ausschnitt gegeben werden, für Bereiche in denen Methoden des Soft Computings und Metaheuristiken ihren Einsatz in der Hafenlogistik bereits gefunden haben.

Autor	Entscheidungsbereich	Methode
Wilson, Roach, 2000	Stauplanung	Tabu Search
Dubrovsky et al., 2002	Stauplanung	Genetische Algorithmen
Kim, Moon, 2004	Liegeplatzplanung	Simulated Annealing
Cordeau et al., 2005	Liegeplatzplanung	Tabu Search
Nishimura et al., 2001	Liegeplatzplanung	Genetische Algorithmen
Lokuge, Alahakoon, 2004	Liegeplatzplanung	NN, ANFI
Lim et al., 2003	Containerbrückenaufteilung	Simulated Annealing
Cao, Uebe, 1995	Yardplanung	Tabu Search
Kozan, Preston, 2001	Yardplanung	Genetische Algorithmen
Böse et al., 2001	Transportplanung	Genetische Algorithmen

Tabelle 1: Soft Computing und Metaheuristiken in der Hafenlogistik⁸

⁸ Die Übersicht stellt lediglich einen Ausschnitt dar und erhebt keinen Anspruch auf Vollständigkeit.

Weitere Ansätze zeigt das Projekt NETRALOG [Stee2000], in dem die Entwicklung und der Einsatz bioanaloger Algorithmen in der Schiffsabfertigung untersucht werden.

Bezogen auf das Beispiel in Kapitel 3, in dem ein fiktiver Van Carrier-Terminal untersucht werden soll, sollen im Folgenden die dort zum Einsatz kommenden Verfahren näher betrachtet werden.

Im Rahmen der Liegeplatzplanung muss entschieden werden, wann und wo ein Containerschiff am Kai anlegen soll. [GuCh2004] stellen eine zusammengesetzte Heuristik vor, die auf paarweisem Austausch von Containerschiffen beruht, wenn sich dadurch eine Verbesserung erzielen lässt und einer *tree-search-procedure*, die die Schiffe wieder anordnet. Als Zielgröße verwenden die Autoren die Minimierung der gewichteten Aufenthaltszeit der Containerschiffe im Hafen, nennen zugleich aber die Bedeutung der Untersuchung weiterer Zielfunktionen.

Auf die Containerbrückenaufteilung wird in diesem (einfachen) Beispiel verzichtet. Es wird stattdessen angenommen, dass an jeder Liegeplatzsektion jeweils eine Containerbrücke zur Verfügung steht.

Ist eine Lösung für die Liegeplatzplanung und die Aufteilung der Containerbrücken vorhanden, dann lässt sich der erwartete Containerfluss für den geplanten Zeitraum bestimmen. Aus dem erwarteten Containerfluss kann dann die zieloptimale Zuordnung innerhalb der Yard-Blöcke berechnet werden. [MLWLR2005] schlagen hierfür einen *fill-ratio equalization*-Ansatz vor, mit dem Ziel Staus und Blockierungen an einem Block durch ungleiche Beanspruchung zu vermeiden. Dieses Vorgehen ist Teil der Entwicklung eines DSS für Operationen im Terminal. Die Zuordnung zu konkreten Blöcken wird dann mit dem Ziel der Minimierung von Transportdistanzen zwischen Kai und Yard bestimmt.

Das Problem der exakten Positionierung des Containers innerhalb des Yard-Blocks ist in diesem Beispiel nicht Gegenstand der Betrachtung.

Die Zuordnung von Container-Jobs zu Van Carriern und Containerbrücken untersucht [Boe2001] mit Hilfe von genetischen Algorithmen. Ein Containerjob ist gekennzeichnet durch seinen Start- und Zielort sowie der Reihenfolge der Entstehung. Diese Modellparameter ergeben sich aus den vorangehenden Planungsergebnissen. Unterschiede hinsichtlich Containergröße, -gewicht und/oder -typ werden zur Vereinfachung nicht berücksichtigt. Zielsetzung des von [Boe2001] vorgeschlagenen Ansatzes ist die Maximierung der Containerbrückenproduktivität über eine Minimierung der VC-Wartezeiten zu erreichen.

In dem folgenden Kapitel soll mit Hilfe dieser Verfahren ein einfaches Szenario geplant und deren Ergebnisse betrachtet und analysiert werden.

3 Untersuchung am Beispiel sequentieller Koordination

In dem folgenden Beispiel soll untersucht werden, wie die einzelnen Entscheidungsbereiche zusammenwirken und wo Koordinationsbedarf auftreten kann. Vereinfachend werden hierfür lediglich Modelle aus der Liegeplatzplanung, der Transportplanung und im Rahmen der Yardplanung, die Zuordnung von Blöcken für die Containerlagerung betrachtet.

Beginnend mit der Liegeplatzplanung wurde zur Lösung des Modells die Heuristik von [GuCh2004] implementiert. Als Zielfunktion wird die Minimierung der gewichteten Aufenthaltszeit der Containerschiffe betrachtet. Die Operationszeit des Schiffes ist gegeben. Die sich so ergebende Lösung ist in Abbildung 3-1 dargestellt. Der Planungszeitraum umfasst eine Woche, zwölf Schiffe in drei Kategorien erreichen in dieser Zeit den Hafen.⁹

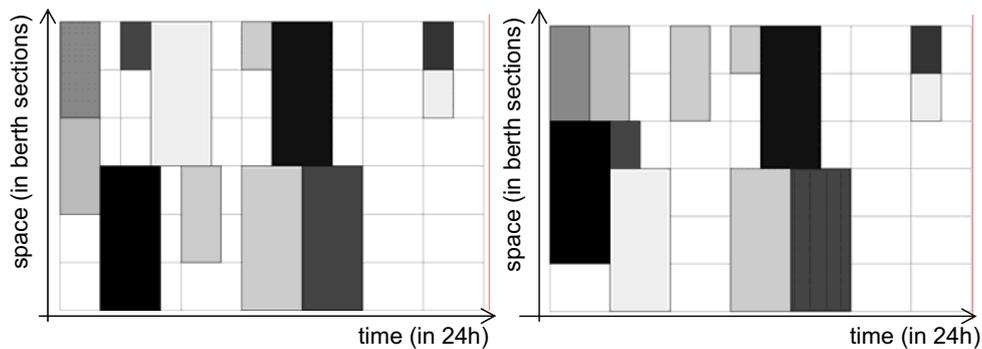


Abbildung 3-1: Ausgangslösung und verbesserte Tauschlösung: Liegeplatzplanung nach [GuCh2004]

Die Abbildung zeigt die typische Darstellungsform des Ergebnisses der Liegeplatzplanung in Form eines time-space-Diagramms. Auf der Abzisse ist die (geplante) Aufenthaltszeit („Wann?“) des Containerschiffs zu erkennen, die Ordinate zeigt die vom Schiff zu besetzenden (geplanten) Liegeplatzsektionen („Wo?“). Der Liegeplatzverantwortliche entscheidet sich, unter Berücksichtigung der vorgegebenen Zielfunktion, für die rechts dargestellte Zuordnung, ohne nachgelagerte Bereiche in die Planung einzubeziehen.

Aus dieser gegebenen Lösung lassen sich nun für den Yardplaner die Containerströme planen. Für dieses Beispiel existieren drei unterschiedliche Yardblöcke, an jedem Tag ist der Terminal mit Input und Outputströmen konfrontiert. Abbildung 3-2 zeigt als Ausschnitt für $t=0$ und $t=1$ die sich aus der fixierten Liegeplatzplanung ergebenden Container Ein- und Ausgänge. Die Bezeichner in der Abbildung richten sich nach dem Modell von [MLWL2005]. Es wird angenommen, dass alle in $t=t^*$ den Terminal erreichenden Importcontainer, diesen in $t=t^*+2$ landseitig wieder verlassen.

Für das gegebene Layout in diesem Beispiel zeigt sich eine ungünstige Situation. In der Liegeplatzplanung findet die Ausgangssituation der im Yard lagernden Container keine Berücksichtigung. Block 1 ist zu diesem Zeitpunkt mit 1430 lagernden Containern, im Gegensatz zu Block 2 (0 Container) und Block 3 (500 Container) bereits stark ausgelastet. Aus diesem Grund müssen die Container der in $t=0$ geplanten Schiffe, die an den Sektionen 2-6 liegen, relativ weite Wege zurücklegen, bevor sie gelagert werden können.¹⁰ Dieses Problem löst sich auch in den Nachfolgeperioden mit hoher Auslastung nicht auf: In $t=1$ muss ein Containerschiff der größten Kategorie auf den Sektionen 1-3 platziert werden, der nah gelegene Yardblock 2 ist jedoch nun mit den Containern aus der vorangegangenen Periode stark ausgelastet. Die Zuordnung von Containern zu einer Yardposition erfolgt zeitlich

⁹ Die Parameter der Untersuchung sind im Anhang gegeben.

¹⁰ vgl. Abbildung 3-1 sowie Distanzmatrix im Anhang.

sogar noch vor der (tatsächlichen) Zuweisung von Liegeplätzen, da zu exportierende Container den Terminal in der Regel vor Ankunft des Containerschiffs erreichen.¹¹ Die Möglichkeit, die Yarauslastung und -planung im Rahmen der Liegeplatzzuordnung zu berücksichtigen, ist somit theoretisch gegeben. Es wird schnell deutlich, dass sich bereits in diesem einfachen Beispiel ein komplexer Containerfluss ergibt.

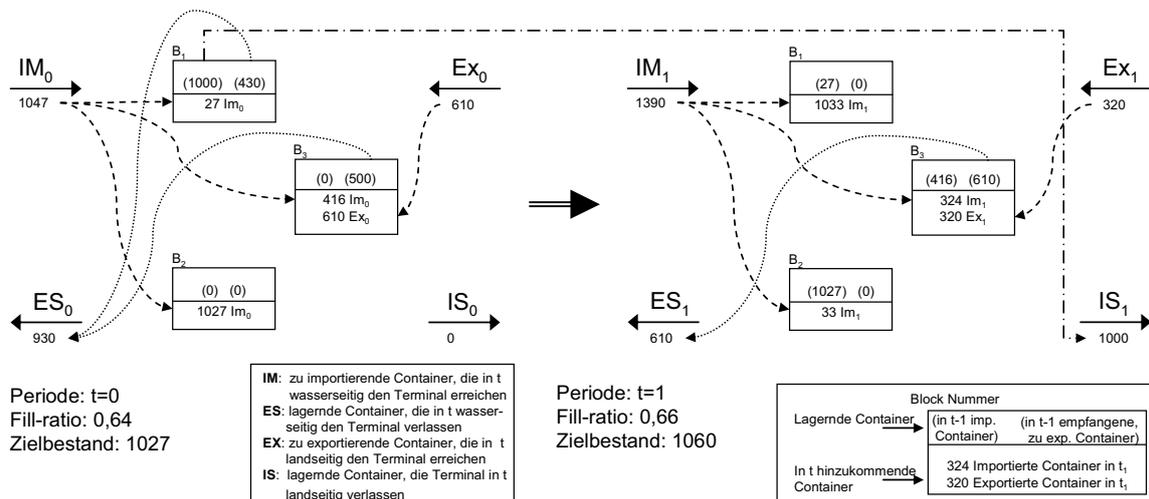


Abbildung 3-2: Beispiel Containerflussplanung für t=0 und t=1 nach [MLWL2005]

Die Problematik in diesem Beispiel zeigt auch, dass es nicht sinnvoll sein kann, lediglich nach einer Zielgröße zu optimieren. Die von [GuCh2004] vorgeschlagene Zielfunktion sollte lediglich im Zusammenhang mit anderen Zielen betrachtet werden, wobei es zu untersuchen gilt, in welcher Zusammensetzung und mit welcher Gewichtung dies erfolgen sollte.

Für die Liegeplatzplanung kommen neben der im Beispiel betrachteten Zielgröße u.a. folgende Ziele in Betracht:

- Minimierung der Summe der Transportdistanzen zwischen Kai und Yard
- Maximierung der Liegeplatzauslastung
- Minimierung der Wartezeit vor dem Hafen
- Maximierung der Anzahl „Berth on arrival“: „Null“-Wartezeiten vor dem Hafen
- Minimierung der Unzufriedenheit durch Zuordnungsreihenfolgen (nicht FIFS)

Für die Yardplanung kommen u.a. diese zusätzlichen Ziele in Betracht:

- Minimierung der Summe der Transportdistanzen zwischen Kai und Yard
- Maximierung der Yard-Auslastung
- Minimierung der Anzahl (bzw. der Wahrscheinlichkeit) ungewollter Containerüberlagerungen

¹¹ In asiatischen Terminals ist eine Schranke von 48h/24h vorgeschrieben. Europäische Terminals gewähren oftmals mehr Kundenflexibilität und nehmen auch noch Container an, die den Terminal nach Ankunft des Containerschiffes erreichen [StVS2004].

- Minimierung der Zugriffszeit
- Minimierung der Wartezeit von Transportern (z.B. VC, Lkw)

Weitere praxisrelevante Kriterien zur Beurteilung der Container Terminal-Performance nennt [Saan2004].

Im Rahmen der Transportplanung gilt es nun, die vorhandenen Van Carrier den Containerbrücken zieloptimal zuzuordnen. [Boes2000] ordnet jedem Containerjob k die von einem VC benötigte Transportzeit zu. Dieser Modellparameter lässt sich aus den vorangegangenen Lösungen bestimmen, da nun die Start- und Zielpositionen und somit die Distanz zwischen beiden Standorten bekannt ist. Mit gegebener Geschwindigkeit der VC lässt sich die Transportzeit für k berechnen und das Modell von [Boes2000] dadurch einsetzen. Auch für die Transportplanung kommen weitere Ziele in Betracht, beispielsweise die Minimierung von Leerfahrten.

Zusammenfassend lässt sich zu diesem Zeitpunkt folgendes festhalten. Neben den in diesem Beispiel eingesetzten Verfahren existiert ein relativ großer Pool an Lösungsansätzen für Teilbereiche, auf die zurückgegriffen werden kann und die sich austauschen lassen.¹² Ansätze, die die Koordination als auch die Prozessintegration behandeln, existieren hingegen kaum, obwohl die Bedeutung von [Pfoh2004] im Allgemeinen und u.a. [SVSt2004] für die Hafenlogistik unterstrichen wird. Für eine Untersuchung zur Verbesserung der Koordination integrierter Logistikprozesse bieten sich in gleich bleibender Umgebung folgende Stellschrauben an:

- Lösungskomponenten: Verfahren, mit denen das Teilproblem optimiert werden soll. Durch Austausch dieser Komponente lässt sich *ceteris paribus* die Qualität des Verfahrens im Prozesszusammenhang bewerten. Zum Einsatz kommen können, neben wissenschaftlichen Verfahren auch aus der Praxis abgeleitete Methoden oder einfache Entscheidungsregeln. Letztere werden zum Beispiel im Rahmen einer Simulation von [MeLa2005] eingesetzt, die die Koordination von ‚handshake operations‘ in einem Container Terminal behandelt.
- Zielsysteme: Durch Variation der Gewichtung des Zielsystems oder durch die Hinzunahme neuer Zielsetzungen werden gefundene Lösungen unterschiedlich bewertet. Es lassen sich so auch übergreifende Zielbeziehungen untersuchen.
- Organisationsstruktur: Im Rahmen der Gestaltung der Organisationsstruktur ist zu klären, wie die gemeinsame Lösungssuche auszusehen hat, z.B. welche Entscheidungsträger in einen Rückkopplungsprozess eingebunden werden sollen und auch wie ein Koordinationsprotokoll zu gestalten ist. Auch die Frage zur Problemlösung im Konfliktfall muss behandelt werden, um Konvergenz sicherzustellen. Im betrachteten Fall sequentieller Koordination ist der Liegeplatzverantwortliche dem Yardplaner weisungsbefugt, der Yardplaner wiederum dem Transportplaner. Es findet keine Rückkopplung statt, ein Konflikt kann in dieser Form nicht auftreten. Mit dem Konzept einer austauschbaren Organisationsstruktur kann das Modell realitätsnah abgebildet werden. Ergebnisse der Untersuchung sind verifizierbar und vereinfachen die spätere Übertragbarkeit.

Es stellt sich die Frage, wie sich der sequentielle Ansatz ersetzen lassen kann, wenn, wie bereits erwähnt, der Output einer Teillösung den (notwendigen) Input für

12 vgl. Tabelle 1.

die nachgelagerte Problemstellung darstellt. Aus den folgenden Gründen ist es möglich, diese Koordinationsform aufzugeben bzw. zu verändern:

- Die in der von [ZLWL+2003] angegebene hierarchische Struktur stellt nur dadurch einen Ansatz zur Lösung des Problems dar, dass Abhängigkeiten lediglich in eine Richtung betrachtet werden. Tatsächlich sind die Entscheidungsbereiche jedoch wechselseitig voneinander abhängig. Beispielsweise ist die Operationszeit des Schiffes stark von der Anzahl operierender Containerbrücken sowie von der Transportdistanz der VC zwischen Kai und Yard abhängig. Zur Lösung von vorgelagerten Problemstellungen müssen demnach zusätzliche Annahmen getroffen werden. Eine Rückkopplung mit *nachgelagerten* Entscheidungsbereichen ist jedoch sinnvoll.
- Die Planungen beginnen nicht, wie vereinfachend angenommen wird, immer wieder „von vorne“ im Zeitpunkt $t=0$, d.h. müssen beispielsweise aufgrund verspäteter Ankunft eines Containerschiffes Änderungen an der Zuordnung von Schiffen zu einem Liegeplatz vorgenommen werden, so beeinflusst diese Entscheidung die anderen bereits getroffenen und umgesetzten Entscheidungen. Container liegen bereits im Yard an den Positionen, die sich aus der ursprünglichen Planung ergaben. Anfallende *rescheduling*- Maßnahmen werden sich sequentiell nicht optimal koordinieren lassen, wie in dem Beispiel gezeigt werden konnte.
- Viele Eigenschaften und Beziehungen zwischen einzelnen Prozessen sind nicht so einfach abzuleiten, wie in dem einfachen Beispiel in Kapitel 3. Die Simulation stellt hier ein geeignetes Instrument dar, mit dem sich solche Beziehungen analysieren lassen.

Auf diese Aspekte soll in dem folgenden Kapitel näher eingegangen werden.

4 Softwareagenten zur Koordination integrierter Logistikprozesse

Bei der sequentiellen Wiederherstellung des Gesamtproblems können die Ergebnisse nachgelagerter Teilprobleme keinen Einfluss auf die vorangehenden Problemstellungen ausüben. Diese Nicht-Berücksichtigung relevanter Beziehungen kann, wie gezeigt wurde, zu ungünstigen Ergebnissen führen. Es ist deshalb zu überlegen, ob ein multiagentenbasierter Ansatz zum Einsatz kommen kann, und wie ein solcher Ansatz aufgebaut sein müsste.

Zur Klärung des Begriffs Softwareagent soll im Folgenden die Definition von [Wool2002] herangezogen werden. [Wool2002] definiert einen Agenten wie folgt: „An agent is a computer system that is *situated* in some *environment* and that is capable of *autonomous action* in this environment in order to meet its design objectives.“ Nach [Davi2005] stellen Agenten eine natürliche Erweiterung zu Objekten dar. Verglichen mit einem Objekt ist ein Agent u.a. in der Lage, in einer zielgerichteten Art zu handeln und mit anderen Agenten zu interagieren. Die multiagentenbasierte Simulation unterscheidet sich von der „herkömmlichen“ Simulation in der Hinsicht, dass (einige) beteiligte Entities als Agenten abgebildet sind [Hene2004].

Die Eigenschaften der Agenten lassen sich für die hier gegebene Problemstellung sehr gut einsetzen. Eine erste Übertragung in Anlehnung an die Definition von

[Wool2002] sieht wie folgt aus: Agenten sind als Entscheidungsträger in der Umgebung „Terminal“ zu autonomen Handlungen fähig und werden versuchen, ihr Zielsystem zu optimieren. Zudem sind sie flexibel, d.h. sie stehen in ständiger Interaktion mit ihrer Umwelt, sie weisen pro-aktives Verhalten auf und besitzen soziale Fähigkeiten zur Kommunikation, Koordination und Kooperation mit anderen Entscheidungsträgern. Im Rahmen der Organisationsstruktur und der austauschbaren Komponenten werden diese Fähigkeiten definiert.

Bevor weiter auf den Aufbau der Agenten eingegangen werden kann, sollen zunächst bestehende Ansätze der Literatur zusammengefasst werden. Tabelle 2 gibt einen Überblick bestehender agentenbasierter Ansätze zur Optimierung von Prozessen in der Hafenlogistik.

Autor	Entscheidungsbereich	Ansatz
Yi, Kim, Kim, 2002	organisationsübergreifend	Betrachtung der Supply Chain
Henese, 2004	übergreifend	marktbasierter, dezentraler Ansatz zur automatisierten Containerzuordnung
Rebollo et al., 2000	übergreifend	Verteilte Problemlösung zur automatisierten Containerzuordnung
Lokuge, Alahakoon, 2004	Liegeplatzplanung	BDI Agents using Neurol Network and Adaptive Neuro Fuzzy Inference zur dynamischen Planung
Thurston, Hu, 2002	Transportplanung	Vermeiden von Staus und Blockierungen, VC-Routing

Tabelle 2: Agentenbasierte Ansätze in der Hafenlogistik¹³

Vor allem in der Transportlogistik finden sich viele weitere Ansätze [Davi2005].

Im Gegensatz zu den in Tabelle 2 aufgeführten Ansätzen sollen die Agenten hier als Entscheidungsträger im Rahmen der Planung und Steuerung des Unternehmens zum Einsatz kommen. Durch Variation der Zielsysteme sowie Vorgabe unterschiedlicher Organisationsstrukturen können Erkenntnisse zur Verbesserung der Koordination gewonnen werden.

Die Umgebung „Terminal“ ist näher zu untersuchen, da sie für die Gestaltung der Agenten eine wichtige Rolle spielt. Sie ist

- Nicht-zugänglich: Der Agent besitzt keine vollständigen, genauen und zeitnahen Informationen über den Umgebungszustand und das Verhalten anderer Agenten sowie stochastische Einflüsse
- Nicht-episodisch: Das gegenwärtige Verhalten des Agenten beeinflusst auch den Erfolg in zukünftigen Perioden

¹³ Die Übersicht stellt lediglich einen Ausschnitt dar und erhebt keinen Anspruch auf Vollständigkeit.

- Nicht deterministisch: Es kann keine Garantie für die Wirkung einer bestimmten Handlung gegeben werden, da stochastische Einflüsse und die Handlungen anderer Agenten die Umgebung beeinflussen
- Dynamisch: In der Terminalumgebung arbeiten verschiedene Entscheidungsträger und verändern diese, ohne dass der einzelne Agent dies steuern könnte
- Kontinuierlich: Die Umgebung weist keine festgelegte, endliche Zahl an Aktionen und wahrnehmbaren Zuständen auf

Konkret kann die Umgebung¹⁴ über die folgenden grundlegenden Elemente definiert werden:

- Terminallayout, das zudem Transporter mit Eigenschaften, Distanzen und den Startzustand umfasst („Was ist vorhanden?“)
- im Simulationszeitraum geplante Containerschiffe, inklusive Lade- und Stauplan („Beeinflussung von außen?“)
- Stochastizität, Zufallsverteilung für den Ausfall von Entities, Verteilung von Ankünften, etc. („Verhalten im Inneren?“)

Mit Hilfe der in dieser Umgebung situierten Agenten lässt sich die Koordination gezielt, auch sukzessiv erweitern, von der ursprünglich in Abbildung 2-1 aufgezeigten Ordnung, hin zu einer realistischeren Abbildung der Vorgänge. Als Benchmark für die Qualität der Lösung kann jeweils das sequentielle Ergebnis dienen. Die dargestellten Planungssysteme werden von den Agenten repräsentiert.

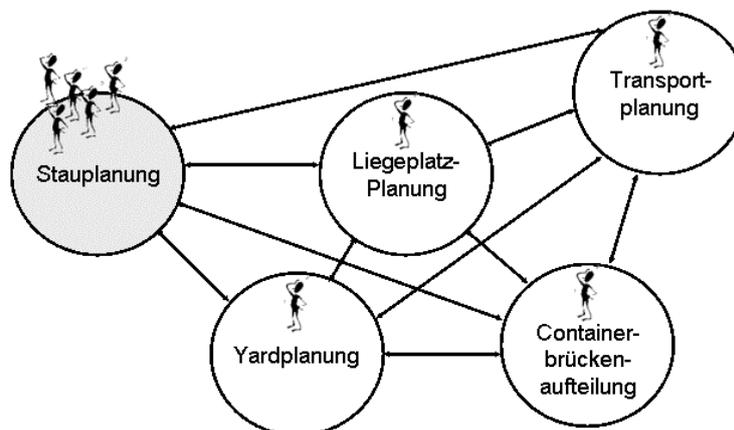


Abbildung 4-1: Agenten in der Terminal-Umgebung

Das Planungssystem „Stauplanung“ ist hervorgehoben, da es in der Regel vom Reeder vorgegeben wird und dessen Zielsetzungen bzgl. der weiteren Route des Containerschiffs vorgegeben ist. Ein Containerschiff kann sowohl geplant als auch ungeplant in die Umgebung „Terminal“ eintreten und jeweils unterschiedliche Eigenschaften aufweisen. Ungeplante Schiffe verursachen erhöhte Anforderungen an die Planung. Zudem verlangen Schiffe großer Reedereien aufgrund ihrer Marktmacht

14 vgl. Anhang für die Definition der Umwelt aus dem Beispiel.

häufig eine bevorzugte Behandlung und könnten so versuchen, Vorteile zu gewinnen. Diese Ansätze können in dem vorgestellten Modell von [GuCh2004] im Rahmen der Gewichtung der Containerschiffe berücksichtigt werden. Weitergehend können sie das System jedoch nicht manipulieren, da die endgültige Entscheidung über die Zuordnung von Ressourcen bei den terminalinternen Planungssystemen verbleibt.

Somit lassen sich Agenten durch die folgenden Elemente charakterisieren. Agenten repräsentieren die einzelnen Planungssysteme und sind mit einer austauschbaren Problemlösungskomponente ausgestattet, z.B. kann eine Fähigkeit des Liegeplatzplaners die Heuristik von [GuCh2004] sein. Sie verfolgen ein bestimmtes variierbares Zielsystem, über das sie in der Lage sind, den Nutzen ihrer Aktionen zu beurteilen und verfügen über eine Definition der Verantwortlichkeiten und Entscheidungsbefugnisse im Konfliktfall, die sich aus der Organisationsstruktur ableiten lassen. [Laßm1992] definiert Koordination als „die gesamtzielkonforme Abstimmung interdependenter Teilaufgaben durch die beteiligten Aufgabenträger“. Über die Organisationsstruktur wird das Koordinationsprotokoll definiert. Diese Struktur ist auch für die Sicherstellung von Konvergenz verantwortlich.

Nachdem ein gemeinsamer Plan verabschiedet wurde, der sich gemäß der Definition der Organisationsstruktur ergibt und den die Agenten gemeinsam zu akzeptieren bereit sind, legen sie ihr Handeln auf die Umsetzung dieser Pläne aus. Die im Plan verankerte Lösung enthält die Soll-Werte der einzelnen Teilprobleme. Im Rahmen der Durchsetzung nehmen die Agenten mit Sensoren die Umwelt wahr und können Soll- und Istwerte miteinander vergleichen. Beispielsweise übergibt der Liegeplatzplaner dem Yardplaner seine Vorschläge zur Zuordnung der Containerschiffe und teilt sich ergebende Änderungen während der Durchsetzung sofort mit. Je nach Koordinationsprotokoll muss er auf die Bestätigung anderer Planungssysteme warten und gegebenenfalls weitere Umstellungen an dem Plan vornehmen.

Kommt es beispielsweise aufgrund Verspätung eines Containerschiffs zu einer Abweichung in der Liegeplatzplanung, so könnte durch ein *rescheduling* das Zielsystem des Liegeplatzplaners verbessert werden. Zugleich hätten diese Maßnahmen jedoch Auswirkungen auf die anderen Entscheidungsbereiche. Änderungen am Plan sind mit Kosten verbunden, zum Beispiel, wenn dadurch Container im Yard umzustapeln sind. Die Änderung sollte also nur dann erfolgen, wenn sich die Agenten auf eine neue gemeinsame Lösung einigen können, bzw. die neue Lösung durchgesetzt werden kann. Über die Veränderung der Zielgrößen lässt sich der Nutzen der Maßnahmen abschätzen. Es lassen sich verschiedene Methoden zur Koordination implementieren und untersuchen. Dabei ist im Rahmen einer Simulation zunächst weniger die Zeit, in der die Problemlösung erfolgt relevant, als die durch das Zusammenspiel gewonnenen Erkenntnisse für eine verbesserte Koordination.

Grundsätzlich werden die Agenten als kooperativ angenommen, da sie gemeinsam die Minimierung der Aufenthaltszeit der Containerschiffe im Hafen, die Maximierung der Containerbrückenproduktivität und/oder eine Kostenminimierung, d.h. Verzicht auf den Einsatz unnötiger Ressourcen, in der Zielsetzung verfolgen. Unkooperative Schiffe und starke Marktmacht der Reeder können den Planungsprozess lediglich erschweren, nicht aber unterwandern.

Trotz dieser gemeinsamen Ziele besitzt jeder Entscheidungsträger aber weitere Zielsetzungen, die zu Konflikten mit anderen Bereichen führen können. Beispiels-

weise könnte die Operationsgeschwindigkeit eines Schiffes im Bedarfsfall gesteigert werden, wenn kurzfristig mehr Yard-Lagerkapazitäten in der Nähe des Schiffes eingeräumt werden könnten, z.B. indem die nach [MLWL2005] für einen Block geplante Menge überschritten wird. Für den Yardplaner erhöht diese Maßnahme jedoch die Wahrscheinlichkeit ungewollter Containerüberlagerungen und verursacht eine erhöhte Zugriffszeit auf einen bestimmten Container und zusätzliche Kosten.

Das Zusammenspiel der Entscheidungsträger bildet den Planungs- und Steuerungsprozess in einem Container Terminal ab. Ausgestattet mit den hier angedeuteten Fähigkeiten werden in diesem Modell relevante Zusammenhänge im Rahmen des Prozesses betrachtet und können so umfassend analysiert werden.

Im Rahmen dieses MAS lässt sich somit u.a. untersuchen:

- Die Qualität eines Lösungsverfahrens im Prozesszusammenhang, über den Vergleich mit anderen Verfahren.
- Die Beziehungen zwischen Zielsystemen in Teilproblemen als wichtige Erkenntnis für deren Gestaltung.
- Unterschiedliche Organisationsstrukturen und damit verbunden unterschiedliche Koordinationsmuster. Es kann untersucht werden, wie sich Machtverhältnisse auswirken, d.h. wer beispielsweise in letzter Instanz eine Lösung durchsetzen können sollte.
- Die Kosten einer Komplexitätsreduzierung in Form einer fixierten Einstellung in Entscheidungsbereichen. Beispielsweise reduziert die vertragliche Zusicherung von q Containerbrücken für die Operationen an einem bestimmten Schiff das Planungsproblem, der Terminal verliert aber auch Flexibilität hinsichtlich eigener Planungsfreiheit zur Prozessgestaltung. In diesem Fall ist es sinnvoll abzuschätzen, was diese vertraglich zugesicherte Leistung „wert“ ist, d.h. wie sie sich auswirken kann.

Es zeigt sich, dass Multiagentensysteme in diesem Rahmen viel versprechend erscheinen, neue Verfahren zur Koordination integrierter Logistikprozesse zu untersuchen.

5 Ausblick

Das in diesem Beitrag skizzierte Modell dient der Analyse von wissenschaftlichen, aber auch aus der Praxis abgeleiteten Ansätzen im Prozesszusammenhang und zur Untersuchung von Verfahren zur verbesserten Koordination integrierter Logistikprozesse in Container Terminals. Es konnte gezeigt werden, dass eine einfache sequentielle Koordination zu unerwünschten Ergebnissen führen kann, und es wurde die Bedeutung einer Prozessoptimierung mit Hilfe koordinierender Maßnahmen verdeutlicht.

Nachdem unterschiedliche Lösungsansätze implementiert und isoliert getestet werden konnten, sollen diese Verfahren zukünftig in ein agentenbasierten Modell eingegliedert werden. Mit Hilfe dieser Lösung ist es möglich, auf Basis existierender Modelle und Lösungsverfahren, (neue) Verfahren zur Koordination zu untersuchen und zu beurteilen. In so fern bietet der Einsatz multiagentenbasierter Simulation für

die dargestellte Problemstellung einen innovativen Ansatz zur Gestaltung, Steuerung und Kontrolle von Geschäftsprozessen.

Literaturverzeichnis

- [BMRu2004] Biethahn, J.; Mucksch, H., Ruf, W.: Ganzheitliches Informationsmanagement, 6. Auflage. Oldenbourg: München, 2004.
- [Boes2000] Boese, J. et al.: Vehicle Dispatching at Seaport Container Terminals Using Evolutionary Algorithms. Proceedings of the 33rd Hawaii International Conference on System Sciences, 2000.
- [Davi2005] Davidsson, P. et al.: Agent-Based Approaches to Transport Logistics. Transportation Research Part C. In: Emerging Technologies, 4 (2005) 13, S. 255-271.
- [GRZa1998] Gambardella, L.M.; Rizzoli, A.E.; Zaffalon, M.: Simulation and Planning of an Intermodal Container Terminal. In: Special Issue of Simulation Journal in Harbour & Marine Simulation 2 (1998) 21.
- [GuCh2004] Guan, Y., Cheung, R.K.: The berth allocation problem: models and solution methods. In: OR Spectrum, (2004) 26, S. 75-92.
- [Hene2004] Henesey, L.E.: Enhancing Terminal Performance: A Multi Agent Systems Approach. Blekinge Institute of Technology: Karlshamn, 2004.
- [Laßm1992] Laßmann, A.: Organisatorische Koordination. Konzepte und Prinzipien zur Einordnung von Teilaufgaben. Dissertation, Univ. Köln. Gabler: Wiesbaden, 1992.
- [LWWa2005] Liu, J.; Wan, Y.; Wang, L.: Quay Crane scheduling at Container Terminals To Minimize the Maximum Relative Tardiness Of Vessel Departures. Naval Research Logistics, (2005) 53, S. 60-74.
- [MeLa2005] Meier, L., Lackner, A.: Simulation und Optimierung von 'handshake operations' eines Container Terminals. 8. Göttinger Symposium Soft Computing, In: Biethahn, J.; Lackner, A.; Nissen, V. (Hrsg.): Information-Mining und Wissensmanagement in Wissenschaft und Wirtschaft, 2005.
- [MLWL2005] Murty, K.G.; Liu, J.; Wan, Y.W., Linn, R.: A decision support system for operations in a container terminal. Decision Support Systems, 39: 309-332, 2005.
- [Pfoh2004] Pfohl, H.C.: Logistikmanagement. Springer: Berlin, 2004.
- [Saan2004] Saanen, Y.A.: An approach for designing robotized marine terminals. PhD thesis. Technische Universität Delft, 2004.
- [Stee2000] Steenken, D.: NETRALOG- Machbarkeitsprojekt für die Entwicklung und den Einsatz bioanaloger Algorithmen zur Steuerung des Horizontaltransports bei der Schiffsabfertigung auf dem Container Terminal Burchardkai. BMBF 01 IB 803 0, HHLA, TU Braunschweig, 2000.
- [SVSt2004] Steenken, D., Voß, St., Stahlbock, R.: Container terminal operation and operations research – a classification and literature review. OR Spectrum, 26 (2004), S. 3-49.
- [Wool2002] Wooldridge, M.J.: An introduction to Multiagent Systems. British Library: London, 2002.

[ZLWM+2003] Zhang, C.; Liu, J.; Wan, Y.; Murty, K.; Linn, R. Storage space allocation in container terminals. *Transportation Research Part B*, 37:883–903, 2003.

6 Anhang

Parameter der Untersuchung

- Containerschiffe in dem Modell (geplant im Simulationszeitraum von 7 Tagen):

5 x Containerschiff-Kategorie I:	3 x Containerschiff-Kategorie II:	4 x Containerschiff-Kategorie III:
- Länge: 300 m (3 Sektionen)	200 m (2 Sektionen)	100 m (1 Sektion)
- .Aufenthalt: 6*4h = 1 Tag	4*4h	3*4h
- Gewicht = 1	Gewicht = 1	Gewicht = 1
- Ankünfte = { 1,2,3,3,4 }	Ankünfte = { 1,1,3 }	Ankünfte = { 2,4,7,7 }
- Container = 1440 pro Sektion = 480	640 320	240 240
Import, Export = 330, 150	160, 160	240, 0

- Terminallayout

- Anzahl Containerbrücken: 6, $v = 120$ Sekunden pro Move
- Anzahl Van Carrier: 18, $v = 8$ m/s
- Layout des Terminals: 3 Yard-Blöcke mit Kapazität 1600 TEU (hier: 1 Container = 1 TEU)
- Belegung $t=0$:
 - Block 1 – 1000 TEU Import Container + 430 TEU Export Container
 - Block 2 – 0
 - Block 3 – 500 TEU Export Container
- Exportcontainer erreichen den Yard 1 Tag vor Ankunft des Containerschiffs
- Importcontainer werden 2 Tage in dem Yard lagern, bevor sie landseitig den Terminal verlassen

Distanzmatrix

	QC1	QC2	QC3	QC4	QC5	QC6	Yrd1	Yrd2	Yrd3
QC1	0	100	200	300	400	500	450	850	950
QC2	100	0	100	200	300	400	350	750	850
QC3	200	100	0	100	200	300	250	650	750
QC4	300	200	100	0	100	200	150	550	650
QC5	400	300	200	100	0	100	150	550	550
QC6	500	400	300	200	100	0	250	650	450
Yrd1	450	350	250	150	150	250	0	400	500
Yrd2	850	750	650	550	550	650	400	0	500
Yrd3	950	850	750	650	550	450	500	500	0

- Stochastizität wurde in dem Beispiel nicht berücksichtigt

Integration von Software-Agenten und Soft-Computing-Methoden für die Transportplanung

Hagen Langer¹, Ingo J. Timm¹, Jörn Schönberger², Herbert Kopfer²

¹Technologie-Zentrum Informatik (TZI), Universität Bremen
Am Fallturm 1, 28359 Bremen
hlanger@tzi.de, i.timm@tzi.de

²Lehrstuhl für Logistik, Universität Bremen
Wilhelm-Herbst-Straße 5, 28359 Bremen
sberger@logistik.uni-bremen.de, kopfer@uni-bremen.de

Zusammenfassung: Anwendungen in der Transportlogistik zeichnen sich durch eine besonders hohe Komplexität und Dynamik aus. In diesem Rahmen haben sich Soft-Computing-Ansätze für die Disposition etabliert, da sie eine effiziente Planung ermöglichen. Daneben werden in der Forschung Ansätze zur Transportplanung mit Hilfe von Multiagentensystemen untersucht, die eine natürliche Repräsentation bieten und hierbei autonomes und robustes Verhalten ermöglichen. Die Integration von Multiagentensystemen und Soft-Computing-Methoden wird in diesem Beitrag als Lösungsansatz beschrieben, um robuste und flexibel reagierende selbststeuernde logistische Prozesse in dieser Domäne zu realisieren. Es wird zunächst beschrieben, wie eine kompatible Wissensdarstellung erreicht werden kann. Wesentlicher Bestandteil hybrider Planungsansätze ist eine Koordination von zentralen und dezentralen Planungskompetenzen zur Erreichung übergeordneter und individueller Ziele. Dabei ist das Rollen-basierten Wissensmanagement von besonderer Bedeutung.

1 Einleitung

Effiziente und zuverlässige Transportprozesse sind für moderne verteilte und arbeitsteilige Produktionskonzepte von zentraler Bedeutung. Daher sind in den letzten Dekaden weit reichende überregionale und internationale Transportnetzwerke entstanden, die eine räumliche Überbrückung zwischen Beschaffungs-, Produktions- und Nachfrageorten zu marktgerechten Preisen ermöglichen.

Nur durch eine Nutzung hoch entwickelter Planungs-Unterstützungssoftware ist es noch möglich, aus der Vielzahl möglicher Prozessausprägungen die jeweils bestmögliche für jede Transportressource bzw. jeden Auftrag zu identifizieren, ohne eines der beiden Ziele Profit- oder Servicemaximierung übermäßig zu vernachlässigen [CrLa1998]. In der wissenschaftlichen Literatur sind zwei Klassen von

Planungsmethoden zu finden, die in aktuellen Anwendungen zur Planung von Transportprozessen eingesetzt werden: Soft-Computing und Software-Agentensysteme.

Unter dem Begriff *Soft-Computing* [Lipp2005] sind Methoden zusammengefasst, die im Gegensatz zu vielen klassischen Ansätzen aus der Künstlichen Intelligenz (KI) keine Repräsentationen auf der Basis einer klassischen zweiwertigen Logik voraussetzen, sondern auch und gerade „weiche“, d.h. unscharfe, mehrwertige oder approximative Repräsentationen einsetzen, um für komplexe Entscheidungsprobleme hochwertige Lösungen zu finden. Die wichtigsten Verfahren sind dabei Evolutionäre Verfahren, Lokale Suche, Fuzzy-Systeme sowie Neuronale Netze, die zentralisiert global umzusetzende Entscheidungen treffen.

Software-Agentensysteme basieren hingegen auf dem Paradigma der dezentralen Entscheidungsfindung. Agenten repräsentieren Einheiten, die autonom entscheiden können und hierbei auf der Basis von Interaktion und Kommunikation mit ihrer Umwelt (insbesondere anderen Agenten) die Handlungsfähigkeit der logistischen Objekte herstellen.

In dieser Arbeit soll überlegt werden, wie diese bisher weitgehend getrennt verfolgten methodische Planungsparadigmen mit dem Ziel der Verbesserung und Ausweitung von Transportplanungskompetenzen in hybriden Ansätzen integriert werden können, um eine Dispositionsunterstützung für zukünftige Transportplanungs-Szenarien zur Verfügung zu stellen.

Im Einzelnen werden die folgenden Fragestellungen beantwortet:

- Vor welchen Herausforderungen steht das Transportprozess-Management und warum können diese Herausforderungen weder durch vollständig zentralisierte noch durch rein dezentrale Planungsunterstützungs-Ansätze bewältigt werden? (Abschnitt 2)
- Welche konzeptionellen Ansätze verfolgen Methoden des Soft-Computing sowie Software-Agenten zur Lösung von Entscheidungsproblemen? (Abschnitt 3)
- Welche Ansatzpunkte stehen für eine Kombination zentraler sowie dezentraler Planungsansätze zur Verfügung? (Abschnitt 4)

2 Grenzen zentraler Transportprozess-Planungsansätze

Technische Innovationen, Management-Trends sowie neue Marktbedürfnisse wirken verändernd auf alle heutigen Wertschöpfungssysteme ein. Somit sind auch Transportsysteme als Wertschöpfungssysteme diesen Einflüssen unterworfen. In diesem Abschnitt werden zunächst Auswirkungen technischer sowie organisatorischer Innovationen auf die Planung von Transportsystemen analysiert (2.1). Anschließend werden Grenzen für die Anwendbarkeit zentraler (2.2) sowie dezentraler Planungsansätze zur Steuerung von Transportsystemen aufgezeigt (2.3).

2.1 Aktuelle Herausforderungen an die Transportplanung

Dynamisierung der Prozessplanung. Die Berücksichtigung aktuellster Planungsdaten bei der Bestimmung der durchzuführenden Transportaktivitäten stellt einen wesentlichen Wettbewerbsvorteil dar. Deshalb müssen alle relevanten Einfluss-

größen, die auf die ablaufenden Prozesse einwirken, zeitnah bei der Prozessfestlegung berücksichtigt werden. Basierend auf möglichst vollständiger Information über die verfügbaren Ressourcen und Aufträge erfolgt eine aktualisierende Prozessplanung mit dem Ziel, die anstehenden Aufgaben (Transportaufträge) unter den gegebenen Rahmenbedingungen mit maximaler Effizienz zu erfüllen. Die festgelegten Prozesse werden anschließend im Transportnetzwerk zur Ausführung freigegeben. Bei der Planung der Prozesse ist einerseits eine Robustheit gegenüber Störungen anzustreben. Andererseits ist jedoch auch sicherzustellen, dass die Prozesse flexibel an veränderte Situationen angepasst werden können.

Notwendigkeit zur Verbesserung des Informationsflusses. Moderne Datenerfassungs-, Verarbeitungs- sowie Kommunikationstechnologien ermöglichen die Bereitstellung aktuellster Informationen, oftmals in Echtzeit über bestehende Kommunikationssysteme. Als Konsequenz muss eine Prozessanpassung umgehend erfolgen und die Informationen über die neuen Prozessschritte müssen zielgerichtet in das Transportsystem propagiert werden [HSEH2005]. Die bestehenden Kommunikationsnetzwerke ermöglichen häufig jedoch nur spezielle Dienste (z.B. eine Positionsbestimmung). Der Ausbau bestehender Kommunikationssysteme ist jedoch mit hohen Investitionen verbunden, die sich nur über einen längeren Zeitraum amortisieren.

Steigerung der Komplexität der Planungsaufgaben. Um die Wünsche nach immer schnelleren und zuverlässigeren Transportdienstleistungen im globalen Kontext erfüllen zu können, werden existierende Transportnetze sukzessive durch Erweiterungen und Verknüpfungen ausgedehnt. Typischerweise sind solche Transportsysteme als Hub-and-Spoke-Netze organisiert. Vom Abholort bis zum Zielort werden einzelne Transportaufträge mehrmals umgeschlagen und mit anderen Aufträgen zusammengefasst. Ein einzelner Auftrag erfordert dabei die Realisierung einer Transportkette aus aufeinander folgenden Non-Stop-Transporten sowie Umschlagsvorgängen und Zwischenlagerungen. Die Berücksichtigung der Verfügbarkeiten von Umschlagsressourcen sowie die Notwendigkeit, die Ankunfts- und Abfahrtzeiten in den Hub-to-Hub-Transporten zu synchronisieren, steigert einerseits die Komplexität der Planungsaufgabe und andererseits das Risiko, dass die Transportkette während ihrer Ausführung gestört wird [ChMu1999].

Dezentrale Entscheidungskompetenz. Aufgrund der überregionalen Ausdehnung der Transportnetze werden regional tätige Dienstleister eingesetzt, die spezifische Aufgaben in der Transportkette übernehmen. Diese Subunternehmen werden oftmals kurzfristig mit der Übernahme von Abschnitten der Transportketten für eine kurze Zeit betraut. Die Verantwortung für den übertragenen Transportabschnitt liegt beim beauftragten Subunternehmer, der für sich autark eigene Prozesse plant. Der Auftraggeber hat keine Möglichkeit, die Subunternehmer-Prozesse festzulegen oder zu beeinflussen. Eine Integration in die Kommunikations- und Informationsnetze ist daher oftmals nicht möglich, da diese mit unverhältnismäßig hohen Investitionskosten verbunden wäre. Somit ist eine zentrale Informationsbasis nicht realisierbar. Daher können fremdvergebene Abschnitte einer Transportkette nicht in die zentrale Planung integriert werden [KrKo2006].

2.2 Grenzen zentraler Planungsansätze

Die Anwendung automatisierter zentralisierter Verfahren zur Planung von Betriebsprozessen im Allgemeinen und von Transportprozessen im Speziellen bedingt zunächst eine Formalisierung des zu bewältigenden Anwendungsproblems. Dafür wird dieses Problem in ein vereinfachtes und abstraktes formales Entscheidungsmodell abgebildet, das das globale Realwelt-Problem repräsentiert. Dieses Modell wird einem Lösungsalgorithmus zugeführt, der nach vorbestimmten Logiken eine Prozess-Alternative ermittelt, die im motivierenden Anwendungsproblem umzusetzen ist. Auf diesem Paradigma der Modell-basierten Planung fundieren fast alle Entscheidungsmethoden des Soft-Computing insbesondere Meta-Heuristiken (Genetische Suche [BeBa2004], Local Search [DeKa2006], Neuronale Netze [1995], Ameisenalgorithmen [Lack2004], ...), die die wesentlichen Verfahren zur Entscheidungsunterstützung in der Transportprozess-Planung darstellen.

Die oben skizzierten Anforderungen an die Planung zukünftiger Transportsysteme korrumpieren sowohl die Datensammlung in ihrer bisherigen Form als auch die Aufstellung des Modells nach bisher verfolgten Paradigmen sowie die Modell-Lösung.

Die Akquisition und Aufbereitung der benötigten Planungsdaten ist im Gegensatz zu statischen Situationen bei sich verändernden Problemsituationen durch mindestens drei Aspekte erschwert: Die verfolgten Planungsziele können sich ändern, so dass a priori unklar ist, welche Daten zur Messung des Zielerreichungsgrades notwendig sind (dynamisches Zielsystem). Es werden zusätzliche Daten benötigt, die die Geschwindigkeit und Richtung der Änderung der Problemstellung hinreichend repräsentieren. Redundante Daten sind zu eliminieren und ggf. fehlende Daten müssen in Echtzeit beschafft werden.

Die Definition des benötigten Entscheidungsmodells ist mit heutigen Methoden nur unzureichend möglich. Einerseits fehlen flexible Lösungsrepräsentationen, andererseits ist unklar, wie kurzfristige und längerfristige Aspekte in einem Modell für ein dynamisches Entscheidungsproblem integriert werden können. Schließlich sind heutzutage keine Methoden bekannt, um autonome Subsysteme in einem globalen Entscheidungsmodell für ein globales System adäquat zu berücksichtigen.

Die Modell-Lösung wird einerseits durch die Notwendigkeit zur Anpassung des Lösungsverfahrens an die geänderte Problemstellung erschwert. Andererseits ist die für die Lösungssuche zur Verfügung stehende Zeit beschränkt, so dass häufig nur die Wiederherstellung der Prozessdurchführbarkeit erzielt werden kann, nicht jedoch eine Prozessoptimierung stattfindet. Die Verwendung von einfachen Update-Regeln wird jedoch der Komplexität der Planungsaufgabe nicht gerecht.

2.3 Grenzen eines verteilten Transportprozess-Managements

Die Künstliche Intelligenz (KI) untersucht Systeme, deren Eigenschaften analog zu den Eigenschaften sozialer Systeme, wie Intelligenzleistung, beschrieben werden können. In diesem Rahmen hat sich die Verteilte Künstliche Intelligenz (VKI) als Teilgebiet der KI insbesondere für dezentrale Planungsaufgaben etabliert [Müll1993]. Die Forschung der VKI bemüht sich, Systeme der KI in Hinsicht auf Skalierbarkeit, multiple Problemlösungsstrategien und Wiederverwendbarkeit durch autonome und

kooperative Systeme zu erweitern [Find1991]. Für die Logistik ist diese Forschungsrichtung insofern interessant, als dass die Gestaltung monolithischer Systeme für die Lösung komplexer Planungs- und Dispositionsaufgaben zunehmend unangemessener wäre und auch eine reine Modularisierung komplexer Systeme in lose gekoppelte Subsysteme keine Vorteile bieten kann.

In der Transportlogistik steht das Potential der VKI im Vordergrund, Ressourcen (Transportmittel) und Aufträge mit intelligenten Stellvertretern zu repräsentieren. Die explizite Berücksichtigung und Ausbeutung dezentraler Entscheidungssysteme wird dabei angestrebt. Ziel ist es, ein System auf der Basis lokal autonom entscheidender Einheiten (Mikro-Dynamik), die miteinander kooperieren, global zu optimieren (Makro-Dynamik). Diese Fähigkeit wird häufig als Emergenz beschrieben.

Die Anfänge der VKI waren durch Probleme gekennzeichnet, die im Wesentlichen dem verteilten kooperativen Problemlösen zuzuordnen waren. Hierbei waren die Agenten benevolent, verhielten sich rational und gingen nach Möglichkeit immer Kooperationen ein, auch wenn diese für den Agenten nicht vorteilhaft sind. In den 90er Jahren hat sich eine Verschiebung des Fokus auf die Koordination kompetitiver Akteure eingestellt.

Das Problem der Transportlogistik unterscheidet sich von der Annahme kompetitiver Umgebungen nochmals, da hier nicht nur Mikrostrukturen (Interaktionen zwischen Agenten) berücksichtigt werden müssen, sondern auch Mesostrukturen, nämlich Unternehmen, die aus unterschiedlichen Akteuren bestehen. Hierbei sind Akteure innerhalb einer Unternehmung kooperativ eingestellt und stehen im Wettbewerb zu Akteuren anderer Unternehmen.

In Systemen der kooperativen Problemlösung stand die Optimierung des Gesamtsystems im Vordergrund, wohingegen in den wettbewerblichen Systemen die individuelle Optimierung der Agenten fokussiert behandelt wird. Die Anforderungen der Transportlogistik benötigen sowohl eine individuelle Optimierung auf Akteursebene als auch eine globale Optimierung auf der Mesoebene. In diesem Bereich sind die Ansätze der VKI noch nicht hinreichend untersucht bzw. entwickelt worden.

3 Vergleich der konzeptionellen Grundlagen

Im folgenden werden die konzeptionellen Grundlagen von Entscheidungsansätzen des Soft-Computing (3.1) und agentenbasierter Entscheidung (3.2) dargestellt und miteinander verglichen.

3.1 Paradigmen des Soft-Computing

Heutige Transportsysteme sind häufig das Resultat von Outsourcing- und Fusionsvorgängen. Im Zuge derer wurde der Werkverkehr einzelner Firmen aus dem Firmenverbund herausgelöst und in eigenständige, profit-orientierte Dienstleistungsfirmen umgewandelt.

Entscheidungsstrukturen wurden zur Planung der gewerblichen Transportdienstleistungen unverändert übernommen. Da der Werksverkehr lediglich ein Ausführungsinstrument darstellte und der Fuhrpark einer zentralen Leitung unterstellt war, wurde typischerweise streng hierarchisch geplant.

Für die typischen Planungsszenarien wie dem Problem des Handlungsreisenden, dem Standardproblem der Tourenplanung oder aber auch deren Verallgemeinerungen wurden insbesondere im Operations Research aber auch in der Künstlichen Intelligenz Planungsmethoden entwickelt, die über lange Jahre zufrieden stellende Planungsergebnisse lieferten. Alle erprobten Entscheidungslogiken basieren jedoch auf den gleichen Paradigmen:

Sie sind *global systembezogen*, d.h., sie treffen alle Entscheidungen unter Berücksichtigung der Wirkungen auf das gesamte Transportsystem. Die dafür erforderlichen Informationen über die relevanten Eigenschaften des Transportsystems stehen der planenden Instanz vollständig und fehlerfrei zur Verfügung.

Sie setzen ein *statisches* Transportnetzwerk voraus: Während des Planungsprozesses und ebenso bei der anschließenden Ausführung des Plans ändern sich die relevanten Eigenschaften des Transportnetzwerks nicht mehr unvorhergesehen.

Sie sind *nicht-kommunikativ*, d.h., es findet kein weiterer Informationsaustausch statt, sobald die zu lösende Planungsaufgabe spezifiziert ist und die Planungsmethode gestartet ist.

Sie sind *koordinierend*, d.h., die Aktivitäten der beteiligten logistischen Objekte (Fahrzeuge, Pakete,...) werden aufeinander abgestimmt, ohne dass ein Feedback der Objekte eingeholt wird, das zu einer Entscheidungsrevision führen könnte.

Sie nutzen eine *eigene spezifische interne Problemrepräsentation*, die nicht mit anderen Objekten bzw. Methoden geteilt wird oder notwendigerweise zu deren Repräsentation kompatibel ist.

Sie kommen in *hierarchisch organisierten Entscheidungssystemen* zum Einsatz, in denen eine zentrale Entscheidungskompetenz vorgesehen ist.

Sie benötigen eine externe Anweisung zur Initiierung einer Planung oder einer Planüberarbeitung.

Aufgrund der globalen Sichtweise wird durch die Soft-Computing-Methoden die Approximation eines globalen Optimums für die auszuwählende Prozessentscheidung angestrebt.

3.2 Konzeption der Software-Agenten-Methodik

Software-Agenten repräsentieren ausgewählte Objekte, die für die betrachtete Entscheidungssituation relevant sind. Jeder Agent stellt eine autonome Entscheidungsentität dar und versucht durch Kommunikation und Verhandlungen mit den übrigen Software-Agenten dezentral eine für alle beteiligten Agenten akzeptable Problemlösung zu finden. Hierbei ist die Autonomie eine zentrale Eigenschaft, die Softwareagenten zugeschrieben wird, d.h., die Agenten agieren selbständig ohne direkte Intervention von Menschen und anderen Entitäten und haben zumindest eine partielle Kontrolle über ihr Verhalten und ihren Zustand [WoJe1995, Ferb1999, Tecu1998]. Agenten werden insbesondere in solchen Anwendungen eingesetzt, in denen die Komplexität der Aufgabe und die Dynamik der Umgebung einer vorprogrammierten Verhaltensweise nicht adäquat sind. Daher werden Agenten so gestaltet, dass sie eine gewisse Autonomie realisieren, d.h., dass der Auftraggeber dem Agenten nicht einen Plan mit auf den Weg gibt, sondern dass der Agent über eigenständige Planungsalgorithmen verfügen muss (vgl. [KaJe1997]). Im Verlauf

seiner Arbeit wird der Agent also nicht direkt von außen gesteuert, sondern erhält durch die eigene Planung und seine Unabhängigkeit im Handeln einen gewissen Grad an Autonomie [Weiß1999]. Russel und Norvig beschreiben Autonomie wie folgt: "To the extent that an agent relies on the prior knowledge of its designer rather than on its own percepts, we say that the agent lacks autonomy." [RuNo2003, S. 37]

Realisierungen von Software-Agenten-Systemen in der Transportplanung basieren insbesondere auf einer Abbildung der einzelnen Transportressourcen bzw. –Aufträge auf Agenten. Durch eine Abstimmung der Agenten untereinander werden umsetzbare Transportprozesse festgelegt.

Die konzeptionelle Basis von Software-Agenten unterscheidet sich aufgrund des dezentralen Ansatzes wesentlich von den Paradigmen des Soft-Computing:

Agenten entscheiden *autonom*, d.h., sie treffen Entscheidungen auf individueller Datenbasis, ohne notwendigerweise auch die Wirkungen auf das Gesamtsystem zu betrachten.

Sie sind *kommunikativ*, d.h., sie erhalten und/oder geben Informationen von/an anderen Agenten, um für sich ein individuelles Entscheidungsmodell zu erhalten.

Sie sind (zumeist) *kooperierend*, d.h., es werden Entscheidungen angestrebt, die nicht nur individuellen Zielen, sondern den Zielen mehrerer Agenten gerecht werden.

Sie sind *interoperabel* und können Wissen weitergeben bzw. empfangen.

Sie kommen in *heterarchisch organisierten Systemen* zum Einsatz, bei denen eine zentrale Entscheidungskompetenz nicht möglich oder nicht gewollt ist.

Sie sind *proaktiv* und können aus eigenem Antrieb Entscheidungen treffen.

Eine Approximation eines globalen und durch alle Objekte verfolgten monolithischen Ziels wird nicht notwendigerweise angestrebt. Vielmehr steht die Verfügbarkeit eines reaktiven Systems, das sich an neue Situationen anpassen kann, im Vordergrund. Dabei können in einem einzigen Multiagentensystem (MAS) sowohl kooperierende als auch konkurrierende Agenten kombiniert werden. Aus diesem Grunde eignen sich Multiagentensysteme auch für die Simulation von komplexeren Szenarien, in denen nicht nur *eine* Organisation (z.B. ein Logistikdienstleister), sondern *mehrere* Organisationen mit unterschiedlichen Kapazitäten, Fähigkeiten und Interessen vorkommen. Die Untersuchung der organisationalen Prozesse in Multiagentensystemen, die Sozionik, ist inzwischen zu einem eigenen, sehr dynamischen Untersuchungsgebiet geworden [FFMa2004] und hat im Paradigma des Soft-Computing keinerlei Pendant. Neben der Möglichkeit, besonders komplexere Szenarien bis hin zu sozialen Systemen zu simulieren, bieten Multiagentensysteme – wie auch andere Verfahren aus der Verteilten KI – eine Robustheit, die durch zentral gesteuerte Systeme kaum erreichbar ist.

3.3 Synopsis

Die beiden bisher zur Transportprozessplanung verwendeten Methodenklassen basieren auf verschiedenen, teilweise sogar konträren Grundannahmen. Dies ist insbesondere durch die unterschiedlichen Planungsziele bedingt. Während die Methoden des Soft-Computing die Erfüllung eines globalen Ziels der Verfolgung individueller Zielsetzungen der beteiligten Objekte überordnen, sehen Agenten-

Methoden die Abstimmung der individuellen Zielsetzungen als Chance zur Erreichung eines übergeordneten Ziels.

4 Integration der Informationsstrukturen

Die Integration der beiden Paradigmen Soft-Computing und Multiagentensysteme kann auf verschiedene Arten erfolgen. Zunächst ist es möglich, dass Agenten als Komponente ihres Entscheidungssystems über Soft-Computing-Methoden verfügen. Soft-Computing-Methoden können z.B. bei der individuellen Handlungsplanung eines Agenten eingesetzt werden, aber auch bei der Erkennung von Mustern in Perzepten sowie bei Lern- und Adaptionsprozessen. Die Integration von Soft-Computing ist in diesem Fall die Integration der Technologie *in* einen einzelnen Agenten. Agentenarchitekturen, die „crispe“ Verfahren der Wissensrepräsentation und Reasoning-Verfahren traditioneller Tertium-non-datur-Logiken zwingend voraussetzen, müssen dementsprechend angepasst werden. Aber eine solche Integration von Soft-Computing-Methoden muss nicht auf die interne Struktur eines einzelnen Agenten beschränkt sein, sondern kann auch erhebliche Auswirkungen auf die Agentenkommunikation haben. Dies gilt z.B., wenn die Semantik der Agentenkommunikation durch eine gemeinsame Ontologie definiert werden soll: Wenn in einem Multiagentensystem Agenten interagieren müssen, die sich darin unterscheiden, dass einige von ihnen unscharfe, vage oder kontinuierliche Repräsentationen ihrer internen Zustände verwenden, andere hingegen nicht, müssen Repräsentationsmodelle prinzipiell unterschiedlicher Art miteinander verknüpft werden. Hier kann sicherlich zum Teil auf etablierte Methoden aus dem Soft-Computing, z.B. Fuzzifizierung/Defuzzifizierung, zurückgegriffen werden, die sich für einfache Steuerungssysteme bewährt haben; für den allgemeinen Fall wäre aber die Integration unterschiedlicher Agenten auf der Basis einer einheitlichen, zugleich jedoch hybriden Ontologie zu leisten. Dies geht natürlich substantiell über Verfahren zur Diskretisierung kontinuierlicher Werte hinaus und muss u.a. qualitative Abstraktionen umfassen. Bereits die Integration von in einer einheitlichen Repräsentationssprache definierten und inhaltlich sehr ähnlichen Ontologien (unter Erhaltung von Informationsgehalt, Redundanzfreiheit und Konsistenzeigenschaften) stellt ein äußerst anspruchsvolles und durchaus noch nicht vollständig gelöstes Problem dar [Wach2003].

Swarm Intelligence kann als ein alternativer Ansatz der direkten Integration von Ideen der beiden Paradigmen Soft-Computing und Multiagentensysteme angesehen werden. Die interne Struktur der „Agenten“ ist in diesem Fall jedoch so einfach, dass intelligentes Verhalten ausschließlich aus dem Zusammenwirken der Agenten emergieren soll. Trotz verschiedener Parallelen zwischen Swarm-Intelligence-Ansätzen und Multiagentensystemen (z.B. Verteiltheit von Wissen, relative Robustheit des Gesamtsystems bei Ausfall einzelner Individuen und Emergenz als zentrales Wirkungsprinzip) sind Swarm-Intelligence-Systeme eine besonders extreme Ausformung von Multiagentensystemen.

5 Ereignisgesteuerte integrierte globale und lokale Planung

Globale Offline-Planung ist die Methode der Wahl, wenn alle für die Lösung eines Planungsproblems benötigten Informationen bereits vor der Ausführung des Plans

korrekt und vollständig vorliegen. Im Bereich der Transportlogistik zählen z.B. die grundsätzlich zur Verfügung stehenden Routen (das Straßen- oder Schienennetz) zu diesen a priori bekannten und während des Planungsprozesses und der Planausführung unveränderlichen Informationen. Dynamisch sind hingegen die Verkehrssituation und die Auftragseingänge. Sie können entweder vorab approximiert werden (hierzu eignen sich insbesondere Soft-Computing-Methoden) oder sie werden online in einen vorhandenen Rahmenplan integriert. Letzteres setzt allerdings voraus, dass die handelnden Einheiten während der Ausführung des Prozesses in der Lage sind, die sich verändernde Situation mittels eigener Sensoren und/oder auf der Basis von Kommunikationsprozessen wahrzunehmen und zu analysieren.

Darüber hinaus müssen sie auch über die Fähigkeit verfügen, die ursprünglichen Pläne an die neuen Rahmenbedingungen anzupassen. Diese Anforderung wird besonders durch Agententechnologie erfüllt. So lässt sich bereits erkennen, dass eine vielversprechende Kombination auf der Basis einer vorgelagerten zentralen Planung durch Soft-Computing und einer nachgelagerten Feinplanung bzw. Laufzeitkontrolle durch Agententechnologie beruht.

5.1 Integration lokaler und globaler Planungsziele

Der integrierte Ansatz von Soft-Computing und VKI soll so erfolgen, dass auf der Ebene von Unternehmen Grobplanungen mit Hilfe von Soft-Computing Ansätzen durchgeführt werden. Diese werden als Zielvorgaben an Intelligente Agenten verteilt, die physikalische Ressourcen (z.B. Transportmittel) repräsentieren und eine Feinplanung unter Zuhilfenahme aller verfügbaren Informationen durchführen. Im Anschluss daran werden die Pläne von den Agenten ausgeführt und die Durchführung kontrolliert. Sollte ein Plan nicht mehr erfüllbar sein, so wird eine Neuplanung auf lokaler Ebene durchgeführt. Durch die individuelle Optimierung und lokale Berechnung kann es jedoch passieren, dass Ziele des Unternehmens oder Ziele eines Agenten nicht mehr erreichbar sind, wenn sich zum Beispiel die Rahmenbedingungen seit der ursprünglichen Planung verändert haben. Daher ist es notwendig, auch die Ausführung und die Umplanschritte zu überprüfen, damit bei größeren Problemen eine erneute Grobplanung angestoßen werden kann. Für diesen Benachrichtigungsmechanismus wird auf das strategische Management für autonome Softwaresysteme zurückgegriffen, welches in [Timm2006] und [TiHi2006] eingeführt wurde.

In aktuellen Forschungsarbeiten der VKI werden unterschiedliche Grade der Autonomie von Softwareagenten betrachtet [RoWe2004], [Müll1996]. Bereits Castelfranchi und Conte haben Ansätze für die dynamische Anpassung von Autonomie durch Normen, Verhaltensweisen und Prozeduren vorgeschlagen [CaCo1992]. In der Gestaltung von realitätsnahen Systemen in der Transportlogistik ist die Planung und Entscheidungsfindung in einer sich schnell ändernden und dynamischen Umwelt vorzusehen. Zusätzlich besteht über diese Änderungen und die neuen Randbedingungen oft Unsicherheit (Risikomanagement). Der konventionelle Ansatz zur Berücksichtigung strategischer Aspekte innerhalb eines Systems führt über die Manipulation des Systems durch menschliche Nutzer, wie es beispielsweise im Konzept der interaktiven Tourenplanung vorgesehen ist [KoSc2002].

Eine erste Möglichkeit der Flexibilisierung ist die Nutzung von Systemkomponenten, die die einzelnen Subsysteme überwachen und bei Zielverfehlung oder unge-

wünschten Systemzuständen in die Autonomie der Subsysteme eingreifen. Schönberger und Kopfer haben in diesem Zusammenhang einen Planungsagenten entwickelt, der solange frei disponieren kann, bis ein vorgegebenes Serviceniveau unterschritten wird. In diesem Fall werden durch eine übergeordnete Autorität Dispositionsentscheidungen vorgegeben, die zwar dem Ziel des Planungsagenten widersprechen, aber eine Wiedererreichung des benötigten Service-Niveaus garantieren [ScKo2007].

Dieser zentrale Ansatz würde aber zur Laufzeit einen potentiellen Engpass darstellen. Das strategische Management (vgl. Abbildung 1) für autonome Softwaresysteme geht davon aus, dass ein übergeordnetes und formalisiertes Zielsystem vorliegt und dieses durch eine weitgehend autonome Interaktion der Agenten eingehalten werden kann [Timm2006]. Abweichend von dem ursprünglichen Ansatz, in dem der Grad der Autonomie während der Laufzeit dynamisch und autonom durch die Agenten selbst angepasst wird, soll im Rahmen des hier vorgestellten Ansatzes nicht die Autonomie beeinträchtigt werden, sondern eine Benachrichtigung der zentralen Einheit durchgeführt werden, die dann eine erneute Planung anstoßen kann. Die Agenten eines Unternehmens interagieren miteinander, um die Einhaltung der Ziele des Unternehmens festzustellen oder adäquat gegenzulenken. Die Zielspezifikation auf Unternehmensebene sollte für die Nutzung im strategischen Management eine Funktion zur Messung der Zielerreichung wie Kennzahlen oder Kennlinien enthalten, vgl. [NyWi1999]. Diese sollte zumindest drei Ausprägungen von Zielerreichung unterscheiden können: Ziel erreicht (grüner Bereich), Ziel leicht verfehlt (gelber Bereich) und Ziel stark verfehlt (roter Bereich). Das strategische Management wird durch einen vierstufigen Prozess realisiert (vgl. Abbildung 5-1).

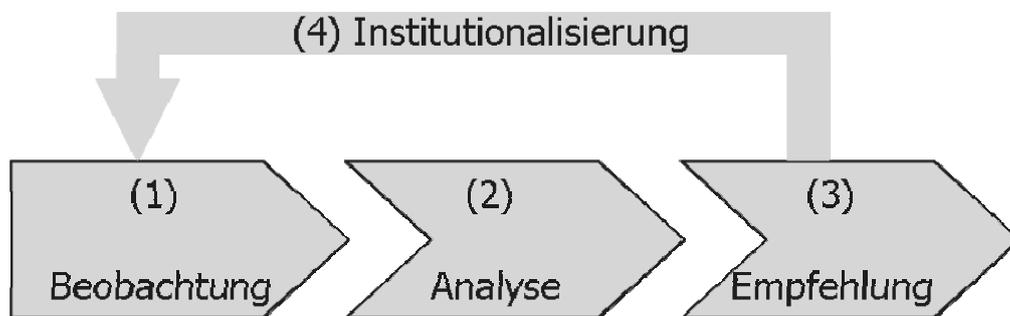


Abb. 5-1: Phasen des Strategischen Managements in Multiagentensystemen

Im Normalbetrieb befindet sich das System in der ersten Phase (Beobachtung). In dieser Phase melden alle Agenten ihre individuellen Erfolgsparameter an eine spezielle Instanz, die die Systemziele sowie deren Erfüllungsgrad auf der Basis dieser individuellen Erfolgsparameter berechnet. Die Zielerfüllung sowie die aktuellen Ziele werden allen Teilnehmern des Systems zur Verfügung gestellt. In dieser Phase sollen die individuellen Agenten, ihre Feinplanung in Hinblick auf ihre lokalen Ziele optimieren. Wird über längere Zeit ein Systemziel leicht verfehlt (gelb), sollen die Agenten bei der Feinplanung auf das leicht verfehlt Gruppensziel besondere Rücksicht nehmen, ohne jedoch eine kooperative Planung anzustoßen. Hierbei wechselt das System in Phase 2 (Analyse). In vielen denkbaren und realen Szenarien wird jedoch eine unkoordinierte individuelle Handlung bei einer globalen Verfehlung nicht zu zufrieden stellenden Ergebnissen führen. Problematisch wird diese Situation

dann, wenn viele Agenten versuchen, auf die gleiche Weise ein Globalziel zu erreichen. Dann kann das Globalziel übererreicht werden, was wiederum dazu führen kann, dass ein anderes Ziel verfehlt wird. Eine solche Wechselwirkung lässt sich nur durch koordinierte Aktionen verhindern. Daher wird in einem Fall andauernder starker Zielverfehlung (roter Bereich) die Phase 3 des strategischen Managements aktiviert. In dieser Phase wird die zentrale Planung eingeschaltet, damit eine Umplanung durch eine zentrale Instanz zu einer Rekonfiguration der Ziele und Aufträge zwischen den Ressourcen führt. Die Institutionalisierung beschreibt dann wieder den Schritt der Rückführung von der zentralen Planung in das dezentrale System.

Die in diesem Kontext vorgestellten Methoden der Beobachtung und Analyse bedürfen in ganz besonderer Weise Wissen. Als Wissen werden neben expliziten Fakten und Regeln auch unsichere, indirekte, kontextabhängige und implizite Annahmen verstanden (z. B. Erfahrungswerte, Heuristiken und Schätzungen). Da logistische Prozesse inhärent dynamisch sind, ist die schnelle und verlässliche Bereitstellung von Wissen erforderlich. Für die Modellierung bedeutet dies, dass statische Methoden, z. B. die Repräsentation statischen Wissens mit Ontologien, durch dynamische Methoden des Wissensmanagement begleitet und mit ihnen integriert werden müssen.

5.2 Koordinations-Rolle des Wissensmanagements

Je dynamischer die Umgebung logistischer Prozesse ist, desto wichtiger werden Infrastrukturen, die es erlauben, die planenden und operativ tätigen Komponenten möglichst schnell mit den jeweils aktuellen und relevanten Informationen zu versorgen. In [LGH2006a] und [LGH2006b] wird eine Wissensmanagement-Infrastruktur für Multiagentensysteme vorgeschlagen, die auf einem System von Wissensmanagement-Rollen beruht. Das Modell verzichtet auf eine strikte 1-zu-1-Zuordnung von Agenten und Wissensmanagement-Funktionalitäten. Unter der „Rolle“ eines Agenten wird in diesem Zusammenhang ein komplexes Verhaltensmuster verstanden, das unabhängig von der primären logistischen Funktion des jeweiligen Agenten (z. B. der Funktion eines Transportmittels oder der eines Disponenten) eine sekundäre Funktion innerhalb eines verteilten Wissensmanagement-Systems realisiert. Diese sekundäre Funktion kann sich entweder auf Prozesse beschränken, die nur genau einen Agenten betreffen (in diesem Falle sprechen wir von einer „internen“ Rolle), oder aber ein Interaktionsmuster festlegen, in das zwei oder mehr Agenten involviert sind („externe“ Rolle). In Abbildung 5-2 sind die vier von uns angenommen externen Rollen und die jeweiligen Interaktionsschemata dargestellt.

Rollen sind nicht direkt an einzelne Agenten gebunden, d. h., grundsätzlich kann ein und derselbe Agent in verschiedenen Situationen unterschiedliche Rollen ausfüllen. In einem sehr elementaren Szenario gibt es zwei Rollen, die eines „Knowledge-Consumers“ und die eines „Knowledge-Providers“.

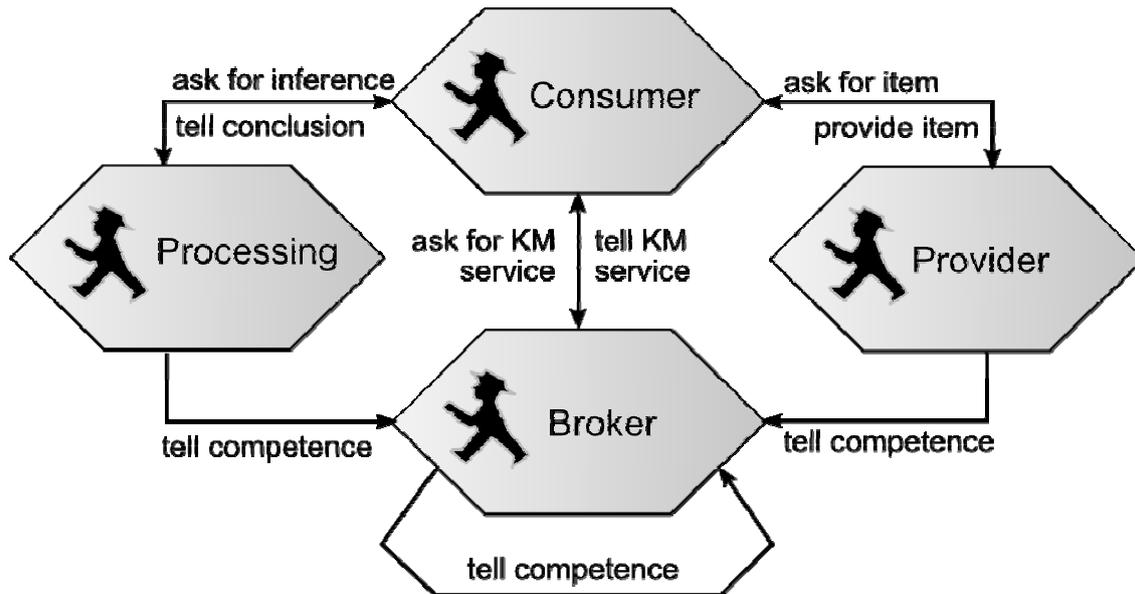


Abb. 5-2: Externe Rollen

In diesem einfachen Szenario findet ein Wissenstransfer von dem Agenten in der Knowledge-Provider-Rolle zu dem Agenten in der Knowledge-Consumer-Rolle statt. Dieses einfache Szenario wurde auf der Basis eines *Iterated-Contract-Net*-Protokolls als „Iterated Knowledge Transfer Protocol“ implementiert (vgl. [LGH2006a]). Neben den beiden Basis-Rollen des Knowledge-Consumer und des Knowledge-Provider wurden fünf weitere, teils deutlich komplexere Rollen definiert, die u. a. Wissensmanagementprozesse wie Wissensakquisition und Wissenswartung ausführen. Das System der Rollen wird durch eine beschränkte Zahl von Parametern definiert, deren Werte festlegen, unter welchen Bedingungen welche Agenten miteinander auf der Basis welcher Wissensmanagement-Rollen interagieren. Als Beispiele für diese Parameter seien *Transferkosten*, *Relevanz* eines Wissenspakets und *Vertrauen* in den Kooperationspartner genannt. Eine detailliertere Beschreibung findet sich in u. a. [LGH2006a] und [LGH2006b].

6 Zusammenfassung und Ausblick

In den letzten Jahren hat sich gezeigt, dass die verfolgten Planungsansätze und die entwickelten Methoden häufig den aktuellen Anforderungen moderner Logistiksysteme nicht mehr genügen. Die Größe und Komplexität der heutigen Transportsysteme ist selbst beim Einsatz massivster Rechenleistungen nicht mehr zufrieden stellend zu beherrschen. Ein Überdenken der bisher streng hierarchisch angelegten Planungskonzepte sowie die Entwicklung und Erprobung neuer Planungsansätze stellt daher eine wesentliche Herausforderung für die Unterstützung der Planung zukünftiger Logistiksysteme dar.

In diesem Beitrag skizzierten wir Wege zur Integration von Soft-Computing-Methoden, Multiagentensystemen und rollenbasiertem Wissensmanagement als Perspektive für die Simulation selbststeuernder logistischer Prozesse in dynamischen Umgebungen. Neben der direkten Integration von Soft-Computing-Methoden in die Architektur individueller Agenten und Swarm Intelligence haben wir einen Ansatz zur ereignisgesteuerten Kombination globaler und lokaler Planungsmecha-

nismen vorgestellt, der uns insbesondere unter dem Aspekt der Robustheit für transportlogistische Anwendungen sehr vielversprechend erscheint. In einem solchen Ansatz kommt einem effektiven und flexiblen Wissensmanagement eine besondere Bedeutung zu. In diesem Zusammenhang skizzierten wir in ihren Grundzügen eine rollenbasierte Infrastruktur für MAS-basiertes Wissensmanagement.

Literaturverzeichnis

- [BeBa2004] Berger, J.; Barkaoui, M.: A parallel hybrid genetic algorithm for the vehicle routing problem with time windows. In: *Computers & Operations Research* 12 (2004) 31, S. 2037-2053
- [CaCo1992] Caltelfranci, C.; Conte, R. (1992): Emergent functionality among intelligent systems: Cooperation within and without minds. In: *Journal on Artificial Intelligence and Society*, 1 (1992) 6, S. 78–87.
- [CHMu1999] Cheung, R.K.; Muralidharan, B.: Impact of dynamic decision making on hub-and-spoke freight transportation networks. In: *Annals of Operations Research* 87, 1999, S. 49-71.
- [CrLa1998] Crainic, T.G.; Laporte, G. (Hrsg.): *Fleet Management and Logistics*. Kluwer, 1998
- [DeKa2006] Derigs, U.; Kaiser, R.: Applying the attribute hill climber heuristic to the vehicle routing problem. Accepted for publication in: *European Journal of Operational Research*, 2006
- [Ferb1999] Ferber, J.: *Multi-Agent Systems - An Introduction to Distributed Artificial Intelligence*. Addison-Wesley: Harlow, 1999.
- [Find1991] Findler, N.: Uttam sengupta: An overview of some recent and current research in the ai lab at arizona state university. *AI Magazine*, 3 (1991) 12, S. 22–29.
- [FFMa2004] Fischer, K.; Florian M.; Malsch T. (Eds.): *Socionics: Its Contributions to the Scalability of Complex Social Systems*. Lecture Notes in Computer Science, Lecture Notes in Artificial Intelligence, Springer Verlag, 2004.
- [HSEH2005] Herrmann, T.; Schöpe, L.; Erkens, E.; Hülder, M. (Hrsg.): *Mobile Speditionslastunterstützung*. Shaker-Verlag: Aachen, 2005.
- [KaJe1997] Kalenka, S.; Jennings, N.: Socially responsible decision making by autonomous agents. In K. Korta et al. (Hrsg.): *Cognition, Agency and Rationality*, Dordrecht, The Netherlands: Kluwer, 1997, S. 135–149.
- [KoSc2002] Kopfer, H.; Schönberger, J.: Interactive Solving of Vehicle Routing and Scheduling Problems: Basic Concepts and Qualification of Tabu Search Approaches. In: Sprague Jr., R.H. et al. (Hrsg.): *Proceedings of HICCS 35*. 2002, S. 84-93.
- [KrKo2006] Krajewska, M.; Kopfer, H.: Collaborating freight forwarding enterprises – request allocation and profit sharing. In: *OR Spectrum* 3 (2006) 28, S. 301-317.
- [Lack2004] Lackner, A.: *Dynamische Tourenplanung mit ausgewählten Metaheuristiken*. Cuvillier Verlag: Göttingen, 2004.
- [LGH2005] Lorenz, M.; Gehrke, J. D.; Hammer, J.; Langer, H.; Timm, I. J.: Knowledge Management to Support Situation-aware Risk Management in Autonomous, Self-

- managing Agents. In: Czap, H. et al. (Hrsg.): Self-Organization and Autonomic Informatics (I). Frontiers in Artificial Intelligence and Applications, Bd. 135. IOS Press: 2005, S. 114-128.
- [LGH2006a] Langer, H.; Gehrke, J. D.; Hammer, J.; Lorenz, M.; Timm, I. J.; Herzog, O.: A Framework for distributed knowledge management in autonomous logistic processes. International Journal of Knowledge-Based and Intelligent Engineering Systems, 4 (2006) 10, S. 277-290.
- [LGH2006b] Langer, H.; Gehrke, J. D.; Herzog, O.: Distributed Knowledge Management in Dynamic Environments. 2006 (to appear).
- [Lipp2005] Lippe, W.-M.: Soft-Computing. Springer; 2005.
- [Müll1993] Müller, H.-J.: Verteilte Künstliche Intelligenz - Methoden und Anwendungen. Mannheim: BI Wissenschaftsverlag; 1993.
- [NyWi1999] Nyhuis, P.; Wiendahl, H.-P.: Logistische Kennlinien – Grundlagen, Werkzeuge und Anwendungen, Springer. Berlin, 1999.
- [Retz1995] Retzko, R.: Flexible Tourenplanung mit selbstorganisierenden Neuronalen Netzen. Unitext Verlag: Göttingen, 1995
- [RoWe2005] Rovatsos, M.; Weiß, G.: Autonomous Software. In Chang, S. K. (Hg.): Handbook of Software Engineering and Knowledge Engineering, volume 3: Recent Advances, River Edge. World Scientific Publishing: New Jersey, 2005.
- [RuNo2003] Russel, S. J.; Norvig, P.: Artificial Intelligence: A Modern Approach. 2nd Edition, Englewood Cliffs, New Jersey: Prentice Hall, 2003.
- [ScKo2007] Schönberger, J.; Kopfer, H.: On Decision Model Adaptation in Online Optimization of a Transport System. Angenommen zur Veröffentlichung im Tagungsband der Konferenz „Supply Networks and Logistics Management, 2007.
- [Tecu1998] Tecuci, G.: Building Intelligent Agents – An Apprenticeship Multi-strategy Learning Theory, Methodology, Tool and CaseStudies. Academic Press: San Diego, California, 1998.
- [TiHi2006] Timm, I.J.; Hillebrandt, F.: Reflexion als sozialer Mechanismus zum strategischen Management autonomer Softwaresysteme. In: Florian, M. et al. (Hrsg.): Reflexive Soziale Mechanismen. VS Verlag für Sozialwissenschaften, Wiesbaden, 2006, Buch im Druck.
- [Timm2006] Timm, I.J.: Strategic Management of Autonomous Software Systems. TZI Technical Report. Habilitation. Fachbereich Mathematik und Informatik. Technologie-Zentrum Informatik. Universität Bremen, 2006.
- [Wach2003] Wache, H.: Semantische Interpretation für heterogene Informationsquellen. DISKI-Dissertationen zur Künstlichen Intelligenz. Bd.261. Köln: infix, 2003
- [Weiß1999] Weiß, G. (Hrsg.): Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence. Cambridge, Massachusetts: The MIT Press, 1999.
- [WoJe1995] Wooldridge, M.; Jennings, N.: Intelligent Agents: Theory and Practice. In: The Knowledge Engineering Review 2 (1995) 10, S. 115–152.

Softwareagenten im Krankenhaus – Ubiquitous Computing zwischen Ordnung und Chaos

Christoph Niemann

Lehrstuhl für Wirtschaftsinformatik (BWL VII)

Universität Bayreuth

Universitätsstr. 30

95447 Bayreuth

christoph.niemann@uni-bayreuth.de

Zusammenfassung: Ubiquitous Computing stellt ein Paradigma der Informationstechnologie dar, das in der Lage ist, neben geschlossenen auch offene Systeme, in denen die Anzahl der abgebildeten Entitäten variabel und nicht ex ante bekannt ist, der realen Welt detailliert in Informationssystemen abzubilden. Auf Grundlage der Annahme von allgegenwärtiger Sensoren, Verarbeitungskapazität und Effektoren stellt der Artikel das Projekt EMIKA vor, mit dem die Effekte des Ubiquitous Computing untersucht werden. Durch Kooperation zwischen menschlichen Akteuren und Softwareagenten entsteht hybride Intelligenz, in der Mensch und Maschine gegenseitig aufeinander angewiesen sind.

1 Einleitung

Die Vision des Ubiquitous Computing (UC) [Weis1991] postuliert als Folge der fortschreitenden Miniaturisierung von IT die zukünftige Allgegenwärtigkeit von Informationsverarbeitung in der Umgebung. Computer werden kleiner und ihre Anzahl erhöht sich. Dadurch müssen sich zwangsläufig ihre Eingabekanäle verändern. Nicht mehr jeder Computer kann durch menschenzentrierte Eingabemöglichkeiten wie Tastatur und Maus gesteuert werden, da die Anzahl der Rechner pro Person zu groß wird. Andererseits ermöglichen Techniken wie RFID (Radio Frequency Identification) die weitgehend automatisierte Wahrnehmung der Umwelt. Neuartige Eingabekanäle können an die Stelle der bisherigen treten und diese erweitern oder ersetzen. Allgegenwärtige Sensorik ermöglicht so die Wahrnehmung der Umwelt in kurzen zeitlichen Abständen und erzeugt dadurch einen kontinuierlichen Datenstrom an Eingaben, der dem Informationssystem (IS) zur Verfügung steht.

In der Konsequenz kann UC es ermöglichen, die Umwelt detaillierter als bisher in IS abzubilden, da die Sensorik in der Lage ist, die physische Welt in Echtzeit mit ihrer informationstechnischen Abbildung zu synchronisieren. Der physischen Welt wird eine *logische Welt* als ihre detailgenaue Abbildung gegenüber gestellt. In der logischen Welt repräsentieren Softwareagenten die Entitäten der physischen Welt. Die Zunahme an Sensorik und – damit einhergehend – an Eingaben führt allerdings auch zu einer größeren Menge an Daten, die verarbeitet werden müssen, um daraus

sinnvolle Information zu generieren. Die *Datenmenge* nimmt zu, die *Informationsqualität* steigt allerdings ohne Verarbeitung nicht in gleichem Maße.

Um den Menschen nicht mit der Flut an neu gewonnener Information zu belasten, müssen IS zunehmend autonom agieren. Nicht mehr jede Aktion kann von Menschen angestoßen oder jede Entscheidung von Menschen getroffen werden, sondern Computer müssen auf Grundlage der logischen Welt Entscheidungen treffen und entsprechende Aktionen auslösen können. Autonome IS können den Menschen von Entscheidungen entlasten, die momentan noch dessen Aufmerksamkeit benötigen. Der Mensch kann sich dadurch stärker seiner eigentlichen (nicht IT-bezogenen) Aufgabe widmen und wird von Datenpflege, -eingabe und -verarbeitung entlastet.

Dieser Artikel gliedert das Paradigma des UC in seine beiden Bestandteile *Allgegenwärtigkeit* (engl. ubiquitous) und *Verarbeitung* (engl. computing) auf. Diese Aufteilung bildet die Grundlage für EMIKA, ein Entscheidungsunterstützungssystem, das UC in der Domäne Krankenhaus einführt, um IS-gestützte Terminplanung zu ermöglichen. Neben den technischen Aspekten steht vor allem die Frage nach der Umsetzung einer „calm technology“ [Weis1996] und nach der Schnittstelle zwischen Wirklichkeit und deren logischem Abbild im Vordergrund.

2 Ubiquitous Computing

Ausgehend vom Paradigma der Personal Computer, in der jeder Computer in der Regel einem Menschen zugeordnet ist, legt das UC eine n:m-Beziehung zwischen Mensch und Computer zu Grunde [Weis1996]. Die Computer werden durch Miniaturisierung und Einbettung in der Umgebung unsichtbar und verrichten ihren Dienst, ohne die Aufmerksamkeit der Menschen über Gebühr zu beanspruchen. Sie bewegen sich aus der zentralen Wahrnehmung in die periphere; der Mensch muss ihnen nur dann Aufmerksamkeit widmen, wenn Ausnahmestände auftreten, die es nicht erlauben, ohne menschlichen Eingriff mit der Ausführung fortzufahren. Gleichzeitig verändern sich die Eingabekanäle der Computer von menschenzentrierten Möglichkeiten wie Tastatur und Maus hin zu allgegenwärtiger Sensorik, die den Computern einen kontinuierlichen Strom an Eingaben liefert. Einerseits entsteht so eine detailgenaue Abbildung der Wirklichkeit, die zu jedem Zeitpunkt die Realität wiedergibt, andererseits müssen IS den Datenstrom verarbeiten und so von einfachen Fakten auf relevantes Wissen schließen. Erst dieses Wissen ermöglicht sinnvolle Entscheidungsunterstützung. Eymann identifiziert drei Trends in der Informatik: (1) die Miniaturisierung, (2) die Veränderung der Schnittstellen zwischen Mensch und Computer (und zwischen Umwelt und Computer) und (3) die universelle Verfügbarkeit von Computernetzwerken [Eyma2002]. Alle drei Trends verstärken die Transformation aus der Ära der Personal Computer in die Ära des UC.

UC stellt durch Sensorik und Verarbeitung das detailgenaue Abbild der Wirklichkeit in einem IS bereit. Anwendungen des UC nutzen die Information dieser logischen Welt, um menschliche Aufgaben zu unterstützen, zu übernehmen oder erst zu ermöglichen. Durch die Allgegenwärtigkeit der IT kann UC auch offene Systeme abbilden, die sich durch die Charakteristika *concurrency*, *asynchrony*, *decentralized control*, *inconsistent information*, *arm's length relationships* und *continuous operation* auszeichnen [Hewi1998]. Die IS ermöglichen Entscheidungsunterstützung, die wieder in die physische Welt kommuniziert werden muss. Dazu dienen Effektoren, die die Schnittstelle zwischen logischer und physischer Welt darstellen.

Die Bestandteile des UC lassen sich in einer Hierarchie anordnen, in der die Sensorik unterhalb der Verarbeitung angesiedelt ist. Abbildung 2-1 zeigt ein solches Schichtenmodell. Sensorik bildet die unterste Schicht der Pyramide, die darauf aufbauende Verarbeitung lässt sich weiter in die Schichten Intelligenz, Integration und Optimierung aufteilen. Erst die Optimierungsschicht ermöglicht Entscheidungsunterstützung, da die Verarbeitung der Daten erst dort eine Perspektive auf das Gesamtsystem ermöglicht, aus der heraus Planung stattfinden kann.

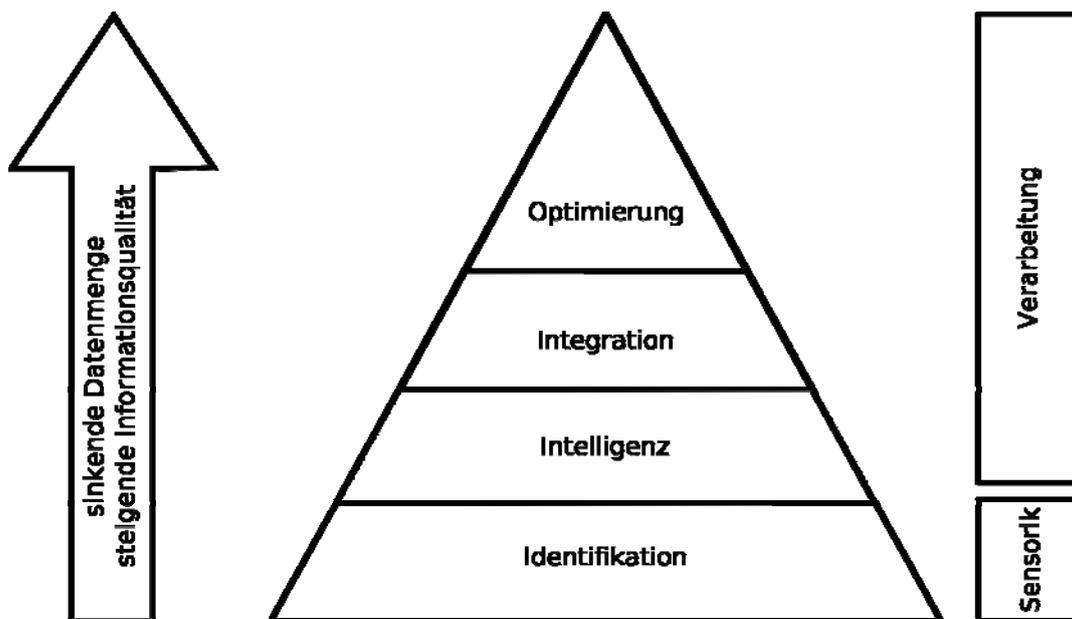


Abbildung 2-1: Schichtenmodell zur Verarbeitung von Kontextinformation [ZREy+2006]

Im Folgenden werden die UC-Bestandteile und die darauf aufbauenden IS entlang der Schichten der Pyramide erläutert. Die Grundlage bildet die allgegenwärtige Sensorik, die erst die logische Abbildung der Wirklichkeit ermöglicht. Die Verarbeitung baut darauf auf, reichert die schon vorgefilterten Daten mit weiteren Daten an und interpretiert diese. Die Integrationsschicht führt die Daten aus verschiedenen Quellen zusammen und erlaubt so weitergehenden Schlussfolgerungen, die zu einer erhöhten Informationsqualität führen. Die Optimierungsschicht schließlich erlaubt die Optimierung auf hoher Ebene. Dort ist schließlich das Entscheidungsunterstützungssystem angesiedelt.

Die Resultate kommuniziert das IS über Effektoren zurück in die physische Welt. Zusammen bilden diese drei Bestandteile die Grundlage für UC.

2.1 Allgegenwärtige Sensorik

Allgegenwärtige Sensorik ermöglicht die kontinuierliche Sammlung hoch aufgelöster Daten, die zu einer virtuellen Abbildung der Wirklichkeit verknüpft werden können. Aus der logischen Welt kann das IS dann den Kontext einzelner Entitäten bestimmen, um – ausgehend von diesem Kontext – die Handlungsmöglichkeiten des einzelnen Akteurs beurteilen zu können. Kontext ist nach [Dey2001] jegliche Information, die zur Charakterisierung der Situation dient. Ausgewählte, häufig

genannte Kontextmerkmale sind *Ort*, *Identität*, *Zeit* und *Rolle* [MFGG2003]. Um sie zu erfassen, können entweder Sensoren eingesetzt werden, deren Zweck es ist, genau dieses Kontextmerkmal zu erfassen, oder die Merkmale können aus den Sensordaten abgeleitet werden, die ursprünglich einem anderen Zweck dienen [Hase2005]. Diese Kontextmerkmale lassen sich direkt durch Sensorik messen und bilden damit die Grundlage weiterer Kontextmerkmale, die erst auf höheren Ebenen erschlossen werden können.

Das Merkmal Identität als grundlegendes Merkmal auf der Identifizierungsschicht kann auf verschiedenen fein granulierten Ebenen abgebildet werden. Je nach Anzahl der Entitäten im System lässt sich die Identität individuell oder auf Klassenebene festlegen. Dazu eingesetzte Technologien sind RFID (Radio Frequency Identification), Magnetkarten oder Barcodes. Im Rahmen der UC gewinnt RFID zunehmend an Bedeutung, da diese Technologie sowohl die individuelle Identifizierung als auch die Identifizierung auf Klassenebene ermöglicht. RFID ist zudem relativ kostengünstig; die Beschaffung der Lesegeräte verursacht den größten Anteil der Kosten. Die Preise der in großer Stückzahl notwendigen Transponder (sog. „Tags“) liegen im Centbereich. Diese Kostenstruktur ermöglicht den Einsatz auch in Anwendungsgebieten, in denen sehr viele Entitäten erfasst werden müssen. Die Technik hat im Gegensatz zu Barcodes oder anderen Identifizierungstechniken den Vorteil, dass sie berührungslos und ohne notwendigen Sichtkontakt funktioniert. Es reicht aus, einen Transponder durch das vom Lesegerät ausgeleuchtete Feld zu bewegen, um die Identifikation zu ermöglichen.

Auch das Kontextmerkmal Ort lässt sich über RFID erfassen: Sobald sich ein Chip in die Nähe eines Lesegerätes bewegt, löst er ein Leseereignis aus, durch das eine Entität lokalisiert wird. Das Leseereignis enthält die eindeutige Identifikation des Tags sowie einen Zeitstempel. RFID-Lesegeräte lösen nur dann ein Leseereignis aus, wenn der Transponder durch das vom Gerät ausgeleuchtete Feld *bewegt* wird. Ein ruhender Chip löst keine Ereignisse aus, selbst wenn er sich in Reichweite befindet. Mittels RFID lassen sich also Bewegungen von Entitäten erkennen. Durch die Platzierung der Lesegeräte zwischen Zonen verschiedener Bedeutung lassen sich Übergänge von Entitäten erkennen. In Gebäuden bieten sich Türen als Übergänge zwischen Räumen zur Erfassung an. Die Leseereignisse entsprechen dann der Bewegung einer Entität von einem Raum in den anderen.

Die Merkmale Zeit und Rolle lassen sich in der Regel direkt abbilden, da diese Information normalerweise bereits im IS vorhanden ist. Sie dienen vor allem der Anreicherung der weiteren Kontextdaten.

Zusätzlich zur automatisierten Erfassung der Kontextmerkmale durch Sensorik kann der Kontext immer auch explizit vom Nutzer erfragt werden. Diese Methode läuft allerdings dem Ziel der „calm technology“ zuwider, so dass sie nur dann angewandt werden sollte, wenn die Wichtigkeit der Information die Beeinträchtigung des Nutzers rechtfertigt.

2.2 Verarbeitung

Die Verarbeitung muss aus die große Menge an Sensordaten schrittweise filtern, aggregieren und durch weitere Information anreichern, so dass im logischen Abbild der Wirklichkeit nur Information enthalten ist, die bereits verknüpft und interpretiert ist

[ZREy2006]. Exemplarisch zeigt Pradhan [Prad2000], wie geographische Ortsinformation durch die Verknüpfung mit semantischer Ortsinformation mit Bedeutung versehen werden kann. Analog müssen auch andere Kontextmerkmale mit Semantik angereichert werden, denn erst diese Information ermöglicht die Verarbeitung im Rahmen der Entscheidungsunterstützung.

Sofern die Sensoren über Verarbeitungskapazität verfügen, kann die Verarbeitung schon auf Ebene der Hardware beginnen. Die Signale können so vorgefiltert werden, dass nur solche, die eine Veränderung der logischen Welt nach sich ziehen, an die höheren Ebenen weitergeleitet werden. Anstelle von kontinuierlicher Anpassung der logischen Welt muss diese nur dann nachgeführt werden, wenn sich der Zustand der Wirklichkeit tatsächlich geändert hat. Nur dann wird eine Veränderung in der logischen Welt sichtbar. So lange lediglich die Zeit verstreicht, alle weiteren Kontextmerkmale jedoch konstant bleiben, bleibt das Bild der logischen Welt konstant.

Jede Schicht filtert und aggregiert die Information lokal ohne Einbeziehung der anderen Entitäten auf der gleichen Schicht. Auf Ebene der einzelnen Lesegeräte kann ein Lesegerät nur Ereignisse filtern, die direkt an diesem Gerät auftreten. Erst die jeweils nächsthöhere Ebene kann Leseereignisse zusammenführen, die von zwei unterschiedlichen Geräten stammen, aber zur Bewegung einer Entität gehören. In der logischen Welt müssen so Ortsübergänge abgebildet werden.

Die Verknüpfung der Leseereignisse der beiden Lesegeräte einer Tür durch eine Entität (2x die gleiche Identität) ermöglicht die Erkennung einer Bewegung. Diesen Interpretationsschritt übernimmt die Intelligenz-Schicht und propagiert nur das Verlassen eines Raumes und das Betreten des nächsten an die nachfolgende Integrations-Schicht.

2.3 Effektoren

Um die physische Welt auf Grundlage der Entscheidungen in der logischen Welt zu verändern, verfügt das IS über Effektoren, mittels derer sich die Wirklichkeit beeinflussen lässt. Eine wichtige Unterscheidung ist dabei zwischen der Kommunikation zwischen Mensch und Maschine bzw. zwischen Maschine und Maschine zu treffen.

Die Maschine-Mensch-Kommunikation stellt die komplexere Variante dar, da die Entscheidungsgewalt oft bei den Menschen liegt. Die Optimierungsschicht kann Vorschläge generieren, die der Mensch dann annehmen oder ablehnen kann. Effektoren zwischen Maschine und Mensch müssen daher die Information in ein menschenlesbares Format umwandeln und gegebenenfalls auch über einen Rückkanal verfügen, um die Entscheidung wieder in die logische Welt zu kommunizieren. Mobile Endgeräte wie PDAs (Personal Digital Assistants) eignen sich als Effektoren: Sie können Informationen in einem menschenlesbaren Format darstellen, sind für die Menschen einfach zu handhaben und können in einer 1:1-Beziehung einem Menschen zugeordnet werden. So entfällt die komplizierte Identifizierung eines einzelnen Nutzers, wenn mehrere Personen auf dasselbe Endgerät zur Kommunikation mit dem IS angewiesen sind (wie es bei stationären Endgeräten der Fall wäre).

Durch das Zusammenspiel von Sensoren, Verarbeitung und Effektoren entsteht ein Kreislauf, der Wechselwirkungen zwischen logischer und physischer Welt ermöglicht. Diesen Kreislauf stellt Abbildung 2-2 dar. In der logischen Welt agieren Softwareagenten, die im folgenden Abschnitt erläutert werden. Sie erfassen die physische Welt über Sensoren, verarbeiten die Information und beeinflussen die physische Welt über Effektoren.

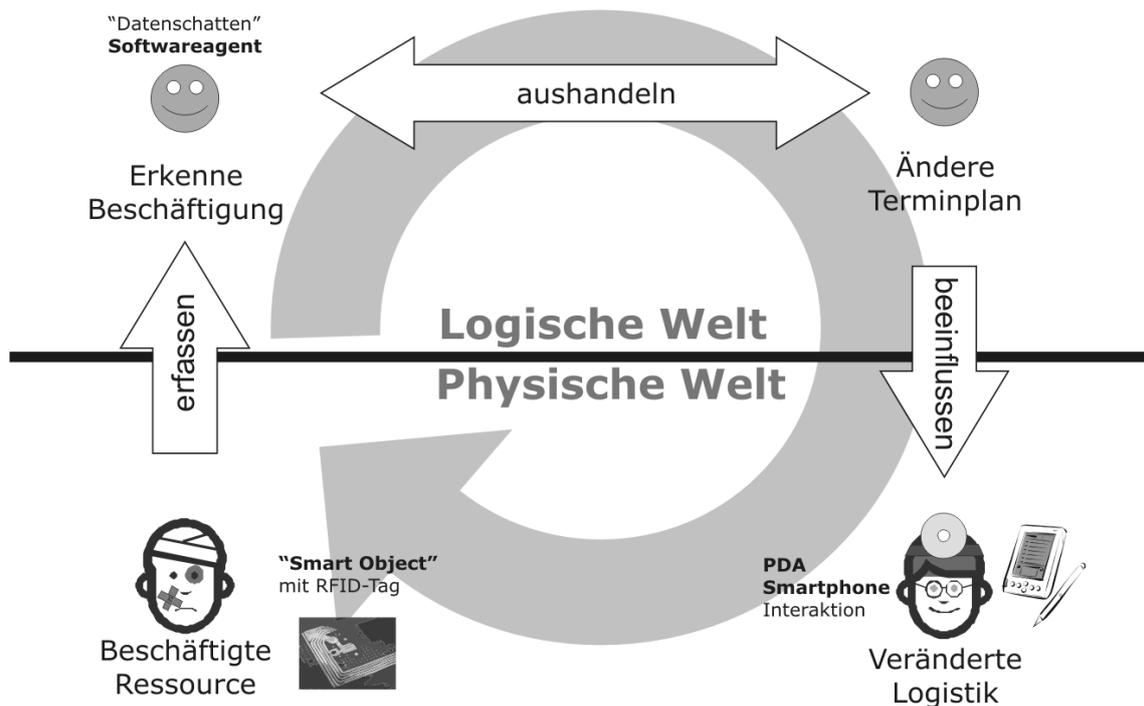


Abbildung 2-2: Feedback-Loop zwischen Mensch und IS bzw. physischer Welt und deren logischer Abbildung

Die Maschine-Maschine-Kommunikation beschränkt sich in der Regel auf die Transformierung der Nachrichten in das richtige Austauschformat. Insbesondere haben Maschinen in der Regel kaum Entscheidungsmöglichkeiten, so dass der Informationsfluss hier einseitig erfolgen kann: Die Ergebnisse aus der Optimierungsschicht werden an die Maschinen in der physischen Welt übermittelt, diese passen daraufhin ihre Aktionen entsprechend an.

3 Softwareagenten im UC

Ein Paradigma zur Verarbeitung der anfallenden Information sind Softwareagenten und – als Kooperation mehrerer Agenten – Multiagentensysteme (MAS). Neben einer zentralen Verarbeitungskomponente eignen sich MAS als Entwicklungsansatz innerhalb des UC sowohl zur Modellierung als auch zur Implementierung, da sie flexibel auf die sich ändernde Umweltbedingungen reagieren können. Diese Flexibilität wird oft als das wichtigste Merkmal der Agententechnologie angesehen [Kirn2006]. In Anwendungsgebieten, in denen eine sich ständig ändernde Anzahl Akteure aktiv ist, können MAS diese Eigenschaft durch den Lebenszyklus einzelner Agenten

abdecken. MAS bieten zusätzlich Flexibilität gegenüber sich ändernden Systemeigenschaften oder -anforderungen, da in diesen Fällen nur einzelne Agenten anders konzipiert werden müssen, das Gesamtsystem jedoch bestehen bleiben kann.

Viele der den offenen IS eigenen Charakteristika werden von Softwareagenten direkt unterstützt. Lokale Informationsverarbeitung auf Ebene der einzelnen Agenten führt zu Nebenläufigkeit der Agenten, die asynchron durch Nachrichten miteinander kommunizieren. Die Agenten unterliegen in MAS keiner zentralen Kontrolle, sondern entscheiden autonom über ihre Aktionen. Als Koordinationsmechanismus können ökonomische Verhandlungen zum Einsatz kommen [ESMü2003], die zu selbst organisierenden Systemen führen. Durch geeignete Aktionsmöglichkeiten können Softwareagenten auch mit inkonsistenter Information umgehen und geeignete Routingprotokolle erlauben es ihnen, Kommunikationsketten im MAS aufzubauen. Agenten können darauf ausgelegt werden, eine bestimmte Aufgabe zu erfüllen oder sie können kontinuierlich ausgeführt werden und ihre Aktionen aufgrund der sich ändernden Bedingungen in der logischen Welt planen und durchführen. Die ex ante nicht feststehende Zahl von Akteuren innerhalb eines Systems lässt sich durch Softwareagenten abbilden: Sobald ein neuer Akteur in der Wirklichkeit auftritt, wird dem MAS ein weiterer Agent hinzugefügt. Gleichermaßen können Agenten aus dem System gelöscht werden, wenn der Akteur in der Wirklichkeit nicht mehr auftritt. Im Kontext einer Klinik können so Patienten, die aufgenommen werden durch Agenten abgebildet werden, die bei Entlassung des Patienten wieder aus dem MAS genommen werden. Das Optimierungsverfahren selbst ändert sich durch die variable Agentenanzahl nicht, da die dezentrale Entscheidungsfindung von der Anzahl der Agenten unabhängig ist.

Die physische Welt nehmen Agenten über Sensoren wahr und können sie über Effektoren beeinflussen. Agenten planen und agieren zielgetrieben. Ausgehend von einem Ziel beeinflussen sie die Wirklichkeit so, dass diese sich in Richtung des Ziels bewegt. Schrittweise kommt es zur Veränderung der Welt, die Aggregation der einzelnen Veränderungen führt zu einem neuen Zustand der Wirklichkeit, der näher am Idealzustand liegt als der vorherige.

Bezüglich des Endzustands erfüllen MAS keine Optimalitätskriterien; sie erreichen die bestmögliche Lösung eines Problems nicht notwendigerweise, sondern kommen ihr nur nahe. Der Grund dafür liegt in der lokalen Information der Agenten, die dazu führt, dass nicht jede Wechselwirkung von Information in die Planung mit einbezogen wird. Zentrale Verfahren, die mit vollständiger Information arbeiten, sind daher nach reinen Optimalitätskriterien den dezentralen Verfahren überlegen, bieten allerdings nicht die gleiche Flexibilität und Möglichkeit, in offenen Systemen zu agieren.

Softwareagenten kommunizieren mit der Umwelt in verschiedenen Stufen. Die System-Umwelt-Beziehung lässt sich in drei Stufen einteilen [Ramm2003]:

1. *Systemumweltbezogene Intelligenz*: Die Agenten arbeiten situiert und angepasst an ihre Umwelt. Allerdings agieren sie isoliert und ohne Kooperation. Ein Beispiel ist ein reaktiver Agent, der die Leistung einer Klimaanlage in Abhängigkeit von der Raumtemperatur reguliert.
2. *Kooperative Intelligenz*: Zusätzlich zur systemumweltbezogenen Intelligenz findet Kooperation von Agenten untereinander statt. Zur Problemlösung können sie zusammenarbeiten. Ein Beispiel sind die im Team spielenden Roboter in der Robocup-Meisterschaft.

3. *Sozialreflexive Intelligenz*: Die dritte Stufe der Intelligenz erreichen MAS, wenn sie nicht nur untereinander kooperieren, sondern zusätzlich mit Menschen interagieren und diese Interaktion in ihre Aktionen einbeziehen. Dazu müssen die Agenten neben der Möglichkeit, in der logischen Welt zu agieren, auch über geeignete Kommunikationskanäle zur physischen Welt verfügen, um ihre Entscheidungen mit Menschen abstimmen zu können. Eine Möglichkeit, diese Kommunikation zu realisieren, sind mobile Endgeräte wie PDAs oder Handys. Beide bieten bidirektionale Kommunikation zwischen Mensch und IS und ermöglichen so die Einbeziehung der menschlichen Entscheidung in die Planung.

UC baut auf der dritten Stufe der System-Umwelt-Beziehungen auf. Softwareagenten agieren dezentral und sind in der Lage, über geeignete Schnittstellen mit der physischen Welt mit Menschen zu kommunizieren. Die Intelligenz teilt sich dabei zwischen Mensch und Agent auf, so dass „hybride Intelligenz“ [Ramm2003] entsteht. Agenten können in der logischen Welt planen und agieren. Die Ergebnisse dieser Planungen geben sie über ihre Effektoren an die Menschen weiter. Deren Einfluss auf die Wirklichkeit erkennen die Softwareagenten durch die Sensorik und sie können die geänderten Umweltbedingungen in ihre zukünftigen Planungen mit einbeziehen. Abbildung 2-2 zeigt diesen Zusammenhang am Beispiel eines Terminplanungssystems in Krankenhausumgebungen: Die Beschäftigung einer Ressource wird durch die Lokalisierung mittels RFID erkannt. Im MAS werden die Ressourcen durch Softwareagenten repräsentiert, die den Status der ihnen zugeordneten Entität erkennen. Der Agent stellt fest, dass der geplante Terminplan nicht mehr eingehalten werden kann und initiiert in der logischen Welt Terminverhandlungen mit anderen Agenten. Das Ergebnis einer erfolgreichen Verhandlung ist ein veränderter Terminplan, den die Agenten an die Akteure in der physischen Welt kommunizieren. So beeinflussen die Agenten in der logischen Welt die Akteure in der physischen.

Die hybride Intelligenz stellt auf die Kombination beider Welten ab. Weder agieren die Agenten losgelöst von der Wirklichkeit noch agieren die Menschen ohne Unterstützung eines IS. Beide bedingen sich gegenseitig und ermöglichen so die effektivere Problemlösung.

4 EMIKA – Entscheidungsunterstützung im Krankenhaus

Den Einfluss von UC auf die Wirklichkeit untersucht das Projekt EMIKA (Echtzeitgestützte Mobile Informationssysteme in Krankenhausanwendungen). Im Projekt entsteht eine Software, die die Terminplanung im Krankenhaus unterstützt. Dabei kommen Technologien des UC zum Einsatz, die sowohl auf Ebene der allgegenwärtigen Sensorik als auch auf Ebene der Informationsverarbeitung angesiedelt sind. Als Modellierungs- und Implementierungsparadigma dient der Ansatz eines MAS, in dem Agenten über ökonomische Verhandlungen einen vorhandenen Terminplan optimieren.

Krankenhausumgebungen sind offene, dynamische Systeme, in denen sich jederzeit kurzfristige Änderungen ergeben können, aufgrund derer der Terminplan angepasst werden muss. Alle Charakteristika für offene Systeme treffen in Kliniken zu, so dass sie sich gut als Anwendungsfall für UC eignen. Insbesondere handelt es sich um Umgebungen, in denen immer wieder Akteure hinzukommen oder wegfallen. Jeder Patient im Krankenhaus entspricht einem Akteur. Da Patienten autonom agieren,

kommt es immer wieder zu „no-shows“ (Patienten, die nicht zu einer terminierten Untersuchung erscheinen) oder zu Notfällen, deren Behandlung keinen Aufschub duldet und sofort erfolgen muss. Andere, zu diesem Zeitpunkt terminierte Patienten können dann nicht untersucht werden.

Stattfindende Behandlungen können zeitlich variieren, wenn eine Behandlung wegen unvorhergesehener Komplikationen länger dauert als angenommen oder zusätzliche Ressourcen in Form von Ärzten oder Geräten beansprucht werden. Diese Veränderungen werden ad hoc, ohne vorherige Information des IS von menschlichen Akteuren entschieden. Die Ärzte, Schwestern und Pfleger sind außerdem mit der Pflege der Patienten beschäftigt, so dass sie diese Änderungen nicht explizit in das IS einpflegen können.

In Krankenhausumgebungen sind immer menschliche Entscheidungsträger verantwortlich. Jegliche informationsverarbeitenden Systeme können nur entscheidungsunterstützend arbeiten und Vorschläge für Entscheidungen generieren. Die Entscheidung selbst muss von einem Menschen getroffen werden, der sich für sie auch im Falle einer Fehlentscheidung verantworten muss. Das EMIKA-System generiert daher Terminvorschläge, der behandelnde Arzt muss diese Vorschläge dann explizit annehmen und sich so für oder (durch ex- oder implizite Ablehnung) gegen den Vorschlag entscheiden.

Das Projekt EMIKA befindet sich im Modellstadium, in dem die Effekte des UC in kleinem Maßstab untersucht werden können. Zu diesem Zweck existiert ein Miniaturmodell einer Klinik, in dem sich verschiedene Terminplan-Szenarien untersuchen lassen. Abbildung 4-1 zeigt den Grundriss des Modells mit den Positionen der RFID-Lesegeräte.

Das Modell berücksichtigt keine technischen Anforderungen auf niedrigem Level. EMIKA setzt RFID als einsetzbare und praktikable Technologie für die Lokalisierung und Identifizierung von Ressourcen im Krankenhaus voraus. Im Modellmaßstab bewährt sich die Technik, trifft dort allerdings auch nicht auf die der Technik eigenen Schwierigkeiten. Die von RFID verwendete Strahlung wird von Metall reflektiert und von Flüssigkeiten absorbiert [Fink2002]. Beide Materialien kommen in Krankenhäusern häufig zum Einsatz, so dass sich Flächen nicht immer uniform ausleuchten lassen. Diese Probleme blendet das Projekt jedoch bewusst aus, da das System nicht an die spezifische RFID-Technologie gebunden ist. Voraussetzung für die erfolgreichen Verhandlungen ist lediglich die verlässliche Lokalisierung der Akteure. Alternative Techniken wie die WLAN-Lokalisierung haben andere spezifische Vor- und Nachteile, können RFID jedoch ersetzen, falls sich RFID in Krankenhäusern aus technischen Gründen als nicht geeignet erweist.

Im System repräsentieren Softwareagenten die menschlichen Akteure. Die Akteure teilen sich dabei in die Gruppe der Patienten, die Leistungen nachfragen und in die Gruppe der Ärzte, Pfleger und Schwestern, die Leistungen anbieten. Zusätzlich können auch technische Geräte wie Instrumente für die Magnetresonanztherapie (MRT) oder für Elektroenzephalogramme (EEG) Leistungen anbieten, so dass sie mit der Gruppe der Anbieter zur Gruppe der Ressourcen zusammengefasst werden. Jede Ressource und jeder Nachfrager innerhalb des Systems wird von einem Softwareagenten repräsentiert, der in der logischen Welt den eigenen Terminplan optimiert.

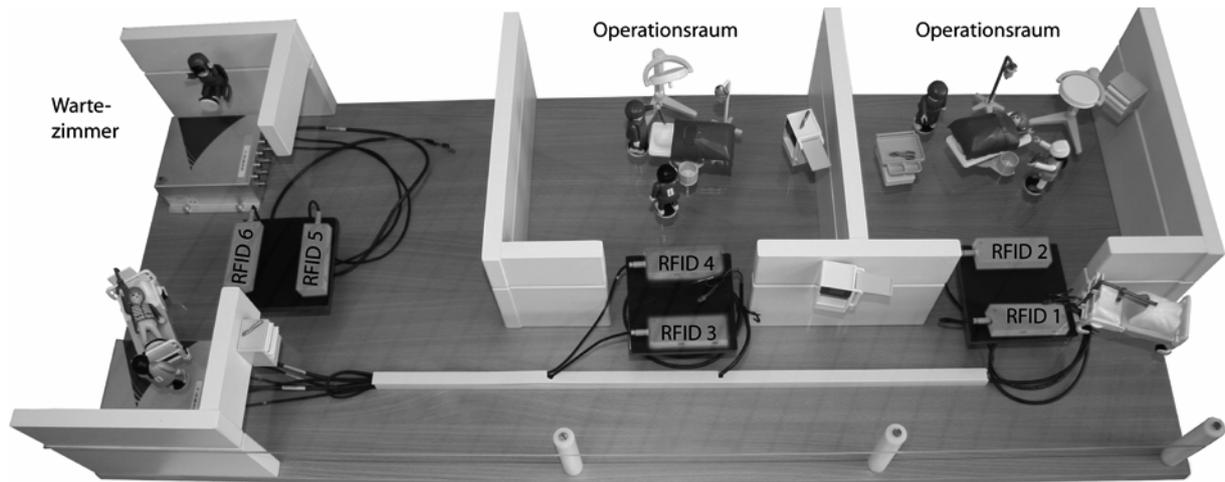


Abbildung 4-1: Grundriss des EMIKA-Modells. RFID 1–RFID 6 zeigen die Positionen der Lesegeräte

Entsprechend des Feedback-Loops *Sensoren*, *Verarbeitung* und *Effektoren* gliedert sich dieser Abschnitt in die Sensoren als Schnittstelle zwischen physischer und logischer Welt, die Verarbeitung auf Grundlage eines künstlichen Marktes und die Effektoren als Schnittstelle zwischen logischer und physischer Welt.

4.1 Sensoren

Die Sensorik erfolgt in EMIKA auf unterster Ebene durch RFID-Technologie. In den nachfolgenden Verarbeitungsschichten reichert EMIKA die Sensordaten durch weitere Information an, die im System erfasst ist und versieht sie mit Bedeutung.

Um Entscheidungsunterstützung zu ermöglichen, sind mehrere Kontextmerkmale notwendig, die sich aus der Kombination der Sensordaten mit im System hinterlegter Information ergeben.

4.1.1 Position und Identität

Zur Bestimmung der Position und der Identität kommt RFID zum Einsatz. Jeder menschliche Akteur trägt einen Transponder bei sich (z. B. als Patientenarmband oder eingebettet im elektronischen Heilberufsausweis), der den Träger eindeutig identifiziert. Die RFID-Lesegeräte sind an Ortsübergängen angebracht, so dass jeder Wechsel zwischen Räumen Leseereignisse auslöst.

Ein Leseereignis allein lässt noch keinen Schluss auf die Bewegungsrichtung zu. Diese kann entweder durch Hinzunahme der vorherigen Position erschlossen werden oder durch weitere Sensorik festgestellt werden. Zusätzliche Sensorik ist dabei die robustere Variante, da sie Probleme umgehen, die entstehen, wenn eine Bewegung nicht erfasst wird und die logische Welt nicht konsistent zur physischen ist [HNEM2004]. Ein Ansatz ist die Ausstattung der Entitäten mit zwei oder mehr Transpondern [HNEM2004]. Die Transponder bewegen sich dann nacheinander durch das Feld. Aus der zeitlichen Verzögerung kann das IS die Bewegungsrichtung herleiten. Eine weitere Möglichkeit ist die Duplizierung der Lesegeräte. Die Auftrittsreihenfolge der Leseereignisse bestimmt dann die Bewegungsrichtung. Die

Methode ist kostspieliger, hat allerdings den Vorteil, dass die Lesegeräte in jedem Fall nacheinander auslösen, da ihre Positionen fixiert sind. Die Verwendung von mehreren Chips ist dagegen fehleranfällig, da sie parallel durch das Feld bewegt werden können. Dies kann nur durch weitere Chips ausgeglichen werden, was allerdings den Verarbeitungsaufwand erhöht. EMIKA arbeitet mit zwei Lesegeräten pro Tür, so dass sich die Bewegungsrichtung erkennen lässt. Die zeitliche Reihenfolge der Leseereignisse legt fest, ob ein Akteur einen Raum betritt oder verlässt.

Die Lesegeräte identifizieren anhand der Seriennummer des Transponders die Identität der Entität und informieren den zugeordneten Agenten mittels einer Nachricht über das Leseereignis. Der Agent speichert dieses Ereignis zwischen. Nur wenn sich die Entität innerhalb einer sehr kurzen Zeitspanne (5 Sekunden) auch über das zweite Lesegerät der Tür bewegt, schließt der Agent aus der Kombination beider Ereignisse auf eine Bewegung zwischen zwei Räumen. Wenn kein korrespondierendes zweites Signal auftritt, verwirft der Agent das zwischengespeicherte Signal und registriert keine Bewegung. Eine Bewegung wird an alle Agenten, deren Terminpläne betroffen sind, weitergegeben, so dass diese ihre Sicht der Welt anpassen können. Falls erforderlich, können Agenten die Positionen anderer Agenten bei den jeweiligen Agenten nachfragen.

4.1.2 Behandlungs- und Terminpläne

Die Patienten-Agenten haben Zugriff auf den Behandlungsplan des Patienten, den sie vertreten. Dieser Plan liefert die Grundlage ihrer Planungen und Aktionen. Sie gleichen ihn mit der Position der benötigten Ressourcen ab und ergreifen, wenn die Pläne nicht mit der Wirklichkeit übereinstimmen, entsprechende Maßnahmen.

Akteure, die Leistungen im Krankenhaus zur Verfügung stellen, werden durch die Ressourcenagenten repräsentiert. Unter diese Gruppe fallen sowohl Ärzte, Pfleger und Schwestern als auch Maschinen oder Räume (die Ressourcen in Form von Raumnutzungsrechten anbieten. Ressourcenagenten können auf ihren Terminplan zugreifen, in dem die geplanten Behandlungen mit allen dafür vorgesehenen Ressourcen eingetragen sind. Grundsätzlich können die Agenten nur auf jeweils ihren eigenen Termin- oder Behandlungsplan zugreifen und auch nur sie selbst dürfen ihn verändern. Bei den Plänen handelt es sich um private Information.

Behandlungs- und Terminpläne setzen sich aus dem Behandlungsnamen, dem geplanten Zeitraum (Anfangs- und Endzeit), dem Behandlungsraum und allen beteiligten Ressourcen zusammen.

4.2 Verhandlungen zwischen den Agenten auf Basis eines künstlichen Marktes

Die Optimierungsschicht bildet in EMIKA ein MAS, welches die Koordinierung der Terminpläne über einen marktbasierten Mechanismus realisiert: Um Fehler in Terminplänen zu beheben, verhandeln die Softwareagenten über die Behandlungstermine. Auf einem künstlichen Markt treffen sich die Agenten, um dort über Ressourcennutzung zu verhandeln. Patienten fragen Leistungen nach, Ärzte, Schwestern, Pfleger und Räume bieten Leistungen an. Patienten als Nachfrager verfügen über ein Budget, mit dem sie Leistungen auf dem Markt einkaufen können, die

Ressourcen haben verschiedene Preise, die ihre Verfügbarkeit signalisieren. Die Patienten suchen so nach einer Lösung für ein Koallokationsproblem, in dem sie unterschiedliche Ressourcen für einen bestimmten Zeitraum zur gleichen Zeit in Anspruch nehmen müssen.

Die Währung, mit der im System verhandelt wird, ist ein Geldsubstitut, das die zur Terminplanung notwendigen Informationen in Geldeinheiten umrechnet über die dann verhandelt werden kann.

Die wichtigste Einflussgröße für das Budget der Patienten, mit dem sie auf dem Markt ihre Behandlungen einkaufen, ist die Wartezeit. Wenn Patienten auf ihre Behandlung warten müssen, steigt ihr Budget. Dieser Anstieg verhindert, dass Patienten nie behandelt werden, wenn andere Patienten ein größeres Budget zur Verfügung haben. Der zweite Einflussfaktor auf das Budget ist die Diagnose der Patienten. Bei der Aufnahme eines neuen Patienten im Krankenhaus wird die erste Diagnose erstellt und der behandelnde Arzt legt fest, wie dringlich die weitere Behandlung ist. Die Dringlichkeit drückt sich in der Höhe des Anfangsbudgets aus. Einem Notfall kann der Arzt bei Aufnahme ein hohes Budget zuweisen, so dass der Patientenagent in die Lage versetzt wird, alle benötigten Ressourcen sofort auf dem Markt einzukaufen. Einem Patient mit niedriger Dringlichkeit wird dagegen ein niedriges Anfangsbudget zugewiesen. Er muss je nach Auslastung der Klinik unter Umständen Wartezeit in Kauf nehmen, bis sein Budget das aktuelle Preisniveau der spezifischen Behandlung erreicht.

Anbieterressourcen verfügen über einen variablen Preis, der sich aus ihrer Auslastung ergibt. Sind sie momentan beschäftigt, so steigt ihr Preis an, sind sie gerade nicht beschäftigt, so sinkt ihr Preis. Dem liegt die Annahme zu Grunde, dass Ärzte, die gerade eine Behandlung durchführen, nicht aus dieser Behandlung „herausgekauft“ werden sollen. Je länger die Behandlung andauert, desto schwieriger soll es werden, diesen Ärzten eine andere Behandlung zuzuweisen. In der Realität findet eine solche Verschiebung von Ärzten aus einer laufenden Behandlung nicht statt. Diese Annahme schwächt EMIKA ab, indem dieser Fall lediglich erschwert, aber nicht verhindert wird. Ein Verbot des Kaufs eines behandelnden Arztes lässt sich ebenfalls über den Preismechanismus realisieren: Bei Behandlungsbeginn stiege dann der Preis sprunghaft an und bliebe während der Behandlung auf hohem Niveau konstant.

Nach Ende der Behandlung sinkt der Preis langsam wieder. Ärzte sollen nach Behandlungen möglichst erst eine Ruhephase durchlaufen, in der ihr Preis wieder auf Marktniveau absinkt, bevor sie eine neue Behandlung durchführen. Ist die Nachfrage im Gesamtsystem allerdings gerade sehr hoch, so kann es sein, dass die Ärzte direkt die nächste Behandlung vorgeschlagen bekommen, da das Preisniveau gestiegen ist. Je nach Angebot und Nachfrage stellen sich so unterschiedliche Gleichgewichte ein. Bei Nachfrage- oder Angebotsüberhängen steigt bzw. sinkt das Preisniveau. Eine genaue Beschreibung des verwendeten künstlichen Marktes liefert [NiEy2006].

Das in EMIKA verwendete Geldsubstitut entspricht nicht der tatsächlichen Bezahlung der Behandlung durch die Patienten bzw. Krankenkassen an die Klinik. Das Geldsubstitut dient nur der Koordination und Terminplanung. Die eigentliche Abrechnung erfolgt davon unabhängig nach Fallpauschalen und ist nicht Teil des Projekts.

4.3 Effektoren

Die Agenten sind in der Lage, die physische Welt über Effektoren zu verändern. Da im Krankenhaus Menschen die Entscheidungsverantwortung tragen und sich diese nicht ohne weiteres auf IS übertragen lässt, generieren die Agenten Vorschläge, die sie an die menschlichen Akteure kommunizieren. Technische Geräte haben keine Entscheidungsmöglichkeiten, so dass zwischen ihnen und den sie repräsentierenden Agenten keine Kommunikation stattfindet. Die Effektoren beschränken sich in EMIKA daher auf die Schnittstellen zwischen Agent und Mensch.

Im Projekt EMIKA kommen PDAs zur Kommunikation mit Ärzten zum Einsatz, auf denen ihr aktueller Terminplan angezeigt wird. Kommt es zu einer Veränderung im Terminplan, so erscheint diese auf dem PDA und der Arzt kann die Veränderung bestätigen oder ablehnen. Im Sinne einer „calm technology“ wird dabei auch das Nichtbeachten einer Meldung innerhalb einer Zeitspanne als implizite Ablehnung gewertet. Zu Grunde liegt die Annahme, dass Ärzte, die nicht auf Meldungen reagieren, beschäftigt sind und dem PDA keine Aufmerksamkeit widmen können.

Stehen keine mobilen Endgeräte mit Rückkanal zur Verfügung, so können auch Endgeräte zum Einsatz kommen, die den Arzt nur über eine Veränderung informieren. Als Annahme oder Ablehnung des Vorschlags kann dann die Bewegung innerhalb des Krankenhauses gewertet werden. Betritt ein Arzt nach Vorschlag eines Behandlungstermins in Operationssaal 1 eben diesen Behandlungsraum, so wird diese Bewegung als Annahme gewertet und der Agent geht davon aus, dass der Arzt die Behandlung durchführt. Verlässt der Arzt den Raum wieder, ohne die Behandlung abgeschlossen zu haben, so entsteht eine Komplikation im Terminplan, die der Agent des Patienten durch Neuverhandlung des Termins zu beheben versucht. Echte Kommunikation mittels PDAs ist dieser Variante auf jeden Fall vorzuziehen; sie dient nur als Möglichkeit der Kommunikation, wenn keine bidirektionale Kommunikation zwischen Arzt und Agent über die mobilen Endgeräte möglich ist.

Patienten können ebenfalls über mobile Endgeräte mit dem MAS kommunizieren. Als Möglichkeiten eignen sich hier Handys, über die SMS an den Patienten geschickt werden können.

5 Zusammenfassung und Ausblick

Das EMIKA-System implementiert die Optimierung der Terminpläne bislang auf Ebene der Substitution einer Ressource. Wenn die Behandlung eines Patienten länger dauert als geplant, erkennen die Softwareagenten, dass der Arzt nachfolgende Behandlungen nicht wahrnehmen kann und substituieren den Arzt mittels Neuverhandlungen durch den unter den veränderten Bedingungen optimal geeigneten Arzt. Die Optimalitätskriterien beziehen sich dabei nur auf Variablen, die Eingang in das Geldsubstitut als Grundlage der Verhandlung gefunden haben.

Die Preise und Budgets der Softwareagenten entwickeln sich den Erwartungen entsprechend: Je nach Nachfrage- und Angebotsmenge bewegt sich das Preisniveau des Gesamtmarktes.

Einzelne Ressourcen lassen sich durch das System effizient ersetzen und wegen der einfachen Entscheidungsregeln über marktbasierter Verhandlungen verläuft die Umlanung des Gesamtsystems in Zeitabständen, die im Anwendungsfeld Kranken-

haus Echtzeitbedingungen bieten. Eine Untersuchung der Skalierungsfähigkeit des Systems kann auf Modellebene nicht erfolgen, da die hybride Intelligenz viele menschliche Eingriffe erfordert, die sich nicht in großem Maßstab simulieren lassen. Die marktbasierende Verhandlung eignet sich als Koordinationsmechanismus, der die Abstimmung der Agenten aufeinander ermöglicht. Durch Einkauf und Verkauf von Ressourcen führt UC neben dem Primäreffekt der Allgegenwärtigkeit und der smarten Dinge zum Sekundäreffekt der Selbstorganisation [MKS+2003].

EMIKA ermöglicht in der aktuellen Entwicklungsstufe die Substitution einer Ressource. Ein Arzt kann in einem Behandlungstermin ersetzt werden, den Termin selbst betrachtet das System als nicht veränderbar. Falls für einen Termin keine geeignete Ersatzressource gefunden wird, verfällt der Termin. Dieses Manko kann ausgeglichen werden, wenn die Agenten nicht nur über die Substitution einer einzelnen Ressource verhandeln, sondern auch Behandlungstermine ersetzen dürfen. Um das zu ermöglichen, müssen allerdings weitere Randbedingungen berücksichtigt werden. Die Behandlungspläne einzelner Patienten sind zwar weitestgehend spezifiziert, da Krankenhäuser nach standardisierten klinischen Behandlungspfaden behandeln, die als Vorschlag die „best practices“ der Behandlung bei bestimmten Diagnosen enthalten. Innerhalb der Pfade treten allerdings Bedingungen auf, die der Verschiebung von Behandlungsschritten enge Grenzen setzen. Bestimmte Behandlungen müssen in festgelegten Zeitabständen nach anderen Teilschritten erfolgen, andere setzen einen bestimmten Teilschritt als zwingend voraus. Durch die mögliche Verschiebung von Behandlungsterminen stellt sich die Terminplanung als durch viele Nebenbedingungen eingeschränkte Geschäftsprozessoptimierung dar. Die Erweiterung der Planung auf diese Ebene ist notwendig, da das System erst dann absichern kann, dass Behandlungstermine nicht verfallen.

EMIKA ermöglicht die nahezu vollständig autonome Veränderung der Terminpläne durch das MAS. Agenten handeln weitestgehend selbständig, so dass die Menschen nur in den Entscheidungsprozess einbezogen werden, wenn der Agenten auf explizite Rückmeldungen aus der physischen Welt angewiesen ist. Durch die Softwareagenten entsteht so die von Weiser geforderte „calm technology“. Demgegenüber steht allerdings die Bevormundung der Menschen durch die Technik. Wie ruhig und autonom die Software handeln sollte, ohne dass die Menschen das Vertrauen in die Technik verlieren, kann erst die Simulation in größeren Maßstäben zeigen.

Literaturverzeichnis

- [AbMy2000] Abowd, G. D.; Mynatt, E. D.: Charting Past, Present, and Future Research in Ubiquitous Computing. In: ACM Transactions on Computer-Human Interaction 7 (2000), S. 29-58.
- [Dey2001] Dey, A. K.: Understanding and Using Context. In: Personal and Ubiquitous Computing, 5 (2001), S. 4-7.
- [Eyma2002] Eymann, T.: Digitale Geschäftsagenten. Springer: Berlin, 2002.
- [ESMü2003] Eymann, T.; Sackmann, S.; Müller G.: Hayeks Katallaxie – Ein zukunftsweisendes Konzept für die Wirtschaftsinformatik?. In: Wirtschaftsinformatik 45 (2003), S. 491-496.

- [Fink2002] Finkenzeller, Klaus: RFID-Handbuch. Hanser: München, 2002.
- [Hase2005] Haseloff, S.: Context Awareness in Information Logistics. Dissertation Technische Universität Berlin, Fakultät IV – Elektrotechnik und Informatik, 2005.
- [Hewi1998] Hewitt C.: Offices Are Open Systems. In: Bond, A. H.; Gasser, L. (Hrsg.): Readings in Distributed Artificial Intelligence. Morgan Kaufman Publishers: San Mateo, 1998, S. 321-329.
- [HNEM2004] Hohl, A.; Nopper, N.; Eymann, T.; Müller, G.: Automatisierte und Interaktive Kontextverarbeitung zur Unterstützung der Patientenlogistik. In: Kirn, S.; Anhalt, C.; Heine, C. (Hrsg.): Mobiles Computing in der Medizin – Proceedings zum 4. Workshop der GMDS-Projektgruppe Mobiles Computing in der Medizin, Universität Hohenheim, Stuttgart, 19.-20. April 2004. Shaker Verlag: Aachen, 2004, S. 14-26.
- [Kirn2006] Kirn, S.: Flexibility of Multiagent Systems. In: Kirn, S.; Herzog, O.; Lockemann, P.; Spaniol, O. (Hrsg.): Multiagent Engineering – Theory and Applications in Enterprises. Springer: Berlin, 2006, S. 53-69.
- [MFGG2003] Munoz, M. A.; Favela, J.; Garcia, A. I. M.; Gonzalez, V. M.: Context-Aware Mobile Communication in Hospitals. In: Computer 36 (2003), S. 38-46.
- [MKS+2003] Müller, G.; Kreutzer, M.; Strasser, M.; Eymann, T.; Hohl, A.; Nopper, N.; Sackmann, S.; Coroama, V.: Geduldige Technologie für ungeduldige Patienten – Führt Ubiquitous Computing zu mehr Selbstbestimmung? In: Mattern, F.: Total Vernetzt. Springer: Berlin, 2003. S. 159-186.
- [NiEy2006] Niemann, C.; Eymann, T.: Softwareagenten in der Krankenhauslogistik – ein Ansatz zur effizienten Ressourcenallokation. In: HMD: Praxis der Wirtschaftsinformatik, 251 (2006), S. 77-87.
- [Prad2000] Pradhan, S.: Semantic Location. In: Personal and Ubiquitous Computing, 4 (2000), S. 213-216.
- [Ramm2003] Rammert, W.: Die Zukunft der künstlichen Intelligenz: verkörpert – verteilt – hybrid. Working Paper TUTS-WP-4-2003, Technische Universität Berlin, 2003.
- [Weis1991] Weiser, M.: The Computer for the 21st Century. In: Scientific American, 265 (1991), S. 66-75.
- [Weis1996] Weiser M.: Some computer science issues in ubiquitous computing. In: Communications of the ACM, 36 (1991), S. 74-84.
- [ZREy2006] Zwicker, F.; Reiher, M.; Eymann, T.: RFID im Krankenhaus – neue Wege in der Logistik. In: Eymann, T.; Hampe, J. F.; Koop, A.; Niemann, C. (Hrsg.): Mobiles Computing in der Medizin – Proceedings zum 6. Workshop der GMDS-Arbeitsgruppe Mobiles Computing in der Medizin, Frankfurt, 2. Juni 2006. Shaker Verlag: Aachen, 2006.

Softwareagenten im Wissensmanagement

Volker Nissen; Mathias Petsch

Technische Universität Ilmenau
PF 100565; 98694 Ilmenau
volker.nissen|mathias.petsch@tu-ilmenau.de

Zusammenfassung: In diesem Beitrag wird der Frage nachgegangen, welche Einsatzmöglichkeiten sich für die Agententechnologie im Wissensmanagement bieten. Ausgangspunkt sind einerseits die Aufgaben und Anforderungen des Wissensmanagements, andererseits die besonderen Eigenschaften von Softwareagenten. Über die heute bestehenden, noch relativ wenigen Anwendungsbeispiele hinaus, werden einige zukünftige Einsatzpotenziale sowohl für intelligente Agenten als auch für einfach strukturierte Agenten im Kontext von „Schwarm Intelligenz“ identifiziert.

1 Einleitung

1.1 Begriff des Wissens

In der Literatur finden sich unterschiedlichste Definitionen des Begriffes „Wissen“. Hilfreich für eine Abgrenzung ist die bei Krcmar und Rehäuser formulierte Begriffshierarchie (siehe Abb. 1).

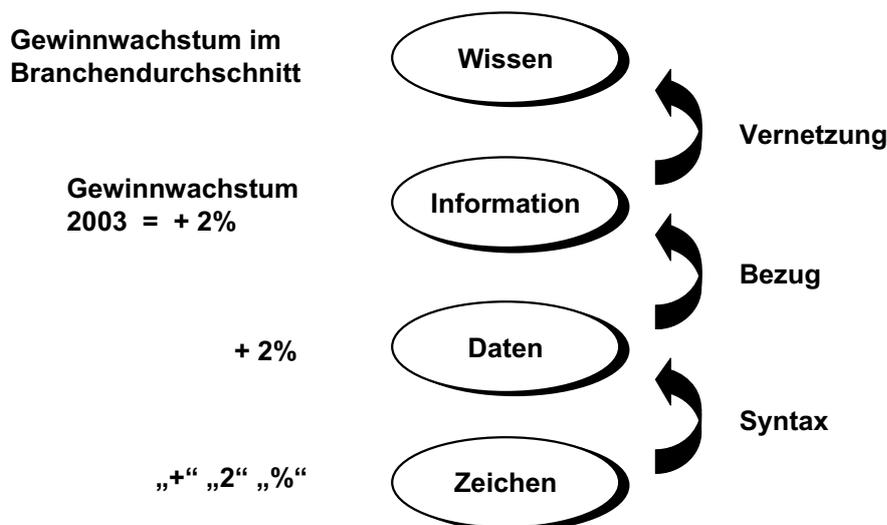


Abbildung 1: Begriffshierarchie (angelehnt an [ReKr1996, 6])

Auf der untersten Hierarchieebene stehen Zeichen, die einem definierten Zeichenvorrat (z.B. allen Buchstaben und Ziffern) entstammen und zunächst zusammenhanglos nebeneinander stehen. Verknüpft man Zeichen mit Hilfe von Ordnungsregeln (Syntax, Code) so entstehen Daten. Informationen dienen im betriebswirtschaftlichen Kontext der Vorbereitung von Entscheidungen oder Handlungen [Nort2002, 38]. Durch Herstellen eines Bezuges werden Daten zu Informationen. Aus den Daten „+ 2 %“ kann beispielsweise durch Bezug zum Unternehmensgewinn die Information entstehen, dass ein positives Gewinnwachstum vorliegt. Wissen entsteht durch zweckorientierte Vernetzung von Informationen. Verbindet man zum Beispiel das Gewinnwachstum von „+ 2 %“ mit der Information des durchschnittlichen Gewinnwachstums der Branche im betrachteten Geschäftsjahr von 3,5%, so weiß man nun, das beim betrachteten Unternehmen ein unterdurchschnittliches Wachstum des Gewinnes vorliegt und wird sich der Ursachenforschung zuwenden. Dieses Wissen konnte nur durch Vernetzung verschiedener Informationen gebildet werden. Genauso kann auch die Vernetzung einer Information mit bereits vorhandenem Wissen zu neuem Wissen führen.

Je nach Differenzierungsziel können verschiedenste Formen von Wissen unterschieden werden. In Abb. 2 sind einige Beispiele aufgeführt.

Kriterium	Unterscheidung
Verbreitung	individuelles - kollektives Wissen
Transferierbarkeit	explizites - implizites Wissen
Spezifität	Allgemein-, Branchen-, Berufs-, Firmenwissen
Ursprung	Erfahrung - rationale Überlegung
Ort	externes - internes Wissen
Wichtigkeit	Kritisches Erfolgswissen - peripheres Wissen

Abbildung 2: Beispielhafte Wissenskategorien

Besonders bedeutsam ist die Differenzierung nach der Transferierbarkeit des Wissens. *Explizites Wissen*, auch deklaratives Wissen genannt, liegt in kodifizierter Form auf Medien vor oder kann leicht in diese Form gebracht werden, beispielsweise als Text, Formel oder Zeichnung und lässt sich somit problemlos kommunizieren. *Implizites Wissen* hingegen entzieht sich der Kodifizierung und Kommunikation weitgehend. Es wird auch als prozedurales Wissen bezeichnet und beruht wesentlich auf persönlichen Erfahrungen und früheren Handlungen. Schüppel spricht im Zusammenhang mit implizitem Wissen von „Tiefenwissen“, das die Prozesse der Realitätswahrnehmung steuert und so die Handlungen und Verhaltensweisen bestimmt [Schü1995, 195 f.].

Wissen wird immer häufiger als eigenständiger Produktionsfaktor betrachtet.¹⁵ In vielen Branchen kommt Wissen als knappem Gut eine Schlüsselrolle bei der Wertschöpfung zu und substituiert dabei klassische Faktoren wie menschliche Arbeit und Kapital [SMSG1998, 224]. Andererseits nutzen viele Unternehmen nur einen Bruchteil des in der Organisation vorhandenen Wissens.¹⁶ Dieses Defizit hat in den vergangenen Jahren dazu geführt, dass eigene Managementtechniken für Wissen als Produktionsfaktor entwickelt wurden.

1.2 Begriff und Strategien des Wissensmanagements

Ebenso vielfältig wie die Vorstellungen vom Wissen sind auch die Definitionen des Wissensmanagements.¹⁷ Wissensmanagement soll hier verstanden werden als die gezielte Beeinflussung aller Unternehmensaspekte, bei denen das Ziel im Mittelpunkt steht, Wissen als Produktionsfaktor optimal zu nutzen.

Wissensaufbau und Wissensmanagement müssen letztendlich den Geschäftszielen des Unternehmens dienen. Als Erfolge von Wissensmanagement werden beispielsweise genannt:¹⁸

- schnellere Anpassung an veränderte Kundenwünsche (höhere Kundenzufriedenheit),
- Kosten- und Zeitersparnis,
- Qualitätsverbesserungen (in Produkten und Prozessen),
- gestiegene Mitarbeiterzufriedenheit,
- Wettbewerbsvorteile durch besseren Wissenstransfer und höhere Innovationsfähigkeit,
- größere Effizienz durch Vermeiden früherer Fehler in vergleichbaren Situationen,
- Sicherung von Wissen für das Unternehmen bei Mitarbeiterfluktuation.

Jedoch ist der Zusammenhang zwischen Aktivitäten des Wissensmanagements und geschäftlichen Erfolgen beziehungsweise Unternehmenszielen schwer exakt zu quantifizieren, da andere Einflussfaktoren ebenfalls wirken. Dies stellt ein Grundproblem bei der Rechtfertigung von Investitionen in Wissensmanagement dar. Die Notwendigkeit, Wissen als wichtige Unternehmensressource zu betrachten und entsprechend zu handeln, wird aber kaum noch bestritten. Ein Kernelement bildet dabei die verfolgte Wissensmanagement-Strategie.

Unternehmen verwenden grundsätzlich zwei verschiedene Wissensmanagement-Strategien [HNTi1999,106-110]. Die *Kodifizierungsstrategie* zielt darauf ab, möglichst

15 Als Produktionsfaktor werden Güter bezeichnet, die der Produktion anderer Güter dienen und daher das quantitative und qualitative Potenzial des Unternehmens zur Leistungserstellung beinhalten.

16 Die Angaben zur Höhe des genutzten Anteils schwanken in der Literatur stark, liegen aber im Durchschnitt unter 50%. Siehe hierzu stellvertretend [SMSG1998, 223].

17 Für einen tabellarischen Überblick verschiedener Definitionen des Wissensmanagements siehe [BoBe2003, 41 – 43].

18 vgl. hierzu [Herb2000, 27], [Skyr1998, 112] und [Niss2003, 55].

viel Wissen explizit zu machen und in Datenbanken und Dokumentenmanagementsystemen zu speichern, damit es jedem Unternehmensmitglied problemlos zur Verfügung steht, um wieder verwendet zu werden. Bei der *Personalisierungsstrategie* wird Wissen als eng an die Person gebunden betrachtet, die es geschaffen hat. Wissen wird hauptsächlich durch direkte Interaktion von Personen weitergegeben. Die Rolle der IT besteht hauptsächlich darin, Dialoge und somit den Austausch von implizitem Wissen zu fördern.

Von besonders großer Bedeutung und gleichzeitig hoher Komplexität ist das Thema Wissensmanagement bei wissensintensiven, interorganisationalen Kooperationen (Wissensnetzwerken).

1.3 Wissensnetzwerke

Ein Wissensnetzwerk kann als *„Zusammenarbeit zwischen mehr als zwei wissenschaftlichen und/oder wirtschaftlichen Akteuren (...) mit dem Ziel, strategische Vorteile zu erlangen durch Produktion und/oder Nutzung und/oder Diffusion von vorhandenem und neuem Wissen“* [KrGr2004, 111] verstanden werden. Es handelt sich in erster Linie um wissensorientierte Formen der Kooperation.

Seufert et al. verstehen unter einem Wissensnetzwerk in ähnlicher Art eine *„Anzahl von miteinander verbundenen Personen, Ressourcen, und Beziehungen, (...) welche zusammengezogen werden, um Wissen zu sammeln, zu generieren, zu teilen und zu nutzen“* [Seuf2002, 140]. In Wissensnetzwerken findet also durch die Diffusion von altem Wissen und „Produktion“ von neuem Wissen ein Prozess des interaktiven Lernens zwischen individuellen Wissensträgern statt, bei dem Teile des impliziten Wissens expliziert und übertragen werden.

Wissen liegt heute in der Regel in fragmentierter Form über verschiedene Wissensträger verteilt vor. Es wird von spezialisierten Wissensproduzenten weiterentwickelt und ist häufig nicht ohne weiteres kompatibel und transformierbar [Bick2004, 1]. Hierin liegt ein wichtiger Grund für das Entstehen von Wissensnetzwerken. Die Zusammenführung der einzelnen Wissensfragmente zu einer sinnvoll strukturierten Gesamteinheit verlangt eine sorgfältig organisierte Netzwerkzusammensetzung und stellt hohe Anforderungen an das Wissensmanagement. Die Herausforderung ist umso größer, aus je mehr verschiedenen Organisationsstrukturen die Akteure stammen und mit verschiedenen Perspektiven und Methoden an die Lösung ihrer jeweiligen Teilprobleme herangehen [Bick2004, 1].

2 Intelligente Agenten

Die Entwicklung der Informationstechnologie der letzten Jahre, die sich insbesondere durch die zunehmende Vernetzung und die damit einhergehende Verteilung von Wissen und Ressourcen manifestiert, ermöglichte starke Veränderungen der Organisation und Gestaltung von weltweit verteilten Geschäftsprozessen. Neue Organisationsformen, wie virtuelle Unternehmungen [AFMS1995] und dynamische Netzwerke [MiSn1986] sowie die damit zusammenhängenden internationalen Koordinations- und Kooperationsbeziehungen, bieten jedoch auch laufend neue Herausforderungen für die Gestaltung der Informationstechnologie. Seit Ende der achtziger Jahre hat sich das Forschungsgebiet der Verteilten Künstlichen Intelligenz (VKI)

entwickelt, das interessante Perspektiven für betriebswirtschaftliche Anwendungen bietet. Zur VKI zählen insbesondere Intelligente Agenten und Multiagentensysteme (MAS).

Der Begriff des Intelligenten (Software-)Agenten ist in der Literatur nicht eindeutig definiert [FrGr1996]. Verschiedene Eigenschaften werden damit assoziiert, wie u.a. Reaktivität, Autonomie, Proaktivität, Kontinuität, Kommunikationsfähigkeit, Adaptivität, Flexibilität und „Persönlichkeit“ [MuJo2000, 29]. Grundsätzlich können Agenten beschrieben werden als: „... a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives“ [Wool2002, 15]. Für den weiteren Verlauf der Arbeit wird die Definition von Wooldridge und Jennings [WoJe1995, 1] zugrunde gelegt, wonach Intelligente Agenten durch folgende Eigenschaften bestimmt sind:

- Autonomie (Agenten agieren ohne direkten Eingriff eines Benutzers und haben Kontrolle über ihre eigenen Aktionen),
- Reaktivität (Agenten nehmen ihre Umwelt sowie deren Dynamik wahr und reagieren auf dort eintretende Ereignisse oder Zustandsänderungen),
- Soziale Fähigkeiten (Agenten sind in der Lage, mit anderen Agenten zu kommunizieren, interagieren bzw. kooperieren),
- Proaktivität (Agenten agieren zielgerichtet und reagieren nicht nur auf Veränderungen der Umwelt).

Insbesondere die Eigenschaften der Autonomie und der Proaktivität unterscheiden Agenten von herkömmlicher Software. Intelligente Agenten sollen über explizite Ziele verfügen, die es ihnen ermöglichen, nicht nur einfach auf Veränderungen der Umwelt zu reagieren, sondern vorausschauend zu handeln. Es werden folglich zielorientierte Pläne generiert, die ein autonomes und zielgerichtetes Handeln unterstützen. Wesentlich dabei ist, dass ein Agent seine Ziele und Pläne an die sich ändernden Umweltbedingungen anpassen kann. Intelligente Agenten können als selbständige Akteure handeln oder aber in so genannten Multiagentensystemen auftreten. Multiagentensysteme bestehen aus mindestens zwei Agenten, die über Kommunikation ihr Verhalten mit dem Ziel der gemeinsamen Lösung einer Problemstellung koordinieren [BoGa1988, 3].

3 Anwendungsmöglichkeiten von Softwareagenten im Wissensmanagement

3.1 Aufgaben und Anforderungen des Wissensmanagements

Probst und Mitarbeiter haben ein ganzheitliches, praxisorientiertes Konzept zum Wissensmanagement entworfen, das sich aus mehreren Bausteinen (= Handlungsfeldern) zusammensetzt (Abb. 3).¹⁹ Es dient hier als Grundlage, um die wesentlichen Aufgaben und die daraus resultierenden Möglichkeiten an eine Unterstützung durch Softwareagenten zu identifizieren.

¹⁹ vgl. zum Folgenden [PRRo1999]. Für eine ähnliche Aufteilung in Wissensbausteine beziehungsweise Kernprozesse siehe auch [Nort2002, 79–154].

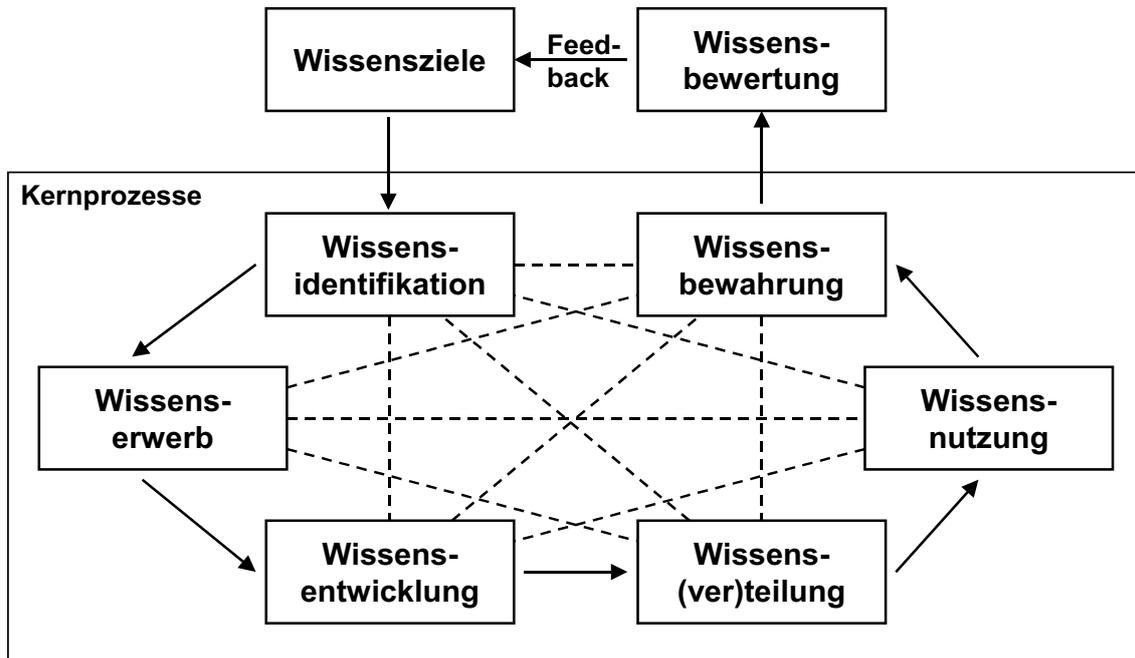


Abbildung 3: Bausteine des Wissensmanagements (nach [PRRo1999, 58])

Die Handlungsfelder sind alle mehr oder weniger eng miteinander verbunden, was dazu führt, dass Maßnahmen in einem Bereich sich auf andere Handlungsfelder ebenfalls auswirken. Folgende sechs Kernprozesse des Wissensmanagements werden unterschieden:

1. Wissensidentifikation

Dieser Prozess führt durch Analyse und Beschreibung des Wissensumfeldes eines Unternehmens zu mehr Transparenz über internes und externes Wissen. Dadurch sollen Ineffizienzen und schlecht fundierte Entscheidungen vermieden werden. Dazu gehört auch herauszufinden, wo Wissenslücken und Fähigkeitsdefizite bestehen. Probst und Mitarbeiter betonen aber, dass keine Transparenz um jeden Preis anzustreben ist, sondern die Anstrengungen sich auf „kritische Fähigkeiten der Organisation“ konzentrieren sollten.

2. Wissenserwerb

Hierbei steht der Einkauf von Wissen, das ein Unternehmen nicht aus eigener Kraft entwickeln kann oder will im Vordergrund. Dies kann auf der personellen Ebene zum Beispiel dadurch geschehen, dass neue Mitarbeiter mit dem benötigten Spezialwissen eingestellt und dazu gegebenenfalls von der Konkurrenz abgeworben werden. Ein Weg, benötigtes Wissen auf organisationaler Ebene zu erwerben besteht unter anderem darin, sich durch Kooperationen einen Zugang zu den Wissensbasen anderer Firmen zu sichern.

3. Wissensentwicklung

Komplementär zum Erwerb externen Wissens ist die interne Wissensentwicklung zu sehen. Es geht um die Produktion neuer Fähigkeiten und Produkte sowie leistungsfähigerer Prozesse in Übereinstimmung mit den Wissenszielen des Unternehmens. Dazu sind der allgemeine Umgang des Unternehmens mit neuen Ideen und der Kreativität der Mitarbeiter kritisch zu analysieren. Wissensentwicklung hat mit Lernprozessen auf der individuellen wie organisationalen Ebene zu tun. Hierfür müssen die richtigen Rahmenbedingungen geschaffen und Transparenz über das vorhandene Wissen hergestellt werden.

4. Wissens(ver)teilung

Dieser Prozess hat drei Teilaufgaben. Erstens geht es um die zentral gesteuerte Multiplikation von Wissen durch Verteilung auf eine Vielzahl von Mitarbeitern. Zweitens müssen isoliert vorhandene Erfahrungen der gesamten Organisation verfügbar gemacht werden, um möglichst großen Nutzen zu stiften. Die dritte Teilaufgabe betrifft den simultanen Wissensaustausch, der in die Entwicklung neuen Wissens mündet. Im Rahmen der Wissensverteilung ist genau zu planen, wer was in welchem Umfang wissen muss und wie dies durch Verteilen von bereits vorhandenem Wissen erreicht werden kann.

5. Wissensnutzung

Dieser Prozess soll gewährleisten, dass wertvolle Fähigkeiten und Wissensbestände im Unternehmen effektiv genutzt werden. Hierin liegt der eigentliche Zweck und das letztendliche Ziel des Wissensmanagements. Probst und Mitarbeiter empfehlen, den Wissensnutzer als Kunden des Wissensmanagements zu betrachten und daher Kontexte zu schaffen, in denen das erarbeitete Wissen auch tatsächlich genutzt wird. Dazu gehört zum Beispiel eine nutzerfreundliche Infrastruktur.

6. Wissensbewahrung

Wissensbewahrung soll das Unternehmen vor Wissensverlusten schützen, wie sie beim Ausscheiden von Mitarbeitern auftreten, aber auch bei Verkäufen von Unternehmensbestandteilen oder größeren Reorganisationen entstehen können. Hierzu muss das Bewahrungswürdige ausgewählt (Teilprozess Selektion), in adäquater Form gespeichert (Teilprozess Speichern) und regelmäßig aktualisiert werden (Teilprozess Aktualisieren). Eine besondere Rolle spielen Schlüsselmitarbeiter, die kritisches Unternehmenswissen repräsentieren und möglichst eng an das Unternehmen gebunden werden sollten. Hierzu können Anreizsysteme und Austrittsbarrieren einen Beitrag leisten. Die gezielte Explizierung und Dokumentation von Expertenwissen vermindert darüber hinaus die Abhängigkeit des Unternehmens von Einzelpersonen.

Den eben geschilderten und für das hier behandelte Thema vorrangigen sechs Kernprozessen fügen Probst und Mitarbeiter noch zwei unterstützende Bausteine hinzu, um den Managementregelkreis des Wissensmanagements zu schließen:

7. Wissensziele

Die Kernprozesse des Wissensmanagements müssen an die Wissensziele des Unternehmens gekoppelt sein, also diese Ziele erfüllen helfen. Wissensziele werden vom Management festgelegt. Sie geben den Aktivitäten des Wissensmanagements eine Richtung, wobei normative, strategische und operative Ziele unterschieden werden.

8. Wissensbewertung

Dieser Prozess ist komplementär zur Definition der Wissensziele und wird als wesentlich erachtet, um zielgerichtete Anpassungsmaßnahmen des Wissensmanagements vorzunehmen. Dazu ist es erforderlich, Veränderungen der organisationalen Wissensbasis zunächst zu messen und anschließend im Hinblick auf die Wissensziele zu bewerten (interpretieren). Da hierbei nicht auf eine etablierte Menge von Mess- und Bewertungsmethoden zurückgegriffen werden kann, wird die Wissensmessung und -bewertung als große Herausforderung gesehen.

Das Baustein-Modell von Probst und Mitarbeitern liefert wertvolle Hinweise zur Strukturierung und Vernetzung der Aktivitäten im Wissensmanagement. Es ist damit besonders für wissensintensive Organisationen, wie beispielsweise Beratungsunternehmen, eine wichtige Hilfe. Das Modell beschreibt alle Handlungsfelder, um Wissen als Schlüsselressource zu managen und zeigt, wo Interventionsmöglichkeiten seitens des Managements bestehen.

Die gesteckten Erwartungen an Wissensmanagementprojekte in Unternehmen blieben bislang häufig unerfüllt [Hatt2002, 180]: So waren die Konzepte anspruchsvoller in der Umsetzung als erwartet. Die Akzeptanz der Anwender war geringer als geplant. Prognostizierte Effizienzsteigerungen konnten nicht erfüllt werden, und die Rolle moderner Technologien blieb unklar.

Einer empirischen Studie [Rugg1998, 87 f.] zufolge liegen die Hauptschwierigkeiten des Wissensmanagements neben notwendigen Verhaltensänderungen der Organisationsmitglieder in der Problematik, den Wert und die Leistung von Wissensressourcen zu messen. Als Haupthindernis des Wissenstransfers wird die (falsche) Unternehmenskultur aufgeführt.

3.2 Nachteile heutiger zentralisierter Wissensmanagementsysteme

Gemäß der Analyse bestehender Wissensmanagementsysteme (WMS) durch Bonifacio und Bouquet [BoBo2002, 300-301] sind die heutigen Lösungen durch folgende Grundannahmen gekennzeichnet:

- Das organisatorische Wissen kann (und sollte) zentralisiert und in einer gemeinsam nutzbaren Wissensbasis abgelegt werden.
- Wissen muss es in ein abstraktes und allgemeines Objekt transformiert werden, um es nutzen zu können.

Es kann festgestellt werden, dass sich die derzeitigen Systeme in ihrem Aufbau am Ansatz des zentralen Wissensmanagements orientieren (Abb. 4). Dies kann zu einer geringen Nutzung von WMS führen, da die Prozesse zur Genierung und Teilung von

Wissen bei verteilten sozialen Gemeinschaften und zentral verwalteten Systemen unterschiedlich sind [BBTR2002, 24].

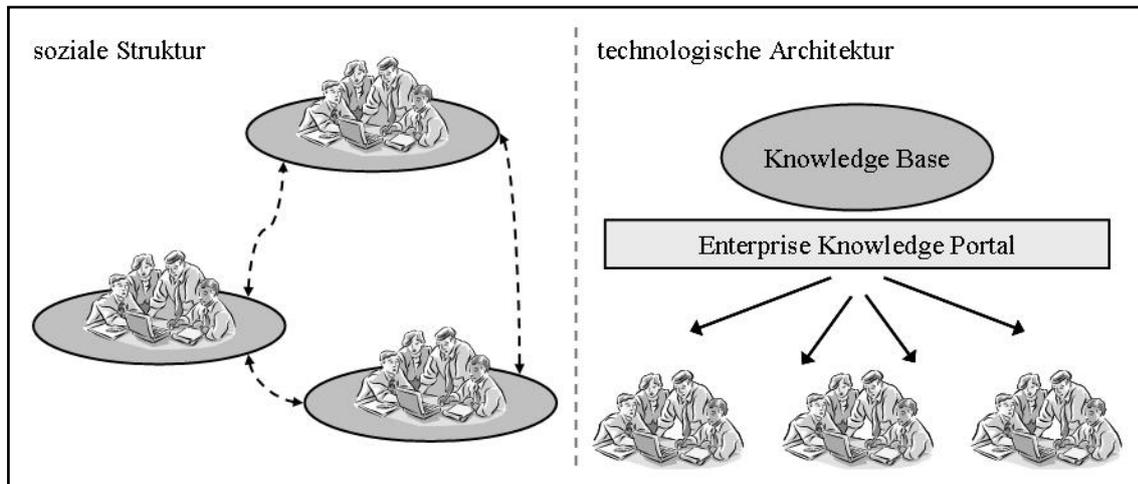


Abbildung 4: Unterschied zwischen der sozialen Struktur in einer Organisation und der technologischen Architektur eines WMS [BBTR2002, 24]

Neumann et al. sowie Bonifacio et al. kritisieren weiter, dass der zentrale Ansatz die Komplexität einer erfolgreichen Wissensintegration in unterschiedlichen Kontexten unterschätzt. Er geht davon aus, das Wissen weitgehend kontextfrei dargestellt werden kann, indem lokale Konzepte in ein einheitliches, gemeinsam nutzbares Konzept überführt werden. In der Realität ist dies aber nicht immer möglich. Es bleibt außerdem unklar, wie in einem dynamischen und hochkomplexen Umfeld mit permanenten Veränderungen umzugehen ist. [NSBo2000, 78;BBMa2000, 4].

Die hier genannten Kritikpunkte und Schwächen zentralisierter WMS lassen sich grundsätzlich durch einen agentengestützten, verteilten Ansatz des Wissensmanagements überwinden. Auf Beispiele aus der Literatur wird im folgenden Abschnitt eingegangen. Das verteilte Wissensmanagement kann durch folgende Grundannahmen charakterisiert werden [NSBo2000, 78; BBMa2000, 4; BBTr2002, 25-26]:

- Wissen ist immer an seinen Kontext gebunden. Solche kontextuellen Faktoren können bspw. Hintergrundannahmen, verfügbare Ressourcen und der verfolgte Zweck sein.
- Sprachliche Heterogenität und semantische Differenzierung sind keine Störgrößen, die es zu entfernen gilt, sondern Eigenschaften des Wissens, die man berücksichtigen sollte.

3.3 Anwendung von Agenten im Wissensmanagement

3.3.1 Informationsagenten

Seit einigen Jahren werden verstärkt Überlegungen angestellt, für welche Anwendungsdomänen die Agententechnologie besonders geeignet ist. Hier haben sich die Informationsgewinnung, -verarbeitung und -bereitstellung als mögliche Gebiete für agentenbasierte Anwendungen herauskristallisiert [KaK11997]. In diesem Zusammenhang werden seit geraumer Zeit s.g. Informationsagenten thematisiert, d.h.

Agenten, die auf Grund ihrer Eigenschaften der Proaktivität und Reaktivität in der Lage sind, auf heterogene und verteilte Datenressourcen zuzugreifen und Informationen im Auftrag des Nutzers zu erwerben, zu vermitteln und zu pflegen [Klus2001, 337]. Die Aufgabe von Informationsagenten besteht vor allem darin, Informationsquellen zu identifizieren, Information zu extrahieren, diese entsprechend dem jeweiligen Nutzerprofil zu filtern, dabei gegebenenfalls Informationen aus unterschiedlichen Quellen zusammenzuführen und in adäquater Form dem Nutzer zu präsentieren [Zarn1999, 26]. Wesentliche Anforderungen an Informationsagenten sind demzufolge, dass diese über interne Modelle der Struktur entsprechender Informationsquellen verfügen, Strategien besitzen, geeignete Informationsquellen auszuwählen, auf diese zuzugreifen, Konflikte zu lösen und Informationen zusammenzuführen [SDPW⁺1996, 36]. Das heißt, Informationsagenten verfügen über die Fähigkeiten, Informationen zu suchen, zu analysieren, zu bearbeiten und zu verschmelzen und diese einem Nutzer aufzubereiten und zu präsentieren, bzw. diesen durch die bereitgestellten Informationen zu führen [Klus2001, 337].

Derzeit sind verschiedene Arten von Informationsagenten zu finden. Im Folgenden wird eine von Franklin und Graesser [FrGr1996] vorgeschlagene und von Klusch [KIBe1998, 8] für Informationsagenten erweiterte Klassifikation von Agenten verwendet (Abb. 5).

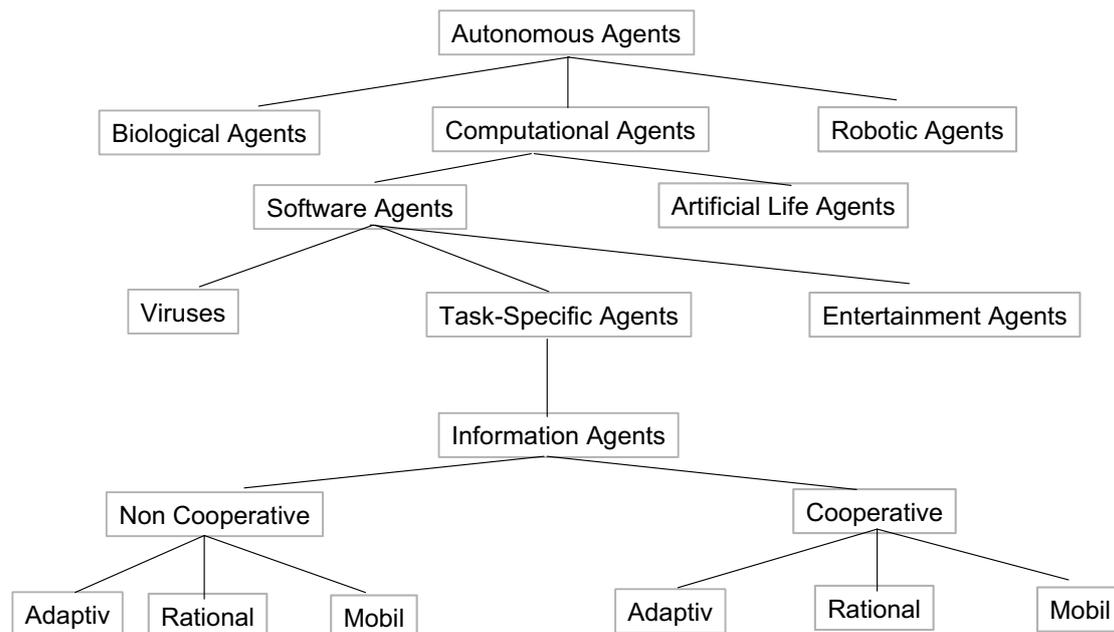


Abbildung 5: Klassifizierung Autonomer Agenten

Grundsätzlich wird demnach zwischen kooperativen (z.B. BIG agent, PLEIADES, ABS und RETSINA [Klus2001, 349]) und nicht kooperativen Informationsagenten (z.B. Letizia, Travel Agent [Klus2001, 351]) differenziert, die sich vor allem darin unterscheiden, ob der Agent in der Lage ist, im Sinne seiner Aufgabenstellung mit anderen Agenten zu interagieren, um zum Beispiel Zugriff auf verteilte Informationsquellen zu erhalten. Diese Agenten können jeweils sowohl adaptiv sein, d.h. anpassbar an auftretende Umgebungsveränderungen, als auch rationales, kooperatives Verhalten zeigen. [KIBe1998, 8].

Es lässt sich im Ergebnis feststellen, dass Informationsagenten selbst zwar noch kein WMS darstellen, jedoch eine geeignete Technologie sind, Teilbereiche in WMS

zu realisieren. Hier ist vor allem an die Bausteine Wissensidentifikation, Wissenserwerb, Wissensverteilung und Wissensnutzung zu denken.

3.3.2 Agentenarchitekturen im Wissensmanagement

In der Vergangenheit wurden verschiedene Vorstellungen darüber entwickelt, in welchem Rahmen Agententechnologie beim Wissensmanagement genutzt werden kann. Die beiden folgenden Architekturen beschreiben auf konzeptioneller Ebene deren Einsatz.

Architektur nach Bonifacio et al.

Die Architektur nach Bonifacio et al. basiert auf den Prinzipien der Autonomie und Koordination [BBCu2002, 3]. Unter Autonomie wird verstanden, dass das Wissen, jeder Gemeinschaft, in der es entsteht und verwendet wird, auch autonom zu verwalten ist [BBTr2002, 26]. Koordination wird beschrieben als Möglichkeit des Austausches des Wissens zwischen lokalen autonomen Gemeinschaften [BBTr2002, 26]. Hierbei ist vor allem die semantisch korrekte Interoperabilität eine notwendige Voraussetzung, die durch Extraktion des Wissens in ein definiertes Format und durch Erstellen von Beziehungen zwischen den Kontexten der autonomen Gemeinschaften (*context matching*) erreicht wird. Da das *context matching* in der Praxis jedoch nicht ausreichend ist und Wissensaustausch in der Regel Ergebnis eines sozialen Prozesses ist, wird durch Bonifacio et al. ein Abgleich vorgeschlagen, der als Bedeutungsverhandlung (*meaning negotiation*) bezeichnet wird. Dieser wird durch autonome Intelligente Agenten realisiert. Sie besitzen Wissen über den Kontext der jeweils vertretenen Gemeinschaft und können nun diesen mit anderen Agenten der weiteren Gemeinschaften durch Kommunikationsprotokolle und Interaktionen abgleichen. Agenten haben in diesem Kontext also vornehmlich die Aufgabe, als intelligente Schnittstellen zwischen den Gemeinschaften zu agieren.

Architektur nach Dignum

In der von Dignum entwickelten Architektur analysieren und modellieren Agenten die betrachtete Organisation und ihren Bedarf an Wissen. Weiterhin wird mit Hilfe der Agententechnologie eine wiederverwendbare Architektur für die Entwicklung von WMS angestrebt [Dign2004, 11].

Die Architektur unterscheidet zwischen Wissensapplikation, -beschreibung und -darstellung. Durch eine gemeinsame, einheitliche Beschreibung der Wissensquelle und des Wissensbedarfs soll es gelingen, flexibel auf die unterschiedlichen Quellen des Wissens und die Bedürfnisse der Nutzer reagieren zu können. Agenten haben dabei die Aufgabe, kollaborativ mit anderen Agenten den Abgleich zwischen Wissensangebot und -nachfrage unter Berücksichtigung der Bedürfnisse und Präferenzen der Nutzer unter der jeweiligen Wissensanforderung einer Aufgabe vorzunehmen. Durch die Zusammenarbeit mit anderen Agenten und dem Benutzer lernt der Agent. So wird er in die Lage versetzt, flexibel auf veränderte Anforderungen zu reagieren.

Im Vergleich mit der Architektur von Bonifacio et al. ist jene von Dignum deutlich komplexer, da unterschiedliche Arten von Agenten wirksam werden. Die Koordination der Agentengemeinschaft erfolgt mit Hilfe eines Netzwerkmodells.

3.3.3 Bestehende Agentenanwendungen im Wissensmanagement

Im folgenden werden ausgewählte agentenbasierte Anwendungen im Wissensmanagement vorgestellt und den Bausteinen des Wissensmanagements nach Probst et al. [PRRo1999, 58] zugeordnet.

KennisNet

KennisNet basiert auf der Architektur von Dignum [Dign2004] und ist als Gemeinschaftsunterstützungssystem (*community support system*) für einen Versicherungs- und Finanzdienstleister konzipiert worden [Dign2002]. Grundsätzliches Ziel des Systems ist es, das gemeinsam genutzte Wissen in einer Gemeinschaft zu dokumentieren, strukturieren und organisieren, womit die breite Verfügbarkeit von aktuellen Informationen und Wissen im Unternehmen gesichert werden soll.

Hierbei wird ein agentenunterstützter Wissensmarktplatz eingesetzt, der folgende Funktionen beinhaltet:

- Wissen gemeinsam zu nutzen, welches (noch) nicht in der Wissensbasis beschrieben wurde,
- Unterstützung bei der Bildung von Koalitionen, um Problemlösungen zu erarbeiten, wenn Wissen noch fehlt und
- Unterstützung für den direkten Austausch von zwischen Parteien, bei dem über die Bedingungen für den Austausch verhandelt werden kann.

Mit Hilfe eines persönlichen Assistenz-Agenten ist es dem Nutzer (als Wissenssuchendem oder Wissensbietendem) möglich, nach Kooperationspartnern in *KennisNet* zu suchen. Um Partner zu finden und über die Bedingungen für den Wissensaustausch zu verhandeln, verwendet der Agent vom Benutzer festgelegten Vorgaben. Dabei kann es sich beispielsweise um die Art der bevorzugten Wissensquelle, einen geplanten Austauschtermin oder die eigene Zahlungsbereitschaft handeln. Über einen s.g. Matchmaker Agent kann der Agent im Folgenden seinen Wissensbedarf mit den entsprechenden Wissensquellen abgleichen und in Verhandlung über die Nutzung des Wissens treten. Als Grundlage der „Bezahlung“ wird im Sinne eines fairen Austausches ein Punktesystem vorgeschlagen, das den Bezahlten in die Lage versetzt, seinerseits Wissen anderswo zu erwerben oder anderweitige Nutzeffekte „einzukaufen“.

DIAMS

DIAMS (Distributed Multi-Agent System for Collaborative Information Management and Sharing) ist eine Entwicklung, die vorwiegend dem kollaborativen Informationsmanagement dient, jedoch ursprünglich in einer Weiterentwicklung als WMS geplant war [CWWr2000]. Die agentenbasierte Architektur des Systems soll den Benutzern bei Suche, Zugang und Organisation von relevanten Informationen aus dem World Wide Web (WWW) sowie deren Austausch und gemeinsamer Nutzung innerhalb von Interessensgruppen helfen.

Persönliche Assistenz-Agenten im System unterstützen ihre jeweiligen Nutzer bei der Präsentation, Organisation und Verwaltung von Informationen. Weiterhin arbeitet jeder Assistenz-Agent mit anderen Agenten zusammen, um Informationen zu teilen und auszutauschen. Dazu kann der Agent externe Informationssammlungen

lokalisieren und für seinen Nutzer zugänglich machen. Hierbei ist ein Matchmaker-Agent behilflich, dessen Funktion der im System KennisNet entspricht.

DIAMS unterstützt die dynamische hierarchische Organisation einer Sammlung von Informationen durch die Integration von nutzerdefinierten, indexierten Kategorien. Eine Kategorie in *DIAMS* ist ein Ordner für die Speicherung des Informationsinhaltes und ein Index für die Suche und Kommunikation. Mittels Kategorien und Schlüsselwörtern kann sowohl der Nutzer Anfragen an die Informationssammlung formulieren als auch Agenten untereinander kommunizieren und Nachrichten über verfügbares Wissen austauschen.

In einer geplanten Ausbaustufe, bei der unklar ist, ob sie realisiert wurde, war ein zusätzlicher Knowledge Agent vorgesehen. Er verfügt über eine Wissensbasis, die komplexe semantische Beziehungen zwischen Informationen speichert und ihm somit „Sachkenntnis“ in verschiedenen Wissensgebieten verleiht. Vorgesehen war auch der automatische Wissenserwerb des Knowledge Agent von den Assistenz-Agenten.

K-InCA

Bei der Entwicklung des Systems K-InCA (Knowledge Intelligent Conversational Agent) wurde vor allem das Ziel verfolgt, die Komplexität beim Umgang mit WMS durch den Einsatz von Agenten zu reduzieren und intelligente Schnittstellen zu schaffen, die den Nutzer bei der Verteilung und Nutzung des Wissens unterstützen. Weiterhin sollen die Agenten bei der Suche nach geeigneten Wissensquellen (i.d.R. Wissen bestimmter Mitarbeiter) behilflich sein [ANRR2001, RANa2001].

Die Agenten im System (s.g. Change Management Agents) verhalten sich wie persönliche Wissensmanagementtrainer, die den Mitarbeitern bei der Einführung in die Praktiken der Wissensmanagementprozesse behilflich sind. Die Agenten unterstützen den Benutzer dahingehend, dass sie ihn durch die verschiedenen Konzepte im WM führen (Agenten schlagen u.a. Ressourcen vor, die benutzt werden können), ihn überwachen (Agenten schlagen Übungen zum besseren Verständnis der Praktiken im WM vor) und fördern (Agenten hinterfragen proaktiv die Überzeugungen des Nutzers). Jeder Agent verfügt über ein Benutzermodell seines Besitzers, mit deren Hilfe der Agent sich an Präferenzen und Aktionen des Nutzers anpassen kann.

Neben dieser Nutzersteuerung existiert auch ein Ressourcen Management Agent, der Ressourcen der Organisation verwaltet, z.B. die Informationssammlungen (*information repositories*) über externe Mitarbeiter und deren Kompetenzen, formelle/nicht-menschliche organisatorische Systeme, Wissensbasen oder Trainingssysteme. Wenn ein Change Management Agent erkennt, dass sein Nutzer als Wissensbesitzer oder Wissenssuchender auftreten kann, initiiert er einen als „Konsultation“ bezeichneten Prozess des Kompetenzabgleichs mit anderen Change Management Agents, der vom Ressourcen Management Agenten gesteuert wird. Jeder Agent, der an dieser Konsultation teilnimmt, kann bei dem Wissensaustausch entscheiden, ob er als Suchender oder Besitzer von Wissen auftreten möchte.

FRODO

Frodo (Framework for Distributed Organizational Memories) ist ein agentenbasiertes Workflow-System, das schwach strukturierte und wissensintensive Prozesse unterstützt [BDEL⁺2004]. Das System basiert auf dem Konzept der Distributed Organizational Memories (DOM). Der DOM-Ansatz beschreibt Vorgehensweisen und Techniken der zeit-, raum- und personenunabhängigen Nutzung, Bewahrung und Handhabung von Wissen [BDEL⁺2004, 59]. Dabei wird untersucht, wie operative Prozesse und WMS gekoppelt werden können, wie auf Wissensobjekte zugegriffen und diese mit Hilfe einer Wissensbeschreibungsschicht verarbeitet werden können und wie unstrukturierte Wissensquellen durch Informationsextraktion und Dokumentenanalyse aufbereitet werden können [DABE⁺2002]. Das Frodo-Projekt zielte auf die lokale Entwicklung unternehmensgedächtnisartiger Strukturen und deren Verknüpfung für eine globale Wissensnutzung. Die Agenten im System realisieren u.a. das Zusammenführen und Aufbereiten von Informationen aus unterschiedlichen Quellen, die Pflege und Nutzung von Ontologien, fungieren als Mediatoren zwischen unterschiedlichen lokalen Wissensquellen auf Basis von Ontologien und unterstützen den Workflow des Systems.

Wissensmanagement-anwendung	Wissens-ident.	Wissens-erwerb	Wissens-entw.	Wissens-(ver-)teilg.	Wissens-nutzung	Wissens-be-wahrung
<i>KennisNet</i> (knowledge sharing market)	X	(X) ¹		(X) ²	(X) ²	
<i>DIAMS</i>	X	X		X		
<i>K-InCA</i>	X	X		X	(X) ³	(X) ⁴
<i>FRODO</i>	X	X		X	X	

¹ Während der Suche nach Personen, wird entsprechend auch Wissen erworben, welches dem Agenten dazu dient, neue Beziehungen herzustellen.

² Nach erfolgreich zustande gekommener Kommunikation zwischen den Mitarbeitern, wird das Wissen danach verteilt und gegebenenfalls auch anderen Personen zugänglich gemacht.

³ Die Agenten helfen passiv bei der Wissensnutzung, indem sie ihrem Besitzer gezielte Hinweise in Bezug auf die Nutzung des Wissens geben bzw. ihm mitteilen, dass neues Wissen vorhanden ist.

⁴ Durch die gezielten Hinweise zur Erfassung und Speicherung von Wissen, unterstützen die Agenten indirekt die Wissensbewahrung.

Abbildung 6: Unterstützung der Bausteine des Wissensmanagement durch die hier vorgestellten agentengestützten Systeme (nach [Ethn2006, 77])

Weitere Ansätze bzw. agentenbasierte Systeme, wie z.B. KRAFT (Knowledge Reuse And Fusion/Transformation) [PHGM⁺1999] oder KAoS (Knowledgeable Agent-oriented System) [EDAb2003] zeigen, dass agentenbasierte Anwendungen im

Umfeld des Wissensmanagements einen Mehrwert gegenüber „herkömmlichen“ Ansätzen erbringen können.

In Abbildung 6 wird dargestellt, wie die vier vorgestellten Systeme sich in die Bausteine des Wissensmanagements einordnen lassen. Dabei wird deutlich, dass Agenten bislang vornehmlich bei der Wissensidentifizierung, dem Wissenserwerb und der Wissensverteilung genutzt werden. Die weiteren Bausteine sind bislang bei agentenbasierten Entwicklungen zum Wissensmanagement unterrepräsentiert. Insgesamt existieren nur wenige Anstrengungen, Agenten für das Wissensmanagement nutzbar zu machen. Diese bewegen sich im allgemeinen auf der Ebene von Prototypen.

3.3.4 Generelle Agentenanwendungen und Anwendungsumgebungen

Obwohl in den vergangenen Jahren eine Vielzahl von Arbeiten das Potenzial von Intelligenten Agenten in betriebswirtschaftlichen Anwendungen untersuchte [Paru2000; KHLS2006], hat sich diese Technologie bislang nicht in der Praxis durchgesetzt. Eine der Ursachen dafür ist, dass nicht eindeutig geklärt ist, für welche Anwendungsklassen Agenten und Multiagentensysteme besonders geeignet sind, bzw. wo sie gegenüber herkömmlichen Ansätzen bessere Ergebnisse erzielen können.

Konferenzen und Workshops zum Thema Agententechnologie weisen zunehmend einen Schwerpunkt in Richtung Anwendungsentwicklung auf. Dennoch fehlen bis heute noch häufig aussagekräftige Leistungsvergleiche, aus denen ersichtlich wird, wo der Einsatz von Agententechnologie vorteilhaft ist. Es wird allgemein akzeptiert, dass Intelligente Agenten durch ihre spezifischen Eigenschaften besonders geeignet sind, hoch dynamische Probleme mit verteilten Informationen und Ressourcen zu bewältigen, bei denen hohe Anforderungen an betriebliche Flexibilität gestellt werden [CoJe1996]. Erste Untersuchungen deuten darauf hin, dass durch den Einsatz von Softwareagenten eine Verbesserung der betrieblichen Flexibilität und Reduzierung von softwarespezifischen Kosten realisiert werden kann (vgl. [CoJe1996; Buss1998; Paru2000; Anth2003]).

Laut Müller [Müll1997] scheint der Einsatz der Agententechnologie sinnvoll, wenn:

- der betrachtete Anwendungsbereich durch eine natürliche verteilte Struktur charakterisiert ist, die nicht künstlich aus einer zentralen Struktur erzeugt wird,
- eine dynamische Umwelt vorliegt, in der strukturelle Änderungen berücksichtigt und flexibel auf Änderungen reagiert werden muss,
- die abzubildenden Interaktionsbeziehungen derart umfangreich sind, dass eine komplexe Koordination zwischen den einzelnen Entitäten unverzichtbar ist.

Einen Anhaltspunkt für geeignete Domänen bildet die von Russel und Norvig erstellte Kategorisierung von Anwendungsumgebungen [RuNo2003] (siehe Abb. 7).

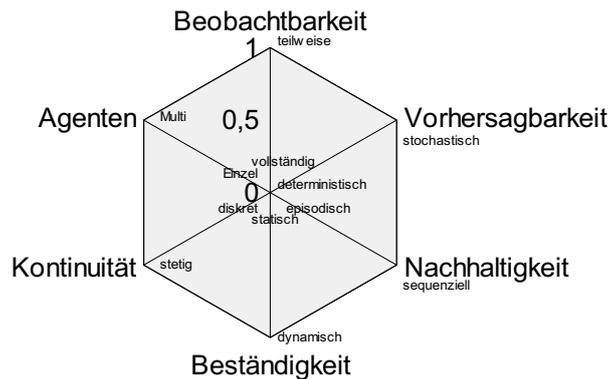


Abbildung 7: Eigenschaften von Arbeitsumgebungen nach [WRZN2006] in Anlehnung an [RuNo2003, 41-44]

Russel und Norvig [RuNo2003, 41-44] haben für Arbeitsumgebungen verschiedene Dimensionen identifiziert, nach denen diese kategorisiert werden können. Demzufolge sind Arbeitsumgebungen:

- vollständig oder teilweise beobachtbar – in welchem Umfang können Zustandsänderungen der Umgebungen wahrgenommen werden,
- deterministisch oder stochastisch – in welchem Grad sind die Auswirkungen von Aktionen auf die Umgebung vorhersagbar,
- episodisch oder sequenziell – besteht die Wahrnehmung und die Reaktion eines Agenten auf Zustandsänderung der Umgebung aus Episoden, oder beeinflussen alle durchführbaren Aktionen die jeweilig nachfolgenden,
- statisch oder dynamisch – in welchem Grad kann die Umgebung sich im Verlauf einer Entscheidung ändern,
- diskret oder stetig – sind Zustandsänderungen der Umgebungen sowie deren Behandlung in festen Zeiteinheiten oder kontinuierlich gegeben,
- Einzelagenten- oder Multiagentenumgebungen – ist nur ein oder sind mehrere Akteure in einer Umgebung gegeben.

Die jeweiligen Kategorien können in einem Kontinuum zwischen 0 und 1 dargestellt werden. Dabei steht eine 0 für vollkommen beobachtbar, deterministisch, episodisch, statisch, diskret und für Einzelagent und eine 1 für teilweise beobachtbar, stochastisch, sequenziell, dynamisch, stetig und Multiagentenumgebungen. Stark vereinfacht lässt sich feststellen, dass je größer der Wert des Hexagons einer Arbeitsumgebung ist, um so mehr sind Agenten für die Aufgabenlösung geeignet [WRZN2006, 78].

Einen wesentlichen Beitrag zum Thema Benchmarking von agentenbasierten Anwendungen stellt [KHLS2006] dar. Dort unterzogen sich im Schwerpunktprogramm (SPP) 1083 der Deutschen Forschungsgemeinschaft organisierte Teilprojekte einer Evaluation bzw. einem Benchmarking. Im Umfeld der Produktionsplanung- und Steuerung wurde festgestellt, dass insbesondere hoch flexible Anforderungen (vor allem im Supply Chain Management oder flexibler Produktion), stochastische Aufgabenstellungen (individuelle Fertigung) oder hoch dynamische Umgebungen (Produktionsflexibilität, Ausfallsicherheit) geeignete Anwendungsbereiche der Agententechnologie sind [WRZN2006, 80].

So wurde durch Lorenzen et al. [LWDS⁺2006] der im Rahmen des InTaPS-Projektes erstellte agentenbasierte Algorithmus zur Prozessplanung, Produktionssteuerung und -kontrolle mit traditionellen Ansätzen verglichen und festgestellt, dass insbesondere hinsichtlich Flexibilität, Robustheit und Fehlertoleranz Vorteile erzielt werden konnten. Wörner und Wörn [WöWö2006, 130] kommen im Vergleich eines agentenbasierten Produktionsplanung und -steuerungs (PPS)-Systems gegenüber einem herkömmlichen Ansatz zu dem Schluss, dass Multiagentensysteme im PPS-Umfeld vor allem Vorteile aufweisen, wenn Aufträge auf leerstehenden Maschinen dynamisch eingeplant werden müssen und das MAS Warteschlangen effizienter verwaltet als zentrale Systeme. Zu ähnlichen Schlussfolgerungen gelangen u.a. auch [MSZI2006, 154], [ZWBO2006, 174] und [SWKö2006, 193], die vor allem die gestiegene Flexibilität und Robustheit der agentenbasierten Systeme gegenüber herkömmlichen Systemen im Planungs- und Steuerungsprozess hervorheben.

Im Umfeld agentenbasierter Anwendungen im Krankenhaus wurden in [KHLS2006] vor allem Projekte vorgestellt, die sich Scheduling-Problemen widmeten, so z.B. OP-Planung, Planung, Steuerung und Koordination von Klinikprozessen und Koordination von Patientenlogistik in der Radiologie. Auch hierbei wurde festgestellt, dass vor allem in hochdynamischen Umgebungen, wie dem Krankenhaus, die Planungsqualität durch verteilte agentenbasierte Planung flexibler, robuster und folglich qualitativ hochwertiger erfolgt als mit konventionellen Ansätzen [BeCz2006; PZRH⁺2006; EMSt2006]. Weiterhin erwiesen sich agentenbasierte Anwendungen als geeignet, eine erhöhte Interoperabilität durch intelligente Schnittstellen zwischen proprietären Geräten bzw. Informationssystemen im Krankenhaus herzustellen [RSKF2006].

Zusammenfassend lässt sich aus der Kategorisierung von Anwendungsumgebungen nach [RuNo2003], den Ausführungen von Müller über geeignete Einsatzgebiete für Agenten [Müll1997] und den Ergebnissen der Evaluation und des Benchmarking agentenbasierter Anwendungen in [KHLS2006] folgern, dass folgende Bedingungen die Anwendung von Agententechnologie sinnvoll erscheinen lassen:

- hohe Dynamik der Umwelt,
- hohe Flexibilität der Aufgabenstellung und Umwelt,
- stochastische Umgebungen,
- eine natürliche Verteiltheit vorliegt,
- ein Vielzahl beteiligter Entitäten bzw. Akteure mit damit verbundenen Interaktionsbeziehungen vorhanden sind.

Hingegen sind Agenten in zentralisierten Systemen, mit einer geringen Anzahl von Akteuren und Entitäten, in stark standardisierten Prozessen und einer Umgebung ohne hohe Dynamik i.d.R. wenig geeignet, da herkömmliche Systeme mit spezialisierten Methoden und Algorithmen, z.B. des Operations Research oder der Simulation, deutlich effizienter und performanter sind. Weiterhin kann festgestellt werden, dass die Implementierung komplexer intelligenter und eventuell sogar lernfähiger Agenten, mit Wissensbasen, Inferenzmechanismen und eigenen Zielvorstellungen hohe Anforderungen an deren Entwicklung stellt und somit aufgrund des Entwicklungsaufwands und Erfolgsrisikos nicht für alle Anwendungsumgebungen angemessen oder geeignet erscheint.

3.3.5 Rückschlüsse auf Anwendungsmöglichkeiten im Wissensmanagement

Bezugnehmend auf o.g. Eigenschaften von Anwendungsumgebungen, die den Einsatz von Agententechnologie nahe legen, lassen sich Rückschlüsse auf Anwendungsfelder im WM identifizieren. Das Wissensmanagement lässt sich in Anlehnung an Abb. 7 ebenso in die Dimensionen der Arbeitsumgebungen einordnen (siehe Abb. 8).

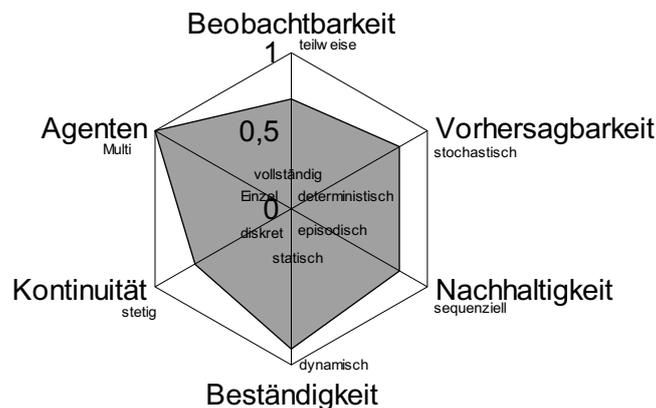


Abbildung 8: Einordnung des Wissensmanagements in die Eigenschaften von Arbeitsumgebungen

Dabei lässt sich feststellen, dass insbesondere hinsichtlich der Anzahl der Akteure bzw. Entitäten im System (Agenten), der Beständigkeit der Arbeitsumgebung (hoch dynamisches Umfeld, im Sinne neuer Wissensquellen, neuer Formate und Änderungen der Inhalte), der Nachhaltigkeit der Umgebung (Wissensbedarf und -änderung erfolgen i.d.R. sequenziell und nicht periodisch) und der Vorhersagbarkeit der Reaktion der Arbeitsumgebung (hoch stochastisch, da nicht exakt bestimmt werden kann, in welcher Weise erworbenes und verwendetes Wissen sich auf die Arbeitsumgebung bzw. Prozesse auswirken) der Schluss nahe liegt, dass das Wissensmanagement eine geeignete Anwendungsumgebung für Agententechnologie ist. Im Folgenden wird untersucht, welchen Beitrag Agenten in den einzelnen Bausteinen des WM leisten können, bzw. wo deren Grenzen bislang liegen.

Wissensidentifikation

Aufgrund der hohen Dynamik des Wissensumfelds und der breiten Verteilung von Wissen und Informationsressourcen in Organisationen bzw. im Internet, ist der Einsatz von Agenten innerhalb dieses Bausteins naheliegend. Dabei sind unterschiedliche Szenarien denkbar, die teilweise schon in Beispielanwendungen in Abschnitt 3.3.3 skizziert wurden. Zunächst können Informationsagenten (Softbots) die Aufgabe der Quellenidentifikation realisieren. Der Einsatz von Agenten erscheint hierbei insbesondere bei stark verteilten Wissensquellen bzw. bei der Identifikation externer Ressourcen sinnvoll, wohingegen bei monolithischen WMS mit nur einer oder wenigen Wissensbasen die Wissensidentifikation durch herkömmliche Systeme i.d.R. effizienter realisiert werden kann. Agentenbasierte Marktplätze und Wissensbroker könnten ebenfalls unterstützen, wenn es um die Suche nach externen Wissensquellen geht. Weiterhin erscheinen agentengestützte Mechanismen des Collaborative Filtering sinnvoll, die das Wissen anderer Teilnehmer eines Wissens-

netzwerkes heranziehen, um gezielte Vorschläge für weitergehende Wissensquellen oder –inhalte geben zu können. Ebenso ist es zweckmäßig, wenn Informationsagenten soweit personalisiert sind, dass sie Aktivitätsschwerpunkte der durch sie repräsentierten Nutzer speichern und daraus auf Interessenschwerpunkte schließen. Dadurch wird die autonome, agentenbasierte Recherche nach Wissensquellen erst sinnvoll möglich.

Wissenserwerb

Sind Wissensquellen identifiziert, dienen Agenten bei verteilten Systemen als Wissensbroker, die als Mediator Bedarf und Nachfrage koordinieren und gegebenenfalls in Verhandlung über die Parameter des Wissensaustausches treten. Eine nicht notwendigerweise monetäre Entlohnung dieses Austausches fördert die Bereitschaft zur Bereitstellung von Wissen. Der Wissenserwerb kann ähnlich wie im vorgestellten KenntnisNet-Projekt quasi automatisch auf Wissensmärkten erfolgen, die zuvor schon der Wissensidentifikation dienen. Die Bausteine der Wissensidentifikation und des Wissenserwerbs werden somit integriert.

Neben der Nutzung von Agenten als Mediatoren sind diese in der Phase des Wissenserwerbs vor allem auch wesentlich für die Etablierung intelligenter Schnittstellen, ähnlich wie im DIAMS- und K-InCA-Projekt skizziert. Hierbei kommt Agenten die Aufgabe zu, Wissensbasen zu repräsentieren und die gespeicherten Inhalte anderen Agenten oder Nutzern in einem standardisierten Format zur Verfügung zu stellen. Die Aufgaben der Agenten sind dabei nicht auf reine Konvertierungsarbeiten beschränkt, sondern sollten eine Aufbereitung von Wissen in Kontexten umfassen. Kommunikation und Koordination (z.B. Klärung von Inhalten) von Agenten in Wissensnetzwerken werden hierbei durch den Rückgriff auf etablierte Ontologien unterstützt. Agenten als intelligente Schnittstellen können wesentlich dazu beitragen, mit der hohen Dynamik im Wissensmanagement fertig zu werden, die durch häufige Veränderungen bei den zu integrierenden Wissensquellen (Offenheit) und –inhalten gekennzeichnet sind sowie durch große Heterogenität der zugrunde liegenden Speicherformate.

Weitere Funktionen die durch Agenten gegenüber herkömmlichen Systemen effizienter zu realisieren sind liegen u.a. im Rating von Wissensbasen und –quellen. Dieses kann einerseits mittels Reputationsmechanismen geschehen. Andererseits sollten hier auch qualitäts- und kostenorientierten Kriterien verwendet werden. Dadurch ist es möglich, ein für den jeweiligen Nutzer optimiertes Preis-Leistungsverhältnis beim Wissenserwerb zu realisieren.

Wissensentwicklung

Wie bei der Untersuchung existierender agentenbasierter WMS festgestellt wurde, ist der Baustein der Wissensentwicklung bislang kaum Betrachtungsgegenstand bei Softwareagenten. Ein Grund liegt in der hohen Komplexität der Thematik. Vorstellbar ist ein Szenario, bei dem jeder Nutzer eines WMS über einen eigenen Agenten repräsentiert wird, der soweit personalisiert ist, dass er über ein Modell des Wissens seines Nutzers verfügt. Dann wäre es möglich, gezielt auf die Wissensentwicklung im Unternehmen einzuwirken, in dem strategische Soll-Vorgaben (Wissensziele) mit dem tatsächlichen Wissensbestand, repräsentiert durch die Agenten, abgeglichen

werden. Grundsätzlich ist dies jedoch kein spezifisch agentenbasiertes Szenario und kann auch durch herkömmliche Systeme realisiert werden.

Agententechnologie könnte einen Mehrwert in dezentralen Wissensentwicklungsansätzen leisten, bei denen der den Nutzer repräsentierende Agent selbständig Defizite in der Wissensentwicklung des Nutzers durch Koordination mit weiteren Agenten erkennt und Maßnahmen vorschlägt. Wirklich intelligente Agenten (eine Vision) wären darüber hinaus in der Lage, ihre eigenen Fähigkeiten laufend zu verbessern und somit die Wissensbasis der Organisation zu erweitern.

Wissensverteilung

Organisationen sind arbeitsteilig strukturiert, was zu einer Verteilung der Fachkenntnis, Problemlösungsfähigkeit und Verantwortlichkeit führt [EDAb2003; 4-5]. Diese nimmt aufgrund der Virtualisierung von Unternehmen und die Entstehung interorganisationaler Wissensnetzwerke weiter zu. Wegen der dezentralen Struktur und Verteilung der Akteure erscheint der Baustein Wissensverteilung besonders geeignet für den Einsatz von Agententechnologie. Der dezentrale Ansatz ermöglicht es, dass Agenten sich selbständig hinsichtlich des Austausches und somit der Verteilung von Wissen koordinieren, wodurch ein hoher Grad an Automatisierung erreichbar ist. Durch den Einsatz von Ontologien und Benutzermodellen ist lokales und globales Wissen gleichzeitig sinnvoll in unterschiedlichen Kontexten zu integrieren.

Wissensnutzung

Im Rahmen der Wissensnutzung dienen Agenten einerseits Aufgaben der personalisierten Wissenspräsentation gegenüber ihrem jeweiligen Nutzer. Hierzu wird ein Benutzermodell herangezogen, das Präferenzen des Nutzers abbildet. Andererseits können Agenten bei der Beschaffung und Integration von Wissen aus räumlich verteilten Quellen behilflich sein. So kann ein Agent automatisch erkennen, wenn für seinen Nutzer in der aktuellen Situation bestimmte Informationen relevant wären, die anderswo im Team oder bei externen Wissensquellen vorliegen. Er weist den Nutzer dann darauf hin bzw. beschafft und präsentiert die Informationen automatisch. Ähnliches gilt für die automatisierte Integration von Inhalten unterschiedlicher Wissensquellen, die für den Nutzer einen Erkenntnisgewinn darstellt.

Der Baustein der Wissensnutzung ist bislang durch agentenbasierte Anwendungen nur indirekt bzw. rudimentär adressiert, da insbesondere die Aufbereitung und Integration von unterschiedlichem Wissen hohe Anforderungen stellt. Durch eine Verzahnung zwischen Wissensbewertung und -nutzung lassen sich jedoch neue Potenziale für agentenbasierte Anwendungen erschließen. Das Ziel besteht darin, dem Nutzer vor allem für ihn hochwertige Informationen zu präsentieren. Dies kann durch qualitätsbezogene Bewertungen für Wissensinhalte gekoppelt mit kooperativen Verhandlungsprozessen von Wissensanbietern und Wissensnachfragern in Multi-agentensystemen erreicht werden.

Wissensbewahrung

Wirklich lernfähige Agenten wären grundsätzlich in der Lage, automatisch relevantes Wissen des Nutzers abzubilden und dezentral im WMS zur Verfügung zu stellen. Zumindest denkbar sind agentenbasierte Ansätze, die mit Hilfe von Lernalgorithmen Rückschlüsse auf Vorgehensweisen und Kenntnisse des Nutzers ziehen und diese anschließend als explizites Wissen repräsentieren. Jedoch sind solche Lernverfahren bislang hochkomplex und dadurch kaum zu implementieren. Einfacher realisierbar erscheinen dagegen Szenarien, bei denen Agenten eine Überwachungsfunktion in den Workflows von Wissensnetzwerken realisieren. Auf Basis spezifischer Fähigkeiten bemerken sie das Eintreten von Ereignissen, die für das Wissensmanagement potenziell relevant sind. Simple Beispiele sind der Abschluss eines größeren Projektes, das Ende einer Kooperation oder das Ausscheiden wichtiger Mitarbeiter. In so einer Situation sorgen sie dafür, dass Aktivitäten der Wissensbewahrung durchgeführt werden, in dem sie die Information an eine geeignete Stelle im Management weitergeben.²⁰

Zusammenfassend kann festgestellt werden, dass Agententechnologie vor allem in jenen Bausteinen des WM sinnvoll eingesetzt werden kann, die dezentrale, verteilte Strukturen mit einer hohen Dynamik der Umwelt aufweisen, wie Wissensidentifikation, -erwerb und -verteilung. Die weiteren Bausteine des WM weisen prinzipiell auch Potenzial zur Unterstützung durch agentenbasierte Anwendungen auf, ohne dass es hier allerdings bisher größere Anstrengungen gab. Hinzu kommt, dass Anwendungen, die nicht zwingend kooperatives Verhalten fordern und demzufolge auch keine natürliche Verteiltheit aufweisen, nicht notwendigerweise durch Agententechnologie realisiert werden sollten, sondern vielmehr auch durch herkömmliche Systeme unterstützt werden können.

3.4 Anwendungsmöglichkeiten von „Schwarmintelligenz“ im Wissensmanagement

Die bisherigen Überlegungen betrachteten intelligente Softwareagenten. Diese sind relativ komplex strukturiert und folgen meist dem Belief-Desire-Intention (BDI) Schema. Davon zu unterscheiden sind einfach strukturierte Stimulus-Response Agenten, auch als reaktive Agenten bezeichnet, die über keine komplexen internen Wissensrepräsentationen oder Schlussfolgerungsmechanismen verfügen, sondern relativ direkt auf eingehende Reize ihrer Umwelt reagieren. Solche einfachen Agenten werden aktuell intensiv im Kontext des Themas Schwarmverhalten bzw. kollektiver Intelligenz diskutiert. Unter kollektiver Intelligenz versteht man ein emergentes Phänomen. Durch Kommunikation innerhalb einer sozialen Gemeinschaft entsteht intelligentes Verhalten auf der Makroebene des sogenannten „Superorganismus“, d. h. der Summe aller Einzelindividuen [BDTh1999]. Ein typisches Beispiel hierfür ist der Ameisenstaat. Die für die beobachteten Fähigkeiten der Gruppe, z.B. im Nestbau oder beim Ausweichen von Hindernissen, notwendige Intelligenz ist bei keinem der beteiligten Individuen einzeln festzustellen, sondern das komplexe Verhalten wird mit Hilfe von Pheromonen erreicht.

20 Hierbei können bereits realisierte agentenbasierte Lösungen für andere Workflow-Kontexte, z.B. das Supply Chain Event Management [ZWBo2006], wichtige Hinweise zur Umsetzung liefern.

Als Schwarmintelligenz bezeichnet man ein Forschungsfeld der Verteilten Künstlichen Intelligenz (VKI), das auf Agententechnologie basierend versucht, künstliche Softwaresysteme nach dem Vorbild staatenbildender Lebewesen (Insekten, Vögel, Fische) zu entwerfen. Außerdem wird der Begriff Schwarmintelligenz benutzt, um eine in Schwärmen von Organismen bzw. Agenten entstehende Problemlösungsfähigkeit zu beschreiben, die sich durch eine kollektive Intelligenz im Schwarm erklären lässt. Typisch für alle in einem Schwarm enthaltenen Individuen ist ihre homogene Struktur und die Beschränkung auf einfache Verhaltensmuster, welche im wesentlichen auf die Operationen Bewerten, Vergleichen und Nachahmen reduziert werden können. Durch die Kombination dieser drei Operationen lassen sich auch komplexe Problemstellungen lösen.

Gemäß Bonabeau zeichnen sich schwarmintelligente Systeme durch eine hohe Flexibilität, Robustheit, sowie die Fähigkeit zur Selbstorganisation aus [BoMe2001]. Prototypische praktische Anwendungen von Schwarmintelligenz mit betriebswirtschaftlichem Bezug finden sich heute v.a. in komplexen Optimierungsproblemen, wie z.B. Job Shop Scheduling, Quadratic Assignment, Vehicle Routing und Travelling Salesman Problemen [Saft2006]. Daneben entstanden bereits frühzeitig erste Erklärungsmodelle für soziales Verhalten in größeren Gruppen (z.B. [EpAx1996]).

Aufgaben des Wissensmanagements können durch den Einsatz schwarmintelligenter Systeme bzw. durch Abstraktion bestimmter Verhaltensweisen von Schwärmen z.B. in folgender Weise unterstützt werden:

Wissensidentifikation:

Bei der Suche von Dokumenten oder bspw. Nachrichten wird der Suchende (Person oder Agent) oft mit für ihn unwichtigen Informationen konfrontiert. Eine Bewertung von Dokumenten durch jedes Individuum eines Kollektivs, die jeweils unmittelbar nach der Suche stattfindet, könnte dem Abhilfe schaffen. So ließe sich z.B. Expertenwissen zu bestimmten Themen in einer Datenbank mit „Pheromonen“ versehen, die proportional zur erreichten Befriedigung der Nachfrage nach dem Wissen zum Thema vergeben werden. Wissens Elemente ließen sich dann bei zukünftigen Recherchen proportional zu ihrer erreichten Bewertung hervorheben.

Wissenserwerb und –verteilung:

Soll im Rahmen kooperativer Modelle des Wissenserwerbs bzw. der Wissensverteilung auf einen zentralen Matchmaker zwischen Anbietern und Nachfragern von Wissen verzichtet werden, ohne gleichzeitig die Komplexität der Kommunikationsbeziehungen zu stark zu steigern, so bietet sich ein Verfahren an, das Kennedy und Eberhart [KeEb2001] entwickelt haben. Dieses so genannte Adaptive Culture Modell, das auf Arbeiten von Axelrod aufbaut, sieht vor, dass jeder Agent nur mit k Nachbarn (k liegt zwischen 2 und 4, kann aber auch anders gewählt werden) kommuniziert. Mit dieser Vorgehensweise des gezielten nachbarschaftlichen Informationsaustausches gelang es Kennedy und Eberhart, ohne methodische Komplexität optimale Lösungen zu Problemstellungen zu finden. Der Grundgedanke kann dabei als das Imitieren besserer Ergebnisse von Nachbarn beschrieben werden.

Alternativ bietet sich für selbstorganisatorische Elemente im Wissensmanagement das holonische Prinzip an, wie es bereits im Umfeld der Produktionssteuerung [Brus1994] erprobt worden ist. Der auf Koestler [Koes1967] zurückgehende Begriff des Holons bezeichnet ein Grundelement in Organisationen oder lebenden Organismen. Ein Holon ist zugleich Teil und Ganzes. Ein Teil in Bezug auf die ihm über-

geordneten Instanzen in der Hierarchie und ein Ganzes im Verhältnis zu den untergeordneten Subsystemen. Ein System weist holonisches Verhalten auf, wenn sich seine Subsysteme selbstständig steuern und gleichzeitig ihre Fähigkeiten bei der Verfolgung eines systemübergreifenden gemeinsamen Ziel kombinieren. Ameisenkolonien sind hier wiederum ein Beispiel.

Brussels Holonic Manufacturing System kann als Kombination von Elementen aus schwarmbasierten Überlegungen und intelligenten Agenten aufgefasst werden. Dort werden Holone zu autonomen, kooperativen und intelligenten Einheiten des Produktionssystems. Damit soll die herkömmliche hierarchische Kontrolle in Produktionssystemen umgangen werden, die ein genaues Modell der Produktionsanlage erfordert. Die Verteiltheit von Informationen und Kontrollfunktionen ist für holonische Systeme grundlegend. Jedes Holon ist Bestandteil eines übergeordneten Holons, kann seinerseits aber wiederum in verschiedene Holone unterteilt werden. Holone besitzen Fähigkeiten, verfügen über Wissen und verfolgen individuelle Ziele, können sich selbstständig anpassen und konfigurieren. Somit wird die Erreichung mehrerer Ziele (durch unterschiedliche Holone) möglich.

Es liegt nahe, dass auch in Wissensnetzwerken eine natürliche holonische Strukturierung gefunden und damit Selbststeuerung als Grundprinzip realisiert werden kann. Sind die Nutzer eines Wissensnetzwerkes durch Agenten repräsentiert, lassen sich Strukturierungs- und Kommunikationsprozesse in den Bereichen Wissensidentifikation, -erwerb, und -verteilung auf dieser Grundlage IT-technisch unterstützen oder automatisieren. Auch die Kombination verteilter Wissensselemente bei der Lösung konkreter Aufgabenstellungen (im Sinne der Wissensnutzung) ist nach diesem Prinzip unterstützbar, wie die Anwendung im Produktionsmanagement belegt.

Wissensentwicklung und -verteilung

Die Fähigkeit von Ameisen, Objekte auf dem Wege der Selbstorganisation zu sortieren, bietet einen weiteren Anhaltspunkt für das Wissensmanagement. Zunächst zufällig verteilte, gleichartige Elemente werden über die Zeit sortiert, in dem sie von Ameisen aufgenommen und in der Nähe von möglichst großen Ansammlungen jener Elemente wieder abgelegt werden [Dene91, S. 356 ff.]. Bonabeau et al. [BDTh1999; S.154-164] stellen hierzu weitere Ansätze vor. Letztlich können sich daraus Anwendungsmöglichkeiten für Gruppen einfach strukturierter Agenten im Data Mining (Wissensentwicklung) ergeben, z.B. bei der Identifikation von Kunden- oder Produktklassen. Die Elemente (Kunden/Produkte) werden dabei meist anhand ihrer, über alle untersuchten Merkmale aggregierten Unterschiede (Distanzen) zueinander in verschiedene Klassen verteilt, wobei die Verschiedenheit der Objekte innerhalb einer Klasse sehr gering, und die Verschiedenheit der Objekte zwischen zwei Klassen sehr groß sein soll. In analoger Weise könnten sich agentengestützt auch automatisch Wissenscluster innerhalb großer Netzwerke bzw. Wissenssammlungen identifizieren lassen, die dann zu einer gezielten Verteilung oder Recherche von Informationen genutzt werden können.

Die hier genannten Beispiele sind erste Vorschläge. Wir haben gerade erst begonnen, die vielfältigen Anwendungsmöglichkeiten von Schwarm Intelligenz im Wissensmanagement zu identifizieren.

Literaturverzeichnis

- [AFMS1995] Arnold, O.; Faisst, W.; Härtling, M.; Sieber, P.: Virtuelle Unternehmen als Unternehmenstyp der Zukunft? Handbuch der modernen Datenverarbeitung, 1995, S. 8-23.
- [Anth2003] Anthes, G. H.: Agents of Change. www.computerworld.com, 2003.
- [ANNR2001] Angehrn, A. A.; Nabeth, T.; Razmerita, L.; Roda, C.: K-InCA: Using Artificial Agents for Helping People to Learn New Behaviours. In: Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT 2001). Madison 2001, S. 225-226.
- [Bick2004] Bickhoff, N.; Bieger, T.; Caspers, R.: Wissen, Netzwerke, Wissensnetzwerke: Zur Aktualität des Themas. In: Caspers, R.; Bickhoff, N.; Bieger, T. (Hrsg.): Interorganisatorische Wissensnetzwerke – Mit Kooperation zum Erfolg. Springer, Berlin, 2004, S. 1-13.
- [BDTh1999] Bonabeau, E.; Dorigo, M.; Theraulaz, G.: Swarm Intelligence. From natural to artificial systems., Oxford University Press, New York 1999.
- [BoMe2001] Bonabeau, E.; Meyer, C.: Swarm Intelligence. A whole new way to think about business. In: Harvard Business Review, May 2001, Harvard Business School, 2001.
- [BoBo2002] Bonifacio, M.; Bouquet, P.: Distributed Knowledge Management : A Systemic Approach. http://www.dthink.biz/egportal/DThink/documenti_sito/DKM-3.pdf, 2002, Abruf am 2006-03-06.
- [BBCu2002] Bonifacio, M.; Bouquet, P.; Cuel, R.: The role of classification(s) in distributed knowledge management. <http://edamok.itc.it/documents/papers/kes20021.pdf>, 2002, Abruf am 13.11.2006.
- [BBTr2002] Bonifacio, M.; Bouquet, P.; Traverso, P.: Enabling Distributed Knowledge Management: Managerial and Technological Implications. In: Informatik : Zeitschrift der schweizerischen Informatikgesellschaft (2002) 1, S. 23-29.
- [BeCz2006] Becker, M.; Czap, H.: Artificial Software Agents as Representatives of their Human Principals in Operating-Room-Team-Forming. In: Kirn, S.; Herzog, H.; Lockemann, P.; Spaniol, O.: Multiagent Engineering. Theory and Applications in Enterprises. Springer, Berlin, 2006, 221-237.
- [BDEL+2004] Bernardi, A.; Dengel, A.; van Elst, L.; Lauer, A.; Maus, H.; Schwarz, S.: FRODO : Ein Agentenframework für verteilte Unternehmensgedächtnisse. In: KI – Künstliche Intelligenz 18 (2004) 3, S. 59-61.
- [BoBe2003] Bodrow, W.; Bergmann, P.: Wissensbewertung in Unternehmen: Bilanzieren von intellektuellem Kapital, Berlin: Erich Schmidt, 2003.
- [BoGa1988] Bond, A.; Gasser, L.: Readings in Distributed Artificial Intelligence, Morgan Kaufmann Publisher Inc., San Mateo, 1988.
- [BBMA2000] Bonifacio, M.; Bouquet, P.; Manzardo, A.: A Distributed Intelligence Paradigm for Knowledge Management. <http://eprints.biblio.unitn.it/archive/00000221/01/68.pdf>, 2000, Abruf am 2006-03-08.
- [BoZi2005] Bodendorf, F.; Zimmermann, R.: Proactive Supply Chain Event Management with Agent Technology. In: International Journal of Electronic Commerce 9 (2005) 4, S. 57 – 90.

- [Brus1994] Brussel, H. Van: Holonic Manufacturing Systems. The Vision Matching the Problem. 1994. <http://citeseer.ist.psu.edu/vanbrussel94holonic.html>, Abruf 22.09.2005.
- [Buss1998] Bussmann, S.: Agent-oriented programming of manufacturing control tasks. In: Proceedings of Third International Conference on Multi-Agent Systems (ICMAS'98). IEEE Computer Society, 1998, 57-63.
- [CoJe1996] Cockburn, D.; Jennings, N.R.: Archon: A Distributed Artificial Intelligence System for Industrial Applications, In G.M.P. O'Hare, N.R. Jennings (Hrsg.), Foundations of Distributed Artificial Intelligence, New York, John Wiley, 1996, S. 319-344.
- [CWWr2000] Chen, J. R.; Wolfe, S. R.; Wragg, S. D.: A Distributed Multi-Agent System for Collaborative Information Management and Sharing. In: Proceedings of the 9th ACM International Conference on Information and Knowledge Management. ACM Press, New York 2000.
- [DABE*2002] Dengel, A.; Abecker, A.; Bernardi, A.; van Elst, L.; Maus, H.; Schwarz, S.; Sintek, M.: Konzepte zur Gestaltung von Unternehmensgedächtnissen. In: KI – Künstliche Intelligenz 16 (2002) 1, S. 5-11
- [DGFr91] Deneubourg, J.-L.; Goss, N.; Franks, A.: The Dynamics of Collective Sorting, In: Proceedings of the First European Conference on AI, MIT Press, Cambridge 1991, S. 356-365.
- [Dign2002] Dignum, V.: A Knowledge Sharing Model for Peer Collaboration in the Non-Life Insurance Domain. In: Minor, Mirjam; Staab, Steffen (Hrsg.): Sharing experiences about the sharing of experience : 1st German Workshop on Experience Management. Gesellschaft für Informatik. Berlin 2002.
- [Dign2004] Dignum, V.: An Overview of Agents in Knowledge Management. <http://www.cs.uu.nl/~virginia/publications/UU-CS-2004-017.pdf>, 2004, Abruf am 28-03-2006.
- [EDAb2003] van Elst, L.; Dignum, V.; Abecker, A.: Towards Agent-Mediated Knowledge Management. In: van Elst, L.; Dignum, V.; Abecker, A. (Hrsg.): Agent-Mediated Knowledge Management : International Symposium AMKM 2003, Springer, Berlin 2003.
- [EpAx1996] Epstein, J. M.; Axtell R.: Growing Artificial Societies. Social Science from the bottom up., Brookings Institution Press, 1996.
- [Ethn2006] Ethner, F.: Verteiltes Wissensmanagement am Beispiel des Einsatzes von Intelligenen Softwareagenten. Unveröffentlichte Diplomarbeit am Fachgebiet Wirtschaftsinformatik für Dienstleistungen (Prof. Dr. Volker Nissen), TU Ilmenau, 2006.
- [EMSt2006] Eymann; T.; Müller, G.; Strasser, M.: Self-Organized Scheduling in Hospitals by Connecting Agents and Mobile Devices. In: Kirn, S.; Herzog, H.; Lockemann, P.; Spaniol, O.: Multiagent Engineering. Theory and Applications in Enterprises. Springer, Berlin, 2006, 319-337.
- [FrGr1996] Franklin, S.; Graesser, A.: Is It an agent, or just a program?: A taxonomy for autonomous agents. In: Müller, J.; Wooldridge, M.; Jennings, N. (eds.): Intelligent Agents III. Springer, Berlin, 1996, S. 21-36.
- [Hatt2002] Hattendorf, M.: Wissensmanagement – Problemprozesse und Problem Prozesse, in: Mohe, M.; Heinecke, H.J.; Pfriem, R. (Hrsg.): Consulting – Problemlösung als Geschäftsmodell: Theorie, Praxis, Markt, Stuttgart: Klett-Cotta, 2002, S. 180 – 194.
- [Herb2000] Herbst, D.: Erfolgsfaktor Wissensmanagement, Berlin: Cornelsen, 2000.

- [HNTi1999] Hansen, M.T.; Nohria, N.; Tierney, T.: What's Your Strategy For Managing Knowledge?, In: *Harvard Business Review* 77 (1999) 2, S. 106 – 116.
- [KeEb2001] Kennedy, J.; Eberhart, R. C.: *Swarm Intelligence*, Morgan Kaufman Publishers, San Francisco, 2001.
- [KaKI1997] Kandzia, P.; Klusch, M. (Hrsg.): *Cooperative Information Agents. Proceedings of First International Workshop on Cooperative Information Agents*. Springer, Berlin, 1997.
- [KHLS2006] Kirn, S.; Herzog, H.; Lockemann, P.; Spaniol, O.: *Multiagent Engineering. Theory and Applications in Enterprises*. Springer, Berlin, 2006.
- [KIBe1998] Klusch, M; Benn, W.: Intelligente Informationsagenten im Internet. In: *KI Künstliche Intelligenz, Themenheft Intelligente Informationsagenten*, (1998) 3, S. 8-17.
- [Klus2001] Klusch, M.: Information Agent Technology for the Internet: A Survey. In: *Journal on Data and Knowledge Engineering, Special Issue on Intelligent Information Integration*, D. Fensel (Ed.), Vol. 36 (2001) 2, S. 337-372.
- [Koes1967] Koestler, A.: *The Ghost in the Machine*. Hutchinson, London 1967.
- [KrGr2004] Kreis-Hoyer, P.; Grünberg, J.: Definition und Typologie von Wissensnetzwerken. In: Caspers, R.; Bickhoff, N.; Bieger, T. (Hrsg.): *Interorganisatorische Wissensnetzwerke – Mit Kooperation zum Erfolg*. Springer, Berlin 2004, S. 109-148.
- [LWDS⁺2006] Lorenzen, L.E.; Woelk, P.-O.; Denkena, B.; Scholz, T.; Timm, I.; Herzog, O.: Iterated Process Planning and Production Control. In: Kirn, S.; Herzog, H.; Lockemann, P.; Spaniol, O.: *Multiagent Engineering. Theory and Applications in Enterprises*. Springer, Berlin, 2006, 91-113.
- [MiSn1986] Miles, E.R. ; Snow, C.C.: *Organization: New Concepts for New Forms*, California Management Review, 28-3, 1986, S. 62-73.
- [MSZi2006] Mönch, L.; Stehli, M.; Zimmermann, J.: Distributed Hierarchical Production Control for Wafer Fabs Using an Agent-Based System Prototype. In: Kirn, S.; Herzog, H.; Lockemann, P.; Spaniol, O.: *Multiagent Engineering. Theory and Applications in Enterprises*. Springer, Berlin, 2006, 135-156.
- [MuJo2000] Murch, R.; Johnson, T.: *Agententechnologie: Die Einführung*, Addison Wesley, München, 2000.
- [Müll1997] Müller, J.P.: Towards Agents Systems Engineering. *International Journal on Data and Knowledge Engineering*, In: Special Issue on Distributed Expertise 23. 1997, S. 217 – 245.
- [Nort2002] North, K.: *Wissensorientierte Unternehmensführung: Wertschöpfung durch Wissen*, 3. Aufl., Wiesbaden: Gabler, 2002.
- [Niss2003] Nissen, V.: Unterstützung ausgewählter Aspekte des Wissensmanagements, in: Biethahn, J.; Kuhl, J.; Lackner, A. (Hrsg.): *Soft-Computing in Wissenschaft und Wirtschaft. Tagungsband 6. Göttinger Symposium Soft Computing*, Göttingen: Cuvillier, 2003, S. 53 – 75.
- [NSBo2000] Neumann, S.; Schuurmans, L.; Bonifacio, M.: Verteilte Intelligente Systeme im Wissensmanagement. In: *IM Information Management & Consulting* 15 (2000) 2, S. 75-82.

- [oV2006a] o.V.: Integration vernichtet Arbeitsplätze. Prozessoptimierung trifft vor allem die Verwaltung in großen Konzernen. In: Computer Zeitung (2006) 32-33 (14.08.2006), S. 4.
- [Paru2000] Parunak, H.V.D.: A Practitioners' Review of Industrial Agent Application. In: Autonomous Agents and Multi-Agent Systems 3 (2000) 4, S. 389-407.
- [PHGM⁺1999] Preece, A.; Hui, K.; Gray, A.; Marti, P.; Bench-Capon, T.; Jones, D.; Cui, Z.: The KRAFT Architecture for Knowledge Fusion and Transformation. In: Ellis, R.; Moulton, M.; Coenen, F. (Hrsg.): Applications and innovations in intelligent systems VII : Proceedings of ES99. Springer, Berlin u.a., 1999, S. 113-120.
- [PRRo1999] Probst, G.; Raub, S.; Romhardt, K.: Wissen managen: Wie Unternehmen ihre wertvollste Ressource optimal nutzen, 3. Aufl., Frankfurt: FAZ Verlag und Wiesbaden: Gabler, 1999.
- [PZRH⁺2006] Paulussen, T.; Zöller, A.; Rothlauf, F.; Heinzl, A.; Braubach, L.; Pokahr, A.; Lamersdorf, W.: Agent-Based Patient Scheduling in Hospitals. In: Kirn, S.; Herzog, H.; Lockemann, P.; Spaniol, O.: Multiagent Engineering. Theory and Applications in Enterprises. Springer, Berlin, 2006, 239-254.
- [RANa2001] Roda, C.; Angehrn, A. A.; Nabeth, T.: Matching Competencies to Enhance Organizational Knowledge Sharing: An Intelligent Agents Approach. In: Proceedings of the 7th International Netties Conference. Fribourg 2001, S. 931-937.
- [ReKr1996] Rehäuser, J.; Krcmar, H.: Wissensmanagement im Unternehmen, in: Schreyögg, G.; Conrad, P. (Hrsg.): Managementforschung 6: Wissensmanagement, Berlin und New York: Walter de Gruyter, 1996, S. 1 – 40.
- [Romh1998] Romhardt, K.: Die Organisation aus der Wissensperspektive: Möglichkeiten und Grenzen der Intervention, Wiesbaden: Gabler, 1998.
- [RSKF2006] Rose, T.; Sedlmayr, M.; Knublauch, H.; Friesdorf, W.: Agent-Based Information Logistics. In: Kirn, S.; Herzog, H.; Lockemann, P.; Spaniol, O.: Multiagent Engineering. Theory and Applications in Enterprises. Springer, Berlin, 2006, 239-254.
- [Rugg1998] Ruggles, R.L.: The State of the Notion: Knowledge Management in Practice, in: California Management Review, Bd. 40, Nr. 3, 1998, S. 80 – 89.
- [RuNo2003] Russel, S.; Norvig, P.: Artificial Intelligence : a modern approach. 2. ed. Prentice-Hall, New Jersey 2003.
- [Saft2006] Saft, D.: Schwarm Intelligenz: State-of-the-Art und betriebswirtschaftliche Anwendungspotenziale. Unveröffentlichte Projektarbeit am Fachgebiet Wirtschaftsinformatik für Dienstleistungen (Prof. Dr. Volker Nissen) der TU Ilmenau, 2006.
- [Schü1995] Schüppel, J.: Organisationslernen und Wissensmanagement, in: Geißler, H. (Hrsg.): Organisationslernen und Weiterbildung, Neuwied u.a., Luchterhand, 1995, S. 185 – 220.
- [Seuf2002] Seufert, A.; Back, A.; von Krogh, G.: Wissensnetzwerke: Vision – Referenzmodell - Archetypen und Fallbeispiele. In: Götz, K. (Hrsg.): Wissensmanagement – Zwischen Wissen und Nichtwissen. 4., verb. Aufl., Rainer Hampp, München 2002, S. 129-154.
- [SDPW⁺1996] Sycara, K.; Decker, K.; Pannu, A.; Williamson, M.; Zeng, D.: Distributed Intelligent Agents. In: IEEE Expert, Nr.6, Vol. 11, 1996, S. 36-46.

- [SMSG1998] Schüppel, J.; Müller-Stewens, G.; Gomez, P.: The Knowledge Spiral, in: Krogh, G.v.; Roos, J.; Kleine, D. (Hrsg.): *Knowing in Firms: Understanding, Managing and Measuring Knowledge*, London et al.: SAGE Publications, 1998, S. 223 – 239.
- [Skr1998] Skyrme, D.: *Measuring the Value of Knowledge: Metrics for the Knowledge-Based Business*, London: Business Intelligence Ltd., 1998.
- [SWKö2006] Stockheim, T.; Wendt, O.; König, W.: Trust-Based Distributed Supply-Web Negotiations. In: Kirn, S.; Herzog, H.; Lockemann, P.; Spaniol, O.: *Multiagent Engineering. Theory and Applications in Enterprises*. Springer, Berlin, 2006, 177-195.
- [WoJe1995] Wooldridge, M.; Jennings, J. R.: Agent Theories, Architectures, and Languages: a Survey. In: Wooldridge, M.; Jennings J. R. (Hrsg.): *Intelligent Agents*. Springer, Berlin, 1995, S. 1-22.
- [Wool2002] Wooldridge, M.: *An Introduction to MultiAgent Systems*. John Wiley & Sons, New York, 2002.
- [WöWö2006] Wörner, J.; Wörn, H. Benchmarking of Multiagent Systems in a Production Planning and Control Environment. In: Kirn, S.; Herzog, H.; Lockemann, P.; Spaniol, O.: *Multiagent Engineering. Theory and Applications in Enterprises*. Springer, Berlin, 2006, 115-133.
- [WRZN2006] Woelk, P.-O.; Rudzio, H.; Zimmermann, R.; Nimis, J.: *Agent Enterprise in a Nutshell*.
- [Zarn1999] Zarnekow, R.: *Softwareagenten und elektronische Kaufprozesse – Referenzmodelle zur Integration*. Deutscher Universitätsverlag, Wiesbaden, 1999.
- [Zimm2006] Zimmermann, R.: *Agent Based Supply Network Event Management*. Basel u.a.: Birkhäuser, 2006.
- [ZWBo2006] Zimmermann, R.; Winkler, S.; Bodendorf, F.: Supply Chain Event Management With Software Agents. In: Kirn, S.; Herzog, H.; Lockemann, P.; Spaniol, O.: *Multiagent Engineering. Theory and Applications in Enterprises*. Springer, Berlin, 2006, 157-175.

Robustheit von Reputationssystemen: Ein evolutionärer Bewertungsansatz auf Basis eines Multiagentensystems

Ivo Reitzenstein

Lehrstuhl für E-Business des Instituts für Wirtschaftsinformatik und
Operations Research der Martin-Luther-Universität Halle-Wittenberg
Universitätsring 3
06108 Halle (Saale)
ivo.reitzenstein@wiwi.uni-halle.de

Zusammenfassung: Im Internet werden Reputationssysteme eingesetzt, um Vertrauen zwischen Akteuren zu erzeugen. Diese Systeme besitzen jedoch Schwachstellen und können daher manipuliert werden. Dieser Beitrag stellt einen evolutionären Bewertungsansatz vor, mit dem die Manipulierbarkeit eines Reputationssystems auf Basis einer Multi-Agenten-Simulation untersucht werden kann.

1 Einleitung

1.1 Problemstellung

Um Vertrauen zwischen Akteuren zu erzeugen, werden im Internet Reputationssysteme eingesetzt [DeRe2003, 1; Dela2003a, 1407]. Das wohl bekannteste Beispiel hierfür stellt das auf eBay eingesetzte Peer Rating dar, bei dem sich die Teilnehmer des Marktplatzes gegenseitig nach einer Geschäftstransaktion bewerten können. Eine andere Anwendung für Reputationssysteme sind Diskussionsforen oder Experten-Websites wie „askme.com“, bei denen die Beiträge der Teilnehmer bewertet werden. Neben dem Consumer-Bereich werden Reputationssysteme auch im Business-Bereich eingesetzt. Ein Beispiel hierfür stellt die Bewertungsplattform „Better Business Bureau“ dar, bei der Beschwerden über Firmen aggregiert werden.²¹

Wie bereits anhand der aufgeführten Beispiele deutlich wird, werden in einem Reputationssystem Informationen über das bisherige Verhalten von Akteuren gesammelt, aggregiert und schließlich anderen Akteuren zugänglich gemacht [RZFK2000, 46]. Dadurch ist es möglich, sich im Vorhinein über den potenziellen Transaktionspartner zu informieren, z. B. ob sich dieser in bisherigen Transaktionen vertrauenswürdig verhalten hat. Damit lassen sich Transaktionen mit den Partnern

²¹ Ein Überblick über weitere Anwendungen von Reputationssystemen findet sich in [Dela2003, S. 1408].

vermeiden, die eine zu geringe Vertrauenswürdigkeit aufweisen. Umgekehrt entsteht für den Transaktionspartner der Anreiz, sich vertrauenswürdig zu verhalten, da sich eine Schädigung der Reputation negativ auf zukünftige Transaktionen auswirkt. Axelrod bezeichnet diese Situation deshalb auch als den „shadow of the future“ [Axel1990, 126-133]. Andererseits können opportunistisch handelnde Akteure auch bestrebt sein, sich durch eine Manipulation des Reputationssystems Vorteile zu verschaffen. Auf dem Marktplatz eBay zeigen sich die Auswirkungen solchen Verhaltens in der Praxis. Hier lässt sich beobachten, dass Betrugsfällen meist eine Manipulation des Reputationssystems vorausgeht.²²

In der Literatur werden Manipulationsmöglichkeiten von Reputationssystemen bereits ausführlich diskutiert.²³ Ein Grossteil der Publikationen konzentriert sich jedoch ausschließlich auf Reputationssysteme, wie sie bei eBay eingesetzt werden. Wie sich solche Manipulationen bei anderen Reputationssystemen auswirken, ist kaum bekannt. Außerdem ist es schwierig vorherzusagen, ob die bereits vorgeschlagenen Maßnahmen zur Behebung der beschriebenen Probleme auch bei anderen Reputationssystemen wirksam sind. Des Weiteren erlauben die bisherigen Ansätze nur eine isolierte Betrachtung der Probleme von Reputationssystemen. In der Praxis ist jedoch zu erwarten, dass viele verschiedene Angreifer gleichzeitig agieren und somit Interdependenzen zwischen den verschiedenen Problemen auftreten. Dies wird jedoch durch die bisherigen Analysen nicht berücksichtigt. Zwar können die bislang vorgeschlagenen Maßnahmen geeignet sein, eine Manipulationsmöglichkeit, wie z. B. den Identitätsmissbrauch zu beseitigen. Es ist jedoch denkbar, dass die Anwendung dieser Maßnahmen dazu führt, dass Angreifer einfach auf andere Manipulationsstrategien ausweichen. Eine Verbesserung der Funktionsfähigkeit des Reputationssystems ist in diesem Fall nicht gegeben. Ein weiterer Kritikpunkt sind die stark vereinfachenden Annahmen, unter denen diese Analysen vorgenommen werden um eine analytische Lösung zu erzielen. Bisherige Ansätze werden deshalb dem Vorwurf fehlender Realitätsnähe ausgesetzt [Dela2003, 1421; ReZe2002]. Dies stellt somit auch die Praxistauglichkeit der bereits vorgeschlagenen Maßnahmen zur Beseitigung der Manipulationsmöglichkeiten von Reputationssystemen in Frage.

Die hier beschriebenen Grenzen der bisherigen Forschungsansätze zeigen, dass eine ganzheitliche Betrachtung der Manipulationsmöglichkeiten von Reputationssystemen erforderlich ist. Dafür wird jedoch ein neuer Ansatz benötigt. Insbesondere ist es wünschenswert, verschiedene Reputationssysteme in die Analyse einzubeziehen. Des Weiteren sollte dieser Ansatz eine möglichst nahe Abbildung der Realität ermöglichen. Die Anwendung von Konzepten der Evolutionären Spieltheorie im Rahmen einer Multi-Agenten-Simulation bietet eine viel versprechende Methode, um die Robustheit von Reputationssystemen gegenüber Manipulation zu analysieren.²⁴

22 vgl. <http://www.golem.de/0408/33299.html>

23 In Kapitel 2 werden die Forschungsansätze zu Manipulationsmöglichkeiten von Reputationssystemen ausführlich vorgestellt.

24 Es gibt bereits Publikationen, die Konzepte der Evolutionären Spieltheorie anwenden. In [MHMo2002] wird die Eignung unterschiedlicher Reputationssysteme zur Durchsetzung kooperativen Verhaltens mit Hilfe eines evolutionären Ansatzes verglichen. Ein umfassendes Konzept zur Bewertung und dem Vergleich der Robustheit von Reputationssystemen gegenüber Manipulationen existiert jedoch bislang scheinbar noch nicht.

1.2 Ziel des Beitrags

In diesem Beitrag wird das Konzept einer Multi-Agenten-Simulation vorgestellt, mit der die Robustheit von Reputationssystemen gegenüber Manipulationen untersucht werden kann. Dieser Simulation liegt ein Bewertungsansatz zugrunde, der auf Konzepten der evolutionären Spieltheorie basiert. Dadurch soll im Vergleich zu bisherigen Forschungsansätzen eine realitätsnähere Analyse von Reputationssystemen ermöglicht werden.

1.3 Vorgehensweise

Im zweiten Abschnitt wird ein Überblick über Manipulationsmöglichkeiten von Reputationssystemen gegeben. Dabei werden bereits in der Literatur vorgeschlagene Lösungsansätze vorgestellt, mit denen diese Probleme behoben werden sollen.

Im dritten Abschnitt werden die Grundlagen des hier vorgestellten Bewertungsansatzes zur Analyse der Robustheit von Reputationssystemen vorgestellt. Dazu wird das Konzept der Evolutionären Spieltheorie vorgestellt. In einem weiteren Schritt wird auf die Modellierung evolutionärer Spiele eingegangen.

Im vierten Abschnitt wird auf der Grundlage der beschriebenen Konzepte ein Bewertungsansatz zur Ermittlung der Robustheit von Reputationssystemen vorgestellt. Dazu wird ein evolutionäres Modell zur Ermittlung der Robustheit eines einfachen Reputationssystems entworfen. In einem weiteren Schritt werden Strategien der beteiligten Akteure beschrieben.

Im fünften Abschnitt wird diskutiert, wie das im vierten Abschnitt entwickelte Modell in einer Multi-Agenten-Simulation umgesetzt werden kann.

Im letzten Abschnitt werden die zentralen Aspekte dieses Beitrages zusammengefasst und einer kritischen Würdigung unterzogen. Abschließend werden die nächsten Schritte des hier vorgestellten Forschungsansatzes vorgestellt.

2 Manipulation von Reputationssystemen

Die meisten in der Praxis eingesetzten Reputationssysteme basieren auf einfach nachvollziehbaren Konzepten, z. B. der Abgabe positiver, negativer und neutraler Bewertungen bei eBay. Dennoch wirft der Einsatz solcher Systeme eine Reihe von Fragen auf [DeRe2003]. Mit der zunehmenden Bedeutung in der Praxis sind Reputationssysteme deshalb verstärkt in das Blickfeld der Forschung gerückt. Die Forschungsschwerpunkte liegen vor allem in der *Untersuchung der Effizienz bestehender Reputationssysteme*, dem *Entwurf neuer Reputationssysteme* sowie der *Untersuchung der Anwendbarkeit von Reputationssystemen in neuen Anwendungsfeldern*, z. B. im Rahmen von Multi-Agenten-Systemen oder Peer-to-Peer-Netzwerken. Innerhalb dieser Forschungsschwerpunkte hat sich herausgestellt, dass bestimmte Konstruktionsmerkmale von Reputationssystemen eine Grundlage für Angriffe oder Manipulationen darstellen können

Eine Problematik stellt die einfache Erlangung von Identitäten im Internet dar [FrRe1998]. So kann ein potenzieller Angreifer durch den Wechsel seiner Identität seine negative Reputation „abstreifen“ und sich damit den negativen Folgen seines

Handelns entziehen. Ein solches Verhalten wird deshalb auch als „whitewashing“ bezeichnet. Um den Anreiz zum Identitätswechsel zu beseitigen, schlagen einige Publikationen vor, Neulinge mit dem schlechtest möglichen Reputationswert beginnen zu lassen [Dela2003b; ZMMa1999]. Andere Ansätze schlagen vor, den Anreiz für einen Identitätswechsel durch die Einführung einer Anmeldegebühr oder durch den Einsatz starker Authentifizierung²⁵ zu verringern [FrRe1998].

Ein weiteres Problem ist, dass nur schwer nachvollziehbar ist, ob die abgegebene Bewertung das tatsächliche Ergebnis der Transaktion widerspiegelt [DMSy2003]. Potenzielle Angreifer können dies ausnutzen und sich durch die gezielte Abgabe von Bewertungen Vorteile verschaffen. Eine Möglichkeit ist die Drohung mit einer negativen Bewertung, um ein bestimmtes Verhalten des Transaktionspartners zu erzwingen [KeSc2002, 24]. Eine andere Möglichkeit besteht darin, entweder das eigene Reputationsprofil durch positive Bewertungen aufzuwerten (ballot stuffing) oder andere Teilnehmer durch die Abgabe negativer Bewertungen zu denunzieren (bad mouthing) [Dela2000, 2; ZMMa1999, 3]. Ein solcher Angriff kann in Kooperation mit anderen Akteuren oder im Rahmen einer „Sybil Attack“ [Douc2002] unter verschiedenen Identitäten durchgeführt werden. Ansätze zur Behebung dieser Problematik bestehen darin, auffällig abweichende Bewertungen mittels Cluster Analyse aufzudecken [Dela2000] oder im Rahmen einer „scoring rule“ zu bestrafen [MRZe2004]. Andere Autoren schlagen die Verwendung von Metareputations-systemen vor, bei denen die „Richtigkeit“ abgegebener Bewertungen wiederum von anderen Teilnehmern bewertet werden kann.

Eine andere Problematik betrifft die mangelnde Aussagekraft der Reputation über das zukünftige Verhalten der Akteure. Da Reputationssysteme nur Informationen über das bisherige Verhalten der Akteure sammeln, sagen diese nichts über eine mögliche Änderung des Verhaltens in zukünftigen Transaktionen aus. Beispielsweise besteht die Gefahr, dass Akteure ihre gute Reputation gezielt ausnutzen und sich opportunistisch verhalten. Dies wird deshalb auch als „rest-on-the-laurels“-Strategie bezeichnet [Dela2003a, 1419]. Lösungsansätze schlagen eine stärkere Gewichtung neuere Bewertungen bzw. die Abwertung bzw. das Verfallen lassen „älterer“ Bewertungen zur Behebung dieser Problematik vor [Dela2003a, 1419f].

3 Spieltheoretische Grundlagen

Der in diesem Beitrag vorgestellte Ansatz zur Analyse der Robustheit von Reputationssystemen basiert auf Konzepten der Evolutionären Spieltheorie. Diese werden deshalb in diesem Abschnitt kurz erläutert. Insbesondere wird auf die Modellierung evolutionärer Spiele eingegangen.

3.1 Das Konzept der evolutionären Spieltheorie

Mit Hilfe der Evolutionären Spieltheorie lässt sich die Entwicklung von Verhaltensweisen²⁶ untersuchen und deren Erfolg in strategischen Entscheidungssituationen

25 Mit Hilfe starker Authentifizierung soll erreicht werden, dass eine einmal zugewiesene Identität nicht mehr gewechselt werden kann.

26 Im Rahmen der evolutionären Spieltheorie werden Verhaltensweisen als Strategien bezeichnet. Im Folgenden wird dieser Begriff weiter verwendet.

beurteilen [HoSi2002, 3; Aman1999, 1f.; Pete1998, 1f.]. Solche Situationen sind dadurch gekennzeichnet, dass Entscheidungen von Individuen wechselseitig voneinander abhängen und damit Auswirkungen auf das Ergebnis dieser Interaktion haben [Holl2000, S. 1]. In der klassischen Spieltheorie werden ähnliche Fragestellungen untersucht. Allerdings verfolgt die evolutionäre Spieltheorie im Gegensatz dazu einen anderen Ansatz, indem Konzepte aus der Biologie auf ökonomische Fragestellungen angewandt werden [Aman1999, S 1f.; Frie1991; Pete1998, S. 1f.].

Die evolutionäre Spieltheorie relativiert das strenge Konzept des rational handelnden Individuums. Das Konzept der Rationalität besagt, dass sich ein Individuum immer für die Alternative entscheidet, welche den individuellen Nutzen maximiert [Vari1994, 98f.]. Dabei wird angenommen, dass alle zur Verfügung stehenden Informationen und Handlungsmöglichkeiten bei der Entscheidungsfindung berücksichtigt werden [Aman1999, 30]. Da sich Individuen in der Praxis nicht immer rational verhalten, wird dieses Konzept in der Literatur wegen mangelnder Realitätsnähe kritisiert. Der Anlass zu dieser Kritik ist, dass Entscheidungen, die durch Anwendung von Heuristiken, auf der Grundlage gesellschaftlicher Normen, Konventionen oder aus Gewohnheit getroffen werden zu einem Verstoß gegen das Prinzip der Rationalität führen [Samu1997, 15]. Bei der evolutionären Spieltheorie ist solches nicht-rationale Verhalten zulässig. Die Wahl einer Strategie erfolgt nun nach einem anderen Ansatz. So wird angenommen, dass Individuen eine bestimmte Strategie verfolgen, diese jedoch von Zeit zu Zeit überdenken und sie ändern, falls eine andere Strategie erfolgreicher ist [Samu1997, 87; Aman1999, 48].

Dies führt zu einem Selektionsprozess, bei dem nach und nach die weniger erfolgreichen Strategien verdrängt werden. Dabei können sich Gleichgewichte bilden, bei denen die Selektion nicht mehr zu einer Änderung der Anteile der Strategien führt. Es ist jedoch möglich, dass neue Strategien auftreten (Mutation) und ein vorhandenes Gleichgewicht stören [Smit1993]. In diesem Fall stellt sich die Frage, ob der Selektionsprozess wieder zu dem Ausgangszustand vor der Mutation führt und damit das betrachtete Gleichgewicht robust ist oder ob diese Mutation zur Bildung eines neuen Gleichgewichtes führt.

3.2 Modellierung evolutionärer Spiele

Der Ausgangspunkt für die Modellierung evolutionärer Spiele ist das Konzept der Population. Diese setzt sich aus mehreren Individuen zusammen, die jeweils genau eine Strategie aus einer Menge möglicher Strategien verfolgen [Holl2000, S. 353-355; Pete1998, S. 1]. Damit lässt sich eine Strategie immer genau einem Populationsanteil zuordnen. In evolutionären Spielen wird typischerweise von sehr großen Populationen ausgegangen.

Im Rahmen eines evolutionären Spiels treten wiederholt paarweise zufällig ausgewählte Individuen miteinander in Interaktion (random matching) [Holl2000, 353]. Der Erfolg dieser Interaktion wird im Kontext der evolutionären Spieltheorie durch die Fitness bewertet [Weib2002, 33]. Diese hängt von den Strategien ab, welche die an der Interaktion beteiligten Akteure verfolgen. Das Konstrukt der Fitness ist jedoch nicht genau definiert und kann mit Hilfe unterschiedlicher Indikatoren bewertet werden. In ökonomischen Modellen wird die Fitness jedoch häufig anhand der Auszahlung bemessen, die sich aus der beschriebenen Interaktion der Individuen ergibt.

Wie bereits im vorigen Abschnitt beschrieben wurde, wird unterstellt, dass sich Strategien mit einer überdurchschnittlichen Fitness in einer Population stärker ausbreiten und die Populationsanteile mit Strategien unterdurchschnittlicher Fitness abnehmen. Dies lässt sich mit einem dynamischen System modellieren [Aman1999, 47; Pete1998, 10f.]. Der bekannteste Vertreter einer solchen Dynamik-Funktion ist die Replikatordynamik, welche insbesondere für biologische Evolutionsmodelle verwendet wird. Für ökonomische Fragestellungen werden andere Dynamiken verwendet, die jedoch wieder auf die Replikatordynamik zurückgeführt werden können [Aman1999, 51-53; Weib2002, 152-162]. Hier liegt die Annahme zugrunde, dass das Verhalten erfolgreicher Individuen imitiert wird. In der Literatur werden diesbezüglich drei Möglichkeiten unterschieden. Die erste Möglichkeit ist die *Imitation erfolgreicher Spieler*. Hier wird aus der Population zufällig ein Individuum gewählt und dessen durchschnittliche Auszahlung mit der eigenen verglichen. Ist die Strategie des anderen Individuums erfolgreicher, so wird diese einfach übernommen. Analog wird bei der so genannten *Erfolgsbestimmten Imitation* vorgegangen. Im Unterschied zur *Imitation erfolgreicher Spieler* werden Individuen einer Population nicht mehr mit derselben Wahrscheinlichkeit ausgewählt. Bei dieser Variante wird zu einem bestimmten Zeitpunkt das Individuum mit einer höheren Auszahlung mit einer höheren Wahrscheinlichkeit ausgewählt und dessen Strategie übernommen. Beide Dynamiken unterstellen, dass die Auszahlungen der anderen Spieler beobachtet werden können. Bei der *Imitation aus Unzufriedenheit* ist dies nicht notwendig. Hier wird für den Spieler ein Anspruchsniveau definiert. Die von ihm gewählte Strategie wird dann beibehalten, wenn die eigenen Auszahlungen über diesem Anspruchsniveau liegen. Falls dies nicht der Fall ist, wird die Strategie eines zufällig gewählten Spielers übernommen.

Wie bereits beschrieben wurde, kann dieser Selektionsprozess zur Bildung von Gleichgewichten führen, bei denen sich die Anteile der Strategien in der betrachteten Population nicht mehr ändern. In diesem Fall stellt sich die Frage, wie robust solche Gleichgewichte gegenüber einer Invasion neuer Strategien sind. Diese Frage kann durch das Konzept der evolutionär stabilen Strategie (ESS) geklärt werden [Smit1994]. Demnach ist eine Strategie evolutionär stabil, wenn sie bei einer beliebig kleinen Invasionsgröße nicht durch eine andere Strategie verdrängt werden kann. In der Literatur werden verschiedene ESS-Konzepte unterschieden, die jedoch nicht weiter erläutert werden. Ein entsprechender Überblick findet sich bei [Pete1998, 6-10].

4 Bewertung der Robustheit von Reputationssystemen

Basierend auf den im vorangegangenen Abschnitt beschriebenen Konzepten lässt sich die Robustheit von Reputationssystemen im Rahmen eines evolutionären Ansatzes untersuchen. Die in Abschnitt 2 beschriebenen Manipulationsstrategien können in einem evolutionären Rahmen als Mutation verstanden werden. Diese Mutationen stören ein durch ein Reputationssystem für einen bestimmten Anwendungsfall (z. B. elektronischer Marktplatz) hergestelltes Gleichgewicht. Im Rahmen eines evolutionären Modells kann nun untersucht werden, wie Reputationssysteme auf solche Manipulationsstrategien reagieren. Insbesondere lässt sich untersuchen, welche Gleichgewichte sich aufgrund des Selektionsprozesses ergeben und ob sich Manipulationsstrategien erfolgreich in einem Reputationssystem

durchsetzen können. Einen Indikator für die Robustheit eines Reputationssystems stellt diesbezüglich der Anteil der Manipulationsstrategien in einem solchen Gleichgewicht dar. Ein Reputationssystem ist gegenüber Manipulationen dann robust, wenn der Selektionsprozess zu einer Verdrängung der Manipulationsstrategien führt und diese sich damit nicht erfolgreich im Reputationssystem durchsetzen können. Analog gilt, je höher der Anteil der Manipulationsstrategien ist, umso geringer ist demnach die Robustheit des betrachteten Reputationssystems gegenüber Manipulation zu bewerten.

In diesem Abschnitt wird ein Konzept zur Ermittlung der Robustheit von Reputationssystemen vorgestellt, welches als Grundlage für die in Abschnitt 5 beschriebene Multi-Agenten-Simulation darstellt. Zunächst wird das Basismodell mit den grundlegenden Modellannahmen vorgestellt. Danach werden die Strategien der Akteure als Grundlage für das Agentenverhalten spezifiziert.

4.1 Basismodell

Den Ausgangspunkt für das hier beschriebene Modell bildet ein elektronischer Marktplatz mit einer Moral Hazard Situation [Dela2005, 210]. Diese tritt z. B. auf, wenn vom Käufer vor der Lieferung eines ersteigerten Gutes die Zahlung per Vorkasse zu verlangt wird. Der Verkäufer hat dabei den Anreiz, seine eigene Auszahlung auf Kosten des Käufers zu erhöhen, indem er die Auslieferung des Gutes verweigert. Der Käufer muss in dieser Situation darauf vertrauen, dass der Verkäufer diese Situation nicht zu seinem Vorteil ausnutzt. Eine solche Moral Hazard Situation kann auch auf der Seite der Käufer auftreten [SnZi2004, 158-159]. Diese könnten z. B. von einer gewonnenen Auktion zurücktreten oder ihrerseits die Zahlung an den Verkäufer trotz Lieferung des Gutes verweigern. Diese Moral Hazard-Problematik auf der Käuferseite wird in diesem Beitrag nicht berücksichtigt, um die Komplexität des Modells nicht unnötig zu erhöhen.

Die Basis für des Modells bilden zwei gleich große Agenten-Populationen, die sich aus Käufern A^k und Verkäufern A^v , mit $A^k = \{a^k_1, a^k_2, \dots, a^k_n\}$ und $A^v = \{a^v_1, a^v_2, \dots, a^v_n\}$, $n \in \mathbb{N}$ und $n \gg 0$ zusammensetzen. Es wird angenommen, dass Marktplatzteilnehmer nur jeweils auf einer Marktseite, d. h. entweder als Käufer oder als Verkäufer agieren und diese Rolle über den gesamten Simulationsablauf beibehalten.²⁷ Weiterhin wird die Annahme getroffen, dass auf dem Marktplatz ein (hypothetisches) homogenes Gut gehandelt wird, für das alle Käufer dieselbe maximale Zahlungsbereitschaft $v^k = 1$ besitzen. Die Zahlungsbereitschaft kann dahingehend interpretiert werden, dass sie den Wert des Gutes für den Käufer repräsentiert. Im Gegenzug wird für alle Verkäufer ein Reservationspreis von $v^v = 0,1$ angenommen. Dieser stellt die Wertschätzung des Verkäufers dar, sodass er das Gut selbst konsumiert, falls der auf dem Marktplatz erzielte Preis unter dem Reservationspreis liegt [Ocke2003, 299]. Ein Einfluss der Präferenzen der Marktplatzteilnehmer auf den Preis des Gutes ist damit ausgeschlossen. Damit ergibt der Preisintervall $PI = \{p \in \mathbb{R} | 0,1 \leq p \leq 1\}$ für das auf dem Marktplatz gehandelte Gut.

Entsprechend den Ausführungen in Kapitel 3.2 interagieren die Individuen aus der Population der Käufer und Verkäufer paarweise miteinander. Dazu wird in jeder

27 Auf Marktplätzen, wie z. B. auf eBay ist es dagegen möglich, sowohl als Käufer als auch als Verkäufer aufzutreten.

Runde $t \in T = \{1, 2, \dots\}$ jeweils jedem Käufer $a_i^k \in A^k$ ein zufällig ausgewähltes Individuum aus der Menge der Verkäufer $a_j^v \in A^v$ zugeordnet (random matching). Es wird angenommen, dass die Populationen so groß sind, dass eine wiederholte Interaktion mit demselben Partner in der Regel nicht stattfindet. Untersuchungen zeigen dass dies auf elektronischen Marktplätzen häufig der Fall ist [ReZe2002; BKOc2004, 185].

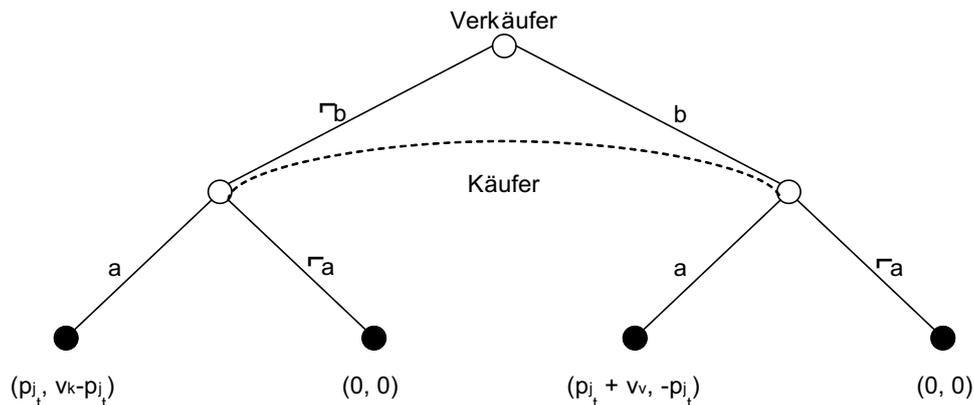


Abb. 4-1: Vertrauensspiel

Die durch random matching gebildeten Paare (a_i^k, a_j^v) spielen zum Zeitpunkt t das in Abbildung 4-1 dargestellte Vertrauensspiel, welches dem von [Ocke2003, 298-300] angelehnt ist. Zu Beginn jeder Runde t trifft jeder Verkäufer die Entscheidung, ob er sich vertrauenswürdig ($\neg b$) oder betrügerisch (b) verhält. Der Käufer kann das Ergebnis dieser Entscheidung erst nach Abschluss der Transaktion beobachten. Entscheidet der Verkäufer, sich betrügerisch zu verhalten, wird er sich weigern, das angebotene Gut an den Käufer auszuliefern. Weiterhin legt jeder Verkäufer j individuell einen Angebotspreis p_t^j fest. Um das Modell weiter zu vereinfachen, wird davon ausgegangen, dass keine Verhandlungen zwischen den Akteuren stattfinden. Der Verkäufer gibt damit einen so genannten „take-it-or-leave-it“-Preis vor, sodass der Käufer nur die Möglichkeit hat, das Angebot des Verkäufers anzunehmen (a) oder abzulehnen ($\neg a$). Wird das Angebot angenommen erhält der Käufer eine Auszahlung von $v^k - p_t^j$, falls der Verkäufer vertrauenswürdig ist. Verhält sich der Verkäufer betrügerisch, erhält der Käufer eine negative Auszahlung in Höhe des Preises p_t^j . Der Verkäufer konsumiert in diesem Fall das Gut und erhält zusätzlich den vom Käufer gezahlten Preis. Für den Verkäufer ergibt sich in diesem Fall eine Auszahlung von $v^v + p_t^j$. Lehnt der Käufer das Angebot ab, so erhalten beide Parteien eine Auszahlung von 0.²⁸ Im Anschluss an die Transaktion erfolgt die Bewertung des Verhaltens des Verkäufers durch den Käufer mit $r_t \in \{+, -, 0\}$. Die Bewertung wird nach der folgenden Regel vergeben: $r_t = +$, falls sich der Verkäufer als vertrauenswürdig erwiesen hat, $r_t = -$, falls sich der Verkäufer betrügerisch verhalten hat und $r_t =$

28 Für den Verkäufer kann in diesem Fall auch eine Auszahlung in Höhe von v_v angenommen werden, wenn man davon ausgeht, dass er das Gut dann selbst konsumiert. Dies ist in dem Modell von [Ocke2003, S. 298-300] der Fall. Auf das Ergebnis der hier modellierten Vertrauenssituation hat dies keine Auswirkung, weshalb für beide Akteure vereinfachend eine Auszahlung von 0 angenommen wird.

0, falls die vorherige Transaktion aufgrund der Ablehnung des Angebotes durch den Käufer nicht zustande gekommen ist.

Für jeden Verkäufer j wird auf dem Marktplatz eine Liste $R^j = \{r_{t=0}, r_{t=1}, \dots, r_{t=t-1}\}$ geführt, welche die für ihn abgegebenen Bewertungen der bisherigen Transaktionen enthält.²⁹ Des Weiteren wird unterstellt, dass jeder Käufer eine Erinnerung an den Erfolg bzw. Misserfolg der vorherigen Transaktionen besitzt. Jeder Käufer führt deshalb zusätzlich eine eigene Liste $R^i = \{r_{t=0}, r_{t=1}, \dots, r_{t=t-1}\}$ welche seine persönliche Erfahrungen den Ergebnissen bzgl. der von ihm durchgeführten Transaktionen enthält.

Welche Handlungen die Individuen in Runde t ausführen, wird anhand der von ihnen verfolgten Strategie festgelegt. Eine Strategie stellt damit eine Entscheidungsregel dar. Für jede Population existiert dabei eine spezifische Strategiemenge. Die Menge aller Strategien lautet $S = (S^k, S^v)$, wobei $S^k = \{s^k_1, s^k_2, \dots\}$ die Strategiemenge der Käufer und $S^v = \{s^v_1, s^v_2, \dots\}$ die Strategiemenge der Verkäufer repräsentiert. Der Erfolg der Strategien wird durch eine Fitnessfunktion f bewertet. Diese berechnet sich aus der Summe der durch die Interaktion erzielten Auszahlungen.

4.2 Strategie der Käufer

Das in Abbildung 4-1 dargestellte Vertrauensspiel bietet nur den Verkäufern einen Anreiz, sich durch betrügerisches Verhalten besser zu stellen. Dies schließt auch eine mögliche Besserstellung durch die Manipulation des Reputationssystems ein. Im Umkehrschluss bedeutet dies für die Käufer-Population, dass diese durch eine Manipulation des Reputationssystems keinen Vorteil in Form höherer Auszahlungen erzielen kann. Im Folgenden wird deshalb davon ausgegangen, dass keine Manipulation des Reputationssystems auf der Käuferseite stattfindet.³⁰ Da für diese Seite somit eine Analyse des Selektionsprozesses unter dieser Annahme nicht sinnvoll ist, wird die Strategiemenge der Käuferseite auf eine einzige Strategie eingeschränkt.

In jeder Runde muss jeder Käufer entscheiden, ob er das Angebot des ihm zufällig zugeordneten Verkäufers annimmt oder ablehnt. Das Ablehnen des Angebotes führt in jedem Fall zu einer Auszahlung von 0. Die Annahme des Angebotes führt zu zwei möglichen Zuständen, nämlich der Auslieferung des Gutes oder dem Verlust in Höhe des Kaufpreises, wenn der Verkäufer das Gut nicht ausliefert. Da nicht bekannt ist, wie sich der Verkäufer verhalten wird, stellt dies eine Entscheidung unter Unsicherheit dar [Vari1994, 172]. Der Käufer erfährt jedoch aus dem Reputationssystem, wie sich der Verkäufer in den bisherigen Runden verhalten hat. Unterstellt man, dass dies einen Rückschluss auf das zukünftige Verhalten des Verkäufers zulässt, liefert das Reputationssystem damit einen Wert für die Erfolgswahrscheinlichkeit der Transaktion.

Eine einfache Möglichkeit zur Bestimmung der Erfolgswahrscheinlichkeit der Transaktion p_{erfolg} ist es, den Anteil der positiven Bewertungen ($\sum r_j^+ / |R^j|$) im Reputationsprofil zu ermitteln. Wie bereits beschrieben wurde, manipulieren Verkäufer das Reputationssystem um dies zu ihrem Vorteil auszunutzen. In diesem Fall

²⁹ Dies entspricht einem Reputationssystem wie es z. B. auf eBay eingesetzt wird.

³⁰ Für zukünftige Untersuchungen ist es jedoch viel versprechend, das Modell dahingehend zu erweitern, dass auch auf der Käuferseite eine Manipulation des Reputationssystems erfolgt.

liefert das Reputationssystem jedoch nicht die korrekte Erfolgswahrscheinlichkeit. Verhält sich ein Verkäufer z. B. trotz einer guten Reputation in einer Transaktion betrügerisch, so ist dies möglicherweise ein Anzeichen dafür, dass das Reputationssystem manipuliert wurde. Es wird davon ausgegangen, dass Käufer dies bei der Erwartungsbildung berücksichtigen und die Erfolgswahrscheinlichkeit basierend auf den eigenen Erfahrungen korrigieren. Unter Berücksichtigung der Erfahrungen des Käufers kann die Erfolgswahrscheinlichkeit als gewichteter Mittelwert aus dem Anteil der vom Käufer erfolgreich durchgeführten Transaktionen ($\sum r_i^+ / |R^i|$) sowie dem Anteil der positiven Bewertungen im Reputationsprofil bestimmt werden:

$$p_{\text{erfolg}} = g \circ \left(\frac{\sum r_i^+}{|R^i|} \right) + (1 - g) \circ \left(\frac{\sum r_j^+}{|R^j|} \right) \quad 0 \leq g \leq 1, g \in \mathbb{R} \quad (1)$$

Der Gewichtungsfaktor g lässt sich als das „Vertrauen“ des Käufers in die Korrektheit des Reputationssystems interpretieren. Eine Erhöhung von g senkt den Einfluss des Reputationssystems bei der Bestimmung der Erfolgswahrscheinlichkeit der Transaktion und bewirkt, dass dabei die eigenen Erfahrungen des Käufers stärker gewichtet werden. Stellt ein Käufer also fest, dass sich Verkäufer häufig abweichend zum Reputationssystem verhalten, erfolgt eine Erhöhung des Parameters g und damit eine Verringerung des Einflusses der Reputation in seinen zukünftigen Entscheidungen.

Auf der Basis der Erfolgswahrscheinlichkeit lässt sich der Erwartungswert der Auszahlung $E(a)$ der Transaktion für den Fall bestimmen, dass der Käufer das Angebot des Käufers annimmt:

$$E(a) = (v^v - p_i^j) \circ p_{\text{erfolg}} - p_i^j \circ (1 - p_{\text{erfolg}}) \quad (2)$$

Damit ist es möglich, eine Entscheidungsregel für den Käufer aufzustellen. Dieser wird sich dann für die Annahme des Angebotes entscheiden, wenn die zu erwartende Auszahlung die sichere Auszahlung bei Ablehnung des Angebotes übersteigt:

$$s^k = \begin{cases} \text{annehmen} & \text{falls } E(a) > 0 \\ \text{ablehnen} & \text{sonst} \end{cases} \quad (3)$$

Diese Entscheidungsregel hat die folgenden Eigenschaften:

- mit steigendem wahrgenommenem Risiko sinkt die Zahlungsbereitschaft des Käufers
- für eine gute Reputation ist ein Käufer bereit, einen höheren Preis zu akzeptieren
- eine schlechte Reputation und damit ein hohes Ausfallrisiko führt zu einer geringen Zahlungsbereitschaft
- verhält sich ein Verkäufer trotz guter Reputation nicht vertrauenswürdig, so bewirkt eine Erhöhung des Faktors g , dass der Einfluss des Reputationssystems zukünftige Entscheidungen des Käufers abnimmt

Die hier beschriebene Entscheidungsregel unterstellt Risikoneutralität der Käufer, die somit keine Präferenzen hinsichtlich der unsicheren bzw. sicheren Alternative aufweisen. Es sind jedoch noch Käufertypen mit einer anderen Risikobewertung denkbar. So tendieren risikoaverse Käufer tendenziell dazu, die sichere Variante der

unsicheren vorzuziehen, während risikofreudige Käufer eher das unsichere Ergebnis bevorzugen. Die Berücksichtigung solcher Risikopräferenzen stellt eine mögliche Erweiterung des Modells dar und lässt sich durch die Verwendung von Erwartungsnutzenfunktionen erreichen. Das Konzept des Erwartungsnutzens ist z. B. in [Bitz1981] beschrieben und wird in diesem Beitrag nicht weiter verfolgt, stellt jedoch eine potenzielle Erweiterung des hier vorgestellten Modells dar.

4.3 Strategien der Verkäufer

Wie auf der Käuferseite muss auch auf der Seite der Verkäufer das Verhalten der Agenten spezifiziert werden. Hier sind unterschiedliche Strategien denkbar. Zum einen können sich Verkäufer kooperativ verhalten. Zum anderen können sich Verkäufer einen Vorteil verschaffen, indem sie die in Abschnitt 2 beschriebenen Manipulationsstrategien anwenden. Im Folgenden werden Beispiele für Strategien auf der Verkäuferseite vorgestellt.

4.3.1 kooperieren

Im Rahmen der Strategie „kooperieren“ wählt ein Verkäufer in jeder Runde $5b$ und versucht dabei, den Angebotspreis an die aktuelle Zahlungsbereitschaft der Käufer anzupassen. Eine solche Anpassung kann erfolgen, indem die Höhe des Angebotspreises abhängig vom Erfolg der Transaktion in der vorangegangenen Runde festgelegt wird. Bei [ZEB+2001, 89-90] werden Agenten mit einem solchen Verhalten als „Derivation Followers“ bezeichnet. Demnach erhöht der Verkäufer den Angebotspreis in der nächsten Runde um einen Betrag b_+ , falls das Preisangebot in der letzten Runde von dem zufällig zugeordneten Käufer akzeptiert wurde, andernfalls verringert er ihn um b_- . Diese Beträge werden dabei in jeder Runde zufällig aus dem Intervall $B = \{b \in \mathbb{R} \mid 0 \leq b \leq s\}$ gezogen, sodass sich für den Verkäufer die folgende Anpassungsregel für den Preis der Runde t ergibt:

$$p_t = \begin{cases} p_{t-1} + b_+ & \text{falls Transaktion in } t-1 \text{ erfolgreich} \\ p_{t-1} - b_- & \text{sonst} \end{cases} \quad (4)$$

Diese Regel führt dazu, dass Verkäufer ständig die Zahlungsbereitschaft der Käufer testen und sich damit den Gegebenheiten des Marktplatzes anpassen. In [Zac+2001, S. 90] wird als weitere Möglichkeit die Anpassung des Preises auf der Grundlage der eigenen Reputation beschrieben. In Experimenten wurde jedoch festgestellt, dass das Verhalten dieser „Reputation Followers“ dem der „Derivation Followers“ ähnlich ist, weshalb die letztgenannte Strategie nicht weiter betrachtet wird.

4.3.2 manipulieren

Neben dem kooperativen Verhalten können Verkäufer auch durch Manipulation des Reputationssystems versuchen eine hohe Fitness zu erzielen. Viele Manipulationsstrategien zielen darauf ab, eine gute Reputation zu einem bestimmten Zeitpunkt auszunutzen. Eine wichtige Rolle spielt daher die in Abschnitt 2 beschriebene rest-on-the-laurels Strategie [Dela2003a, 1419]. Eine einfache Möglichkeit zur Implementierung einer solchen Strategie besteht darin, dass sich ein

Verkäufer eine zufällige Anzahl von Runden entsprechend der Strategie „kooperieren“ verhält, jedoch in der folgenden Runde b (Betrug) wählt, um dadurch eine höhere Auszahlung zu erzielen. Danach verhält er sich eine weitere zufällige Zahl von Runden entsprechend der Strategie „kooperieren“. Die rest-on-the-laurels-Strategie lässt darüber hinaus einige Variationsmöglichkeiten zu. So könnte die Strategie so gestaltet werden, dass Verkäufer ihre gute Reputation mehrere Runden hintereinander ausnutzen. Eine andere Möglichkeit ist die Wahl des Zeitpunktes, ab dem das Ausnutzen der Reputation erfolgt. Statt einer zufälligen Zahl von Runden kann dafür die durch Betrug erzielbare Auszahlung als ein mögliches Kriterium herangezogen werden.

Eine Eigenschaft der rest-on-the-laurels Strategie ist, dass diese zu negativen Bewertungen durch die Käufer führt, was sich wiederum negativ auf die zukünftigen Transaktionen auswirkt. Um diesen negativen Konsequenzen zu entgehen kann in Betracht gezogen werden, die Strategie so abzuwandeln, dass sich ein Verkäufer, nachdem er b spielt, durch eine Neuankündigung auf dem Markt ein „frisches“ Reputationsprofil verschafft. Dadurch wird eine whitewashing-Strategie abgebildet. Im Rahmen des hier beschriebenen Simulationskonzeptes könnte eine Neuankündigung durch ein Löschen des (bisherigen) Reputationsprofils des Verkäufers nachgeahmt werden. Ein leeres Reputationsprofil stellt jedoch einen Indikator für eine solche whitewashing-Strategie dar. Um dies zu verhindern, muss der Markt so erweitert werden, dass eine „Fluktuation“ auf der Verkäuferseite entsteht, bei der ein bestimmter Anteil α der Verkäuferpopulation den Markt verlässt, gleichzeitig jedoch entsprechend viele „neue“ Verkäufer dem Markt beitreten. Eine ähnliche Annahme wird in dem Modell von [FrRe1999, 6] getroffen. Im Rahmen der hier vorgestellten Simulation könnte daher zu Beginn jeder Runde eine Löschung des Reputationsprofils bei α zufällig ausgewählten Verkäufers erfolgen.

Die in Abschnitt 2 beschriebenen Manipulationsmöglichkeiten zeigen, dass über die hier beschriebenen Strategien hinaus noch eine Vielzahl weiterer Strategien zur Manipulation eines Reputationssystems denkbar sind. Darüber hinaus ist es auch möglich, verschiedene Manipulationsformen miteinander zu kombinieren. Andere Strategien, wie z. B. „ballot stuffing“ oder „bad mouthing“ erfordern, dass verschiedene Agenten miteinander kooperieren. Die Untersuchung solcher komplexen Manipulationsstrategien wird im weiteren Verlauf des Forschungsprojektes einen Schwerpunkt bilden.

4.3.3 Imitation

Wie bereits in Abschnitt 3.2 beschrieben wurde, werden (erfolgreiche) Strategien anderer Verkäufer imitiert, was dazu führt, dass eine einmal gewählte Strategie nicht zwangsläufig beibehalten wird. Abschließend stellt sich somit die Frage nach einem geeigneten Imitationsmodell auf der Verkäuferseite. Ein Auswahlkriterium hierfür stellt die Beobachtbarkeit der Auszahlungen der Akteure dar. Da auf einem Markt nur die eigene Auszahlung beobachtet werden kann, wird in dem hier vorgestellten Modell die *Imitation aus Unzufriedenheit* als eine plausible Dynamik für die Simulation erachtet. Als Anspruchsniveau kann die pro Runde durchschnittlich erzielte Auszahlung angenommen werden, welches in der Verkäuferpopulation gleichverteilt im Intervall PI liegt.

5 Simulationskonzept

Die Ermittlung der Robustheit eines Reputationssystems auf Basis des im vorigen Abschnitt beschriebenen Konzeptes erfolgt durch die Analyse des zugrunde liegenden Dynamischen Systems. Dazu müssen so genannte Attraktoren ermittelt werden, welche einen gegen Störung hinreichend stabilen Gleichgewichtszustand eines dynamischen Systems kennzeichnen [Aman1999, 57f.]. Ein solches Modell kann selbst mit starken Vereinfachungen schnell eine hohe Komplexität erreichen, sodass eine analytische Lösung nur mit sehr hohem Aufwand oder auch gar nicht mehr möglich ist. So führen evolutionäre Modelle bei denen mehr als zwei Strategien analysiert werden schon bei einer einzigen Population zu einem schwer vorher-sagbaren Verhalten [Pete1998, 6]. Im Hinblick auf die Grenzen bisheriger Analysen der Manipulationsmöglichkeiten von Reputationssystemen ist es jedoch wünschenswert, möglichst viele solcher Strategien in die Untersuchung einzubeziehen, um möglichst realitätsnahe Ergebnisse erzielen zu können. Wünschenswert ist auch die möglichst realitätsgetreue Abbildung des Verhaltens der betrachteten Populationen, was die Komplexität des dynamischen Systems weiter erhöht. Aus diesem Grund wird die Simulation als ein geeignetes Mittel zur Bestimmung der Robustheit von Reputationssystemen erachtet.

Das globale Verhalten des in Abschnitt 4 modellierten Systems ergibt sich implizit aus der Interaktion der Individuen der beiden Populationen. Daher lässt sich dieses Modell gut auf ein Multi-Agenten-System abbilden, da auch hier auf autonom handelnde Individuen (Agenten) fokussiert wird [Degu2004, 10f.; GiSe2005, 2-5; BeBo2001, 149]. Zur Simulation des in Abschnitt 4 beschriebenen Bewertungsansatzes wird deshalb der Einsatz eines Multi-Agenten-Systems als sinnvoll erachtet. Verschiedene Publikationen [Tesf2000; Degu2004, 25ff.] zeigen, dass dies ein übliches Vorgehen zur Simulation evolutionärer Modelle ist.

Eine Möglichkeit, das in Abschnitt 4.1 beschriebene Modell auf ein Multi-Agenten-System zu übertragen besteht darin, jeden Marktplatzteilnehmer des Basismodells durch jeweils einen Agenten zu repräsentieren. Das Multi-Agenten-System besteht in diesem Fall aus einer Menge von Käufer- und Verkäufer-Agenten. Diese Agenten verhalten sich entsprechend den in den Abschnitten 4.2 bzw. 4.3 beschriebenen Strategien. Neben den Käufer- und Verkäufer-Agenten wird zusätzlich noch ein weiterer Agent benötigt, der die Aufgabe hat, das random-matching durchzuführen sowie den Reputationsmechanismus zu verwalten. Dieser Agent repräsentiert damit den Marktplatz, auf dem Käufer und Verkäufer-Agenten agieren und wird daher im Folgenden als Marktplatz-Agent bezeichnet.

In Abbildung 5-1 ist der prinzipielle Ablauf der Simulation dargestellt. Die Simulation erfolgt dabei rundenbasiert. In jeder Runde führt der Marktplatz-Agent zunächst ein random matching zwischen Käufer- und Verkäufer-Agenten durch. Danach spielen die zufällig gebildeten Agenten-Paare das in Abbildung 4-1 abgebildete Vertrauensspiel. Zunächst sendet der Verkäufer ein Preisangebot an den ihm zugewiesenen Käufer. Der Käufer fragt wiederum die Reputation des Verkäufers vom Marktplatz-Agenten ab und entscheidet entsprechend der in Abschnitt 4.2 beschriebenen Regel, ob er das Angebot des Verkäufers annimmt (accept) oder ablehnt (decline). Danach teilt er dem Verkäufer das Ergebnis dieser Entscheidung mit. Entsprechend der in Abschnitt 4.3 beschriebenen Strategie entscheidet der Verkäufer, ob er betrügt (deliver_nothing) oder nicht betrügt (deliver). Abhängig vom

Ergebnis der Transaktion sendet der Käufer abschließend die Bewertung an den Marktplatz-Agenten. Danach beginnt eine neue Runde, indem ein erneutes random matching durch den Marktplatz-Agenten durchgeführt wird.

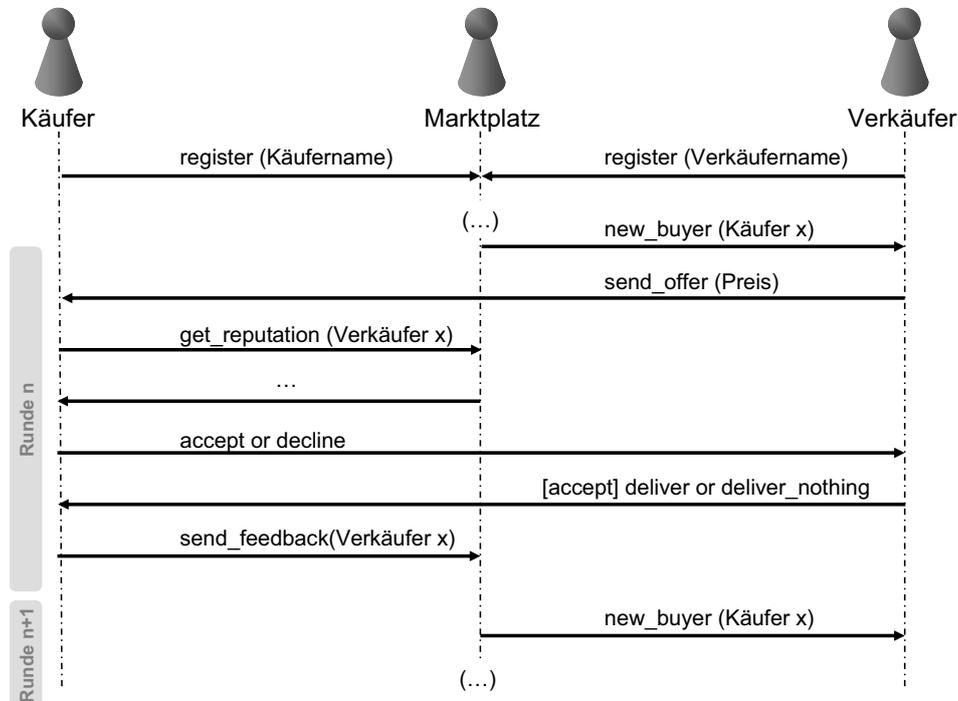


Abb. 5-1: Interaktion der Agenten

Mit Hilfe dieses Multi-Agenten-Systems kann die Entwicklung beider Agenten-Populationen beobachtet werden. Als Simulationsparameter müssen dabei die Größe der Populationen sowie der Populationsanteil der Manipulationsstrategien auf der Verkäuferseite festgelegt werden. Anschließend ist zu untersuchen, welche Gleichgewichte sich in der betrachteten Verkäufer-Population bilden. Wie bereits beschrieben wurde, lässt der Anteil der nicht durch den Selektionsprozess verdrängten Angreifer einen Rückschluss auf die Robustheit des Reputationssystems zu.

6 Zusammenfassung und Ausblick

Das beschriebene Konzept ermöglicht die Untersuchung der Robustheit von Reputationssystemen gegenüber den in Abschnitt 2 beschriebenen Manipulationsformen. Die Anwendung der Multi-Agenten Technologie als Simulationsumgebung erweitert dabei die Grenzen bisheriger analytisch motivierter Ansätze. Gleichzeitig lässt die Verwendung von Konzepten der evolutionären Spieltheorie und die damit verbundene Lockerung der Grundannahme des streng rationalen Verhaltens eine höhere Realitätsnähe erwarten. Außerdem trägt die Möglichkeit der simultanen Untersuchung unterschiedlicher (Manipulations-) Strategien zu einer höheren Realitätsnähe bei. Das hier beschriebene Konzept kann außerdem auf verschiedene Reputationssysteme angewandt werden und ermöglicht damit den Vergleich der Robustheit unterschiedlicher Reputationssysteme. So lässt sich auch ermitteln, welches Reputationssystem gegen bestimmte Manipulationsformen besonders anfällig ist.

Im nächsten Schritt ist die Implementierung der Multi-Agenten-Simulation und darauf basierend die Untersuchung der Robustheit des hier beschriebenen Reputationssystems geplant. Im Anschluss daran soll die Komplexität dieses einfachen Simulationsprototyps schrittweise erhöht werden um sich damit immer weiter der Realität anzunähern. Dies beinhaltet unter anderem die Weiterentwicklung der hier beschriebenen Strategien, sodass auch komplexe Angriffsformen analysiert werden können. Außerdem ist das Basismodell um eine Moral Hazard Situation auf der Käuferseite zu erweitern. Des Weiteren sollen zusätzlich auch Manipulationsstrategien auf der Käuferseite analysiert werden können.

Im weiteren Verlauf des Forschungsvorhabens müssen darüber hinaus weitere grundlegende Fragestellungen geklärt werden. Zum einen soll eine Klassifikation der Manipulationsformen entwickelt werden, anhand derer sich bestehende und eventuell neu auftretende Manipulationsformen einordnen lassen. Zum anderen ist die Entwicklung eines generischen Modells zur Beschreibung von Reputationssystemen geplant. Die Untersuchung soll sich dabei zunächst auf solche Reputationssysteme konzentrieren, wie sie auf elektronischen Marktplätzen eingesetzt werden. Mit fortschreitender Entwicklung des Forschungsprojektes ist geplant, auch Reputationssysteme aus weiteren Anwendungsgebieten in die Analyse mit einzubeziehen. Dazu wird auch das zugrunde liegende Modell ständig weiterzuentwickeln und diesbezüglich auf neu auftretende Aspekte anzupassen sein.

Nach dem Abschluss der Analyse wird man in der Lage sein, die Eignung von Reputationssystemen zur Herstellung des Vertrauens beurteilen zu können. Des Weiteren wird es möglich sein, die Wirksamkeit von Maßnahmen zur Behebung der beschriebenen Probleme von Reputationssystemen beurteilen und ggf. neue Maßnahmen vorschlagen zu können. Damit könnte dieser Ansatz einen Beitrag zur Ableitung von Gestaltungsempfehlungen für Reputationssysteme liefern.

Literaturverzeichnis

- [Aman1999] Amann, E.: Evolutionäre Spieltheorie: Grundlagen und neue Ansätze. Physica: Heidelberg, 1999.
- [Axel1990] Axelrod, R.: The Evolution of Cooperation. Penguin-Verlag: London u. a., 1990.
- [BeBo2001] Bertels, K.; Boman, M.: Agent-Based Social Simulation in Markets. In: Electronic Commerci Research 1 (2001) 1-2, S. 149-158.
- [Bitz1981] Bitz, M.: Entscheidungstheorie. Vahlen: München, 1981.
- [BKOc2004] Bolton, G. E.; Katok, E.; Ockenfels, A.: Trust among Internet Traders. In: Analyse & Kritik. 26 (2004) 1, S. 185-202.
- [Degu2004] Deguchi, H.: Economics as an Agent-Based Complex System: Toward Agent-Based Social System Sciences. Springer: Tokyo u. a., 2004.
- [Dela2000] Delarocas, C.: Immunizing Online Reputation Reporting Systems Against Unfair Ratings and Discriminatory Behavior. Proceedings of the 2nd ACM conference on Electronic commerce. Minneapolis, Minnesota, 2000.
<http://ccs.mit.edu/dell/ec00reputation.pdf>, Abruf am 2006-09-14.
- [Dela2003a] Delarocas, C.: The Digitization of Word of Mouth: Promise and Challenges of Online Feedback Mechanisms. In: Management Science 10 (2003) 49, S. 1407–1424.

- [Dela2003b] Dellarocas, C.: Efficiency and Robustness of eBay-like Online Feedback-Mechanisms in Environments with Moral Hazard. MIT Sloan School of Management, Working-Paper Nr. 4297-03.
<http://ccs.mit.edu/dell/SITE%202002.pdf>, Abruf am 2006-09-13.
- [Dela2005] Dellarocas, C.: Reputation Mechanism Design in Online Trading Environments with Pure Moral Hazard. In: Information Systems Research 2 (2005) 16, S. 209-230.
- [DeRe2003] Dellarocas, C.; Resnick, P.: Online Reputation Mechanisms: A Roadmap for Future Research.
<http://ccs.mit.edu/dell/papers/symposiumreport03.pdf>, Abruf am 2006-09-13.
- [DMSy2003] Dingleline, R.; Mathewson, N.; Syverson, P.: Reputation in P2P Anonymity Systems. Workshop on Economics of Peer to Peer Systems, 2003.
<http://freehaven.net/doc/econp2p03/econp2p03.pdf#search=%22Reputation%20in%20P2P%20Anonymity%20Systems%22>, Abruf am 2006-09-13.
- [Douc2002] Douceur, J.: The Sybil Attack. Proc. of the IPTPS02 Workshop. Cambridge, 2002.
<http://www.cs.rice.edu/Conferences/IPTPS02/101.pdf#search=%22Douceur%20The%20Sybil%20Attack%22>, Abruf am 2006-09-14.
- [Frie1991] Friedmann, D.: Evolutionary Games in Economics. In: Econometrica 3 (1991) 59, S. 637-666.
- [FrRe1999] Friedmann, E.; Resnick, P.: The Social Cost of Cheap Pseudonyms.
<http://www.si.umich.edu/~presnick/papers/identifiers/081199.pdf>, Discussion Paper, 1999, Abruf am 2006-09-13.
- [Holl2000] Holler, M; Illing, G.: Einführung in die Spieltheorie, 4. Auflage. Springer: Berlin u. a., 2000.
- [HoSi2002] Hofbauer, J; Sigmund, K.: Evolutionary Games and Population Dynamics. Cambridge University Press: Cambridge u. a., 2002.
- [KeSc2002] Kennes, J.; Schiff, A.: The Value of a Reputation System. Working-Paper, The University Auckland. Dezember 2002.
<http://econwpa.wustl.edu:80/eps/io/papers/0301/0301011.pdf>, Abruf am 2006-09-14.
- [MRZe2004] Miller, N.; Resnick, P.; Zeckhauser, R.: Eliciting Informative Feedback: The Peer-Prediction Method. In: Management Science 9 (2005) 51, S. 1351 – 1373.
- [MHMo2002] Mui, L; Halberstadt, A; Mohtashemi, M.: Notions of Reputation in Multi-Agents Systems: A Review. In: Proceedings of AAMAS 2002, ACM Press: Bologna, 2002, S. 280-287.
- [Ocke2003] Ockenfels, A.: Reputationsmechanismen auf Internet-Marktplatzformen. In: ZfB: Zeitschrift für Betriebswirtschaft. 3 (2003) 73, S. 295-315.
- [Pete1998] Peters, R.: Evolutionäre Stabilität in sozialen Modellen. Accedo: München, 1998.
- [ReZe2002] Resnick, P.; Zeckhauser, R: Trust Among Strangers in Internet Transactions: Empirical Analysis of eBay's Reputation System. In: Michael R. Baye (Hrsg.): The Economics of the Internet and E-Commerce. Amsterdam, 2002, S. 127-158.
- [RZFK2000] Resnick, P; Zeckhauser, R; Friedman, E; Kuwabara, K: Reputation Systems. In: Communications of the ACM, 12 (2000) 43, S. 45 - 48.

- [Samu1997] Samuelson, L.: Evolutionary Games and Equilibrium Selection. MIT-Press: Cambridge, Mass. u. a., 1997.
- [GiSe2005] Giorgini, P.; Henderson-Sellers, B.: Agent-Oriented Methodologies: An Introduction. In: Henderson-Sellers B.; Giorgini, P.: Agent-Oriented Methodologies. Idea Group Publishing: Hershey u. a., 2005, S. 1-19.
- [Smit1993] Smith, J. M.: Evolution and the Theory of Games. Cambridge University Press: Cambridge u. a., 1993.
- [SnZi2004] Snijders, C; Zijdeman, R.: Reputation and Internet Auctions: eBay and Beyond. In: Analyse & Kritik 1 (2004) 26, S. 158-184.
- [Tef2000] Tesfatsion, L.: How Economists can get Alife. Working-Paper, Oktober 2000. <http://www.econ.iastate.edu/tesfatsi/rgetroot.pdf>, Abruf am 2006-10-02.
- [Vari1994] Varian, H. R.: Mikroökonomie. 3. Auflage, Oldenbourg: München u. a., 1994.
- [Weib2002] Weibull, J. W.: Evolutionary Game Theory, 5. Auflage. MIT-Press: Cambridge u. a., 2002.
- [ZEB+2001] Zacharia, G.; Evgeniou, T.; Moukas, A.; Boufounos, P.; Maes, P.: Economics of Dynamic Pricing in a Reputation Brokered Agent Mediated Marketplace. In: Electronic Commerce Research. 1-2 (2001) 1, S. 85-100.
- [ZMMa1999] Zacharia, G; Moukas A; Maes, P: Collaborative Reputation Mechanisms in Electronic Marketplaces. Proceedings of the 32nd Hawaii International Conference on System Sciences, 1999. <http://csdl.computer.org/comp/proceedings/hicss/1999/0001/08/00018026.PDF>, Abruf am 2006-09-13.

Economic Coordination in Service-Oriented Architectures

Falk Kretzschmar

Martin-Luther-University Halle-Wittenberg
Dept. of Management Information Systems
Universitätsring 3
D-06108 Halle, Germany
kretzschmar@wiwi.uni-halle.de

Abstract: The software infrastructure of today's enterprises is transforming from monolithic application software into more flexible component-based and modular architectures. In this context Service-Oriented Architectures (SOAs) are discussed as a paradigm where application components can be accessed and exchanged through a service broker. Especially in a B2B setting where the involved enterprises are economically independent instances non-functional parameters such as price and Quality of Service (QoS) for the exchanged services are of particular interest. This causes the increasingly important question of how the economic perspective can be added to SOA.

1 Introduction

This article introduces negotiation concepts for extending SOA and suggests current technologies for their implementation. One extension is the negotiation of price and other non-functional parameters such as Quality of Service for the exchanged services. The article is organized as follows: After a short introduction and the definition of fundamental terms the second part of the article introduces main concepts for extending SOA and adding the economic perspective to the basic architecture. This is followed by suggesting technologies for the implementation of the described negotiation concepts with special emphasis on semantic description for the exchanged services. Based on the described concepts and technologies the third part introduces an agent-based approach for simulating the economic coordination between service provider and service requestor. The article ends with a discussion of the proposed concepts and a short outlook on future work.

The main idea of a Service-Oriented Architecture (SOA) is to provide the functionality of applications as a service and to allow a simple way to access this service via a web infrastructure. Important goals for using a SOA are reuse of existing components, interoperability, loose coupling, and the possibility to flexibly adjust the applications to the company's business processes [Alon2003], [Zimm2003].

According to the W3C specification a SOA consists of three main parts: The Service Requestor provides services, the service broker contributes to publishing and finding of the services and the service requestor finds the adequate services. Figure 1-1 shows the mentioned SOA components and their interaction [W3C12004].

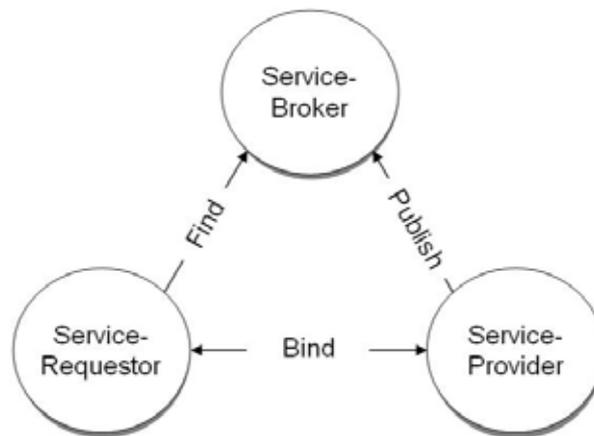


Figure 1-1: Components of a Service-Oriented Architecture

Since the W3C has published the recommendation on the basic architecture many new technologies evolved. Web Services are currently the most popular technology for implementing a SOA. The Web Services standards Simple Object Access Protocol (SOAP) for messaging, Web Service Description Language (WSDL) for interface description, and Universal Description, Discovery, and Integration (UDDI) as an optional technology for implementing the service broker are often referred to as the basic technology stack. Besides this basic set of XML-based technologies there exist many extensions to SOA in terms of composition and coordination of services as well as to ensure transactions and security. In addition to that there are promising extensions with regard to the semantic description of services to finally provide an automated semantic search for web services. Currently this automated semantic search is not possible with the basic SOA standards. One main limitation is that UDDI only allows the search for services based on their functionality. Other search criteria such as price and Quality of Service are not considered, which means the negotiation about economic criteria is still taking place outside the basic architecture in a separate process. This contradicts the above mentioned goals for using SOA, in particular the flexible on-demand access to services and the loose coupling between service provider and service requestor.

2 Economic coordination in SOA

2.1 Definition and classification

Economic coordination in SOA describes the negotiation between service provider and service requestor based on economic criteria. Price and Quality of Service are considered as important negotiation criteria for exchanging services in SOA [Bole2004], [Dan2004], [Ludw2003], [Mena2004], [Sing2005]. Economic coordination

in SOA can be seen as a process consisting of four transaction phases. For the scope of this article the negotiation phase is of special interest. In the negotiation phase transaction partners are selected and the contract for using the service is elaborated. This contract is called Service Level Agreement (SLA), which is finally established and agreed upon in the agreement phase. Figure 2-1 shows the four transaction phases that are similar to the transaction phases in electronic markets [Lind1998].

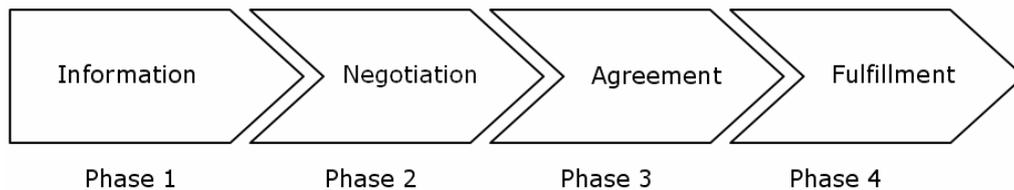


Figure 2-1: Transaction phases

For implementing the negotiation phase there exist many different concepts. Among them are the very prominent auction mechanisms. Often they are referred to as general concepts without an explicit relation to negotiations in SOA. [Sege1999], [Bich2001] mention the following concepts for implementing the negotiation phase:

- 1:1 negotiation, m:n marketplace, brokered marketplace
- simple auction mechanisms: English, Dutch, Vickrey auction
- extended auction mechanisms: double, multi-unit, multi-attribute auction
- negotiation support systems, agent-based negotiations.

These different concepts for implementing the negotiation phase derive from the parameters that characterize a particular negotiation scenario. Many authors state that besides the pure negotiation mechanism the actual setting of a negotiation is of main importance [Kris2002], [Milg2004]. For the classification of negotiation scenarios in SOA we use the following parameters:

- negotiation partners: number of negotiation partners, strategic behavior of negotiation partners
- negotiation mechanism: rules for matchmaking of negotiation partners, rules for pricing the traded service
- negotiation subject: properties of the traded service
- technological setting: network and server performance, standards and communication protocols, semantic description of services and negotiation ontology.

The above mentioned parameters can be used for a general description of negotiation scenarios. Depending on these parameters the outcome of the negotiation phase varies considerably. For instance the number of negotiation partners influences the negotiation power of one particular negotiation partner and leads to specific strategic behavior. Changing the number of negotiation partners means that their original strategy is not appropriate anymore. In addition to the strategy derived from the number of participants it is possible to define a basic

strategy for each actor or for types of actors. This strategy is based on the rules that are defined in the negotiation mechanism.

We explained earlier that in the negotiation phase transaction partners are matched and the price of the traded service is determined. The second category of negotiation parameters describes the actual matchmaking and the pricing of services. In the case of negotiations between service provider and service requestor the properties of the traded service can be important negotiation parameters as well as technical parameters like network and server performance or the availability of semantic descriptions for services.

2.2 Negotiations in SOA

In this section we examine the question of how transaction partners should be matched and what pricing mechanism should be used for trading services in SOA. In 2.1 we named general concepts for implementing the negotiation phase. In real-world negotiations the different negotiation concepts can occur as a combination of the above mentioned categories. For instance, on a brokered marketplace a multi-attribute auction can be used for negotiating between software agents. Our goal is to adapt the general negotiation mechanisms so that they match with the special setting for negotiations in SOA.

SOA is an architectural concept for application software, which is often referred to as software as a service [Elfa2004], [Sing2005]. So in broad terms the negotiation subject is a service, the negotiation partners are applications or organizations that run these applications and the technological setting is the Internet. Because of the SOA goal to flexibly adjust to a company's business processes by selecting the best suited service for the current task, we need automated selection of services based on a semantic search. For developing a concept for negotiations in SOA this means we also need an approach for automated negotiations. Negotiation Support Systems include a human decision and are therefore not matching with the above mentioned SOA goals and the technological setting. But there are also negotiation concepts that allow automated negotiations between negotiation partners. Agent-based negotiations fall into that category. In agent-based negotiations a strategy based on a fixed set of rules can be implemented. This means the software agent is able to fulfill the task of an automated search and selection of services.

For the remainder of this article we want to focus on an extended service broker for automated negotiations in SOA. By definition SOA implies the use of a service broker, but from the technological perspective SOA allows 1:1 negotiation, m:n marketplace as well as a brokered marketplace for negotiations between service provider and service requestor. Until now many companies do not use a service broker when implementing SOA, because it is not necessary for them. Integrating existent applications within a company is usually done without a service broker, although from the technology perspective the service broker can be used for intra-company transactions as well as for inter-company transactions.

Despite the fact that the idea of a global marketplace for service has not been implemented yet, we consider the service broker to be a good starting point for extending SOA, because the broker can be used for storing metadata about services and service providers. Extending the service broker leads to a central instance that

allows automated negotiations of service parameters. In table 2-1 main concepts for an extended service broker in SOA are listed. Applying the different pricing models means that the standard service broker is transformed into a simple catalogue, an extended catalogue, a simple auction mechanism, or an extended auction mechanism. Agent-based negotiations can also be used for automated negotiations in SOA, but they do not determine a special pricing model.

Pricing model:	Service broker:
Fixed price, "take it or leave it" offer	Simple catalogue
Fixed price, differentiated price	Extended catalogue
Dynamic pricing	Simple auction mechanisms
Negotiation of price, Quality of Service and other contract parameters	Extended auction mechanisms, Agent-Based Negotiations

Table 2-1: Concepts for an extended service broker in SOA

The above mentioned concepts for negotiations in SOA describe different pricing models for trading services. Fixed prices and differentiated prices are a simple way to incorporate the economic perspective in SOA. The service provider is getting paid for the functionality of a software component and for the promise to deliver it on-demand. Ignoring the cost of developing and providing services would mean that the service provider will use other channels to distribute his software components. The problem with fixed prices and differentiated prices is that the service provider needs to know the demand function for the traded services. In real-world scenarios this assumption does not hold for trading traditional products and it does not hold for trading software services either. With the different auction mechanisms the pricing of the traded services is done dynamically and according to the auction type specific rules. The service provider is getting information about the demand function for the traded services and usually earns higher profits.

2.3 Existing standards for implementing negotiations in SOA

UDDI is often called a global marketplace for offering web services. In fact it is a catalogue that can be used by service providers to publish their services and by service requestors to find them. Because of the deficit of semantic service description in UDDI, it does not support automated negotiations between the involved transaction partners. Nevertheless UDDI is a good starting point for extending SOA, since it already implements a structure for storing metadata: "white pages" include provider information, "yellow pages" provider and service information, and "green pages" information for calling the service. These categories are matched with the UDDI data structure and in addition to that UDDI finally provides the reference to a WSDL document [Sing2005]. With this reference the binding of the actual service is done

through direct interaction between service provider and service requestor. Service binding is done either statically or dynamically. With the static approach services are connected to each other at design time. This can be problematic, in the case of changes of interfaces, messages, transport protocols, and network addresses. Furthermore static binding is an inflexible and in the case of frequent changes costly way to operate a distributed system. By contrast the dynamic approach allows the binding of services at runtime. With this approach UDDI delivers a set of services that implement exactly the same interface.

For automated negotiations in SOA this is not enough, because the same interface does not guarantee the same semantics of two services. In addition to that the metadata about services and service providers in UDDI is not sufficient. For that reason there already exist a number of extensions to the basic architecture. In majority these extensions were developed in research projects and make use of semantic web technologies to extend SOA. We differ between the two categories “Service broker with semantic extensions” and “Interface description with semantic extensions”. Extending the service broker is a sound method for developing a catalogue that includes more information on services than UDDI. When incorporating negotiation mechanisms like auctions the service broker is the central instance for clearing offers and counteroffers. Research projects and prototypes that fall into the category of a service broker with semantic extensions are UDDIe, OWL-S Matchmaker, and WS-QOS [Shai2003], [Chen2003], [Srin2004], [Paol2002], [Gram2003]. Interface descriptions with semantic extensions match with SOA implementations that do not use a service broker, because the functional and non-functional parameters of services are described by the interface. In this case lightweight approaches of service discovery like Web Service Inspection Language (WSIL) can be used, but implementing UDDI is not necessary. Research projects and prototypes that fall into the category of interface descriptions with semantic extensions are WSOL, SLang, WSML, WSDL-S, and WSLA [Pate2003], [Lama2003], [Verh2004], [Tian2004].

2.4 Extended service broker architecture for automated negotiations in SOA

For implementing automated negotiations in SOA we propose the extended service broker architecture. It consists of a standard UDDI registry extended by Web Ontology Language for Services (OWL-S) service profiles, the OWL-S matchmaker and the negotiation mechanism. A conceptual overview of this extended service broker is depicted in figure 2-2.

Compared to a standard service broker with this architecture it is possible to describe service profiles. With the help of these service profiles the service requestor can define a service request, which contains parameters the service provider should guarantee. For the service provider the service offer contains service parameters he is able to guarantee. Based on that semantic description of services the inference engine of the OWL-S matchmaker matches the two profiles and by using the UDDI registry or the WSIL document it can finally discover and select the requested service. Part of the architecture is the negotiation mechanism that allows a more flexible and more complex interaction between service provider and service requestor. Compared to the simple matching of predefined service profiles, the

negotiation mechanism allows a process of dynamic adjustment to the technological and economic setting. How the actual matching is done is implemented in the negotiation mechanism that defines the negotiation rules and the negotiation strategy of service providers and service requestors. Another important part of the extended service broker architecture is the negotiation ontology. OWL-S can be used to define an ontology that allows a common understanding of the negotiation subject [Buss2002], [Paol2002]. For automated negotiations in SOA we propose a negotiation ontology that consists of several non-functional service parameters.

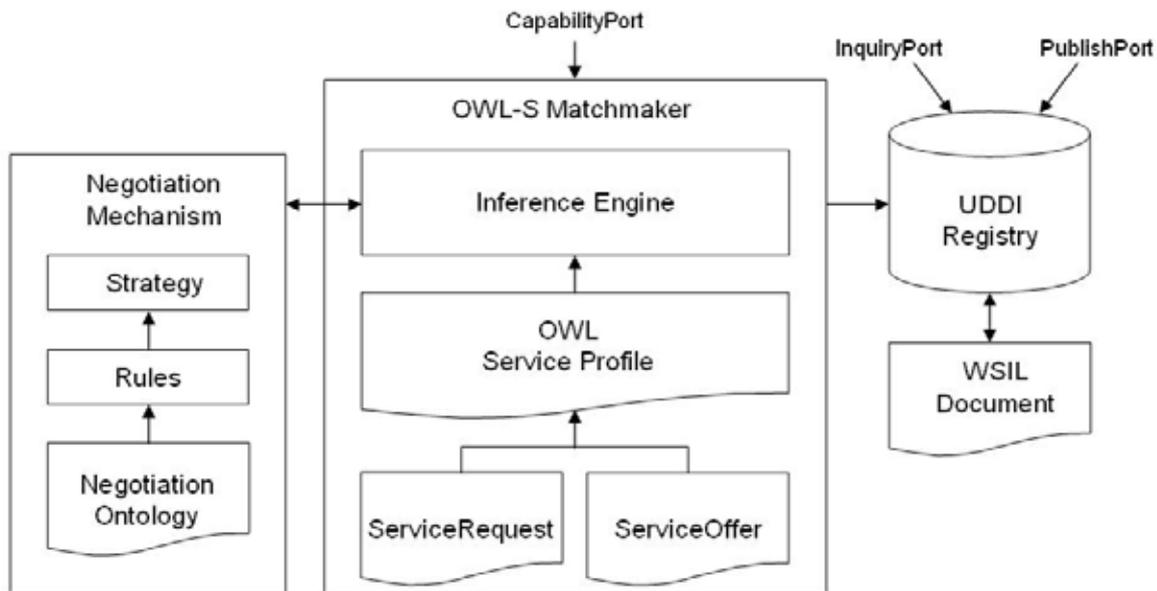


Figure 2-2: Extended service broker architecture for automated negotiations in SOA

Table 2-2 shows the main non-functional service parameters service price and Quality of Service as well as additional parameters like reputation of negotiation partners or duration of service lease.

Negotiating a Service Level Agreement between service provider and service requestor is a complex task. Before the actual negotiation process starts the service profiles for searching services and for providing services have to be defined. In addition to that for the service provider it is necessary to establish an architecture for measuring and monitoring the service parameters. During the negotiation phase all parameters must be understood by both negotiation parties. This is done with the help of the negotiation ontology, which describes the meaning of the above mentioned non-functional service parameters in a machine-readable way. Using this description the rules of the negotiation mechanism consider the service parameters and their meaning for the negotiation parties. We argue that for the main non-functional parameters price and Quality of Service it is likely that service provider and service requestor have a common understanding. Furthermore service provider and service requestor know how to measure these service parameters. With the additional non-functional parameters such as reputation of negotiation partners and security it is difficult to come to an agreement. Measuring and monitoring the additional non-functional service parameters is more complex, because it requires a reputation mechanism and policies for secure service delivery. Besides the

challenges of automatic service selection the semantic description of services and the use of a negotiation ontology are fundamental parts of the extended service broker architecture. For the remainder of this article we assume the existence of the described architecture and examine the economic decisions of service providers and service requestors within this technological setting in more detail.

Non-functional service parameters:	
Service price	
Quality of Service	Server performance: response time, availability, throughput, reliability Network performance: bandwidth, latency, jitter
Security	
Negotiation mechanism	
Number of negotiation partners	
Reputation of negotiation partners	
Duration of service lease	
Sanctions or penalties in case of non-compliance to the service level agreement	

Table 2-2: Non-functional service parameters for negotiations in SOA

3 An agent-based approach for simulating the economic coordination in SOA

3.1 Simulating the economic coordination in SOA

The negotiation concepts described in 2.2 are a set of possible extensions to SOA to finally provide an automated service selection based on economic criteria. In 2.3 we also discussed technologies for implementing these negotiation concepts. One possibility to examine the different negotiation mechanisms and the described technologies in more detail is to build a prototype, which implements the described extended service broker architecture. Based on a prototype it is possible to experiment with a concrete technological setting, the negotiation mechanism, and the negotiation ontology. Another possibility is to model and simulate the described architecture and the behavior of service providers and service requestors. For our research we decided build an agent-based simulation model to examine the economic coordination in SOA. The simulative approach is used, because it allows analyzing different negotiation scenarios and different parameter sets in less time compared to experimenting with a prototype.

On the one hand existing simulations in the field of electronic negotiations and auction theory concentrate on the economic analysis of different negotiation mechanisms [Bich2001], [Chav1996]. They examine what negotiation mechanism

leads to an efficient allocation, what negotiation mechanism leads to higher profit for sellers or what strategy of buyers and sellers performs best for a given negotiation mechanism. The analysis is usually done without considering a specific negotiation domain or a specific technological setting. On the other hand in the field of service-oriented computing and distributed systems the analysis concentrates on technological aspects of SOA [Nara2002], [Sava2005]. Our model includes both, technological aspects as well as economic aspects for exchanging services in SOA. For instance the model considers the specifics for trading digital services in terms specific cost functions, the composition of services, interaction with a service broker or direct interaction, and Quality of Service measures. The considerations and economic decisions of service providers and service requestors as well as their interaction are modeled by using the concept of autonomous software agents [Nwan1996]. For implementing the agent-based simulation we use the open source agent framework JADE (Java Agent Development Framework).

3.2 Model overview

In this section we introduce a model for simulating the economic coordination between service provider and service requestor. The model abstracts the behavior of service providers and services requestors in terms of their economic decisions and reactions to an actual technological setting. Our goal is to use this model for conducting experiments on the impact of important design decisions for implementing negotiations in SOA. In particular it is a question of how the used negotiation mechanism affects the profit of either service providers or service requestors. Another question is if the consideration of different Quality of Service parameters results in a higher delivery rate of services or not.

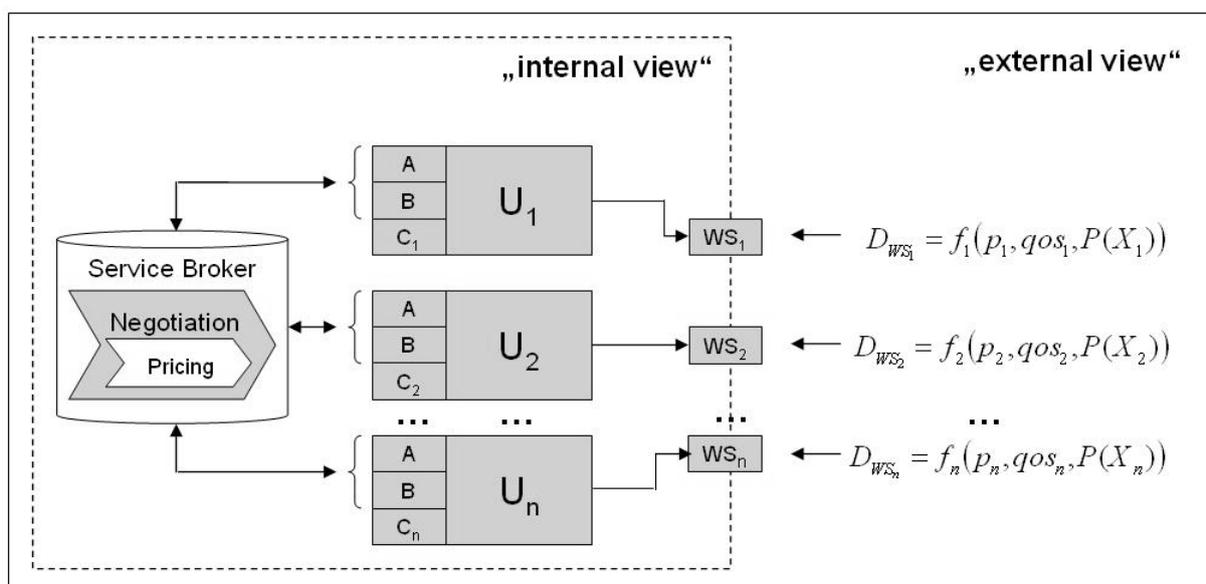


Figure 3-1: Model overview

To examine these research questions we differ between an internal and an external view. The external view only looks at the demand for external services, while the internal view includes the more complex considerations for producing external

services. One reason for that set up is the abstraction from external economic coordination e.g. competition between service providers to attract service requestors. Instead of considering the coordination for consuming external services the model describes the economic coordination for procuring internal services and for producing external services. Figure 3-1 depicts the model overview.

The rules for service providers are as follows: Before a service provider actually delivers external services they need to be built out of internal services A, B and C_n . This process is modeled by the value chains specific for each service provider. For composing external services out of internal ones service providers produce internal services. To do that each service provider is assigned an internal resource pool that can be used to produce internal services A, B and C_n . Instead of producing internal services A and B it is also possible to buy them. By contrast it is not possible to buy the internal services C_n , since they are specific services belonging to each service provider. Trading internal services A and B can be done either in direct interaction between the enterprises or by contacting the service broker. This is where the mentioned negotiation mechanisms come into effect. For a fixed-price scenario where the traded internal services A and B are offered for the price p_A and p_B the cost for procuring A and B can be easily calculated. In case of dynamic pricing the outcome depends at least on the strategic behavior of other enterprises as well as on the own strategy. One strategic consideration can be that a service provider is buying A and B to use its resource pool for the production of the specific internal service C_n , since this will result in a higher output of external services.

Based on the described rules for enterprises acting as service providers or service requestors it is possible that different concepts to negotiate between service provider and service requestor will have an effect on profits for the involved enterprises or delivery rate of external services.

3.3 Model parameters and agent interaction

In JADE the messaging between agents and the management of their behavior is done with the help of standard methods for agent interaction. Based on the framework we implemented different types of agents, their behavior classes and rules for the agent interaction. Besides the enterprises acting as service provider or service requestor the model also includes the abstraction of a service broker. In the case of direct interaction between agents this service broker is not needed. For modeling negotiation mechanisms like simple auctions or double auctions as well as for a fixed price scenario the service broker is represented by a broker agent. To get a better understanding of what is modeled by each agent table 3 shows the model parameters and their description.

model parameter:	description:
$D_{WS_n} = f_n(p_n, qos_n, P(X_n))$	Simulation input: demand function of external service requestors depending on price, Quality of Service and a random variable
$x_1 = f_2(x_A, x_B, x_{C_1})$ $x_A = f_3(res)$ $x_B = f_4(res)$ $x_{C_1} = f_5(res) \quad res < bnd$ $bgt_t = bgt_{t-1} + pi_1$	Agent_type1: Enterprises acting as service provider or service requestor: value chain for producing one unit of x_1 , usage of resource pool in case of producing internal services, budget
$bid = bid(u, S)$ $u = u(x_A, x_B)$ $S = S(bgt_{t-1}, F(Y))$	bidding for internal services A, B based on the utility derived from consuming internal services and based on the bidding strategy specific for the selected negotiation mechanism
$pi_1 = r_1 - c_1$ $r_1 = p_1 \cdot x_1 + trd$ $c_1 = c_A + c_B + c_{C_1} + c_{trans}$ $c_A = p_A \cdot x_A$ $c_B = p_B \cdot x_B$	profit for service providers considers revenue out of providing external services and out of trading internal services as well as costs for producing, procuring and trading
$D_A = f_6(p_A, qos_A)$ $D_B = f_7(p_B, qos_B)$ $S_A = f_8(p_A, qos_A)$ $S_B = f_9(p_B, qos_B)$ $D_A = S_A \quad D_B = S_B$	Agent_type2: Service Broker: holding demand and supply of internal services A, B clearing offers and counteroffers pricing internal services depending on the selected negotiation mechanism
pi_n, p_A, p_B, qos_n	Simulation output: profit of service providers, market price for internal services A and B, Quality of Service for external services

Table 3-1: Model parameters

One simulation run consists of a given number of similar rounds. Each round starts with the demand for external services as the simulation input parameter. Based on that information an enterprise decides whether to provide external services or not. Specific to each enterprise the value chain parameters describe what internal services are needed to finally provide the external services. Depending on the used negotiation mechanism an enterprise can buy internal services for a given price or negotiate the price. Different negotiation mechanisms require different strategies of

enterprises when they are bidding for internal services or just buying them. A simulation round ends with calculating the market price for internal services, the profit of service providers, and Quality of Service for external services.

4 Discussion and future work

This article gave a short overview of existing technologies and further concepts for implementing negotiations in SOA. We argued that in the case of economically independent enterprises the economic perspective for exchanging services is of particular interest. We introduced possible extensions to SOA and suggested current technologies for their implementation. The most important extension is an architecture that allows automated negotiations of price and other parameters such as Quality of Service for the exchanged services. The proposed extended service broker architecture makes use of a negotiation ontology that allows a common understanding of the negotiation subject.

The goal to adapt the mentioned standard negotiation mechanisms so that they match with the technological setting of SOA can be achieved by conducting an analysis on the outcome of negotiations in SOA. As a method for further analysis of design decisions and negotiation parameters we described an agent-based approach for simulating the economic coordination between service provider and service requestor. The model abstracts the behavior of service providers and services requestors in terms of their economic decisions and reactions to an actual technological setting. Our goal is to use this model for conducting experiments on the impact of important design decisions for implementing negotiations in SOA.

References

- [Alon2003] Alonso, G.: *Web Services: Concepts, Architectures and Applications*. Springer, 2003.
- [Bich2001] Bichler, M.: *The Future of E-Markets: Multi-Dimensional Market Mechanisms*. Cambridge University Press, Cambridge, 2001.
- [Bole2004] Boles, D.: *Erlösformen für Web Content und Services*. Informatik Springer, 2004.
- [Buss2002] Bussler, C., Fensel, D., Maedche, A.: A conceptual architecture for semantic web enabled web services. *ACM SIGMOD Record*, v.31 n.4, December 2002.
- [Chav1996] Chavez, A., Maes, P.: *Kasbah: An agent marketplace for buying and selling goods*. In: *First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology*, London, Great Britain, 1996.
- [Chen2003] Chen, Z. et al.: *UX -An Architecture Providing QoS-Aware and Federated Support for UDDI*, ICWS '03, School of Computer Engineering, Nanyang Technological University, 2003.
- [Dan2004] Dan, A. et al.: *Web services on demand: WSLA-driven automated management*. *IBM Systems Journal*, v.43 n.1, p.136-158, 2004.
- [Gram2003] Gramm, A.: *WS-QOS - Ein Rahmenwerk für Dienstgüteunterstützung in Web Services*, Institut für Informatik, Freie Universität Berlin, 2003.

- [Elfa2004] Elfatry, A., Layzell, P.: Negotiating in Service-oriented Environments. *Comm. ACM*, 47 (8), 103-08., 2004.
- [Kris2002] Krishna, V.: *Auction Theory*. Academic Press, 2002.
- [Lama2003] Lamanna, D. et al.: *SLang: A language for Service Level Agreements*, Department of Computer Science, University College London, 2003.
- [Lind1998] Lindemann, M.A., Runge, A.: *Electronic Contracting within the Reference Model for Electronic Markets*, 6th European Conference on Information Systems (ECIS-98). Aix-en-Provence, France, 1998.
- [Ludw2003] Ludwig, H.: *Web Services QoS: External SLAs and Internal Policies Or: How do we deliver what we promise?* In *Proc. of the First Web Services Quality Workshop*, IEEE Computer Society, Rome, 2003.
- [Milg2004] Milgrom, P.R.: *Putting Auction Theory to Work (Churchill Lectures in Economics)*. Cambridge University Press, 2004.
- [Nara2002] Narayanan, S., McIlraith, S.: *Simulation, Verification and Automated Composition of Web Services*. In: *Proceedings of the 11th International World Wide Web Conference*. 2002.
- [Nwan1996] Nwana, H. S.: *Software Agents: An Overview*, *Knowledge Engineering Review*, 11/3 p. 205-244, 1996.
- [Paol2002] Paolucci, M. et al.: *Importing the Semantic Web in UDDI*, in *Proceedings of Web Services, E-business and Semantic Web Workshop 2002*.
- [Pate2003] Patel, K., Tasic, V.: *Web Service Offerings Language (WSOL): Characteristics, Applications, and Tools*. 2003.
- [Sava2005] Savarimuthu et al.: *Agent-based integration of Web Services with Workflow Management Systems*. In *Proceedings of the Fourth international Joint Conference on Autonomous Agents and Multiagent Systems*, ACM Press, New York, 2005.
- [Shai2003] Shaikh A. et al.: *UDDIe: An Extended Registry for Web Services*, Department of Computer Science, Cardiff University, UK, 2003.
- [Sege1999] Segev, A., et al.: *On Negotiations and Deal Making in Electronic Markets*, *Information Systems Frontiers*, v.1 n.3, p.241-258, October 1999.
- [Sing2005] Singh, M., Huhns, M.: *Service-Oriented Computing: Semantics, Processes, Agents*. Wiley, 2005.
- [Srin2004] Srinivasan, N. et al.: *Adding OWL-S to UDDI, implementation and throughput*, *First International Workshop on Semantic Web Services and Web Process Composition*, San Diego, California, USA, 2004.
- [Tian2004] Tian, M. et al.: *A Survey of current Approaches towards Specification and Management of Quality of Service for Web Services*. Freie Universität Berlin, Institut für Informatik, 2004.
- [Verh2004] Verheecke, B. et al.: *AOP for Dynamic Configuration and Management of Web Services*, *International Journal of Web Services Research*, 1(3), 25-41, 2004.
- [W3C12004] Booth et al.: *Web Services Architecture*. W3C Working Group Note, 2004. <http://www.w3.org/TR/ws-arch/>

[Zimm2003] Zimmermann, O., Tomlinson, M., Peuser, S.: Perspectives on web services: applying SOAP, WSDL, and UDDI to real-world projects. Springer, Berlin, 2003.

SIMJADE – Ein Simulationsdienst für die verteilte Ausführung agentenbasierter Modelle

Dirk Pawlaszczyk

Technische Universität Ilmenau
98693 Ilmenau
dirk.pawlaszczyk@tu-ilmenau.de

Zusammenfassung: Agentenbasierte Simulation ist eine Analysetechnik die es erlaubt, das Verhalten komplexer Systeme sowie emergente Phänomene zu studieren. In diesem Beitrag wird ein ereignisbasierter Simulationsdienst vorgestellt, der die Durchführung verteilter Simulationsexperimente auf Basis eines optimistischen Synchronisationsansatzes unterstützt. Ferner werden erste Ergebnisse einer Simulationsstudie gezeigt.

1 Einleitung

Simulation stellt eine universell einsetzbare Analyse- und Planungsmethode für viele Anwendungsbereiche zur Verfügung. Simulationsanwendungen werden in Ingenieur- und Naturwissenschaften, ebenso wie innerhalb der Soziologie und nicht zuletzt in den Wirtschaftswissenschaften eingesetzt. Im unternehmerischen Alltag bildet die Methode der Simulation mittlerweile eine feste Größe, etwa wenn es darum geht, betriebswirtschaftliche Entscheidungsprozesse vorzubereiten und Prognosen zu erstellen. Mit der Anwendung von Simulationsansätzen im betriebswirtschaftlichen Umfeld werden insbesondere Entscheidungen adressiert, bei denen die Handlungsspielräume begrenzt sind und gleichzeitig die Auswertung analytisch nicht durchführbar ist. Dies ist insbesondere dann der Fall, wenn die für eine Entscheidungssituation signifikante Ungewissheit der beteiligten Einflussfaktoren explizit berücksichtigt werden soll.

In jüngster Zeit gewinnt eine spezielle Simulationsmethode - die Agentenbasierte Simulation (ABS) - zunehmend an Bedeutung. Agententechnologie verspricht flexible und adaptive Lösungen für bestehende realwirtschaftliche Problemstellungen. Bislang liegen jedoch nur eingeschränkt Erfahrungen mit dem Einsatz von agentenbasierten Systemen im betriebswirtschaftlichen Umfeld vor. Darüber hinaus fehlt es an Werkzeugen zum Test und der Validierung von agentenbasierten Modellen. Insbesondere in Szenarien, bei denen wirtschaftlich und rechtlich selbständige Organisationen an der Leistungserbringung beteiligt sind, scheint eine akteurgetriebene Sichtweise sinnvoll. Multiagentensysteme (MAS) bieten eine geeignete Metapher, um Handlungsträger und deren Interaktionsbeziehungen in Simulationsmodellen abzubilden.

Um die Komplexität realwirtschaftlicher Phänomene studieren zu können, beispielsweise im Rahmen des Geschäftsprozessmanagements, bedarf es geeigneter Werkzeuge und Techniken. Gerade für die Simulation komplexer agentenbasierter Modelle mit hunderten oder gar tausenden Softwareagenten bestehen bisher kaum Erfahrungen. Die zunehmende Komplexität der zu simulierenden Agentenmodelle erfordert Simulationstechniken die in der Lage sind, den wachsenden Ressourcenbedarf zu decken, d.h., die entsprechend skalierbar sind.

In diesem Beitrag wird ein Simulationsdienst zur Durchführung verteilter Simulationsexperimente vorgestellt. Im Mittelpunkt steht dabei ein Ansatz zur Zeitsteuerung, die mit Hilfe eines optimistischen Synchronisationsansatzes realisiert wurde. An den Anfang der Betrachtung wird jedoch zunächst eine kurze Einführung in den Bereich der agentenbasierten Simulation gestellt. Im Anschluss daran werden grundlegende Ansätze zur Zeitfortschreibung von Simulationsanwendungen dargelegt, sowie verwandte Arbeiten auf diesem Gebiet vorgestellt. Daran schließt sich die Darstellung des eigentlichen Simulationsdienstes an. Erste Ergebnisse einer Simulationsstudie zur Bewertung der Leistungsfähigkeit der Simulationsumgebung werden ebenfalls vorgestellt. Der Beitrag schließt mit einer kurzen Zusammenfassung und einem Ausblick zu weiterem Forschungsbedarf.

2 Agentenbasierte Simulation

2.1 Begriffsbestimmung

Unter Simulation wird allgemein die Nachbildung der Struktur und des Verhaltens eines Systems in einem Computermodell verstanden. Ziel ist es, durch Experimente an diesem Modell zu Erkenntnissen zu gelangen, die auf die Realwelt übertragbar sind (vgl. [VDI2003]). Unter Agentenbasierter Simulation (ABS) wird die Simulation von Modellen verstanden, die sich aus so genannten Agenten zusammensetzen [Davi2000, 97]. Agenten sind Softwareartefakte, in der Lage sind, ihre Umgebung wahrzunehmen und unter Rückgriff auf lokal hinterlegtes Wissen, Schlüsse über diese Wahrnehmungen ziehen. Darauf aufbauend wählt ein Agent anschließend Aktionen aus, mit denen er seine Umwelt gezielt verändern kann. Agenten können mit anderen Agenten Informationen austauschen, wobei ihr Lebenszyklus nicht direkt von dem eines anderen Agenten abhängig ist: *„Agents are situated in an environment, act autonomously, and are able to sense and to react to changes“* [KnTi1999, 2]. Neben Ihrer Autonomie sind Software-Agenten häufig durch soziale Eigenschaften charakterisiert, was sich u.a. in der Fähigkeit widerspiegelt, benachbarte Systeme zu erkennen und mit diesen zu kommunizieren [WoJe1995, 1-32]. Multiagentensysteme (MAS) wiederum bestehen aus Agenten, die zum Zweck der Lösung einer gestellten Aufgabe mit anderen im System vorhandenen Agenten kooperieren.

Agentenbasierte Simulationsansätze sind geeignet, das globale Verhalten von Systemen abzubilden, welches sich aus der Interaktion einer Vielzahl von lokal agierenden, über nicht-lineare Wechselwirkungen miteinander vernetzten Individuen ergibt. ABS erlaubt es, Ordnungszustände in diesen Systemen zu analysieren und insbesondere emergente Phänomene nachzubilden. Unter Emergenz wird allgemein das Entstehen neuer Strukturen bzw. Eigenschaften durch die Wechselwirkung von

Teilen eines Systems verstanden, wobei aus der Beschaffenheit der einzelnen Systembestandteile allein nicht auf das beobachtete Verhalten auf globaler Ebene geschlossen werden kann.

Es existiert eine Reihe von Simulationsumgebungen, die insbesondere für die Bearbeitung sozialwissenschaftlicher Modelle vorgesehen sind. Andere haben einen Bezug zum Bereich des Artificial Life. Agentenbasierte Simulationsansätze werden in den letzten Jahren aber auch verstärkt zur Analyse betriebswirtschaftlicher Problemstellungen eingesetzt. So finden sich Arbeiten zum Supply Chain Management, zur Simulation im Produktions- und Fertigungsbereich, elektronischen Märkten und Transportnetzen ([DaHe2004, 1-2], [GPSa2005, 119-125]). Generell haben diese Systeme zumeist einen stark begrenzten Anwendungsfokus oder untersuchen speziell Fragen der Modellbildung.

Der Entwicklung von Werkzeugen zur effizienten Durchführung von Simulationsexperimenten wird bisher hingegen vergleichsweise wenig Beachtung geschenkt. Auf der anderen Seite nimmt die Komplexität der mittels Agententechnologie zu lösenden Problemstellungen und gleichsam die Größe der zu simulierenden Modelle stetig zu. Agentenbasierte Simulation erlaubt es, ex ante agentenbasierte Modelle und Anwendungen zu testen bzw. zu evaluieren und leistet somit einen wichtigen Beitrag im Rahmen der Softwarequalitätssicherung. Flexible, effizient einsetzbare Lösungen für die Simulation agentenbasierter Systeme gewinnen an Bedeutung [Loga2005, 5].

3 Synchronisationsansätze

Die effiziente Durchführung von Simulationsläufen wird in erster Linie durch das verwendete Zeitfortschreibungsverfahren bestimmt ([Schu2002, 75], [Fuji2000, 30-48]). Das vorliegende Kapitel gibt einen Überblick zu wichtigsten bestehenden Verfahren der Zeitsteuerung innerhalb der Simulationstechnik und bewertet diese in Hinblick auf ihre Eignung für den Einsatz im Rahmen der Agentenbasierten Simulation.

3.1 Der Zeitbegriff in der Simulation

Im Kontext der Simulation sind drei verschiedene Zeitebenen von einander abgrenzbar. Zuvorderst unterscheidet man die so genannte *physikalische Zeit*. Diese beschreibt die reale, auf das Originalsystem bezogene Zeit. Davon begrifflich abgegrenzt wird die eigentliche *Simulationszeit*, welche im Modell die physikalische Zeit nachbildet. Jeder Zeitpunkt im Modellsystem findet somit eine Entsprechung in der Realzeit des Urbildsystems. Die real ablaufende Zeit während eines Simulationslaufs schließlich, aus der sich die Ausführungszeit bestimmt, wird als *Wallclock Time* bezeichnet [Fuji2000, 27-28].

3.2 Zeitfortschreibung innerhalb der Simulation

Der Zustand eines dynamischen Systems kann bezogen auf die Zeit kontinuierlich oder diskret betrachtet werden [Meh1994, 2-8]. Kontinuierliche Modelle sind dadurch gekennzeichnet, dass sich die Werte der zu simulierenden Zustandsvariablen in Bezug auf die Zeit stetig verhalten. Es gibt bei dieser Modellform keine Zeitspanne die ereignislos verläuft. Kontinuierliche Modelle basieren auf Differentialgleichungs-

systemen und finden insbesondere bei der Untersuchung physikalischer Vorgänge Anwendung, beispielsweise in der Klimaforschung. Viele Systeme sind jedoch dadurch geprägt, dass sich ihr Zustand sprunghaft ändert, um dann wiederum für eine gewisse Zeit ereignislos zu verharren. Die dafür notwendige Diskretisierung der Zeitachse erfolgt entweder zeitgesteuert oder ereignisdiskret. In der zeitgesteuerten (*time driven*) Simulation wird die Zeitskala zumeist in äquidistante Abschnitte unterteilt. Zustandsänderungen können entsprechend nur zu bestimmten Zeitpunkten stattfinden. Die Zeitspanne von einem Zeitpunkt zum nachfolgenden wird als ereignislos angenommen und einfach übersprungen (Totzeit). Die besondere Herausforderung und zugleich der größte Nachteil bei dieser Form der Zeitsteuerung besteht darin, eine geeignete Intervalllänge festzulegen [Mehl1994, 2-5], [Fuji2000, 30-33]. So muss einerseits sichergestellt sein, dass ein Ereignis nur in der Zukunft liegende Ereignisse beeinflussen kann, nicht aber solche, die im gleichen Zeitintervall ausgeführt wurden. Andererseits sollte die Intervalllänge möglichst auch nicht zu klein sein, da sonst unnötig viele ereignislose Intervalle mit simuliert werden müssen. Die Mehrzahl derzeit existierender agentenbasierter Simulationsanwendungen nutzt dennoch einen zeitgesteuerten Steuerungsansatz [Loga2005, 5]. Der Hauptgrund hierfür liegt nicht zuletzt in der einfachen Implementierbarkeit dieses Ansatzes, verglichen mit anderen Verfahren der Zeitfortschreibung. Darüber hinaus begnügen sich die meisten derzeit existierenden Simulationsumgebungen für Agenten mit der Abbildung einer relativ kleinen Anzahl von Agenten bzw. simulieren zwar eine große Zahl an Agenten, die dafür wiederum sehr einfach strukturiert sind. Außerdem sind die Systeme häufig im akademischen Bereich angesiedelt und in der Regel nicht für die Abbildung großer realwirtschaftlicher Problemstellungen ausgelegt.

Anders als in der zeitgesteuerten Simulation ist die Länge der Totzeiten - die Zeitspanne, die zwischen einem Ereignis und seinem unmittelbaren Nachfolger liegt - innerhalb ereignisdiskreter Modelle grundsätzlich variabel. Ein Ereignis bezeichnet den Zeitpunkt der Zustandsänderung einer oder mehrerer Modellentitäten. Ereignisse sind neben ihren Eintrittszeitpunkt zusätzlich durch ihren Ereignistyp näher bestimmt. Tritt ein Ereignis ein, so wird eine Zustands-Transformation bei den betroffenen Modellelementen durchgeführt und eventuelle Folgeereignisse generiert. Der Nachteil zeitdiskreter Modelle, naturbedingt lange Zeitintervalle ohne Zustandsänderungen am Modell simulieren zu müssen, wird innerhalb der ereignisdiskreten Simulation dadurch begegnet, dass stets die Eintrittszeit desjenigen Ereignisses mit dem aktuell kleinsten Zeitstempel ausgeführt wird. Ereignisdiskrete Simulationsmodelle sind dadurch sehr flexibel. Diese Modellform besitzt einen hohen Verbreitungsgrad im wirtschaftlichen Bereich.

3.3 Ansätze der verteilten Simulation

Neben den bisher vorgestellten zentralistischen Ansätzen zur Zeitsteuerung von Simulationsanwendungen existieren zusätzlich Verfahren aus dem Bereich der verteilten Simulation. Im Rahmen der verteilten Simulation wird das auszuführende Modell zunächst in Teilmodelle so genannte Logische Prozesse (LPs) untergliedert. Jeder LP wird anschließend einem Rechner in einem Rechnernetz zugeordnet und parallel mit anderen LPs ausgeführt. Durch die verteilte Ausführung des Simulationsmodells wird das Ziel verfolgt, die dem jeweiligen Modell innewohnende Parallelität nutzbar zu machen. Dadurch kann zum einen die Ausführungszeit für

einen Simulationslauf verringert werden. Zum anderen wird das System so skalierbar, da Verteilung eine Grundvoraussetzung für Skalierbarkeit bildet [Law1998, 1-3]. Der letzte Punkt ist umso bedeutender je komplexer das zu simulierende Modell ist, d.h. je mehr Agenten zu simulieren sind.

Mit der verteilten Ausführung des Simulationsmodells werden jedoch auch neue Fragestellungen aufgeworfen. Da jeder Prozess nunmehr seine Ereignisse selbst verwaltet und keiner globalen Zeitsteuerung mehr unterworfen ist, schreiten die Logischen Prozesse während eines Simulationslaufes naturgemäß unterschiedlich weit in der Simulationszeit voran. So ist es beispielsweise möglich, dass der Laufzeitprozess eines Agenten a_1 bereits ein Ereignis mit dem Zeitstempel 19 ausgeführt hat, während ein zweiter Prozess a_2 , der in seiner Ereignisabarbeitung noch nicht so weit voran geschritten ist wie Prozess a_1 , ihm ein Ereignis mit einem früheren Ereigniszeitpunkt, beispielsweise 10, zuschickt (vgl. Abb. 3-1). Im Ergebnis ist die kausale Ordnung der Ereignisse somit möglicherweise nicht mehr gewährleistet. In der Folge könnte der Simulationslauf andere Ergebnisse erzeugen, als es bei einer unverteilter Ablaufsteuerung der Fall gewesen wäre. Um dies zu vermeiden und die kausale Ordnung in jedem Fall zu sicherzustellen, müssen die Agenten-Prozesse entsprechend synchronisiert werden³¹.

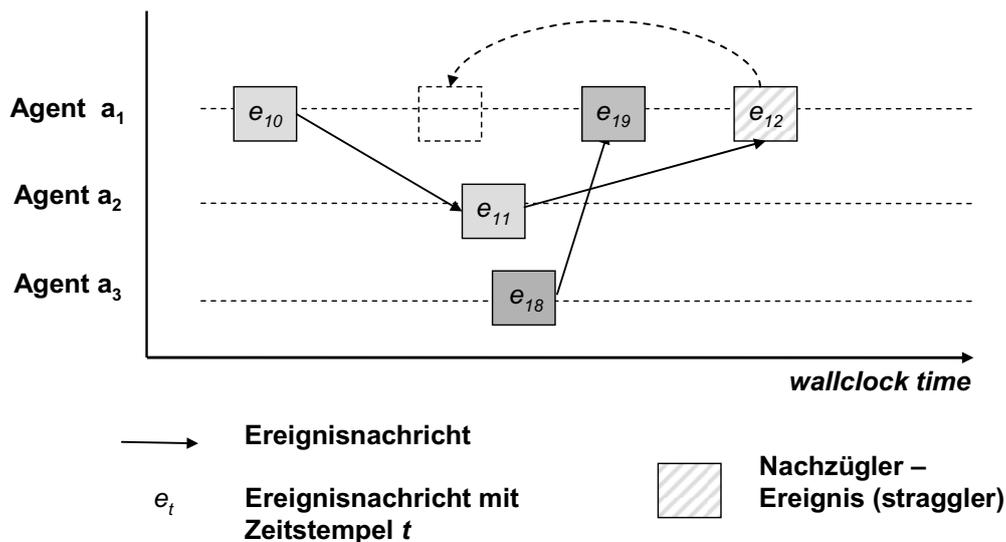


Abb. 3-1: Problem der Kausalitätsverletzung im Rahmen der verteilten Simulation

Zur Lösung dieses Problems wurden unterschiedliche Ansätze entwickelt, die zwei Klassen zuordenbar sind: *Konservative* und *Optimistische* Verfahren ([Fuji2000], [Schu2002]). *Konservative* Synchronisationsverfahren vermeiden Kausalitätsverletzungen von vornherein. Hierfür versendet jeder LP so genannte Garantie-Nachrichten an alle benachbarten Prozesse, mit denen er während eines Simulationslaufs kommuniziert. Eine Garantie ist in diesem Fall als Zusicherung darüber zu verstehen, dass der jeweilige Prozess einen bestimmten Zeitpunkt innerhalb der Simulationszeit bereits überschritten hat, und zeitlich gesehen nicht mehr hinter diesen Zeitpunkt zurückfallen wird. Insbesondere bei der Simulation großer Warteschlangennetze

31 Dieser Umstand wird in der Literatur unter dem Begriff der so genannten *Local Causality Constraint (LCC)* diskutiert [ThLo1999].

konnten mit diesem Verteilungsansatz gute Ergebnisse in Hinblick auf die Beschleunigung der Simulationsexperimente erzielt werden [Fuji2000]. Durch das Bestreben die Kausalität der Ereignisse zu wahren, reduziert sich andererseits die effektiv nutzbare Parallelität des Modells. Hinzu kommt, dass konservative Verfahren implizit ein statisches Modell voraussetzen. Auf Grund dieser Eigenschaften erscheinen konservative Verfahren für die Simulation hochdynamischer Umgebungen, wie sie im Rahmen der agentenbasierten Simulation in der Regel vorzufinden sind, nur bedingt geeignet.

Die zweite Klasse verteilter Simulationsverfahren bilden die so genannten *Optimistischen* Verfahren. Anders als im konservativen Fall lassen diese mögliche Kausalitätsverletzungen zunächst zu. Wird aber im Verlauf der Simulation festgestellt, dass die zeitliche Ordnung der Ereignisse nicht eingehalten wurde, so wird dies im Nachhinein korrigiert. Um dies zu ermöglichen, müssen alle LP von Zeit zu Zeit eine Kopie ihres aktuellen Ausführungszustandes anlegen, auf die im Bedarfsfall zurückgegriffen werden kann. Trifft wie oben geschildert ein Nachzügler-Ereignis mit einem Zeitstempel t ein, wobei t kleiner der aktuellen lokalen Zeit des jeweiligen Prozesses ist, so plant der Prozess das Ereignis zunächst ein. Anschließend setzt der LP seinen Zustand mittels der zuvor angelegten Zustandskopie auf einen Zeitpunkt t' zurück, wobei $t' < t$ gilt. Die Ereignisausführung wird entsprechend an diesem Zeitpunkt fortgesetzt. Dieser Vorgang wird auch als Rollback bezeichnet [Fuji2000, 100-106].

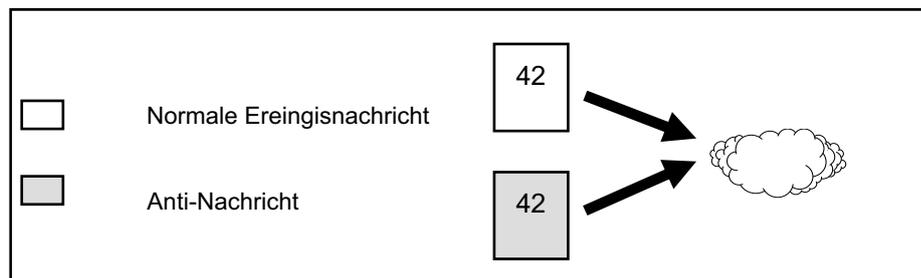


Abb. 3-2: Konzept der Anti-Nachrichten in der optimistischen Synchronisation (Nachrichten-Annihilation)

Zusätzlich müssen alle Ereignisse mit dem Zeitstempel t^* ($t' \leq t^* \leq t$) die der Prozess zwischenzeitlich erzeugt hat ebenfalls rückgängig gemacht werden, da diese bei der wiederholten Ausführung nicht zwingend ebenfalls erneut erzeugt werden müssen. Um eine weitere Ausbreitung einer potentiell falschen Berechnung zu verhindern, sendet der LP deshalb so genannte Anti-Nachrichten an die betroffenen LPs. Eine Anti-Nachricht bildet das exakte Gegenstück zu der ursprünglich versendeten Ereignisnachricht und ist lediglich durch ein zusätzliches Flag gekennzeichnet. Trifft eine Anti-Nachricht auf ihren Konterpart, so heben sich die beiden Nachrichten einfach auf (siehe Abb. 3-2). Wurde die Ereignisnachricht bereits vom empfangenden Prozess verarbeitet, so muss dieser zusätzlich seinen Ausführungszustand zurückrollen und seinerseits Anti-Nachrichten versenden. Dieser Vorgang setzt sich solange fort, bis alle potentiell falschen Berechnungen vollständig rückgängig gemacht worden sind. Den wohl bekanntesten Vertreter optimistischer Synchronisationsverfahren bildet der *Time-Warp*-Ansatz von Jefferson [JeBe1987]. Im Vergleich zu den konservativen Verfahren bietet der optimistische Ansatz eine bessere Aus-

nutzung der potentiellen Parallelität eines Simulationsmodells. Bei einigen Modellen besteht die Gefahr, dass dieser Vorteil durch das gehäufte Auftreten von Rollbacks auf Grund potentiell falscher Berechnungen wieder kompensiert wird. Hinzu kommt ein erhöhter Speicher- und Rechenaufwand für das Anlegen der Zustandskopien (vgl. [Schu2002, 82-88], [Fuji2000, 172-175]).

Dessen ungeachtet erscheint der optimistische Synchronisationsansatz am ehesten geeignet, eine Beschleunigung für ein breites Spektrum an Simulationsmodellen zu gewährleisten ([Fuji2000, 176], [ThLo1999, 58-65]). Darüber hinaus gestattet der Ansatz im Gegensatz zu konservativen Verfahren den dynamischen Aus- und Eintritt von Prozessen zur Laufzeit.

4 Überblick zu verteilten Zeitsteuerungsansätzen innerhalb der Agentenbasierten Simulation

Eines der ersten generischen Systeme zur agentenbasierten Simulation überhaupt war GENSIM von *Anderson* und *Evans* [AnEv1995, 527-528]. In der verteilten Version D-GENSIM wird ein zeitgesteuerter Synchronisationsansatz verfolgt [Ande2000]. Andere Simulationssysteme wie beispielsweise das SYNCER-Framework, eine verteilte Variante der bekannten SWARM Simulationsumgebung, setzt ebenfalls auf eine zeitgesteuerte Zeitfortschreibung [GoSa2001].

Ein Beispiel für die verteilte Ausführung eines Simulationsmodells basierend auf einem ereignisgesteuerten Ansatz bietet der im Rahmen der Projekte *Agent.Enterprise* und *Agent.Hospital* eingesetzte Zeitsteuerungsdienst [BrPo2004]. Die Kopplung der einzelnen Agentensimulatoren erfolgt hier über eine zentral verwaltete Ereignisliste, wobei die Simulatoren selbst auf verschiedene Rechner verteilt werden können.

Anwendungsbeispiele für das konservative Synchronisationsverfahren finden sich beispielsweise beim MPADES-Projekt, einer Simulations-Middleware für die verteilte Simulation agentenbasierter Modelle [Rile2003]. Auch das von *Gasser* et al. entwickelte System MACE3J verwendet einen konservativen Ansatz zur Zeitsteuerung [GaKa2004]. Diese Anwendung wurde explizit zur Unterstützung skalierbarer Simulationsexperimente entwickelt. Für MACE3J geben die Entwickler an, auf einem Shared-Memory-System ein Modell mit rund 5000 Agenten ausgeführt zu haben, und schlussfolgern daraus, dass das System skaliert. Leider wurden für das Testszenario keinerlei Nachrichten zwischen den Agenten ausgetauscht, wodurch sich die Aussagekraft dieses Ergebnisses relativiert.

Eine Reihe weiterer Projekte konnte Simulatoren für Agentenmodelle erfolgreich mit Hilfe der so genannten High Level Architecture (HLA) verbinden ([MiTh2004], [LeLo2004], [WaTu2005]). Bei HLA handelt es sich um einen mittlerweile durch die ISO zertifizierten Standard zur Kopplung unterschiedlicher Simulationsanwendungen. Um seinen Simulator im Verbund mit anderen über HLA koppeln zu können, muss zunächst eine entsprechende Schnittstelle, das so genannte Runtime Interface (RTI), implementiert werden. Die Zeitsteuerung in HLA erfolgt ebenfalls unter Verwendung konservativer Verfahren, obwohl die Kopplung optimistischer Simulatoren ebenfalls möglich ist [Fuji2000, 209-220]. HLA wurde primär unter dem Gesichtspunkt der Interoperabilität entwickelt. Die Beschleunigung von Simulations-

experimenten stand ausdrücklich nicht im Vordergrund. Entsprechend sind die mit diesem Kopplungsansatz erzielten Ergebnisse eher moderat.

Die Testumgebung JAMES bildet als eines von zwei Systemen eine Form der optimistischen Zeitsteuerung ab [UhSc1998]. Das Framework wurde speziell für den Test deliberativer Agenten entwickelt. Es basiert formal auf dem DEVS-Ansatz von Zeigler und erlaubt es, diskrete Ereignisse der Agenten auf eine kontinuierliche Umwelt abzubilden. Da die entsprechenden Planungsvorgänge der zu simulierenden Agenten vergleichsweise viel Rechenzeit in Anspruch nehmen, kann jeweils ein Planungsvorgang spekulativ in die Zukunft ausgeführt werden. Wird ein zeitlich konkurrierender Zugriff auf eine oder mehrere Variablen des gemeinsam genutzten Zustandsraumes festgestellt, so wird der Planungsvorgang entsprechend zurückgerollt und wiederholt ausgeführt. Bei diesem System, wie im Übrigen bei der Mehrzahl derzeit existierender Simulationsansätze im Agentenbereich koordinieren die Agenten sich zumeist über ein explizit vorgegebenes Umweltmodell. Eine direkte nachrichtenbasierte Kommunikation der Agenten ist häufig nicht vorgesehen oder wird indirekt über die wechselseitige Manipulation gemeinsamer Zustandsvariablen realisiert. Hauptgrund dafür ist sicherlich der Umstand, dass viele der ursprünglich entwickelten Systeme speziell zur Abbildung von Agenten entwickelt wurden, die in einer räumlich genau determinierten Umgebung situiert sind, und diese durch ihre Aktionen manipulieren [KIHe2003]. Eine direkte Interaktion der Agenten untereinander, etwa durch den Austausch von Nachrichten, wird zumeist nicht unterstützt. Die zur Abbildung des Umweltmodells genutzten Datenstrukturen werden dabei für gewöhnlich zentral verwaltet. So entsteht zeitlich gesehen ein Engpass, da alle Agenten sich über diesen gemeinsamen, zentral verwalteten Zustandsraum koordinieren müssen. Diesen Umstand tragen die Arbeiten von *Theodoropoulos* und *Logan* entsprechend Rechnung [ThLo1999, 58-65]. Ihr Ansatz sieht vor, den gemeinsamen Zustandsraum entsprechend der Zugriffshäufigkeit zu partitionieren und diese Partitionen entsprechend zu verteilen. Die Manipulation des verteilten Zustandsraumes erfolgt mit Hilfe eines optimistischen Synchronisationsansatzes.

Wie gezeigt wurde, existieren verschiedene Ansätze zur verteilten Simulation agentenbasierter Modelle. Die dabei zum Einsatz kommenden Synchronisationsverfahren reichen von einfachen zeitgesteuerten bis hin zu komplexeren optimistischen Verfahren. Für keines der hier vorgestellten Systeme finden sich in der Literatur Aussagen darüber, ob die Systeme tatsächlich für die Simulation großer Agentenmodelle geeignet sind. Aussagen über das Leistungsvermögen der jeweiligen Umgebung sind selten und beziehen sich in der Regel auf sehr kleine Modelle mit einer geringen Anzahl an Agenten.

5 Der Simulationsdienst

Im Folgenden wird ein Simulationsdienst vorgestellt, der speziell für die verteilte Agentenbasierte Simulation entwickelt wurde. Wie der Name *SIMJADE* bereits andeutet, handelt es sich nicht um ein komplett neues Agenten-Framework. Vielmehr setzt der hier beschriebene Simulationsdienst auf dem JAVA Agent Development Environment (JADE) - Framework auf, einer weit verbreiteten Entwicklungsumgebung für Agentensysteme [JADE2006]. Das Java-basierte Framework stellt dem

Entwickler eine anwendungsbereite FIPA-konforme³² Agentenplattform zur Verfügung, die alle grundlegenden innerhalb der FIPA-Architektur spezifizierten Infrastrukturdienste bereitstellt.

5.1 Anwendungsbereiche von *SIMJADE*

Simulationsumgebungen wie SWARM oder REPAST verwenden zumeist reaktive Architekturen zur Abbildung so genannter Schwarmintelligenz [Klüg2001, 99]. Die Agenten werden in der Regel durch einfache Objekte repräsentiert, um eine möglichst große Anzahl an Elementen simulieren zu können. *SIMJADE* hingegen unterstützt, ähnlich wie das JAMES-System oder MACE3J, den Test komplexer deliberativer Agenten, welche als eigenständige Prozesse bzw. Threads implementiert sind ([UhSc1998], [GaKa2004]). *SIMJADE* ermöglicht somit den Test von agentenbasierten Modellen im Rahmen der Softwareentwicklung, da es auf den selben Architekturkonzepten und Kommunikationstechnologien aufbaut, die bei Entwicklung realwirtschaftlicher, agentenbasierter Systeme zum Einsatz kommen und ist problemlos integrierbar. *SIMJADE* realisiert den derzeit einzigen optimistischen Zeitsteuerungsdienst für FIPA-konforme Agentensysteme.

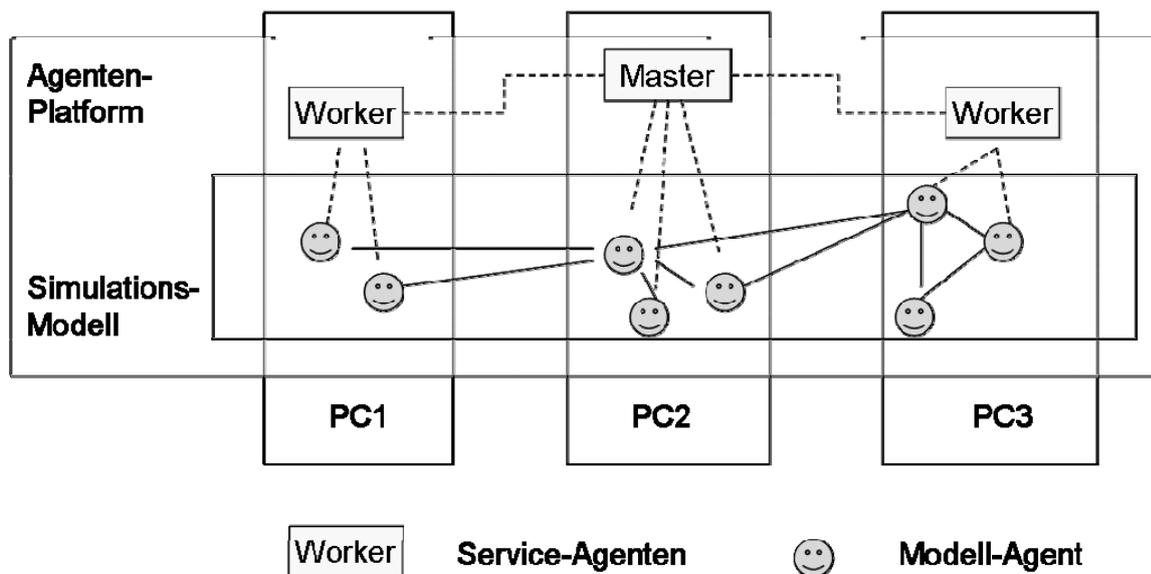


Abb. 5-1: Verteilungsszenario in *SIMJADE*

5.2 Architektur

Aufbauend auf dem JADE-Framework wurde ein zusätzlicher Infrastrukturdienst geschaffen, der die verteilte Simulation agentenbasierter Modelle unter Verwendung eines optimistischen Zeitsteuerungsansatzes unterstützt. Der prinzipielle Aufbau eines verteilten Simulationslaufes mit *SIMJADE* ist in Abb. 5-1 dargestellt. Neben den zu simulierenden Agenten die das eigentliche Simulationsmodell abbilden

³² Die *Foundation for Intelligent Physical Agents* (FIPA) repräsentiert den einzigen akzeptierten Standard im Bereich der Agententechnologie. Kernstück des Standards bildet die FIPA-Referenzarchitektur, die eine Rahmenarchitektur für den Aufbau von Agentensystemen vorgibt, sowie den Nachrichtenaustausch verbindlich regelt [FIPA2002].

existieren spezielle Service-Agenten. Ein *Master-Agent* übernimmt die Aufgabe der globalen Steuerung der Simulationsexperimente. Er liest hierfür eine entsprechende Konfigurations-Datei ein, verteilt die zu simulierenden Agenten auf die verfügbaren Rechner und startet bzw. beendet die eigentlichen Simulationsläufe. Ihm beigeordnet sind weitere so genannte *Worker-Agenten*, die jeweils auf den für die Simulation vorgesehenen Client-Rechnern gestartet werden, und sich zu Beginn automatisch beim Master-Agenten registrieren. Jeder Worker-Agent empfängt die Anweisungen des Masters und startet bzw. beendet entsprechend die auf seinem Rechner auszuführenden Modell-Agenten.

5.3 Implementierung eigener Agenten

Jeder zu simulierende Agent verfügt über eine lokale Ereignisverwaltung, die im Hintergrund die Verwaltung der lokalen Ereignislisten die automatische Zustandsicherung sowie die Freigabe von nicht mehr benötigtem Speicherplatz sicherstellt. Bei der Konzeption des Simulationssystems wurde auf eine einfache Anwendbarkeit und leichte Erweiterbarkeit geachtet, weswegen alle Details der Zeitsteuerung und Speicherverwaltung in einer speziellen Oberklasse (*SimAgent*) zusammengefasst sind. Dadurch ist ein hohes Maß an Transparenz sichergestellt. Die Implementierung eigener Agenten ist entsprechend übersichtlich gestaltet (vgl. Abb. 5-2). Der Anwendungsentwickler kann sich auf die Implementierung der eigentlichen Ereignisbehandlungsroutine konzentrieren. Bei Empfang einer Ereignisnachricht wird diese über eine entsprechende *Callback-Methode* vom Framework automatisch aufgerufen. Die Kommunikation der Agenten zur Simulationszeit erfolgt durch den Austausch von Ereignisnachrichten, die in der FIPA - Agent Communication Language (ACL) abgefasst sind. Für die ereignisbasierte Kommunikation wird jeder Nachricht zusätzlich ein entsprechender Zeitstempel mitgegeben, der den Eintrittszeitpunkt des Ereignisses beim Empfänger kennzeichnet.

```

Class SampleAgent extends SimAgent {
    //Callback-Methode - wird automatisch aufgerufen
    void handleEvent(EventMessage msg){
        //Ereignisbehandlung...
        msg.getContent();
        //Erzeugung von Folgeereignissen...
        EventMessage response = createNewEventMsg();
        //Sende Folgeereignis...
        sendEvent(response, reiceivetime);
    }
}

```

Abb. 5-2: Beispiel-Implementierung eines Agenten in SIMJADE

5.4 Speichermanagement

Ohne geeignete Maßnahmen kann der Speicherplatzbedarf optimistischer Verfahren schnell die Grenzen des zur Verfügung stehenden Hauptspeichers übersteigen. Um den Speicherbedarf für das Anlegen der Zustandskopien möglichst gering zu halten unterstützt *SIMJADE* eine Inkrementelle Zustandsicherung. Beim Anlegen einer Zustandskopie werden entsprechend nur die tatsächlich geänderten Zustandsvariablen des Agentenprozesses gespeichert und so der Speicherbedarf gering gehalten.

Zur Freigabe von nicht mehr benötigtem Speicherplatz und um festzustellen, ob der aktuelle Simulationslauf terminiert ist wird darüber hinaus periodisch der globale Zustand der Simulation ermittelt. Hierfür ist es notwendig die Globale Virtuelle Zeit (GVT) zu bestimmen. Die GVT bildet die untere Grenze innerhalb der Simulationszeit, die alle LP bereits überschritten haben und hinter die kein Prozess mehr zurückfallen kann [Fuji2000, 74]. Alle zeitlich gesehen vor der GVT liegenden Ereignisse und Zustandskopien können aus dem Speicher entfernt werden, ohne das sich dadurch das Ergebnis der Simulation ändern würde.

Zur Bestimmung der GVT wird die zu simulierende Zeitspanne in äquidistante Abschnitte (Quantum) unterteilt. Sobald ein Agentenprozess das Ende eines solchen Zeitabschnittes erreicht, meldet er seine aktuelle lokale Zeit sowie die in diesem Quantum empfangenen bzw. gesendeten Nachrichten an den Worker-Agenten. Die Worker-Agenten wiederum synchronisieren sich wechselseitig mittels einer Barriersynchronisation um im Ergebnis die globale Zeit zu berechnen [Fuji2000, 69]. Diese ergibt sich aus dem Minimum über alle lokalen Zeiten, welche die Modellagenten an die Worker-Agenten zuvor gesendet haben. Auf Grund dieser Information, die an alle Agentenprozesse weitergegeben wird, können alle nicht mehr benötigten zeitlich vor der GVT liegenden Ereignisreferenzen, Nachrichten und Zustandskopien aus dem Speicher entfernt werden. Ein ähnliches Verfahren wird u.a. in [ChSz2005] diskutiert.

Ist die Differenz über alle während des Simulationslaufes empfangenen und gesendeten Ereignisnachrichten gleich 0, d.h. kein Prozess verarbeitet mehr Ereignisse und es existieren auch keine transienten evtl. noch nicht empfangenen Nachrichten im System, kann die Simulation beendet werden [Matt1993, 426-434].

5.5 Zeitsteuerung

Wie in Kapitel 3.3 dargestellt kann es bei einer optimistischen Zeitsteuerung mitunter erforderlich sein, den Zustand eines Prozesses zurückzurollen, um potentiell falsche Berechnungen rückgängig zu machen. Das Auftreten solcher Rollbacks mindert wiederum die effektiv erreichbare Beschleunigung. Um die Häufigkeit von Rollbacks zu reduzieren unterstützt *SIMJADE* ein spezielles Verfahren zur Senkung des Optimismusgrades. Die Grundidee dieses Verfahrens beruht auf der Annahme, dass die Wahrscheinlichkeit der Fehlerfortpflanzung auf Grund von Nachzügler-Nachrichten umso höher ist, je weiter ein Agenten-Prozess in der Simulationszeit anderen Agenten voraus ist. Entsprechend werden kausal zusammengehörige Ereignisse möglichst auch zusammen ausgeführt. Da Agenten über Protokolle miteinander kommunizieren ist es relativ leicht möglich, zusammenhängende Nachrichten zu identifizieren. Die Ereignisausführung des Agenten während eines Simulationslaufes wird entsprechend verzögert, bis alle zu einem Interaktionsprotokoll gehörenden

Nachrichten vom Agenten tatsächlich empfangen wurden. Zwischenzeitlich eintreffende andere Ereignisnachrichten werden zunächst in die Ereignisliste eingeplant, aber anders als beim klassischen Time-Warp-Verfahren nicht sofort ausgeführt und zurückgestellt. Somit wird verhindert, dass sich einzelne Agenten-Prozesse zu weit in der Simulationszeit von den anderen Prozessen entfernen. Die Häufigkeit des Auftretens von Rollbacks kann, verglichen mit dem klassischen Ansatz, signifikant gesenkt werden, wie erste empirische Studien belegen. Das genaue Verfahren ist in [PaTi2006] detailliert beschrieben.

6 Simulationsstudie

Im Rahmen verschiedener Simulationsstudien wurde die Leistungsfähigkeit von *SIMJADE* überprüft. Hierfür wurde zunächst ein Modell bestehend aus 50 Agenten betrachtet, die während eines Simulationslauf 2000 Ereignisnachrichten austauschen. Die Eintrittszeitpunkte der jeweiligen Ereignisse wurden zufällig gewählt und über die Agenten gleichverteilt, so dass jeder Agent durchschnittlich 40 Ereignisnachrichten pro Lauf verarbeiten muss. Der Vernetzungsgrad (Clusteringkoeffizient) beträgt in diesem Szenario entsprechend 1, da jeder Agent mit jedem Agenten verbunden ist (vollständiger Graph). Für die Kommunikation wurden unterschiedliche Kommunikationsprotokolle (*fipa-Request*, *fipa-Query*, *fipa-ContractNet*) eingesetzt.

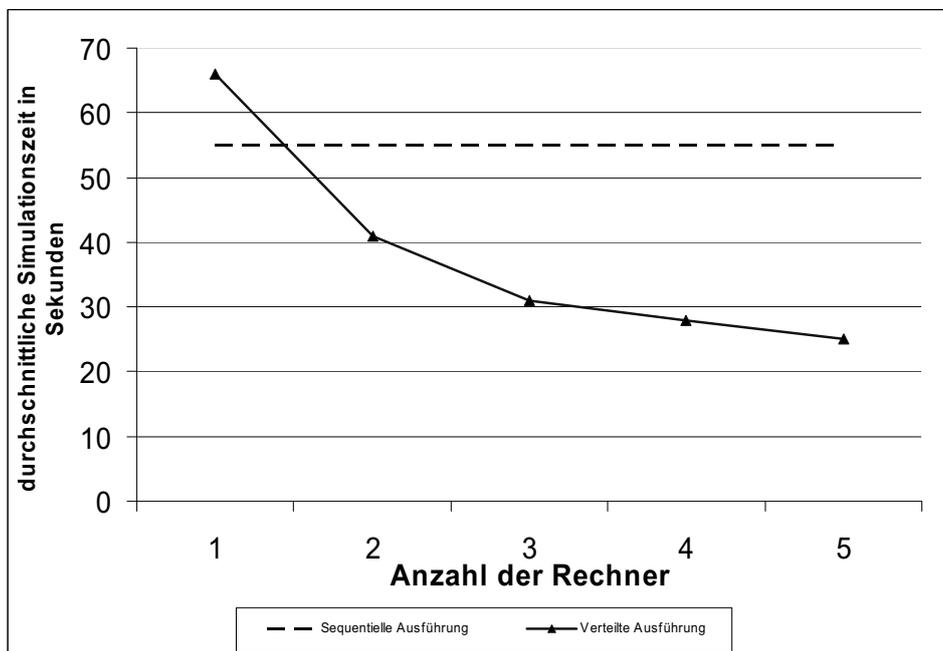


Abb. 6-1: Leistungsvergleich zwischen sequentieller und verteilter Ausführung

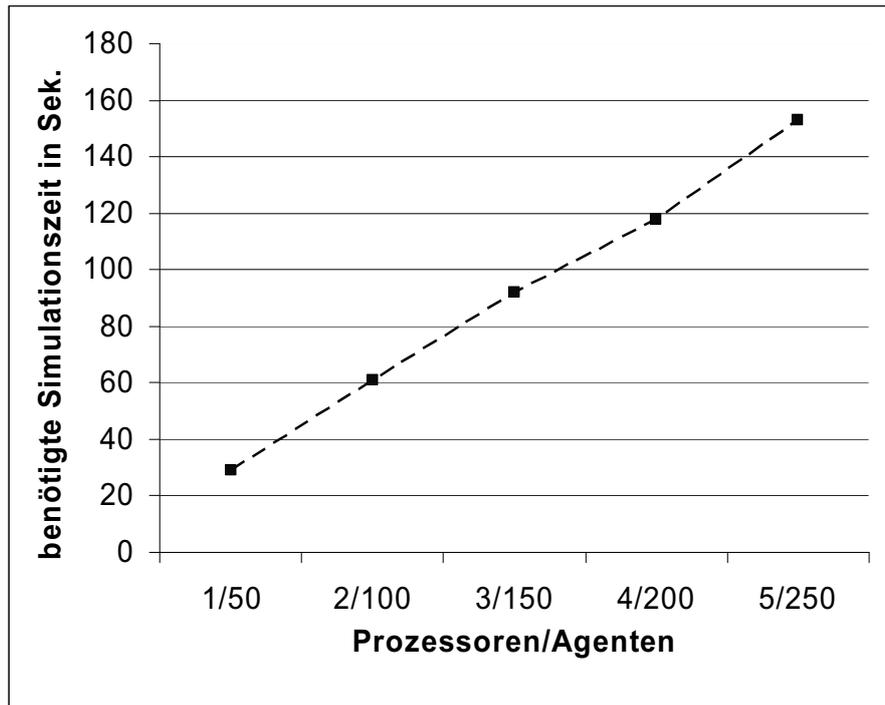


Abb. 6-2: Versuchsergebnisse Skalierbarkeitstest³³

Im Test wurden die erzielten Ausführungszeiten unter Verwendung einer sequentiellen Zeitsteuerung bzw. bei Einsatz des in *SIMJADE* realisierten optimistischen Synchronisationsansatzes ermittelt und anschließend vergleichend gegenübergestellt (siehe Abb. 6-1). Die Testergebnisse weisen für das untersuchte Modell einen Speedup-Wert von durchschnittlich 2,8 bei verteilter Ausführung auf 5 Rechnern aus im Vergleich zum sequentiellen, unverteilter Fall. Die maximal erreichbare Beschleunigung für dieses Modell liegt laut *Amdahlschen* Gesetz bei Faktor 5.

Dieser Wert stellt aber eher eine theoretische Größe dar, da er zum einen durch den Grad der Parallelität des Modells selbst sowie durch den zusätzlich im Rahmen der verteilten Zeitsteuerung entstehenden Synchronisationsaufwand begrenzt ist. Insofern kann das erreichte Ergebnis als sehr gut bezeichnet werden. In einer weiteren Testreihe wurde die Skalierbarkeit des Simulationsdienstes untersucht. Hierfür wurde, anders als in der ersten Versuchsreihe, die Größe des Modells bei wachsender Zahl der Prozessoren ebenfalls um einen konstanten Faktor erhöht. Wiederum wurde die benötigte Durchlaufzeit für einen Simulationslauf als Vergleichswert herangezogen.

Ziel dieses Tests war es, zu zeigen, dass der Simulationsdienst prinzipiell in der Lage ist auch große Modelle effizient zu simulieren. Wie die Testergebnisse deutlich machen, wächst die Ausführungszeit des Modells linear im Vergleich zur Größe des Modells bzw. der Anzahl der eingesetzten Prozessoren (siehe Abb. 6-2). Ein System, das ein lineares Zeitverhalten aufweist wird in der Simulationsliteratur gemeinhin als skalierbar bezeichnet (vgl. [Law1998, 782], [Nico1998]).

³³ Alle Testläufe wurden auf einem Cluster bestehend aus fünf 2,8 GHz P4 Workstations mit jeweils 256 MB Arbeitsspeicher und Suse Linux 8 durchgeführt. Die PCs waren über eine 100 Mbit LAN-Verbindung miteinander gekoppelt.

7 Fazit und Ausblick

Die effiziente Simulation zunehmend komplexer werdender agentenbasierter Systeme stellt erhöhte Anforderungen an die zu Grunde liegende Simulationssoftware. Wie gezeigt wurde, existieren in der Literatur bisher nur vereinzelt Aussagen über die Leistungsfähigkeit bestehender Simulationsanwendungen für Multiagentensysteme. Die Mehrzahl der derzeit verfügbaren Systeme bietet darüber hinaus keine Unterstützung zur Durchführung verteilter Simulationsexperimente. Hinzu kommt, dass die verwendeten Verfahren zur Zeitsteuerung innerhalb existierender Lösungen in der Regel nur bedingt geeignet erscheinen, die inhärente Parallelität agentenbasierter Modelle wirklich auszuschöpfen.

Der in diesem Beitrag vorgestellte Simulationsdienst (*SIMJADE*) wurde speziell für die Durchführung verteilter Simulationsexperimente mit komplexen deliberativen Agenten konzipiert und unterstützt den Test realwirtschaftlicher Agentenanwendungen. Das Hauptaugenmerk liegt dabei auf der Realisierung einer effizienten Zeitsteuerung. Wie die Ergebnisse der hier vorgestellten exemplarischen Simulationsstudie belegen, ist der gewählte Ansatz prinzipiell in der Lage zu skalieren, d.h. auch bei wachsender Modellgröße gute Simulationszeiten zu liefern.

Um eine höhere Aussagekraft bezüglich der Leistungsfähigkeit des vorgestellten Ansatzes treffen zu können sind weitere Leistungsstudien mit verschiedenen Probleminstanzen geplant, u.a. die verteilte Simulation agentenbasierter logistischer Transportnetze [Paw2006]. Darüber hinaus sind zusätzliche Leistungsverbesserungen in Bezug auf eine weitere Verkürzung der Simulationszeiten sowie Maßnahmen zur Reduzierung des Speicherbedarfs der Simulationsanwendung vorgesehen.

Literaturverzeichnis

- [AnEv1995] Anderson, J., Evans, M.: A Generic Simulation System for Intelligent Agent Designs. In: Applied Artificial Intelligence 5 (1995) 9, S. 527-562.
- [Ande2000] Anderson, J.: A Generic Distributed Simulation System for Intelligent Agent Design and Evaluation. <http://www.citeseer.ist.psu.edu/399301.html>, 2000.
- [BrPo2004] Braubach, L.; Pokahr, A. et al: A Generic Simulation Service for Distributed Multi-Agent Systems. In: Trappl, R. (eds.): Cybernetics and Systems (2004) 2, Vienna, Austria, 2004, S. 576-581.
- [Davi2000] Davidsson, P.: Multi Agent Based Simulation: Beyond Social Simulation. In: Moss, S., Davidsson, P. (eds.): 3. Workshop on Multi Agent Based Simulation (MABS) 2000, LNAI 1997, 2000, S. 97-107.
- [DaHe2004] Davidsson, P., Henesey, L. et al: Agent-Based Approaches to Transport Logistics. AAMAS Workshop on Agents in Traffic and Transportation, New York City, 2004.
- [Fuji2000] Fujimoto, R.M.: Parallel and Distributed Simulation Systems. John Wiley & Sons Inc., 2000.
- [FIPA2002] Foundation For Intelligent Physical Agents (FIPA): Interaction Protocol Specification Document no. SC00026H-SC00036H, <http://www.fipa.org/specs/>, 2002.

- [GaKa2004] Gasser, L., Kakugawa, K. et al: Smooth Scaling Ahead: Progressive MAS Simulation from Single PCs to Grids. In: Proceedings of the Joint Workshop on Multi-Agent & Multi-Agent-Based Simulation, Autonomous Agents & Multiagent Systems (AAMAS) New York, 2004, S 1-10.
- [ChSz2005] Chen, G., Szymanski, B. K.: DSIM: Scaling Time Warp to 1,033 Processors. In: Proceedings of the WinterSim Conference 2005.
- [GoSa2001] Goic, J., Sauter, J. A. et al: SYNCER: Distributed simulations using swarm. In: SwarmFest 2001, Santa Fe, NM 2001.
- [GPSa2005] Gentile, M., Paolucci, M., Sacile, R.: Agent-Based Simulation. In: Paolucci, M., Sacile, R.: Agent-Based Manufacturing and Control Systems: new agile manufacturing solution for achieving peak performance. CRC Press LCC, 2005, S. 119-149.
- [JADE2006] JADE Framework <http://sharon.csel.it/projects/jade/>, 2006.
- [JeBe1987] Jefferson, D. R., Beckman, B. et al.: The Time Warp operating systems. In: 11th Symposium on Operating Systems Principles. 21: 1987, S. 77-93.
- [KIHe2003] Klügl, F., Herrler, R. et al: From Simulated to Real Environments: How to use SeSAM for software development In: M. Schillo et al. (eds) Multiagent System Technologies - 1st German Conferences MATES, (LNAI 2831), 2003, S. 13-24.
- [Klüg2001] Klügl, F.: Multiagentensimulation – Konzepte, Werkzeuge, Anwendung. Addison-Wesley, 2001.
- [KnTi1999] Knirsch, P., Timm, I.J.: Adaptive Multiagent Systems Applied on Temporal Logistics Network. In: Muffatto, M., Pawar, K.S. (Hrsg.): Logistics in the Information Age. In: Proceedings of the 4th International Symposium on Logistics (ISL-99), Florence, Padova, 1999.
- [Law1998] Law, D.R.: Scalable means more than more: a unifying definition of simulation scalability. In: Proceedings of the 30th conference on winter simulation table of contents Washington, D.C., United States, S. 781 – 788, 1998.
- [LeLo2004] Lees, M., Logan, B.: Simulating Agent-Based Systems with HLA: The Case of SIM_AGENT - Part II (03E-SIW-076) 2004.
- [Loga2005] Logan, B.: The Simulation of Agent Systems. In: AgentLink News. European Coordination Action for Agent-based Computing, ISSUE 17, April 2005, S. 5-8.
- [Matt1993] Mattern, F.: Efficient algorithms for distributed snapshots and global virtual time approximation. In: Journal of Parallel and Distributed Computing 4 (1993) 18, S. 423-434.
- [MiTh2004] Minson, R., Theodoropoulos, G.: Distributing repast agent based simulations with HLA. In: Proceedings of the 2004 European Simulation Interoperability Workshop, Edinburgh, 2004.
- [Mehl1994] Mehl, H.: Methoden der verteilten Simulation. Vieweg, 1994.
- [Nico1998] Nicol, D. M.: Scalability, Locality, Partitioning and Synchronization in PDES. 1998.
- [PaTi2006] Pawlaszczyk, D.: Timm, I.J.: A Hybrid Time Management Approach to Agent-Based Simulation. In: 29th Annual German Conference On Artificial Intelligence KI2006, 2006 (im Erscheinen).

- [Pawl2006] Pawlaszczyk, D.: Scalable Multi Agent Based Simulation - Considering Efficient Simulation of Transport Logistic Networks. In: S. Wenzel (Hrsg.): Tagungsband zur 12. Fachtagung Simulation in Produktion und Logistik (ASIM 2006), SCS Publishing House, 2006, S. 479-488.
- [Rile2003] Riley, P.: MPADES: Middleware for Parallel Agent Discrete Event Simulation. In: Kaminka, G.A., Lima, P.U. and Rojas, R. (eds.): RoboCup 2002, LNAI Vol. 2752. Springer, 2003, S. 162-178.
- [Schu2002] Schulz, R.: Parallele und Verteilte Simulation bei der Steuerung komplexer Produktionssysteme. Dissertation, TU-Ilmenau, 2002.
- [ThLo1999] Theodoropoulos, G., Logan, B.: A framework for the distributed simulation of agent-based systems. In: Proceedings of the 13th SCS European Simulation Multiconference, 1999, S. 58-65.
- [ToHo2004] Tobias, R., Hofmann, C.: Evaluation of free Java-libraries for social-scientific agent based simulation. In: Journal of Artificial Societies and Social Simulation 1 (2004) 7.
- [UhSc1998] Uhrmacher, A.M., Schattenberg, B.: Agents in Discrete Event Simulation. In: Bargiela, A., Kerckhoffs, E. (eds.): Proceedings of the 10TH ESS'98, SCS Publications Ghent, 1998, S.129-136.
- [VDI2003] VDI, Verein Deutscher Ingenieure: VDI-Richtlinie 3633. August 2003.
- [WaTu2005] Wang, F.; Turner, S.J.; Wang, L.: Agent Communication in Distributed Simulations. In: Davidsson, P. et al (eds.): Workshop on Multiagent Based Simulation (MABS) 2004, LNAI Vol.3415. Springer, 2005, S 11-24.
- [WoJe1995] Wooldridge, M.; Jennings, J.R.: Agent Theories, Architectures, and Languages: a Survey. In: Wooldridge, M.; Jennings, J.R. (Hrsg.): Intelligent Agents. Springer, 1995, S. 1-32.

