

**Beitrittsentscheidungen zu
Multiagenten-Organisationen:
Ein Revenue Management-basierter Ansatz**

Marc Premm



DISSERTATION

zur Erlangung des Grades
Doktor der Wirtschaftswissenschaften (Dr. oec.)

vorgelegt der
Fakultät für Wirtschafts- und Sozialwissenschaften
der Universität Hohenheim

2018

Marc Premm

Beitrittsentscheidungen zu Multiagenten-Organisationen:
Ein Revenue Management-basierter Ansatz

Datum der mündlichen Prüfung:	6. Juni 2018
Erstgutachter:	Prof. Dr. Stefan Kirn
Zweitgutachter:	Prof. Dr. Christian Ernst
Vorsitzender der Prüfungskommission:	Prof. Dr. Klaus Prettnner
Dekan:	Prof. Dr. Karsten Hadwich

Inhaltsübersicht

Abbildungsverzeichnis	viii
Tabellenverzeichnis	ix
Abkürzungsverzeichnis	xi
Symbolverzeichnis	xiii
1 Einleitung	1
2 Analyse des Stands der Forschung	11
3 Verfahrensentwurf	89
4 Evaluation	129
5 Zusammenfassung und Ausblick	153
Literaturverzeichnis	157

Inhaltsverzeichnis

Abbildungsverzeichnis	viii
Tabellenverzeichnis	ix
Abkürzungsverzeichnis	xi
Symbolverzeichnis	xiii
1 Einleitung	1
1.1 Problemstellung	1
1.2 Forschungsansatz	5
1.3 Epistemologische Einordnung	6
1.4 Aufbau der Arbeit	8
2 Analyse des Stands der Forschung	11
2.1 Softwareagenten	14
2.1.1 Definitionen in der Literatur	14
2.1.1.1 Autonomie und Lernfähigkeit	15
2.1.1.2 Soziale Fähigkeiten	18
2.1.1.3 Reaktivität und Umgebung	19
2.1.1.4 Proaktivität und Zielorientierung	21
2.1.2 Der Begriff des maschinellen Aufgabenträgers	22
2.1.2.1 Kompetenz	22
2.1.2.2 Verantwortung und Verantwortlichkeit	23
2.1.2.3 Stelle	23
2.1.2.4 Aufgabe und Aufgabenträger	24
2.1.2.5 Maschineller Aufgabenträger	25
2.1.3 Softwareagenten als maschinelle Aufgabenträger	29
2.2 Organisationsmodelle der verteilten künstlichen Intelligenz	33
2.2.1 Grundmengen von Softwareagenten	33
2.2.1.1 Bekanntschaften	33
2.2.1.2 Multiagentengesellschaft	34
2.2.2 Verteilte Problemlöser	35
2.2.2.1 Blackboard-Systeme	39
2.2.2.2 Nicht-kooperative verteilte Problemlöser	40
2.2.2.3 Kooperative verteilte Problemlöser	43

2.2.3	Multiagentensysteme	47
2.3	Organisationstheoretische Ansätze für Multiagenten-Organisationen	53
2.3.1	Der Organisationsbegriff	53
2.3.1.1	Der Organisationsbegriff der Betriebswirtschaftslehre	53
2.3.1.2	Multiagenten-Organisationen	58
2.3.2	Mitgliedschaft	62
2.3.2.1	Mitgliedschaft in der Organisationstheorie	64
2.3.2.1.1	Bürokratie-Ansatz von Max Weber	64
2.3.2.1.2	Anreiz-Beitrags-Theorie	65
2.3.2.1.3	Entscheidungslogisch-mathematische Ansätze	66
2.3.2.1.4	Neue Institutionenökonomik	69
2.3.2.1.5	Ressourcenbasierter Ansatz	74
2.3.2.1.6	Fazit	76
2.3.2.2	Mitgliedschaft in Multiagenten-Organisationen	78
2.4	Beitrittsentscheidung	80
2.4.1	Beitrittsentscheidung aus organisationstheoretischer Sicht	80
2.4.2	Beitritt zu einer Multiagenten-Organisation durch Vertrag	84
2.4.3	Vertragliche Nebenbedingungen	85
2.4.4	Ressourcenallokation und Mehrfachmitgliedschaft	87
2.4.5	Fazit und Forschungslücke	88
3	Verfahrensentwurf	89
3.1	Anwendbarkeit von Revenue Management auf Beitrittsentscheidungen von Softwareagenten	90
3.1.1	Revenue Management und dessen Anwendungen	91
3.1.2	Voraussetzungen zur Anwendung von Revenue Management-basierten Ansätzen	93
3.1.2.1	Heterogenität der Kunden	93
3.1.2.2	Nachfrageschwankungen und Unsicherheiten	94
3.1.2.3	Inflexibilität der Produktion	95
3.1.2.4	Preis ist kein Qualitätsmerkmal	96
3.1.2.5	Informationstechnische Infrastruktur	97
3.1.3	Voraussetzungen zur Anwendung erweiterter Verfahren des Revenue Management	98
3.1.3.1	Flexibilität der Ressourcennutzung	98
3.1.3.2	Abhängigkeiten zwischen angebotenen Diensten	99
3.1.3.3	Überbuchen von Ressourcen	100
3.1.4	Zusammenfassung zur Anwendbarkeit von Revenue Management	101
3.2	Revenue Management-basiertes Modell der Beitrittsentscheidung	103
3.2.1	Problemklassifizierung	103
3.2.2	Zeithorizont	110
3.2.3	Dienste und Ressourcen	111
3.2.4	Anfragen von Multiagenten-Organisationen	112

3.2.5	Dienstbereitstellungsangebote	113
3.2.6	Mitgliedschaftsverträge	114
3.3	Prognosemodell	115
3.4	Kapazitätssteuerung bei Beitrittsentscheidungen	117
3.4.1	Entscheidungsverfahren	118
3.4.2	Bid-Price Verfahren	119
3.4.3	Bid-Price Berechnung	121
3.5	Überbuchen bei Beitrittsentscheidungen von Softwareagenten	124
4	Evaluation	129
4.1	Evaluationsmethode	129
4.2	Simulationsaufbau	130
4.3	Simulationsergebnisse	135
4.3.1	Verhältnis der nachfragenden Multiagenten-Organisationen	135
4.3.2	Nachfragehäufigkeit und Mitgliedschaftsdauer	140
4.3.3	Ressourcenverfügbarkeit und Nachfragemenge	142
4.3.4	Kompensationen	142
4.3.5	Dienstgüteklassen und Dienste	145
4.3.6	Dienstanbieterstruktur	148
4.4	Diskussion	150
5	Zusammenfassung und Ausblick	153
5.1	Ergebnisse	153
5.2	Kritische Würdigung und weiterer Forschungsbedarf	155
	Literaturverzeichnis	157

Abbildungsverzeichnis

1.1	Austauschbeziehungen zwischen Softwareagenten und Multiagenten-Organisationen	4
2.1	Personelle und maschinelle Aufgabenträger als ER-Diagramm	28
2.2	Problemzerlegung bei nicht-kooperativen verteilten Problemlösern	41
2.3	Abstimmungsbeispiel für kooperative verteilte Problemlöser	44
2.4	ER-Modell der Beziehung zwischen Organisation und deren Mitgliedern . .	77
2.5	Entscheidungsmodell in Organisationen	82
3.1	Revenue Management Ansätze	90
3.2	Vierdimensionale Darstellung der Produkt- bzw. Dienstnachfrage	93
3.3	Beispiel zum Verlauf der freien, vergebenen und genutzten Ressourcen bei Mitgliedschaften in mehreren Multiagenten-Organisationen	126
4.1	Sequenzdiagramm des Anfrage-/Angebotsablaufs im Simulationsmodell . .	131
4.2	Steigerung der erzielten Kompensationen unter Veränderung der Anzahl nachfragender Multiagenten-Organisationen vom Typ 1 und 2	136
4.3	Erzielte durchschnittliche Kompensation abhängig von der Anzahl von Multiagenten-Organisationen vom Typ 2 bei einer vom Typ 1	137
4.4	Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der Anzahl nachfragender Multiagenten-Organisationen vom Typ 1 und 2	138
4.5	Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der Anzahl nachfragender Multiagenten-Organisationen vom Typ 1 und 2 (Fortsetzung von Abbildung 4.4)	139
4.6	Steigerung der erzielten Kompensationen unter Veränderung der durchschnittlichen Mitgliedschaftsdauer und der Nachfragewahrscheinlichkeit je Zeiteinheit	140
4.7	Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der durchschnittlichen Mitgliedschaftsdauer und der Nachfragewahrscheinlichkeit je Zeiteinheit	141
4.8	Steigerung der erzielten Kompensationen unter Veränderung der durchschnittlichen Dienstmenge pro Anfrage und der Ressourcenkapazität	143

4.9	Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der durchschnittlichen Dienstmenge pro Anfrage und der Ressourcenkapazität	144
4.10	Steigerung der erzielten Kompensationen unter Veränderung der Kompensation für Dienst 1 sowie des Rabatts für günstigere Dienstgüteklassen . . .	145
4.11	Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der Kompensation für Dienst 1 sowie des Rabatts für günstigere Dienstgüteklassen	146
4.12	Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der Anzahl der angebotenen Dienste und Dienstgüteklassen .	147
4.13	Steigerung der erzielten Kompensationen unter Veränderung der Anzahl nachfragender Multiagenten-Organisationen und anbietender Softwareagenten vom Typ <i>SimpleAgent</i>	148
4.14	Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der Anzahl der Softwareagenten vom Typ <i>SimpleAgent</i> sowie der Anzahl an Multiagenten-Organisationen vom Typ 1 und 2	149

Tabellenverzeichnis

2.1	Bürokratie-Ansatz	65
2.2	Anreiz-Beitrags-Theorie	66
2.3	Effizienz-Konzept	68
2.4	Delegationswert-Konzept	69
2.5	Übergreifende Annahmen der Neuen Institutionenökonomik	71
2.6	Transaktionskostentheorie	72
2.7	Verfügungsrechtetheorie	73
2.8	Prinzipal-Agenten-Theorie	74
2.9	Ressourcenbasierter Ansatz	76
3.1	Erfüllung der Voraussetzungen zur Anwendung Revenue Management- basierter Ansätze	102
3.2	Erfüllung der Voraussetzungen zur Anwendung erweiterter Verfahren des Revenue Management	102
3.3	Gegenüberstellung von Revenue Management Domänen mittels Taxonomie	109
4.1	Simulationsparameter	134
4.2	Korrelationsanalyse zum Verhältnis der nachfragenden Multiagenten-Organi- sationen	137
4.3	Korrelationsanalyse zur Nachfragewahrscheinlichkeit und Mitgliedschaftsdauer	142
4.4	Korrelationsanalyse zur Ressourcenverfügbarkeit und Nachfragemenge . . .	143
4.5	Korrelationsanalyse zur Kompensationsgestaltung	147
4.6	Korrelationsanalyse zu Diensten und Dienstgüteklassen	147
4.7	Korrelationsanalyse zur Dienstanbieterstruktur	150

Abkürzungsverzeichnis

ACL	Agent Communication Language
EMSR	Expected Marginal Seat Revenue
GE	Geldeinheiten
KI	künstliche Intelligenz
MAS	Multiagentensystem
PEMSR	Prorated Expected Marginal Seat Revenue
RM	Revenue Management
VKI	verteilte künstliche Intelligenz

Symbolverzeichnis

α_t	Gewichtungsvektor zum Zeitpunkt t
β_t^r	Korrekturfaktor für Ressource r zum Zeitpunkt t
b	Dienstgüteklasse
C_t	Vertragsabschlüsse zum Zeitpunkt t
c_b^t	Spaltenvektor b von C_t
$\bar{d}_t$	Beitrittsentscheidung des Softwareagenten zum Zeitpunkt t
d_t	Beidseitige Mitgliedschaftsentscheidung zum Zeitpunkt t
$D_{s,b}^t$	Erwartete Nachfrage für Dienst s in Dienstgüteklasse b zum Zeitpunkt t
$D_t^r(t, \hat{t})$	Erwartete Nachfrage für Ressource r zum Zeitpunkt t für den Zeitabschnitt von t bis $\hat{t}$
$E_t(x_r, t + \delta_t^r)$	Erwartete Restkapazität für Ressource r zum Zeitpunkt t für den Zeitpunkt δ_t
λ^t	Durchschnittliche Mitgliedschaftsdauer zum Zeitpunkt t
M_b	Ressourcennutzung in Dienstgüteklasse b
p_b	Preise für Dienste in Dienstgüteklasse b
r_t	Anfrage zum Zeitpunkt t
t	Zeitpunkt
$V_t(\mathbf{x})$	Wertschätzung für die Ressourcen $\mathbf{x}$ zum Zeitpunkt t
x_t	Verfügbare Ressourcen zum Zeitpunkt t

1. Einleitung

1.1 Problemstellung

Der Forschungsbereich Multiagentensysteme (MAS) hat sich seit den späten 1970er Jahren als Teilbereich der verteilten künstlichen Intelligenz (VKI) etabliert und bietet – insbesondere aufgrund der multidisziplinären Ausrichtung – Anwendungen in zahlreichen Domänen (Weiss, 2013, S. xxxv f.). Gegenstand dieses Forschungsbereichs sowie dieser Arbeit sind Softwareagenten, die als Softwaresysteme zielorientiert agieren und mittels Lernverfahren eine gewisse Autonomie gegenüber ihrem Entwickler erlangen. Dabei handeln Softwareagenten im Auftrag und auf Rechnung einer natürlichen oder juristischen Person (Müller-Hengstenberg und Kirn, 2016, S. 59 f.). Softwareagenten ist es möglich, sich zu Multiagentensystemen zusammenzuschließen, falls sie sich aus der daraus ergebenden Zusammenarbeit einen Vorteil versprechen. Multiagentensysteme sind folglich offene Systeme, aus denen Softwareagenten ein- und wieder austreten können – im Allgemeinen ohne globale Kontrolle (Bond und Gasser, 1988, S. 3). Hieraus ergibt sich eine gewisse Flüchtigkeit sowohl der Mitgliederzahl als auch der Interaktionsstrukturen eines Multiagentensystems und somit eine eingeschränkte Möglichkeit zur Steuerung des nach außen hin wahrnehmbaren Systemverhaltens, welcher es beim Einsatz von Softwareagenten in kommerziellen Anwendungen im Allgemeinen bedarf (Kirn, 1996a, S. 11, 64). Multiagenten-Organisationen bilden einen Ansatz, um die betriebswirtschaftliche Organisationstheorie zur internen Strukturierung von Multiagentensystemen zu nutzen und folglich auch deren Außenverhalten zielgerichtet zu steuern (Kirn, 1996a, S. 11). Diese Arbeit versteht Multiagenten-Organisationen als auf Dauerhaftigkeit ausgelegte Zusammenschlüsse von mehreren unabhängigen Softwareagenten, die durch vertragliche Regelungen an der Erfüllung eines vorgegebenen Organisationsziels mitwirken. Sie bilden somit die Grundlage, um Softwareagenten in kommerzielle Anwendungen einzubinden.

Softwareagenten benötigen Zugriff auf Ressourcen, um einerseits bestimmte Dienste anbieten (z.B. Datenbank-Zugriff) aber auch um ihre eigene Ausführung sicherstellen zu können (z.B. Hardwareressourcen). Softwareagenten können die ihnen zur Verfügung stehenden Ressourcen nutzen, um einer Multiagenten-Organisation im Rahmen einer Mitgliedschaft

Dienste zur Verfügung zu stellen. Die Erbringung von Diensten konsumiert stets einen Teil der einem Softwareagenten zur Verfügung stehenden Ressourcen. Softwareagenten werden daher einer Multiagenten-Organisation Dienste nur gegen eine entsprechende Kompensation zur Verfügung stellen. Die Beitrittsentscheidung zu einer Multiagenten-Organisation wird für den Softwareagenten somit insbesondere von der Verfügbarkeit von notwendigen Ressourcen sowie der angebotenen Kompensation beeinflusst. Dabei ergibt sich das Problem der Maximierung des Nutzens jedes einzelnen Softwareagenten bei der Beitrittsentscheidung zu einer existierenden Multiagenten-Organisation. Das Konzept des *Nutzens* wird dabei den Sozialwissenschaften entlehnt, um den Grad der Zielerreichung von Softwareagenten zu quantifizieren. Zwar kann die Kompensation grundsätzlich auch durch andere als monetäre Zahlungsströme erfolgen (beispielsweise eine gegenläufig gerichtete Dienstbereitstellung der Multiagenten-Organisation an den Softwareagenten), jedoch können diese nicht-monetären Leistungen auch in monetärer Form bewertet werden. Daher wird in dieser Arbeit angenommen, dass die Kompensation stets in monetärer Form erfolgt.

Es obliegt jedem einzelnen Softwareagenten über einen Beitritt zu einer Multiagenten-Organisation zu entscheiden. Dabei handelt es sich jedoch nicht um eine einseitige Entscheidung, sondern die Gegenseite – die Multiagenten-Organisation – entscheidet ebenfalls, ob der Beitritt eines Softwareagenten zu deren Zielerfüllung dienlich ist. Der Beitritt eines Softwareagenten zu einer Multiagenten-Organisation ist somit stets das Ergebnis von Beitrittsverhandlungen zwischen diesen beiden Akteuren. In dieser Arbeit wird angenommen, dass eine bestehende Multiagenten-Organisation bei Bedarf diese Beitrittsverhandlungen initiiert und entsprechende Anfragen an potentiell verfügbare Softwareagenten versendet. Hierzu ist es notwendig, dass der Multiagenten-Organisation durch direkten Kontakt mit dem jeweiligen Softwareagenten oder mittels einer zentralen Dienste-Registrierungsstelle Informationen über potentiell verfügbare Softwareagenten zur Verfügung stehen. Erhält ein Softwareagent eine Anfrage zur Bereitstellung eines Dienstes für eine Multiagenten-Organisation, besteht die Problemstellung für den Softwareagenten in der Bewertung der Anfrage, um hierauf aufbauend eine Beitrittsentscheidung treffen sowie seine hierfür gültigen Bedingungen kommunizieren zu können. Stimmt die Multiagenten-Organisation dem Angebot zu, wird der Softwareagent Mitglied der Multiagenten-Organisation. Da Softwareagenten ihre Dienste verschiedenen Multiagenten-Organisationen anbieten und auch mit diesen Verträge eingehen können, können sie Mitglied in mehreren Multiagenten-Organisationen sein. Dabei können sie stets den gleichen oder auch unterschiedliche Dienste anbieten und erbringen, die dieselben oder verschiedene dem Softwareagenten zur Verfügung stehenden Ressourcen nutzen. Jeder Softwareagent hat hierbei zu entscheiden, ob bzw. welcher Multiagenten-Organisation er beitrifft und in welchem Umfang er die verfügbaren Ressourcen hierfür einsetzt. Dabei hat er bestehende Mitgliedschaften in anderen Multiagenten-Organisationen, für die der Softwareagent bereits Dienste bereitstellt und somit Ressourcen auslastet, bei der Beitrittsentscheidung mit zu berücksichtigen. Das folgende Beispiel in Verbindung mit Abbildung 1.1 veranschaulicht die sich hieraus ergebende Beitrittsentscheidung.

Beispiel: Gegeben sei ein Softwareagent a_1 , der über einen exklusiven Zugriff auf eine beschränkte Menge an Ressourcen verfügt. Durch die ihm zur Verfügung stehenden Ressourcen ist a_1 in der Lage, dauerhaft entweder den Dienst s_1 oder den Dienst s_2 zu erbringen. Durch Marktbeobachtungen ist dem Softwareagenten bekannt, dass er pro Zeiteinheit für den Dienst s_1 bzw. s_2 die Kompensation p_1 bzw. p_2 mit $p_1 > p_2$ fordern kann. Seine Zielfunktion sei definiert als Maximierung der erzielten Kompensation, die ausschließlich durch Mitgliedschaften in Multiagenten-Organisationen erzielt werden kann. Eine einseitige Auflösung der Mitgliedschaftsbeziehung durch den Softwareagenten ist für diesen mit hohen Kosten verbunden.

Eine Multiagenten-Organisation sendet eine Anfrage an a_1 zur Bereitstellung des Dienstes s_2 im Rahmen einer Mitgliedschaft. Der Softwareagent hat in diesem Fall über den Beitritt zu dieser Multiagenten-Organisation zu entscheiden. Dabei bindet er sich sowie die ihm zur Verfügung stehenden Ressourcen dauerhaft und ist durch die beschränkte Menge an Ressourcen nicht mehr in der Lage, andere Mitgliedschaften einzugehen ohne zuvor erstere aufzulösen. Entscheidet sich a_1 für einen Beitritt mit der Kompensation p_2 ist es ihm nicht möglich, später die höhere Kompensation p_1 zu erwirtschaften. Lehnt er die Anfrage hingegen ab, bleiben seine Ressourcen mindestens so lange ungenutzt bis eine weitere Anfrage eintrifft.

Diese Arbeit geht davon aus, dass einer Multiagenten-Organisation die Existenz von Softwareagenten, die nicht in dieser Mitglied, aber potentiell in der Lage sind, benötigte Dienste zu erbringen, bekannt und eine Kommunikation mit diesen möglich ist. Es ist der Multiagenten-Organisation somit auch möglich, diese Softwareagenten bezüglich der Erstellung eines Angebots für eine Dienstleistung anzufragen. Softwareagenten, die eine solche Anfrage erhalten, stehen vor dem Entscheidungsproblem, ob sie ein solches Angebot erstellen und wie dieses gegebenenfalls auszugestalten ist. Da auch die Entscheidung der anfragenden Multiagenten-Organisation über ein gegebenenfalls abgegebenes Angebot entscheidend für den endgültigen Beitritt ist, ist es im Interesse des Softwareagenten, dieses Angebot so zu gestalten, dass es zum einen den eigenen Nutzen maximiert und gleichzeitig mit einer entsprechend hohen Wahrscheinlichkeit angenommen wird. Da die Präferenzen der Multiagenten-Organisation dem Softwareagenten zunächst nicht bekannt sind, hat er ebenfalls die Möglichkeit, mehrere unterschiedliche Angebote gleichzeitig abzugeben, um die Wahrscheinlichkeit zu erhöhen, dass eines dieser Angebote angenommen wird. Verschiedene bindende Angebotsalternativen zu erstellen, aus denen der Dienstabnehmer auswählen kann, ist auch in der betrieblichen Praxis ein gängiges Verfahren. Dabei ist jedes der Angebote bindend. Durch entsprechende Formulierung der einzelnen Angebote wird jedoch verhindert, dass mehrere Angebote angenommen werden können, welche unter Umständen nicht bedient werden könnten.

Hieraus lässt sich folgende Forschungsfrage ableiten: Wie sind Entscheidungsverfahren auszugestalten, die es Softwareagenten ermöglichen, nutzenmaximierende Beitrittsentscheidungen zu Multiagenten-Organisationen zu treffen?

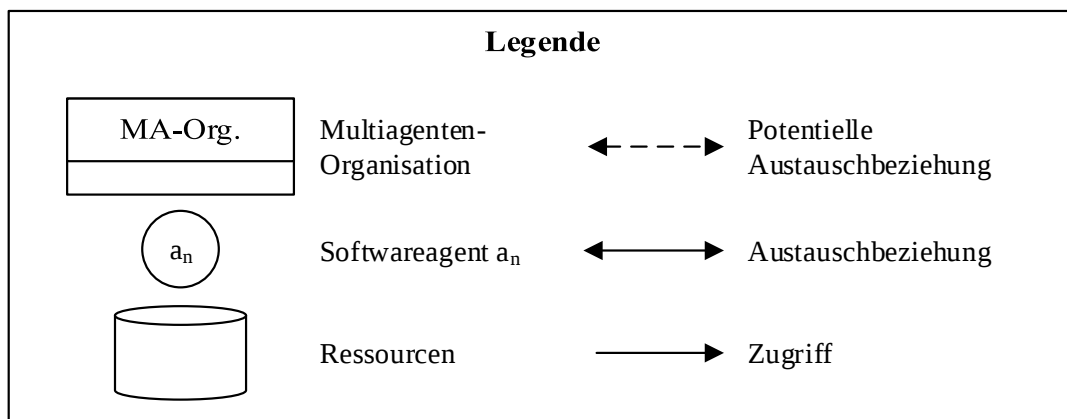
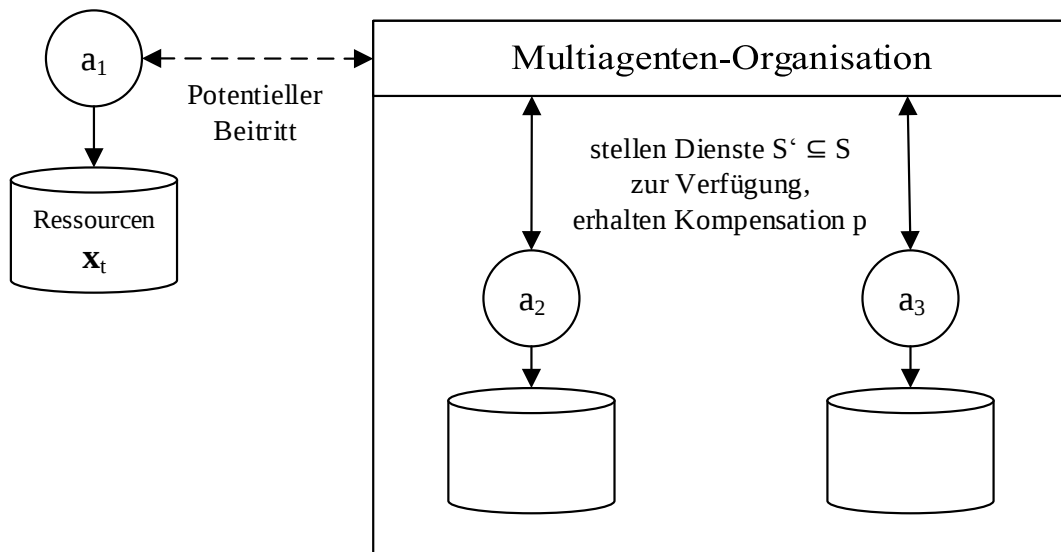


Abbildung 1.1: Austauschbeziehungen zwischen Softwareagenten und Multiagenten-Organisationen

1.2 Forschungsansatz

Als Perspektive werden Verfahren aus dem Revenue Management als Teilbereich des Operations Research verwendet. Ziel ist es, die dem Softwareagenten zur Verfügung stehenden Ressourcen möglichst nutzenstiftend einzusetzen. Hierzu zählt unter anderem die Verwendung von Preisdiskriminierungsansätzen, die es Softwareagenten ermöglichen, den Adressatenkreis der angebotenen Dienste im Hinblick auf potentielle Multiagenten-Organisationen zu erweitern. Im Fokus jeder Beitrittsentscheidung von Softwareagenten steht dabei das Entscheidungsproblem, das mit Hilfe der Entscheidungstheorie formalisiert wird. Wie bereits oben aufgezeigt, hat ein Softwareagent beim Erhalt einer Anfrage zur Dienstbereitstellung die Möglichkeit, mehr als ein Angebot abzugeben. Der Hauptbestandteil des Angebots ist dabei stets identisch: die Bereitstellung des entsprechenden Dienstes. Der Softwareagent kann hingegen die vertraglichen Nebenbedingungen variieren, wie beispielsweise Stornierungskosten oder Dienstgütevereinbarungen (engl. *service level agreements*). Unterschiedliche Kombinationen dieser vertraglichen Nebenbedingungen werden dem Softwareagenten auch unterschiedlichen erwarteten Nutzen stiften. Beispielsweise kann eine Dienstgütevereinbarung die Antwortzeit des Softwareagenten auf eine Dienstanforderung vorsehen: Kürzere vorgeschriebene Antwortzeiten limitieren den Handlungsspielraum des Softwareagenten, da er seine Ressourcen zur Erfüllung dieses Dienstes einsetzen muss, während entsprechend längere Antwortzeiten seine Flexibilität zur Ressourcennutzung erhöhen. Zwar kann der Softwareagent auch ausschließlich ein Angebot unterbreiten, das seine eigene Flexibilität maximiert, indem er keine maximale Antwortzeit fixiert und ganz auf Dienstgütevereinbarungen verzichtet, jedoch senkt dies die Wahrscheinlichkeit, dass die Multiagenten-Organisation dieses Angebot annehmen wird. Das Entscheidungsproblem des Softwareagenten besteht folglich aus der Entscheidung über die anzubietenden Dienste einschließlich der vertraglichen Nebenbedingungen. Dieses Entscheidungsproblem und insbesondere vertragliche Nebenbedingungen adressieren Ansätze des Revenue Management (Talluri und van Ryzin, 2004, S. 1 ff.).

Während die in der Literatur vorhandenen Ansätze des Revenue Management im Allgemeinen auf kurzfristige Vertragsabschlüsse über die Erbringung von Diensten über einen festen Zeitraum ausgelegt sind (Botimer und Belobaba, 1999, S. 1085 ff.), werden für Beitrittsentscheidungen zu Multiagenten-Organisationen hingegen Kriterien benötigt, die die a priori unbekannte Mitgliedschaftsdauer und die Dauerhaftigkeit der Multiagenten-Organisation berücksichtigen. In den klassischen Revenue Management Disziplinen wie der Luftverkehrsindustrie werden beispielsweise einmalige Beförderungsverträge von einer Destination zur nächsten betrachtet. Eine dauerhafte Bindung zwischen Dienstnachfrager und Dienstbringer bildet sich in diesem Kontext nicht. Die bestehenden Verfahren des Revenue Management müssen daher auf die besonderen Bedürfnisse einer a priori unbekannten Bindungsdauer von Softwareagenten an eine Multiagenten-Organisation ausgerichtet werden. Dies erfordert die Anpassung der Modelle an langfristige vertragliche Bindungen, deren Dauer im Allgemeinen a priori nicht bekannt ist. Durch das angepasste Modell erfordern auch die Entscheidungsverfahren des Revenue Management selbst Anpassungen an diese abweichenden Gegebenheiten. Das entwickelte Verfahren ermöglicht

es Softwareagenten, eine erwartungswertmaximierende Entscheidung über den Beitritt zu Multiagenten-Organisationen zu treffen.

Die Überprüfung des entwickelten Verfahrens erfolgt mittels einer quantitativen Erkenntnismethode. Im Rahmen einer Simulationsstudie wird das Verfahren auf seine Wirksamkeit untersucht. Hierzu werden domänenunabhängige synthetische Daten verwendet, die die Wirksamkeit und Wirkrichtung von unabhängigen auf abhängige Variablen des Verfahrens herausstellen.

1.3 Epistemologische Einordnung

Die Forschungsrichtung Wirtschaftsinformatik lässt sich in zwei Hauptforschungsrichtungen unterteilen: Die verhaltensorientierte Wirtschaftsinformatik hat die Erklärung von beobachtbaren Phänomenen und deren Prognose mittels behavioristischer Forschung zum Ziel. Dieses Ziel wird auch als theoretisches Wissenschaftsziel bezeichnet (Hill et al., 1994, S. 22). Die konstruktionsorientierte Forschung der gestaltungsorientierten Wirtschaftsinformatik hingegen folgt dem pragmatischen Wissenschaftsziel, welches insbesondere aus der Nutzbarmachung von Erkenntnissen für Umsetzungen in der Realwelt besteht. Der Erkenntnisgegenstand selbst, über den Erkenntnisse gewonnen und auf dem diese angewandt werden, ist beiden Richtungen gemein: Die Wirtschaftsinformatik befasst sich mit „*Informationssysteme[n] in Wirtschaft und Gesellschaft, sowohl von Organisationen als auch Individuen*“ (Österle et al., 2010, S. 666). Dabei handelt es sich um soziotechnische Systeme, die sowohl Menschen als auch technische Systeme umfassen, einschließlich der im Sinne dieser Arbeit betrachteten Softwareagenten.

Diese Arbeit ordnet sich der gestaltungsorientierten Wirtschaftsinformatik¹ zu und greift bei der Gestaltung des zu entwickelnden Verfahrens auf die sieben Richtlinien nach Hevner et al. (2004, S. 82 ff.) zurück:

1. **Gestaltung eines Artefakts:** Die gestaltungsorientierte Wirtschaftsinformatik entwirft IT Artefakte zur effektiven Nutzung im Erkenntnisgegenstand der soziotechnischen Systeme (Österle et al., 2010; Hevner et al., 2004). IT Artefakte im Sinne der gestaltungsorientierten Wirtschaftsinformatik sind Konstrukte (Vokabeln und Symbole), Modelle (Abstraktionen und Repräsentationen), Methoden (Algorithmen und Verfahren) sowie Instantiierungen (Implementierungen und prototypische Systeme) (Hevner et al., 2004, S. 77). Diese Arbeit entwickelt ein Verfahren, das es Softwareagenten ermöglicht, Beitrittsentscheidungen zu Multiagenten-Organisationen mit Hilfe von Revenue Management-basierten Ansätzen zu treffen. Dieses Verfahren basiert auf einem Modell, welches zwar als eigenes Artefakt aufgefasst werden könnte, hier jedoch als Teil des methodischen Artefakts angesehen wird. Diese Auffassung begründet sich wie folgt: „*Solving a problem simply means representing it so as to make the solution transparent*“ (Simon, 1996, S. 132). Die Modellierung selbst bildet somit einen wesentlich Beitrag zur Lösung des betrachteten Problems.

¹ Der Begriff der gestaltungsorientierten Wirtschaftsinformatik wird hier synonym zum englischen *Design Science* verwendet.

2. **Problemrelevanz:** Das Forschungsziel der gestaltungsorientierten Wirtschaftsinformatik ist die Gewinnung von Wissen über die Entwicklung und Implementierung von IT-basierten Lösungen für Problemstellungen aus dem betriebswirtschaftlichen Bereich (Hevner et al., 2004, S. 84). Diese Arbeit beschäftigt sich mit Softwareagenten, die im Allgemeinen nicht zum Selbstzweck, sondern zur Erfüllung betriebswirtschaftlicher Ziele entwickelt werden. Der Beitritt eines Softwareagenten zu einer Multiagenten-Organisation bindet einen Teil der ihm zur Verfügung stehenden Ressourcen und hat somit Einfluss auf mögliche zukünftige Beitritte zu anderen Multiagenten-Organisationen. Bei der Beitrittsentscheidung müssen somit Mitgliedschaften in anderen Multiagenten-Organisationen sowie potentielle Beitrittsanfragen beachtet werden. Diese Faktoren haben direkten Einfluss auf die vom Softwareagenten erwirtschafteten Kompensationen.
3. **Evaluation:** Das entwickelte Artefakt ist auf seine Nützlichkeit, Qualität sowie Wirksamkeit hin zu überprüfen (Hevner et al., 2004, S. 85). Diese Arbeit evaluiert das entwickelte Verfahren mit Hilfe einer Multiagenten-Simulation. Dabei wird herausgestellt, unter welchen Gegebenheiten das entwickelte Verfahren dem anwendenden Softwareagenten bei Beitrittsentscheidungen Vorteile verschafft.
4. **Forschungsbeitrag:** Das entwickelte Artefakt muss einen Beitrag zum Forschungsbereich des Artefakts leisten (Hevner et al., 2004, S. 87). Der Beitrag dieser Arbeit ist ein neues Verfahren, das es Softwareagenten ermöglicht, den Beitritt zu einer Multiagenten-Organisation zu bewerten und eine entsprechende Entscheidung über den Beitritt zu treffen. Das entwickelte Verfahren basiert auf Ansätzen des Revenue Management als Teilbereich des Operations Research, welche auf Beitrittsentscheidungen zu Multiagenten-Organisationen übertragen und entsprechend angepasst werden. Diese Arbeit lässt sich im „*Design Science Research Knowledge Contribution Framework*“ nach Gregor und Hevner (2013, S. 345 ff.) somit den Exaptationen zuordnen. Die Übertragung bekannter Revenue Management Verfahren auf Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen ist nicht-trivial, da unterschiedliche Gegebenheiten eine Anpassung des verwendeten Modells und damit auch der zu verwendeten Entscheidungsverfahren erfordern.
5. **Wissenschaftliche Sorgfalt:** Die gestaltungsorientierte Wirtschaftsinformatik bedarf sorgfältiger wissenschaftlicher Methoden, sowohl bei der Entwicklung des Artefakts als auch bei dessen Evaluation (Hevner et al., 2004, S. 87 f.). Auf informationstechnischer Seite setzt diese Arbeit auf dem Wissen über Softwareagenten und Multiagentensysteme auf und untersucht Beitrittsentscheidungen zu Multiagenten-Organisationen. Zur Problemlösung wird die Perspektive des Revenue Management gewählt, dessen zur Verfügung gestellte Herangehensweisen rigoros angewandt werden, um das entwickelte Verfahren auf den Gegenstand der Softwareagenten anzupassen. Das Verfahren wird mit Hilfe einer Multiagenten-Simulation evaluiert, was einer gängigen Methode zur Evaluation des Verhaltens von Softwareagenten in der gestaltungsorientierten Wirtschaftsinformatik entspricht.

6. **Gestaltung als Suchprozess:** Die Entwicklung von Artefakten in der gestaltungsorientierten Wirtschaftsinformatik kann als Suchprozess aufgefasst werden, der iterativ zu einer Lösung gelangt. Simon (1996, S. 128 f.) beschreibt diesen Suchprozess als Zyklus aus Entwicklung und Test, der auf dem Weg des Suchprozesses mehrmals durchlaufen wird. Der Problemlösungsprozess selbst nutzt nach Simon (1996, S. 117) dabei Instrumente (engl. *means*), um bei gegebenen Gesetzmäßigkeiten (engl. *laws*) bestimmte Ziele (engl. *ends*) zu erreichen. Da es im Allgemeinen nicht möglich ist, alle Möglichkeiten eines Problemlösungsprozesses in endlicher Zeit zu überprüfen, wird die Entwicklung von Artefakten als Suchprozess ausgestaltet. Diese Arbeit entwickelt ein Verfahren auf Basis bestehender Ansätze aus dem Revenue Management und zeigt mittels einer Multiagenten-Simulation auf, unter welchen Bedingungen das Verfahren wirksam eingesetzt werden kann.
7. **Kommunikation der Forschung:** Forschung im Rahmen der gestaltungsorientierten Wirtschaftsinformatik soll sowohl der technischen als auch der betriebswirtschaftlichen Zuhörer- und Leserschaft zugänglich gemacht werden (Hevner et al., 2004, S. 90). Vorarbeiten zu dieser Arbeit wurden insbesondere auf wirtschaftsinformatischen Konferenzen publiziert, die beide Personengruppen bedienen. Die Übertragung von Revenue Management Ansätzen auf Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen wurde in Premm et al. (2013) vorgenommen. Premm (2015) erweitert das Verfahren auf Beitrittsentscheidungen mit zeitlich unbegrenzter Laufzeit im Kontext der Anwendungsdomäne Cloud Computing. Premm und Kirn (2017) zeigen Problemstellungen bei der Integration autonomer Softwareagenten in die betriebliche Organisation auf.

1.4 Aufbau der Arbeit

Diese Arbeit ist untergliedert in fünf Kapitel.

Kapitel 1 (*Einleitung*) gibt einen Überblick über die Problemstellung, den Forschungsansatz sowie die epistemologische Einordnung dieser Arbeit.

Kapitel 2 (*Analyse des Stands der Forschung*) erarbeitet die für die folgenden Kapitel nötigen Definitionen und Grundlagen. Hierzu zählen insbesondere die Begriffe, die im Zusammenhang mit Beitrittsentscheidungen zu Multiagenten-Organisationen stehen und die aus der Perspektive der sozialwissenschaftlichen Organisationstheorie wie auch der VKI betrachtet werden. Weiterhin wird der Stand der Forschung zur Problemstellung, existierende Lösungsansätze sowie die sich hieraus ergebende Forschungslücke aufgezeigt.

Kapitel 3 (*Verfahrensentwurf*) gibt eine Einführung in das Revenue Management als Teilbereich des Operations Research. Nach Überprüfung der sich hieraus ergebenden Voraussetzungen, werden die Verfahren des Revenue Management auf Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen angepasst. Dieses Kapitel entwickelt zum einen ein Revenue Management-basiertes Modell der Beitrittsentscheidung und zum anderen hierauf aufbauend ein Entscheidungsverfahren für diese Beitrittsentscheidungen.

Kapitel 4 (*Evaluation*) evaluiert das im vorangegangenen Kapitel entwickelte Verfahren mit Hilfe einer Simulationsstudie. Mit Hilfe einer Multiagenten-Simulation werden synthetische domänenunabhängige Daten verwendet, um die Wirksamkeit und -richtung von unabhängigen auf abhängige Variablen des Verfahrens herauszustellen.

Kapitel 5 (*Zusammenfassung und Ausblick*) fasst die Ergebnisse dieser Arbeit zusammen und gibt einen Ausblick auf möglichen anschließenden Forschungsbedarf.

2. Analyse des Stands der Forschung

Der Begriff des Softwareagenten entstammt dem Forschungsgebiet der VKI. Während sich die künstliche Intelligenz (KI) mit intelligentem Verhalten, wie beispielsweise dem heuristischen Lösen von Problemstellungen, einzelner Softwareagenten beschäftigt, widmet sich das Forschungsgebiet der VKI den durch die gleichzeitige Betrachtung mehrerer Softwareagenten entstehenden Nebenläufigkeiten (Bond und Gasser, 1988, S. 3). Als erstes umfassendes Werk der VKI fassen Bond und Gasser (1988) die Entwicklungen des Forschungsbereichs bis Ende der 1980er Jahre zusammen. Die Autoren unterscheiden dabei innerhalb der VKI zwischen den zwei Hauptforschungsgebieten *Verteilte Problemlöser* sowie *MAS*. Auf die Unterteilung dieser Forschungsgebiete unter der Betrachtung als Organisationsmodelle der VKI geht Abschnitt 2.2 vertiefend ein. Die Grundzüge dieser Unterscheidung finden in der VKI weite Verbreitung und lassen sich wie folgt charakterisieren:

Verteilte Problemlöser. In der historischen Entwicklung des Forschungsgebiets VKI tritt die Bezeichnung *verteilte Problemlöser* (engl. *distributed problem solver*) deutlich früher in Erscheinung als der Begriff der Multiagentensysteme. Bereits Ende der 1970er Jahre stellten Smith und Davis das Contract Net Protocol als Verhandlungsprotokoll für verteilte Problemlöser vor: *„we propose a model of a distributed problem solver which consists of a collection of processors connected with communications and control mechanisms that enable them to operate concurrently, and enable them to cooperate in solving complex problems“* (Smith und Davis, 1978, S. 2). Den Ansätzen verteilter Problemlöser ist gemein, dass mehrere Knoten an der Lösung eines a priori gestellten Problems zusammenarbeiten. Die Lösung dieses Problems bzw. die damit zusammenhängende Bearbeitung einer Aufgabe steht somit im Mittelpunkt und zieht die nachgelagerte Entscheidung über die Vorgehensweise der Problembewältigung mit sich. Verteilte Problemlöser sind demnach stets *„top down“* strukturiert und setzen voraus, dass sich die beteiligten Knoten am Problemlösungsprozess in vorgesehener bzw. vorgegebener Weise beteiligen. Durfee und Rosenschein (1994) fassen diese Problemfokussierung der Ansätze zu verteilten Problemlösern bei der Gegenüberstellung zu MAS wie folgt zusammen: *„the emphasis was on the problem, and how to get*

multiple agents to work together to solve it in a coherent, robust, and efficient manner“ (Durfee und Rosenschein, 1994, S. 54).

Die für verteilte Problemlöser verwendeten „Agenten“ erfüllen häufig nicht die in dieser Arbeit gesetzten Kriterien an Softwareagenten (vgl. Abschnitt 2.1). Eine Ausnahme bildet hierbei die Klasse der kooperativen verteilten Problemlöser (vgl. auch Abschnitt 2.2.2.3), deren Akteure je nach Ausprägung ebenfalls den Kriterien für Softwareagenten genügen (Kirn, 1996a, S. 55). Um mehrere Agenten der Bearbeitung einer Aufgabe mitwirken zu lassen, muss die Aufgabe selbst jedoch abhängig vom verwendeten Verfahren einige Voraussetzungen erfüllen: Beispielsweise muss die Aufgabe selbst in Teilaufgaben zerlegbar und deren Teilergebnisse wieder für die Gesamtaufgabe verwendbar sein. Die konkreten Voraussetzungen sind dabei abhängig von der gestellten Aufgabe sowie des verwendeten Verteilungsmechanismus. Dabei kann es auch vorgesehen und notwendig sein, dass mehrere Agenten zur Problemlösung kooperieren. In diesem Fall wird auch von kooperativen verteilten Problemlösern gesprochen (vgl. beispielsweise Durfee et al. (1989)). Auf die Unterscheidung der verschiedenen Typen verteilter Problemlöser, deren Voraussetzungen für die Entwicklung und Ausführung und die sich hieraus ergebenden Auswirkungen für organisatorische Fragestellungen geht Abschnitt 2.2.2 näher ein.

Multiagentensysteme. Während die ersten Ansätze der verteilten Problemlöser diese als eine „Ansammlung von Prozessorknoten“ verstehen (Smith und Davis, 1978, S. 2), entwickelten sich neben Koordinationsmechanismen auch die Fähigkeiten der involvierten Akteure bzw. Knoten weiter. Hieraus bildeten sich Ansätze, deren Akteure keiner expliziten Koordination unterstanden: Klett definiert diesen Typ als *„einen Verbund aus autonomen Systemen, die mit Hilfe eines Transportmediums mit einer allen Verbundmitgliedern zugänglichen Datenbank kommunizieren ... Bei dieser Verbundarchitektur ist an keiner Stelle, lokal oder global, explizit Verhaltenswissen gespeichert, noch sind Mechanismen vorgesehen, die gezielt die Kontrolle über Teilauftragsbearbeitung an zuvor ausgewählte Verbundmitglieder delegieren und deren Ergebnisse zu einem Gesamtergebnis zusammenfassen“* (Klett, 1989, S. 99). Bei dieser Art von Systemen wird die Koordination der beteiligten Akteure von einer durch den Entwickler vorgegebenen hierarchischen Perspektive auf die Akteure selbst übertragen, die die Struktur somit zwischen den Beteiligten aushandeln. Im Gegensatz zu den Agenten verteilter Problemlöser, können Softwareagenten im Allgemeinen schon zeitlich vor MAS existieren, an denen sie beteiligt sind (Bond und Gasser, 1988, S. 3). Bereits aus dieser Tatsache ergibt sich ein wesentliches Unterscheidungskriterium zwischen MAS und verteilten Problemlösern: Während letztere „top down“ auf Basis eines a priori bekannten Problems gebildet werden und dabei *„das vorrangige Interesse auf [der] Gestaltung des Gesamtsystems und auf [den] daraus abzuleitenden Anforderungen an die Architektur, Funktionalität bzw. Form der Interaktionen“* (Kirn, 1996a, S. 64) liegt, finden sich Softwareagenten aus eigener Initiative zu MAS zusammen und strukturieren dieses somit „bottom-up“. Das Entwicklungsziel als Unterscheidungskriterium stellen bereits Durfee und Rosenschein

(1994) heraus: „*the focus of MAS has been on the agent, and getting it to interact meaningfully with other agents*“ (Durfée und Rosenschein, 1994, S. 54 f.). Bei der Entwicklung von Softwareagenten im Forschungsgebiet der Multiagentensysteme steht somit die Entwicklung des Softwareagenten selbst sowie dessen Interaktionsmöglichkeiten mit anderen Softwareagenten und nicht die Entwicklung des Gesamtsystems zur Lösung eines a priori definierten Problems im Vordergrund. Softwareagenten, die in einem MAS interagieren, können somit durch unterschiedliche Entwickler entworfen worden sein, denen im Allgemeinen Ziele durch diesen Entwickler vorgegeben wurden. Dabei müssen die Ziele der beteiligten Softwareagenten nicht übereinstimmen und können unter Umständen auch den Zielen des zugehörigen MAS entgegenlaufen oder zu einem späteren Zeitpunkt entwickelt werden (Bond und Gasser, 1988, S. 3). Insbesondere ergibt sich aus dieser Abgrenzung, dass Interaktionsstrukturen zwischen Softwareagenten nicht bei deren Entwicklung durch den Entwickler festgelegt, sondern zur Laufzeit zwischen den Softwareagenten ausgehandelt werden. Die Anforderungen, die in der Literatur an Softwareagenten im Zusammenhang mit MAS gestellt werden, sind vielfältig und werden in den nachfolgenden Abschnitten analysiert. Abschnitt 2.2.3 geht näher auf den in dieser Arbeit verwendeten Begriff von Multiagentensystemen als Zusammenschluss von Softwareagenten ein und untersucht analog zu den verteilten Problemlösern Voraussetzungen, die sich für die Entwicklung, Ausführung und Organisation von MAS ergeben.

Neben MAS und verteilten Problemlösern erwähnen Bond und Gasser (1988, S. 3) die *parallele KI* als dritte Forschungsrichtung. Diese beschäftigt sich jedoch nicht mit der konzeptionellen Weiterentwicklung von intelligenten Softwareagenten und deren Interaktion, sondern ausschließlich mit der Lösung von Performanzproblemen und hat somit für Beitrittsentscheidungen zu Multiagenten-Organisation auf konzeptioneller Ebene keine Relevanz.

Dieses Kapitel beschreibt und analysiert den Stand der Forschung im Zusammenhang mit Beitrittsentscheidungen zu Multiagenten-Organisationen und erarbeitet die für die folgenden Kapitel nötigen Definitionen und Grundlagen aus den Bereichen der Organisationstheorie und VKI. Hierzu werden die in dieser Arbeit verwendeten Begriffe auf ihr Vorkommen und ihre Verwendung in der Organisationstheorie sowie VKI analysiert und für die weitere Verwendung in dieser Arbeit definiert. Abschnitt 2.1 widmet sich der Definition von Softwareagenten unter Berücksichtigung der Aufgabenträger-Eigenschaft der Organisationstheorie. Abschnitt 2.2 beschreibt Organisationsmodelle der VKI und untersucht diese auf Beitrittsentscheidungen der beteiligten Akteure. In Abschnitt 2.3 wird der Organisationsbegriff der Betriebswirtschaftslehre sowie der damit zusammenhängende Begriff der Mitgliedschaft untersucht und die jeweiligen Verknüpfungen zu Softwareagenten und Multiagenten-Organisationen hergestellt. Abschnitt 2.4 geht abschließend auf die Beitrittsentscheidung ein – ebenfalls aus organisationstheoretischer als auch informationstechnischer Perspektive – und analysiert in diesem Kontext insbesondere vertragliche Nebenbedingungen von Beitrittsentscheidungen und Mehrfachmitgliedschaften.

2.1 Softwareagenten

In der Literatur zu Multiagentensystemen finden sich zahlreiche Definitionen für den Begriff *Agent* die jeweils unterschiedliche Schwerpunkte setzen (vgl. Abschnitt 2.1.1). Ferstl und Sinz (2008, S. 124) verstehen unter einem Agenten „*Personen oder als selbstständig betrachtete automatisierte Systeme*“. Diese Arbeit befasst sich jedoch nicht mit jeglicher Art von Systemen, sondern beschränkt sich auf Softwaresysteme. Um herauszustellen, dass es sich bei den hier betrachteten Agenten um Software handelt und um von Agentenbegriffen anderer Theorien, wie beispielsweise der Prinzipal-Agenten-Theorie, abzugrenzen, wird hier der Begriff *Softwareagent* gewählt (Kirn, 2002).

Da es sich bei Softwareagenten um ein vergleichsweise junges Forschungsfeld handelt, kann eine präzise einheitliche Definition unter Umständen Innovationen hemmen. Dies ist bereits aus anderen Disziplinen wie der Organisationstheorie bekannt: „*Since research in this area is still in its early stages, undue exactness must be avoided*“ (Arrow, 1970, S. 224). Um Softwareagenten als Gegenstand dieser Arbeit von anderen informationstechnischen Systemen abzugrenzen, beschreiben und analysieren die nachfolgenden Abschnitte den Stand der Forschung zu Softwareagenten und deren Verknüpfung zum Begriff des maschinellen Aufgabenträgers der Organisationstheorie und definieren den Begriff *Softwareagent* hinreichend genau, um hierauf aufbauend Beitrittsentscheidungen zu Multiagenten-Organisationen betrachten zu können.

2.1.1 Definitionen in der Literatur

Anforderungen an den Agentenbegriff unterscheiden sich wesentlich im Hinblick auf die Fähigkeiten von Softwareagenten: Shoham (1993) bezeichnet bereits einen Lichtschalter als Softwareagenten, der den Willen eines menschlichen Nutzers durch dessen Bedienung umsetzt: „*It is perfectly coherent to treat a light switch as a (very cooperative) agent with the capability of transmitting current at will, who invariably transmits current when it believes that we want it transmitted and not otherwise; flicking the switch is simply our way of communicating our desires*“ (Shoham, 1993, S. 53). Der Aspekt der Kooperation bezieht sich in diesem Beispiel auf die Interaktion mit dem menschlichen Nutzer und nicht auf die Interaktion mit anderen Softwareagenten. Andere Definitionen setzen hingegen weitergehende Fähigkeiten voraus und fordern von Softwareagenten unter anderem proaktives Verhalten (Wooldridge und Jennings, 1995, S. 116). Inwiefern der oben angeführte Lichtschalter als Softwareagent bezeichnet werden kann, ist somit abhängig von der Definition und den hieraus geforderten Eigenschaften.

Dieser Abschnitt arbeitet die Unterschiede zwischen den in der Literatur gängigen Definitionen heraus. Dabei wird in der Literatur häufig auf Definitionen von Wooldridge und Jennings (1995) und Jennings (2000) zurückgegriffen. Da in beiden Definitionen Jennings mit beteiligt war, weisen diese inhaltliche Parallelen, aber ebenso konzeptionelle Unterschiede auf. Die „*weak notion of agency*“ nach Wooldridge und Jennings (1995, S. 116) sieht Softwareagenten als Hardware- oder Software-basierte Computersysteme mit vier wesentlichen Eigenschaften, auf die in den folgenden Abschnitten näher eingegangen wird:

- Autonomie (vgl. Abschnitt 2.1.1.1): „*agents operate without the direct intervention of humans or others, and have some kind of control over their actions and internal state*“
- Soziale Fähigkeiten (vgl. Abschnitt 2.1.1.2): „*agents interact with other agents (and possibly humans) via some kind of agent-communication language*“
- Reaktivität (vgl. Abschnitt 2.1.1.3): „*agents perceive their environment ..., and respond in a timely fashion to changes that occur in it*“
- Proaktivität (vgl. Abschnitt 2.1.1.4): „*agents do not simply act in response to their environment, they are able to exhibit goal-directed behaviour by taking the initiative*“

Wooldridge und Jennings (1995, S. 117) diskutieren weitere Eigenschaften von Softwareagenten im Rahmen einer „*stronger notion of agency*“, die noch stärker versucht, menschliche Eigenschaften auf Softwareagenten zu übertragen. Dabei liefern sie allerdings keine Definition, sondern geben einen kurzen Überblick über entsprechende Auffassungen anderer Autoren: „*it is quite common in AI to characterise an agent using mentalistic notions, such as knowledge, belief, intention, and obligation*“ (Wooldridge und Jennings, 1995, S. 117). Während die dabei verwendeten Begriffe von *Knowledge*, *Belief* und *Intention* nahe an den Begriffen und dem Verständnis der von Bratman et al. (1988) vorgestellten Belief-Desire-Intention (BDI) Architektur liegen, werden beispielsweise Emotionen von Softwareagenten in dieser Arbeit nicht betrachtet.

Jennings (2000) definiert Softwareagenten hingegen wie folgt: „*an agent is an encapsulated computer system that is situated in some environment and that is capable of flexible, autonomous action in that environment in order to meet its design objectives*“ (Jennings, 2000, S. 280). Statt einzelner nebeneinander stehenden Anforderungen, die jeweils erläutert werden, fasst Jennings (2000) seine Definition von Softwareagenten in einem Satz zusammen. Trotz dieses konzeptionellen Unterschieds gibt es inhaltliche Zusammenhänge und ein Teil der oben genannten Anforderungen findet sich in dieser Definition wieder: Während die Autonomie explizit als „*autonomous action*“ genannt wird, sind für andere Zuordnungen Interpretationen erforderlich. Beispielsweise kann „*capable of ... action ... in order to meet its design objective*“ durch die vorgegebene Zielorientierung als Entsprechung zur *Proaktivität* (vgl. Abschnitt 2.1.1.4) aufgefasst werden. Bereits dieses Beispiel macht deutlich, dass sich eine einheitliche Definition unter anderem durch unterschiedliches Vokabular nicht durchsetzen konnte. Im Folgenden werden die zentralen Begriffe, die in der Literatur Verwendung finden, aufgegriffen und untersucht. Dies bildet die Basis für die Definition von Softwareagenten in dieser Arbeit und somit für die Betrachtung von Beitrittsentscheidungen zu Multiagenten-Organisationen.

2.1.1.1 Autonomie und Lernfähigkeit

Autonomie definiert das Duden-Fremdwörterbuch als „1. *Selbstständigkeit [in nationaler Hinsicht], Unabhängigkeit. 2. (Philos.) Willensfreiheit*“ (Dudenredaktion, 2006, S. 111). *Autonom* leitet sich aus dem griechischen her und lässt sich mit „*nach eigenen Gesetzen lebend*“ (Dudenredaktion, 2006, S. 111) übersetzen. Diese Definition macht bereits deutlich,

dass der Begriff *Autonomie* häufig im Zusammenhang mit staatsbildenden Fragestellungen wie *autonomen Regionen* verwendet wird. Zahlreiche Autoren der VKI entlehnen sich dieses Begriffs, um ihn als Eigenschaft von Softwareagenten zu verwenden. Oftmals wird auf die exakte Bedeutung einer Forderung nach Autonomie von Softwareagenten jedoch nicht eingegangen. Die *Conference on Autonomous Agents and Multiagent Systems (AAMAS)* veranstaltete hierfür einen separaten Workshop unter dem Namen *Agents and Computational Autonomy*, um Autonomie bei Softwareagenten zu untersuchen (Nickles et al., 2003). Trotzdem konnte sich keine einheitliche Definition durchsetzen, so dass es nahe liegt, für eine Betrachtung von Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen zunächst die Eigenschaft *Autonomie* zu überprüfen.

Wooldridge und Jennings (1995) beziehen sich bei der Definition der Autonomie-Eigenschaft auf Castelfranchi (1995) und stellen dabei zwei wesentliche Punkte heraus: (i) Softwareagenten agieren ohne den direkten Eingriff von Menschen oder anderen Softwareagenten und (ii) haben ein gewisses Maß an Kontrolle über ihre Aktionen und ihren internen Zustand (Wooldridge und Jennings, 1995). Dabei definiert (i), gegenüber welchem anderen Objekt Autonomie besteht und setzt somit die Voraussetzung für den von Castelfranchi (1995) verwendeten Autonomie-Begriff, der für Autonomie stets einen Bezugspunkt vorsieht. Dies wird in der Literatur oftmals vernachlässigt und es wird Autonomie als Eigenschaft von Softwareagenten gefordert, ohne eine Konkretisierung des Bezugs der Autonomie. Nach Castelfranchi (1995, S. 58) kann Autonomie jedoch nicht ohne eine Relation definiert werden. In der Literatur zu Softwareagenten werden sowohl – wie oben erwähnt – Menschen und andere Softwareagenten (Wooldridge und Jennings, 1995), aber auch der Entwickler eines Softwareagenten (Russell und Norvig, 2010, S. 39) als Bezugspunkte verwendet. Dabei ist fraglich, inwiefern ein Softwareagent autonom gegenüber seinem Entwickler werden kann, der das Verhalten des ersteren durch Programmierregeln vorgibt. Russell und Norvig (2010, S. 39) setzen dieses Autonomie-Verständnis mit der Lernfähigkeit des Softwareagenten gleich: Ist ein Softwareagent in der Lage, sein bei der Entwicklung vorgegebenes Wissen zu adaptieren, zu verwerfen und zu erweitern, ist der Softwareagenten auf der Ebene seines Wissens autonom von seinem Entwickler. *Autonom* ist in diesem Fall nicht gleichzusetzen mit *unabhängig*, da das Verhalten eines Softwareagenten stets auf seinem aktuellen Wissen beruht und somit das Verhalten und die sich daraus einstellenden Lerneffekte abhängig von Vorwissen, folglich auch von initial vorgegebenem Wissen, sein können. Zwar kann ein Softwareagent autonom gegenüber dem vorgegebenen Wissen des Entwicklers werden, Russell und Norvig (2010) lassen jedoch die Frage offen, ob und in welcher Art ein Softwareagent auch autonom gegenüber den implementierten Lernverfahren werden kann bzw. muss. Wird ein Softwareagent sich keine neuen Lernverfahren aneignen oder bestehende adaptieren, ist seine Autonomie gegenüber dem Entwickler zumindest eingeschränkt. Um einen Softwareagenten vollständig autonom von seinem Entwickler agieren zu lassen, ist es folglich zwingend notwendig, dass der Softwareagent nicht nur Informationen über seine Umgebung sammelt und durch Beobachtung auch Regeln seiner Wissensbasis hinzufügen kann, sondern auch sein Lernverfahren selbst adaptiert und auf neue Gegebenheiten anpasst. Selbst dann ist der Softwareagent noch nicht unabhängig von seinem Entwickler, der sich für eine Programmiersprache entschieden hat und deren Änderung dem Softwareagenten selbst

kaum möglich erscheint. Setzt man dieses Verständnis einer umfangreichen Autonomie voraus, existieren zur Zeit dieser Arbeit keine Softwareagenten. Demnach ist es fraglich, ob die Forderung nach Autonomie bei Softwareagenten zielführend für deren Einsatz ist. Wie oben bereits erwähnt, stellen Russell und Norvig (2010) bei der Autonomie eines Softwareagenten auf dessen Lernfähigkeit ab. Müller-Hengstenberg und Kirn (2016) bezeichnen diese Art der Autonomie als „epistemologische Autonomie“ und ordnen sie der „technischen Autonomie“, folglich der *„tatsächliche[n] Realisierbarkeit intelligenten Verhaltens angesichts begrenzter Mittel, insb. hinsichtlich konkreter Ausstattung an Sensorik und Aktuatorik, Energie, Zeit und Speicherplatz“* (Müller-Hengstenberg und Kirn, 2016, S. 99) zu. Gibt beispielsweise der Entwickler eines Softwareagenten diesem einen gewissen Suchraum zur Lösungsfindung vor, ermöglicht die Lernfähigkeit eines Softwareagenten bei geänderten Umgebungsbedingungen die Anpassung durch Verkürzung oder Erweiterung dieses Suchraums zur Laufzeit. Das Zitat zeigt jedoch einen weiteren Punkt der Autonomie gegenüber dem Entwickler auf: Da es sich bei einem Softwareagenten um Software handelt, benötigt dieser zur Ausführung Zugriff auf entsprechende Hardware. Diese Ausstattung an Ressourcen ist somit essentiell für Softwareagenten und deren Lernfähigkeiten (vgl. Abschnitte 2.1.3 und 2.3.2.1.5).

Das Kriterium der Lernfähigkeit wird auch in dieser Arbeit für die Betrachtung von Beitrittsentscheidungen von Softwareagenten gewählt. Es ist demnach nicht nötig, dass ein Softwareagent vollständig autonom agiert, sondern er muss in der Lage sein, aus beobachteten Informationen zu lernen und durch diese Lernfähigkeit eine gewisse Unabhängigkeit gegenüber dem ihm bei der Entwicklung gespeicherten Vorwissen zu erreichen. Diese Lernfähigkeit deckt somit nur einen Teil der gemeinhin unter Autonomie verstandenen Kriterien ab. Gleichzeitig ist das Ergebnis der Lernfähigkeit und somit die Unabhängigkeit von gespeichertem Vorwissen abhängig von der Zeit, die seit der Implementierung durch den Entwickler vergangen ist: Erst durch eine ausreichend große Zeitspanne für den Einsatz des Lernverfahrens des Softwareagenten ist es diesem möglich, einen gewissen Grad an Unabhängigkeit gegenüber dem bei der Entwicklung gespeicherten Vorwissen zu erreichen.

Softwareagenten erscheinen durch dieses Kriterium aus einer externen Perspektive als nicht-deterministisch, da sie durch die Lerneffekte auf die gleichen Umgebungssituationen unterschiedlich reagieren (vgl. auch Timm und Hillebrandt (2006, S. 4 ff.)). Der nach außen hin scheinende Nicht-Determinismus lässt jedoch keine Schlüsse über die internen Abläufe des Softwareagenten zu. Diese können sowohl durch stochastisch arbeitende Algorithmen nicht-deterministisch sein, als auch deterministische Lernverfahren beinhalten. Diese Lernfähigkeit kommt somit insbesondere bei Beitrittsentscheidungen zu Multiagenten-Organisationen zum Tragen. Die Entwicklung eines Verfahrens zur Maximierung des Nutzens eines Softwareagenten bei der Beitrittsentscheidung zu einer Multiagenten-Organisation bedingt, dass das zu entwickelnde Verfahren nicht auf einfachen deterministischen Regeln basieren darf, sondern dem Softwareagenten neu zugegangene Informationen mit in die Entscheidungsfindung aufgenommen werden müssen.

2.1.1.2 Soziale Fähigkeiten

Das Adjektiv *sozial* wird in der organisationstheoretischen Betriebswirtschaftslehre meist ausschließlich mit Menschen in Verbindung gebracht (vgl. z.B. Heidhardt (1980, S. 2077 f.)). Dabei dient es oftmals auch zur Abgrenzung von rein technischen Systemen. So beschreibt beispielsweise die wortverwandte Form *sozio* im Begriff sozio-technisches System die Verknüpfung von Mensch und Maschine. Im Rahmen der VKI hingegen wird der Begriff *sozial* auch für Beziehungen zwischen Maschinen bzw. Softwareagenten verwendet. Diese unterschiedliche Begriffsnutzung lässt sich durch verschiedene Bedeutungsalternativen erklären. Der Duden gibt fünf verschiedene Definitionsalternativen für das Adjektiv *sozial*: (i) „die menschliche Gesellschaft, Gemeinschaft betreffend“, (ii) „das Gemeinwohl betreffend“, (iii) „auf das Wohl der Allgemeinheit bedacht“, (iv) „die gesellschaftliche Stellung betreffend“ sowie (v) „gesellig lebend“ (Dudenredaktion, 2006, S. 974). Obwohl nur in der Definition (i) *sozial* direkt mit dem Menschen verknüpft wird, stellt der Mensch indirekt auch in den anderen Definitionen den zentralen Bezugspunkt dar. Die Auslegung der Definition in Richtung einer gemeinwohlstiftenden Eigenschaft steht jedoch einer in der Literatur oftmals genannten Forderung nach rationalem und nutzenmaximierendem Verhalten von Softwareagenten entgegen. Daher liegt es nahe, den Begriff des *Sozialen* im Kontext von Softwareagenten als Übertragung der in der Definition (i) angeführten menschenbezogenen Bedeutung auf Softwareagenten auszulegen. Unter sozialen Fähigkeiten von Softwareagenten wird in der Literatur hauptsächlich die Möglichkeit einer Kommunikation mit anderen Softwareagenten oder auch mit Menschen verstanden (Wooldridge und Jennings, 1995, S. 116). Dabei unterscheidet sich die Fähigkeit zu sozialem Verhalten von der Fähigkeit ausschließlich Informationen auszutauschen (Wooldridge, 1999, S. 34). Wooldridge (2009, S. 28) konkretisiert diese Forderung und gibt exemplarisch Verhandlungen und Kooperationen mit anderen Softwareagenten an, um von der rein technischen Möglichkeit der Kommunikation abzugrenzen.

Die technische Möglichkeit zu kommunizieren, ist die Grundvoraussetzung, um Informationen zwischen Softwareagenten auszutauschen. Diese rein technische Ebene ist jedoch nicht ausreichend, um soziale Fähigkeiten von Softwareagenten zu begründen. Hierzu wird im Allgemeinen auf semantischer Ebene eine gemeinsame Agentenkommunikationssprache benötigt (Genesereth und Ketchpel, 1994, S. 48 ff.). Dies ist auch die Voraussetzung, um Beitrittsverhandlungen von Softwareagenten mit Multiagenten-Organisationen zu betrachten: Zunächst muss die technische Möglichkeit der Kommunikation gewährleistet sein, das heißt, die Kommunikationspartner müssen physisch miteinander verbunden sein (Kabel, Funk, Lichtwellen, etc.) und müssen sich auf ein gemeinsames Nachrichtenübermittlungsverfahren, mindestens auf Bit-Ebene (Modulation, Multiplex) oder gar auf höheren Ebenen (IP, TCP, etc.), geeinigt haben. Neben dieser hier als technische Ebene der Kommunikation bezeichneten Verbindung, müssen sowohl der potentiell beitretende Softwareagent, als auch die Multiagenten-Organisation für die übermittelten Nachrichten die gleiche Semantik hinterlegen (semantische Ebene). Die angebotenen Dienste des Softwareagenten oder die Anforderungen der Multiagenten-Organisation müssen für das jeweilige Gegenüber verständlich und eindeutig interpretierbar sein. Dies setzt somit eine gemeinsame Sprache wie

beispielsweise die FIPA Agent Communication Language (ACL)² voraus. Im Folgenden wird angenommen, dass eine gemeinsame Sprache der beteiligten Akteure existiert, mit der Angebote und deren Annahme oder Ablehnung ausgetauscht werden können.

2.1.1.3 Reaktivität und Umgebung

Den Begriff *Reaktivität* bzw. das Adjektiv *reaktiv* mit dem gleichen Wortstamm definieren Wooldridge und Jennings (1995) wie folgt: „*Agents perceive their environment, ... and respond in a timely fashion to changes that occur in it*“ (Wooldridge und Jennings, 1995, S. 116). Reaktivität wird folglich als zeitnahe Reaktion wahrgenommener Veränderungen in der Umgebung eines Softwareagenten verstanden. Die Einschränkung „zeitnah“ wird nicht weiter definiert, wodurch sich die Definition auf das Reagieren auf Veränderungen reduziert. Betrachtet man die durch Sensoren wahrgenommenen Veränderungen in der Umgebung als Eingabe des Softwareagenten und dessen Reaktionen mit Hilfe von Aktuatoren als Ausgabe, ist fraglich, inwiefern sich Reaktivität vom informationstechnischen EVA-Prinzip (Eingabe, Verarbeitung, Ausgabe) unterscheidet. Dabei wird davon ausgegangen, dass eine Rechenanlage Eingabewerte über entsprechende Eingabegeräte entgegennimmt, diese von einer zentralen Recheneinheit verarbeitet und die Ergebnisse mittels Ausgabegeräte zur Verfügung gestellt werden (Dudenredaktion, 2003, S. 232). Gleicht man dieses Prinzip mit den in der Literatur zu Softwareagenten Verwendung findenden Begriffen ab, bilden Sensoren die Eingabegeräte, der interne Entscheidungsprozess die Verarbeitungseinheit und die Aktuatoren die Ausgabegeräte zur Beeinflussung der Umgebung. Über die Art der Verarbeitung sagt die oben angeführte Definition nach Wooldridge und Jennings (1995), außer der Einschränkung „zeitnah“, nichts aus. Diese Einschränkung kann wiederum in die Richtung interpretiert werden, dass der Softwareagent seine Ausgabe mit Hilfe einfacher „Case-Anweisungen“ determiniert und keine zeitaufwendigen deliberative Entscheidungsfindung initiiert. Durch den Mangel an der Konkretisierung von „zeitnah“ lässt sich dieser Punkt jedoch nicht abschließend festlegen. Die Verwendung des Begriffs „zeitnah“ lässt hingegen den Schluss zu, dass hierdurch eine Unterscheidung zwischen fest vorgegebenen (reaktiven) und durch Schlussfolgerung gewonnenen (deliberativen) Aktivitäten beabsichtigt ist (siehe unten). Würde man hingegen Reaktivität nur als reines Reagieren auf Einflüsse von außen interpretieren, erhält diese Eigenschaft von Softwareagenten keine neuen Informationen für deren Definition, da dies bereits auf alle informationstechnischen Objekte und somit auch auf Softwareagenten zutrifft.

Wie Jennings (2000, S. 280) definieren auch Russell und Norvig (2010, S. 34) Softwareagenten im Zusammenspiel mit ihrer Umgebung. Die Umgebung bildet somit einen zentralen Ansatzpunkt für die Definition von Softwareagenten und ist essentiell für das Kriterium der Reaktivität. Dabei nutzen Softwareagenten Sensoren, um die Ereignisse oder Zustände der Umgebung wahrzunehmen und Aktuatoren, um die Umgebung zu beeinflussen. Obwohl die Umgebung somit ein wesentliches Merkmal eines Softwareagenten darstellt, wird die Definition und Abgrenzung in der Literatur oftmals vernachlässigt (Weyns et al., 2005,

² Die Agent Communication Language (ACL) der Foundation for Intelligent Physical Agents (FIPA) ist ein Standard zur Kommunikation zwischen Softwareagenten, der auf der Sprechakt-Theorie nach Searle (1969) basiert.

S. 2). Russell und Norvig (2010, S. 40 ff.) stellen folgende Eigenschaften der Umgebung heraus: (i) Eine Umgebung ist für einen Softwareagenten *vollständig beobachtbar*, falls zu jedem Zeitpunkt der vollständige Zustand der Umgebung über die Sensoren aufgenommen werden kann, andernfalls wird sie als *teilweise beobachtbar* bezeichnet. (ii) Nach der Anzahl an Softwareagenten kann in *Einzelagenten*- oder *Multiagenten*-Umgebungen unterschieden werden. (iii) Ist der Folgezustand der Umgebung durch eine Aktion des Softwareagenten eindeutig bestimmt, handelt es sich um eine *deterministische*, falls auch andere Einflussfaktoren (z.B. andere Softwareagenten die zeitgleich agieren) bestehen, um eine *stochastische* Umgebung. (iv) Hängen die Entscheidungen eines Softwareagenten nicht von vorangegangenen Zeitschritten ab, wird die Umgebung als *episodisch* bezeichnet, andernfalls als *sequenziell*. (v) Bleibt die Umgebung in einem festen Zustand, während der Softwareagent eine Entscheidung trifft, so ist diese *statisch*, kann sich ihr Zustand kontinuierlich ändern, *dynamisch*. (vi) Anhand des Wertebereichs der Umgebungsparameter wird die Umgebung in *diskret* oder *stetig* eingeteilt.

Bei den angeführten Eigenschaften gehen Russell und Norvig (2010) davon aus, dass die betrachteten Softwareagenten in verschiedensten Szenarien eingesetzt werden können, insbesondere auch zur Lösung von Problemen, denen ein festes Regelwerk für ausführbare Aktionen sowie deren Auswirkungen zugrunde liegt. Bei der Betrachtung von Beitrittsentscheidungen von Softwareagenten können unter Umständen sämtliche Einflussfaktoren der realen Umwelt auf den Softwareagenten einwirken. Demnach kann die Umgebung von Softwareagenten in Beitrittsszenarien wie folgt charakterisiert werden:

- (i) Die Umgebung des Softwareagenten ist von diesen stets nur **teilweise beobachtbar**, da nur eine endliche Anzahl an Sensoren mit im Allgemeinen beschränkter Reichweite vorhanden ist.
- (ii) Definitionsgemäß befinden sich in einem Beitrittsszenario mehr als ein Softwareagent und es handelt sich somit um eine **Multiagenten**-Umgebung.
- (iii) Bereits durch die Existenz anderer Softwareagenten, deren Verhalten durch die jeweils anderen beeinflusst werden kann, erscheint jedem Softwareagenten die Umgebung als **stochastisch**.
- (iv) Beitrittsentscheidungen von Softwareagenten sind im Allgemeinen **sequenziell**, hängen folglich von den Entscheidungen vorangegangener Aktionen ab, da bereits vorhandene Mitgliedschaften die Verfügbarkeit von Ressourcen des Softwareagenten beeinflussen.
- (v) Da auch andere Softwareagenten Beitrittsentscheidungen treffen und sich somit die zugrunde liegende Gruppierung und die damit verbundenen Beitrittsangebote verändern können, sehen sich Softwareagenten bei Beitrittsentscheidungen einer **dynamischen** Umgebung gegenüber.
- (vi) Im Allgemeinen können für die zu verhandelnden Parameter eines Beitritts sämtliche reellen Werte Verwendung finden und die Umgebung des beitretenden Softwareagenten somit als **stetig** eingeordnet werden.

In der Literatur wird der Begriff *reaktiv* ebenfalls verwendet, um zwei unterschiedliche Typen von Softwareagenten zu unterscheiden: (i) reaktive und (ii) deliberative Softwareagenten. Wooldridge (1999, S. 38 ff.) bezeichnet erstere Gruppe als „*purely reactive agents*“, Ferber (2001, S. 36) als „*reaktive Agenten*“ sowie Russell und Norvig (2010, S. 48 ff.) als „*simple reflex agents*“. Dabei ist den Definitionen gemein, dass dieser Typus von Softwareagenten im Gegensatz zur zweiten Variante keine Lernfähigkeit besitzt und somit auf die gleichen wahrgenommenen Umweltzustände stets in gleicher Weise reagiert. Diese Unterscheidung führt jedoch zu dem Widerspruch, dass rein reaktive Agenten nach diesem Verständnis nicht die Voraussetzung der Definition von Softwareagenten erfüllen: Für diese werden im Allgemeinen Autonomie (vgl. Abschnitt 2.1.1.1) und Proaktivität (vgl. Abschnitt 2.1.1.4) vorausgesetzt, die sich ohne Lernfähigkeit nicht realisieren lassen. Die in der Literatur auftretenden rein reaktiven Agenten fallen somit nicht unter die in dieser Arbeit verwendete Definition von Softwareagenten. Unabhängig von der Lernfähigkeit der einzelnen (reaktiven) Agenten ist es möglich, dass Multiagentensysteme, welche aus rein reaktiven Agenten bestehen, Lernfähigkeit aufweisen. Diese Lernfähigkeit der so genannten reaktiven Multiagentensysteme verändert jedoch nichts an dem Mangel an Lernfähigkeit der beteiligten rein reaktiven Agenten und schließt diese somit von der Definition als Softwareagenten für diese Arbeit aus.

2.1.1.4 Proaktivität und Zielorientierung

Das Duden-Fremdwörterbuch definiert *proaktiv* als „*eine Situation herbeiführend oder beherrschend, indem man anstatt auf etwas Geschehenes zu reagieren, durch differenzierte Vorausplanung und zielgerichtetes Handeln die Entwicklung eines Geschehens selbst bestimmt*“ (Dudenredaktion, 2006, S. 843). Wooldridge und Jennings (1995) verstehen unter Proaktivität die Fähigkeit zu initiativem zielgerichtetem Verhalten, das sich von einem ausschließlichen Reagieren auf Veränderungen in der Umwelt abhebt. Wie von Wooldridge und Jennings (1995) wird Proaktivität häufig als Antonym zum Begriff der Reaktivität verwendet, tatsächlich ist die Abgrenzung im Kontext von Softwareagenten jedoch schwierig. Während der in Abschnitt 2.1.1.3 beschriebene Begriff der Reaktivität das zeitnahe – und somit ohne aufwendige Schlussfolgerungsprozesse – Reagieren auf Veränderungen der Umwelt beschreibt, soll Proaktivität verdeutlichen, dass nicht nur auf Veränderungen reagiert, sondern auch durch eigenständige Aktionen die Initiative ergriffen wird. Neben der gegenläufigen Intention der Wortbedeutung existieren auch Gemeinsamkeiten, die eine Zuordnung einzelner Aktionen schwierig gestalten: Sowohl die reaktiven als auch die initiativ erzeugten Aktionen werden durch Sensoren und die interne Logik des Softwareagenten initiiert. In beiden Fällen ist die Wahrnehmung der Sensoren somit ursächlich für die Aktionen des Softwareagenten.

Lockemann (2006, S. 24) stellt nicht die Proaktivität, sondern die Zielorientierung einem rein reaktiven Verhalten gegenüber. Das Kriterium von zielorientiertem Verhalten findet sich auch in anderen Quellen: Bereits die zu Beginn dieses Abschnitts erwähnte Definition von Jennings (2000, S. 280) hebt die Erfüllung der Designziele von Softwareagenten hervor. Auch Russell und Norvig (2010, S. 80 f.) nutzen die Definition eines zielbasierten Softwareagenten, um diesen von rein „reflexiven“ abzugrenzen. Bei dieser Auffassung der

Zielorientierung ist diese automatisch mit einer notwendigen Lernfähigkeit verknüpft (vgl. auch Abschnitt 2.1.1.1). Beitrittsverhandlungen und darauf anschließende -entscheidungen von Softwareagenten werden stets durch deren Ziele gesteuert. Durch den Zusammenschluss mit anderen Softwareagenten verspricht sich der beitretende Softwareagent einen Vorteil gegenüber einer alleinigen Zielverfolgung. Dabei kann ein Softwareagent den Beitritt im Allgemeinen nur auf Basis von Erwartungswerten bewerten, während sich dessen Vorteilhaftigkeit in Bezug auf die Zielerreichung erst ex-post abschließend klären lässt.

2.1.2 Der Begriff des maschinellen Aufgabenträgers

Der Begriff des maschinellen Aufgabenträgers wurde von Grochla (1966, 1972) geprägt und war seinerzeit in der deutschen Organisationstheorie stark umstritten. Neben personellen Aufgabenträgern (Menschen) können in Organisationen auch Maschinen Aufgaben bearbeiten und somit als maschinelle Aufgabenträger bezeichnet werden. Ferstl und Sinz (2008, S. 3) teilen als Vertreter der Wirtschaftsinformatik diese Sichtweise und bezeichnen auch Rechner(-systeme) als maschinelle Aufgabenträger in einem betrieblichen Informationssystem. Dieser Abschnitt analysiert den Begriff des maschinellen Aufgabenträgers und dessen Entwicklung, um die Grundlage für die Betrachtung von Softwareagenten als maschinelle Aufgabenträger zu schaffen (vgl. Abschnitt 2.1.3). Um maschinelle Aufgabenträger in einer organisatorischen Struktur betrachten zu können, ist es zunächst notwendig, einige organisatorische Begriffe in diesem Kontext zu untersuchen. Die nachfolgenden Abschnitte gehen daher insbesondere auf die Begriffe *Kompetenz* (Abschnitt 2.1.2.1), *Verantwortung* (Abschnitt 2.1.2.2), *Stelle* (Abschnitt 2.1.2.3) sowie *Aufgabe* (Abschnitt 2.1.2.4) ein, die zur Charakterisierung von maschinellen Aufgabenträgern notwendig sind.

2.1.2.1 Kompetenz

Während Weber (1922, S. 124) Kompetenz mit Zuständigkeit gleichsetzt, versteht Picot (2005, S. 230) unter Kompetenz Rechte und Pflichten und bezieht sich bei der Unterteilung in verschiedene Kompetenzarten auf Hill et al. (1994, S. 125 ff.). Während letztere Kompetenzen nur als Handlungsrechte auffassen – mit dem Begriff somit nicht unmittelbar Pflichten verbinden, da sich diese erst durch die mittelbar resultierende Verantwortung ergeben – und dies auch bei der Unterscheidung in die Kompetenzarten hervorheben, verwendet Picot (2005, S. 230 f.) zur Beschreibung dieser Arten fast ausschließlich ebenfalls den Begriff der Rechte. Einzige Ausnahme stellt hier die Ausführungskompetenz dar, zu deren Definition Picot (2005, S. 230) weder den Begriff der Rechte noch der Pflichten verwendet. Sowohl Ulrich (1969, S. 852), Bleicher (1980a, S. 1056), Thom (1992, S. 2327) als auch Bronner (1992, S. 2507) verstehen unter Kompetenz in Abgrenzung zur im folgenden Abschnitt behandelten Begriff der Verantwortung nur Rechte, die im Allgemeinen in Verbindung mit einer Stelle übertragen werden. Hieraus lässt sich jedoch nicht ableiten, dass Kompetenzen – im Sinne von Rechten – ohne entsprechende Pflichten übertragen werden sollten. Im Gegenteil geht die Organisationstheorie davon aus, dass die Voraussetzung für eine funktionierende Organisation, die Kongruenz zwischen Rechten (Kompetenzen) und Pflichten (Verantwortung) ist und bezeichnet dies als „organisatorisches Postulat“ (Hauschildt, 1969, S. 1697).

Ulrich (1969, S. 852) stellt als weiteres Kriterium für Kompetenz ein Mindestmaß an Handlungsfreiheit des Handelnden voraus und schließt somit vollständig vorbestimmte Handlungsanweisungen aus. In der Typisierung von Hill et al. (1994, S. 125) wird diese Handlungsfreiheit als Teil der Ausführungskompetenz aufgefasst, die stets einen gewissen Spielraum für die Arbeitsmethodik vorsieht. Unabhängig von der Kompetenzart können Kompetenzen sowohl auf Menschen als auch auf Softwareagenten übertragen werden.

2.1.2.2 Verantwortung und Verantwortlichkeit

Ein zentrales Kriterium der Eigenschaft eines Aufgabenträgers ist die Verantwortung für die Erfüllung der übertragenen Aufgabe. Die Organisationslehre sieht einen sehr starken Zusammenhang zwischen Verantwortung und Kompetenz, so sieht Kosiol (1969b, S. 233) eine Abhängigkeit der Verantwortung vom Freiheitsgrad, innerhalb dessen Entscheidungen zur Aufgabenerfüllung getroffen werden können. Während im Falle der Eigenverantwortung die Kongruenz von Aufgabe, Kompetenz und Verantwortung umgesetzt werden kann, sieht das Prinzip der Fremdverantwortlichkeit, z.B. bei Führungspersonen, auch das Entstehen für Handlungen untergeordneter Mitarbeiter vor (Bleicher, 1980b, S. 2284 f.). Hierbei wird bereits die Unterscheidung von Verantwortung und Verantwortlichkeit deutlich, die in der Literatur nicht durchgängig vollzogen wird. Während sich die Verantwortung auf die Durchführung der Handlungsprozesse bezieht, erstreckt sich die daraus resultierende Verantwortlichkeit auf die Verpflichtung, persönlich Rechenschaft abzulegen und umschließt somit insbesondere die juristische Sichtweise (Bleicher 1980b, S. 2283; Hill et al. 1994, S. 124; Luhmann 1976, S. 180 f.). Da eine Delegation von Aufgaben nicht von der Verantwortlichkeit befreit, ist die Verantwortlichkeit selbst nicht delegierbar (Luhmann, 1976, S. 181). Luhmann (1976, S. 188 f.) stellt gleichermaßen klar, dass jede Diskrepanz zwischen Verantwortung und Verantwortlichkeit Konfliktpotential birgt und im Außenverhältnis für Dritte nur schwer erkennbar ist.

2.1.2.3 Stelle

Thom (1992, S. 2321) versteht unter einer Stelle „die kleinste selbstständig handelnde organisatorische Einheit eines soziotechnischen Systems“. Diese wird grundsätzlich unabhängig vom Stelleninhaber (dem Aufgabenträger) gebildet (Picot 2005, S. 230; Thom 1992, S. 2321). Nach dem Grad der übertragenen Kompetenz und Verantwortlichkeit kann in Leitungsstellen und ausführende Stellen unterschieden werden (Schwarz, 1980b, S. 2117). Dabei wird nach Kosiol (1962, S. 89 ff.) und Acker (1969, S. 1577 f.) eine Stelle nur von einem Aufgabenträger besetzt: *„Die Stelle als Organisationseinheit wird zur Erfüllung ihrer Aufgabe (Stellenaufgabe) auf eine einzige menschliche Arbeitskraft bezogen“* (Kosiol, 1962, S. 90). Auch Acker (1969) teilt diese Auffassung und begründet sie wie folgt: *„Die Tatsache, daß in Organisationen häufig gleichartige Stellen mehrfach vorkommen ... führt gelegentlich zu der irrtümlichen Annahme, eine Stelle könne auch mit mehreren Personen besetzt werden. Dieser Irrtum wird dadurch begünstigt, daß für eine Reihe gleichartiger Stellen ... nur eine einzige Stellenbeschreibung existiert. Trotzdem handelt es sich um mehrere Stellen, und zwar um sogenannte Parallelstellen, die durch Mengenteilung einer Gesamtaufgabe entstanden sind“* (Acker, 1969, S. 1578). Die Begründung liegt somit in der

Unteilbarkeit der Stelle als “kleinste organisatorische Einheit“ (Acker, 1969, S. 1577). Diese Auffassung wird in der Literatur jedoch nicht durchgängig geteilt. Während Kosiol (1962, S. 97) selbst beim Zwei-Schicht-Betrieb mit identischen Aufgaben von zwei gesonderten Stellen ausgeht, sieht Thom (1992) hierin die Besetzung einer Stelle mit mehreren Aufgabenträgern: *„Eine Stelle kann einerseits an mehreren Arbeitsplätzen ausgeübt werden, andererseits ist es möglich, daß etwa im Falle der Schichtarbeit oder des Job Sharing mehrere Menschen dieselbe Stelle besetzen“* (Thom, 1992, S. 2322). Bei der Begründung greift Thom (1992, S. 2322) unter anderem auf Grochla (1966) zurück und eröffnet ebenfalls die Möglichkeit der Integration von Maschinen bei der Stellenbesetzung. So erweitert auch Grochla (1966, S. 93 ff.) den Stellenbegriff auf mehrere Aufgabenträger und unterscheidet hierbei in Mensch-Systeme, Maschine-Systeme sowie Mensch-Maschine-Systeme. Insbesondere das reine Maschine-System, bei dem die Aufgaben einer Stelle ausschließlich auf Maschinen übertragen werden, setzt voraus, dass Maschinen dieser Aufgabenträgereigenschaft gerecht werden können. Abschnitt 2.1.2.5 untersucht diese Auffassung anhand den in der Literatur gelieferten Argumenten.

2.1.2.4 Aufgabe und Aufgabenträger

Auf die Bedeutung des Aufgabenbegriffs für die Organisationstheorie wurde erstmals durch Nordsieck (1931) eingegangen. Dabei grenzt Nordsieck (1931) die Aufgabe in einer Organisation deutlich von der sich daraus ergebenden *„betrieblichen Arbeit“* ab, deren *„Träger ... Personen und Hilfsmittel sein [können]“* (Nordsieck, 1931, S. 9). Dieser dispositiven betrieblichen Arbeit hingegen ist die Aufgabe übergeordnet, welche *„in der Regel Teile [umfasst], die in ihrem Ablauf noch unbestimmt sind“* (Nordsieck, 1931, S. 10). Den Inhalt einer Aufgabe definiert Nordsieck allerdings erst später und unterteilt diesen in drei wesentliche Bestandteile: (i) das Ziel, welches in die Struktur der Organisation mit den übergeordneten organisatorischen Zielen eingebettet ist, (ii) das Ausgangsobjekt, welches durch die Aufgabe bearbeitet werden soll, sowie (iii) die Zeit, als Planungs- und Kontrollfaktor (Nordsieck und Nordsieck-Schröer, 1969, S. 191 ff.).

Kosiol (1962, S. 43) hingegen unterteilt Aufgaben ebenfalls in Komponenten, wählt hierfür jedoch die folgenden fünf: (1) Verrichtungsvorgang, (2) Gegenstand (Objekt), (3) sachliche Hilfsmittel, (4) Raum sowie (5) Zeit. Während Kosiol (1962) Raum (4) und Zeit (5) als Grundkategorien zusammenfasst, stellen Nordsieck und Nordsieck-Schröer (1969) ausschließlich die Zeit als planungsrelevante Größe heraus. Daneben unterscheiden sich die beiden Auffassungen jedoch in einem wesentlichen Aspekt: Kosiol sieht eine Aufgabe als *„Zielsetzung“* (Kosiol, 1962, S. 43), die allerdings die konkreten Ausführungsmerkmale enthält, die sich zum einen in den benötigten Hilfsmitteln (3) sowie den durchzuführenden Tätigkeiten (1) manifestieren. Nordsieck und Nordsieck-Schröer (1969) beziehen sich hingegen auf das vorangegangene Werk (Nordsieck, 1931) und gestehen der Aufgabe einen höheren Freiheitsgrad bei der Wahl der nötigen Mittel zu: Teil der Aufgabe ist damit ausschließlich das zu erreichende Ziel (i). Die zur Zielerreichung nötigen Arbeitsschritte hingegen sind Teil der *„betrieblichen Arbeit“* (siehe oben).

Mit Blick auf den Träger einer Aufgabe ist ebenfalls das Aufgabenverständnis Kosiols als „eine Zielsetzung für zweckbezogene menschliche Handlungen“ (Kosiol, 1969b, S. 233) relevant. Bereits aus dieser Definition der Aufgabe wird deutlich, dass diese nach Kosiol (1969b) nur einem Menschen zugewiesen werden kann (zur Abgrenzung zum maschinellen Aufgabenträger siehe Abschnitt 2.1.2.5). Die Übernahme einer Aufgabe ist automatisch mit der Verantwortung einer zieladäquaten Erfüllung verbunden (Kosiol, 1969b, S. 233). Zwar lassen sich die Kompetenz zur Aufgabenerfüllung und die entsprechende Verantwortung auch getrennt übertragen, jedoch sieht die Organisationslehre stets eine Übertragung einer (Teil-)Verantwortung im Zusammenhang mit der Übertragung von Kompetenzen vor (Bleicher, 1980b; Hauschildt, 1969; Kosiol, 1969b). Auf den Begriff der Verantwortung und den damit verbundenen Differenzierungsmöglichkeiten geht Abschnitt 2.1.2.1 näher ein.

Kosiol (1962, S. 44; 1969b) unterscheidet zwischen dem Arbeitsträger und dem Aufgabenträger, liefert allerdings an dieser Stelle keine Definition, sondern grenzt die Begriffe mit Hilfe deren Eigenschaften ab. Dabei sieht er zwei wesentliche Eigenschaften des Aufgabenträgers gegenüber denen des Arbeitsträgers: (1) eigenständige Entwicklung von Aktivitäten sowie (2) das Tragen von Verantwortung. Da nach Kosiol (1962, S. 44; 1969b, S. 234) nur der Mensch diese beiden Eigenschaften erfüllen kann, ist nach dieser Auffassung nur er in der Lage eine Stellung als Aufgabenträger einzunehmen. Arbeitsträger hingegen, denen die Durchführung des Arbeitsprozesses obliegt, der zur Lösung der gestellten Aufgabe führt, können sowohl Menschen als auch – in stetig wachsendem Umfang – Sachmittel sein (Kosiol, 1969a, S. 202). Hieraus ergibt sich folgende Definition des Aufgabenträgers für diese Arbeit:

Definition 1 (Aufgabenträger) *Aufgabenträger ist ein Mitglied einer Organisation, dem eine Aufgabe sowie die Verantwortung für deren Erfüllung durch die eigenständige Entwicklung von Aktivitäten übertragen wurde.*

Präzise Kurzdefinitionen finden sich in der Literatur nicht, daher basiert diese Definition auf den Argumenten von Kosiol (1962, S. 43 ff.; 1969b, S. 232 ff.), der den Begriff des *Aufgabenträgers* prägte. Die Frage nach der Fähigkeit von Maschinen, Aufgaben zu übernehmen und somit ebenfalls Aufgabenträger in einer Organisation zu werden, wurde hier bewusst außen vor gelassen und wird im nachfolgenden Abschnitt diskutiert.

2.1.2.5 Maschineller Aufgabenträger

Voraussetzung für die Zusprennung der Aufgabenträger-Eigenschaft ist die selbstständige Entwicklung von Aktivitäten sowie Verantwortung für diese (vgl. Abschnitt 2.1.2.4). Nach Kosiol (1969a,b) kann diese Eigenschaft nur von einem Menschen übernommen werden und diese Auffassung wurde bis zum Ende der 1960er Jahre auch von anderen Autoren nicht bestritten (Schwarz, 1980a, S. 218). Grochla (1966) lieferte in dieser Zeit erste Argumente, dass in bestimmten Situationen auch Sachmittel Aufgabenträger sein können. Der Begriff des maschinellen Aufgabenträgers hat sich in der Organisationstheorie nicht durchgesetzt, jedoch folgen in der Literatur auch andere Autoren der Argumentation Grochlas: Nordsieck und Nordsieck-Schröer (1969, S. 191 f.) sehen „mannigfaltige“

Möglichkeiten für Träger einer Aufgabe, insbesondere auch Maschinen. Hill et al. (1994, S. 134) sehen eine Anerkennung des maschinellen Aufgabenträgers allerdings nur dann als sinnvoll an, wenn die Maschine einen wesentlichen Anteil an der Aufgabenerfüllung leistet. In der Wirtschaftsinformatik hat sich der Begriff hingegen etabliert (Ferstl und Sinz, 2008, S. 3). Im Folgenden werden die von Kosiol (1969a,b) angeführten beiden wesentlichen Unterscheidungskriterien eines Aufgabenträgers gegenüber einem Arbeitsträger auf deren Anwendbarkeit auf Maschinen untersucht:

Selbstständige Entwicklung von Aktivitäten. Kosiol (1969b, S. 234) argumentiert, dass Maschinen keine eigenen Aktivitäten entwickeln können und deren Tätigkeit durch vom Menschen eingebrachten Programmierungsregeln beschränkt ist. Grochla (1969a, S. 253) sieht hingegen die „*Selbsttätigkeit außermenschlicher Gebilde*“ als Grundgedanke der Automation (nach Grochla (1980) synonym zur Automatisierung). Die selbsttätige Erfüllung von Aufgaben durch technische Mittel ohne menschliche Mitwirkung ist für ihn das wesentliche Ziel der Automation, was diese wiederum von der vorangegangenen Stufe der Mechanisierung abgrenzt. Während in dieser Vorstufe der Mensch Aufgaben der Bedienung, Steuerung und Überwachung durchführte und bei diesen von technischen Gegebenheiten abhängig war (beispielsweise dem Fließband, das den Arbeitsrhythmus bestimmt) wird im Rahmen der Automation versucht, den Menschen von den verbliebenen Funktionen der Aufgabenerfüllung zu ersetzen (Grochla, 1966, S. 30). Setzt man voraus, dass dieses Ziel der Automation in vollem oder weitreichendem Umfang realisiert wird, ist neben der selbstständigen Durchführung der Aufgabe auch die Initiative für deren Durchführung auf die Maschine übergegangen.

Dabei hat die Literatur nach den oben genannten Entwicklungsphasen der Mechanisierung und Automatisierung ebenfalls die Phase der Autonomisierung erkannt (Kirn und Müller-Hengstenberg, 2015, S. 2). Die aus Sicht von Grochla eingetretene Phase der Automation kann demnach unterteilt werden in die Automatisierung und Autonomisierung. Während unter Automatisierung ausschließlich die selbstständige Durchführung von Aktivitäten mit im Allgemeinen vorgegebenen Prozessabläufen verstanden wird, sieht die Autonomisierung insbesondere die oben genannte selbstständige Entwicklung von Aktivitäten und somit nicht nur deren Durchführung vor. Diese Entwicklung von Aktivitäten umfasst sowohl die Initiative zu deren Durchführung als auch die durch Lernverfahren entstehende Möglichkeit, neue Aktivitäten zu gestalten bzw. bestehende Aktivitäten kontinuierlich zu verbessern. Grochla (1966, S. 30) wendet ein, dass zur Zeit der Entstehung seines Textes nicht absehbar war, ob oder in welchem Umfang dieses Ziel erreicht wird. Die selbstständige Aufgabenerfüllung durch Maschinen ist für Grochla (1966, S. 94) ein zentrales Kriterium für die Zurechnung der Aufgabenträger-Eigenschaft. Diese hängt folglich vom Grad der Automation – bzw. im heutigen Sinne der Autonomisierung – eines Arbeitsprozesses ab, den es zu Abgrenzungszwecken zu bewerten und zu quantifizieren gilt.

Verantwortung für die Ausführung von Aktivitäten. Kosiol (1969b, S. 234) setzt für Aufgabenträger Verantwortung für ihr Handeln voraus, welches von Maschinen

nicht übernommen werden kann. Auch Schwarz (1983, S. 28) argumentiert, dass Maschinen, obwohl diesen bereits die Steuerung und Kontrolle über Arbeitsprozesse übertragen wird, in denen „*dem Mensch nur die auslösende und überwachende Bedienungsfunktion bleibt*“ (Schwarz, 1983, S. 28), durch die fehlende Verantwortung für die Aufgabenerfüllung nicht zum Aufgabenträger werden können. So können auch höchst entwickelte Maschinen keine Verantwortung übernehmen (Schwarz, 1980b, S. 2117). Sowohl Kosiol (1969b) als auch Schwarz (1980b, 1983) gehen allerdings nicht auf die in Abschnitt 2.1.2.2 betrachtete Unterscheidung zwischen Verantwortung und Verantwortlichkeit ein. Folgt man dieser gesonderten Betrachtung der Verantwortung für die Durchführung einer Aufgabe und der juristischen Verantwortlichkeit für die Folgen der Bearbeitung, kann Verantwortung sehr wohl von Maschinen übernommen werden, nicht jedoch die Verantwortlichkeit (Kirn, 1996b, S. 40).

Aus den vorangegangenen Überlegungen ergibt sich, dass durch eine Unterscheidung in Verantwortung und Verantwortlichkeit durchaus Argumente existieren, dass auch Maschinen Träger einer Aufgabe sein können, was wiederum Auswirkungen auf die Organisationsstruktur mit sich bringt (Grochla, 1969c, S. 1641 ff.). Analog zur Besetzung von Stellen mit menschlichen Aufgabenträgern, stellt sich die Frage, wie Stellen mit maschinellen Aufgabenträgern besetzt werden sollen. Thom (1992, S. 2328) erkennt zwar die Problematik bei der Stellenbesetzung mit maschinellen Aufgabenträgern, beschränkt sich bei seinen Ausführungen jedoch auf Menschen. Grochla (1972) hingegen verwendet den Begriff *Aktionsträger* synonym zu dem des Aufgabenträgers und stellt hierbei insbesondere auf die Besetzung von Stellen ab. Dabei grenzt sich Grochla (1972) bewusst vom Stellenbegriff ab, da dieser bis dato in der Literatur stets mit der Besetzung durch einen Menschen verknüpft war. Stattdessen führt er den Begriff der *Aktionseinheit* ein, der durch drei mögliche Aktionseinheiten besetzt werden kann: (i) Mensch-Aktionseinheit, (ii) Mensch-Maschine-Aktionseinheit sowie (iii) Maschine-Aktionseinheit (Grochla, 1972, S. 46). Setzt man voraus, dass Aufgabenträger auch Maschinen sein können, ist die Abgrenzung zwischen Aktionseinheiten und Stellen, wie in Abschnitt 2.1.2.3 beschrieben, jedoch nicht notwendig und die Begriffe können synonym verwendet werden.

Während sich der Begriff des maschinellen Aufgabenträgers in der Organisationstheorie nicht durchsetzen konnte, findet er in der Wirtschaftsinformatik hingegen in aktueller Literatur Verwendung. Ferstl und Sinz (2008) stellen den maschinellen Aufgabenträger dem personellen gegenüber und nutzen diese Unterscheidung insbesondere zur Definition des Grades der Automatisierung sowie der Kommunikation zwischen mehreren Aufgabenträgern. Aufgaben sind demnach (i) vollautomatisiert bei ausschließlich maschineller Durchführung, (ii) teilautomatisiert bei Verrichtung durch Menschen und Maschinen sowie (iii) nicht automatisiert bei vollständiger Durchführung durch personelle Aufgabenträger (Ferstl und Sinz, 1984, S. 19 f.). Diese Einteilung deckt sich mit den von Grochla (1972, S. 46) unterschiedenen Arten von Aktionseinheiten, jedoch wird anhand der von Ferstl und Sinz (1984, S. 18) gewählten Beispiele deutlich, dass eine Diskrepanz beim Begriff des Aufgabenträgers besteht: Ferstl und Sinz (1984, S. 18) nennen ebenfalls rein mechanische Systeme wie Schreibmaschinen und Registrierkassen, welche von Kosiol (1969b, S.234)

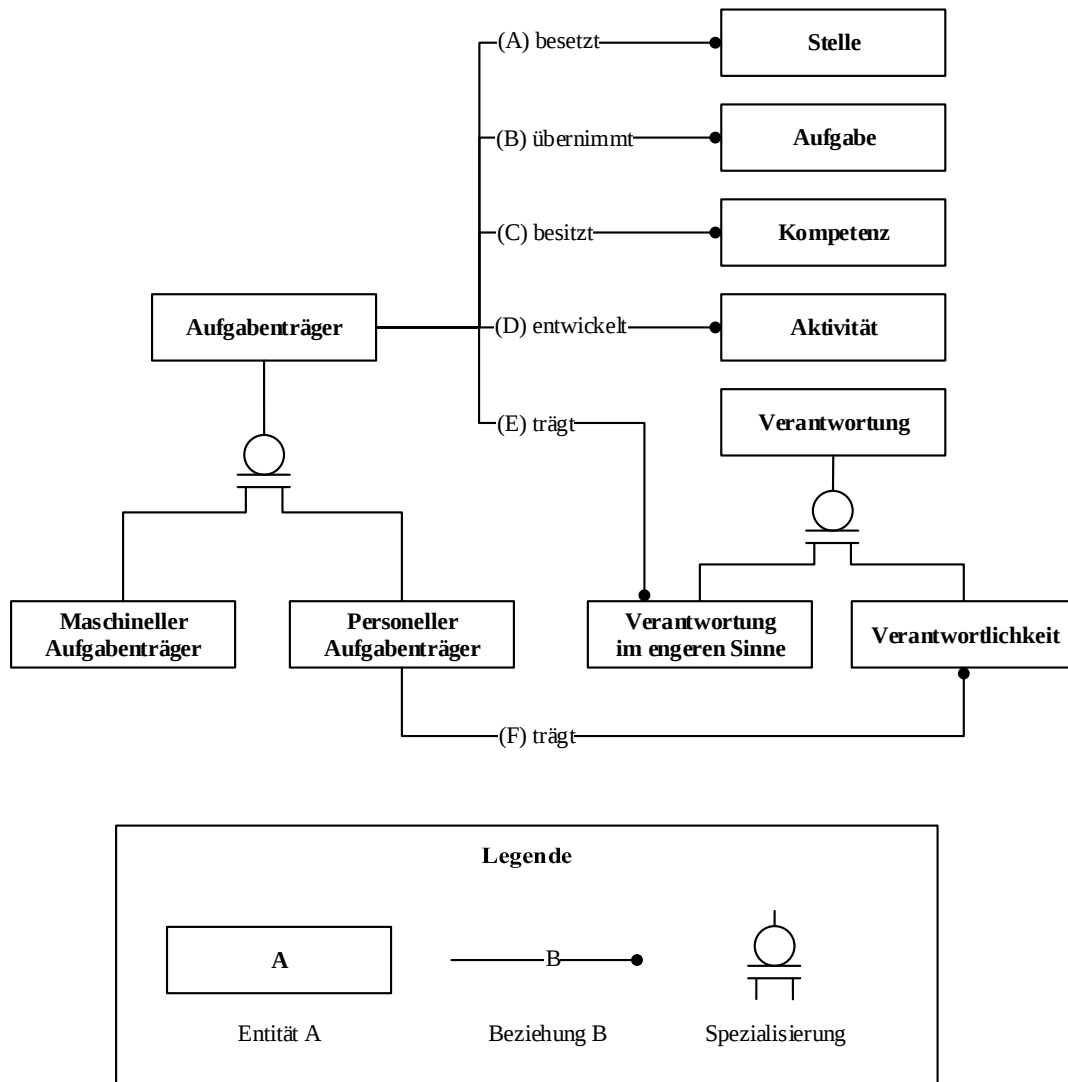


Abbildung 2.1: Personelle und maschinelle Aufgabenträger als ER-Diagramm

nicht als Aufgaben-, sondern als Arbeitsträger bezeichnet werden und auch nicht die von Grochla (1966, 1972) geforderte Selbsttätigkeit an den Tag legen.

Abbildung 2.1 fasst die Erkenntnisse dieses Abschnitts als Entity-Relationship-Diagramm in IDEF1X³ Notation zusammen. Sowohl maschinelle als auch personelle Aufgabenträger (A) besetzen Stellen, (B) übernehmen Aufgaben, (C) besitzen Kompetenzen und (D) entwickeln Aktivitäten. Letztere Eigenschaft wurde in der Literatur stets in Frage gestellt (siehe oben). Die zunehmende Autonomisierung ermöglicht jedoch Maschinen auch selbstständig Aktivitäten zu initiieren, mit Hilfe von Lernverfahren neue Aktivitäten zu gestalten und bestehende weiterzuentwickeln. Die Verantwortung lässt sich als die Verantwortung im engeren Sinne (Durchführungsverantwortung) und die Verantwortlichkeit im juristischen Sinne unterteilen. Hierbei treten Unterschiede zwischen maschinellen Aufgabenträgern, denen aus-

³ Der IDEF1X Standard wurde 1993 durch die *Federal Information Processing Standards Publications* (FIPS PUBS) für die Beschreibung von semantischen Datenmodellen definiert. In dieser Arbeit wird die Notation der Entitäten angepasst, da keine Darstellung von Attributen notwendig ist (siehe Legende in Abbildung 2.1).

schließlich die Durchführungsverantwortung (E) übertragen werden kann, und personellen Aufgabenträgern zu Tage, die zusätzlich die juristische Verantwortlichkeit (F) tragen.

2.1.3 Softwareagenten als maschinelle Aufgabenträger

Abschnitt 2.1.1 hat die unterschiedlichen in der Literatur auftretenden Definitionen des Begriffs Softwareagent aufgezeigt und kritisch hinterfragt. Obwohl Wooldridge (2009, S. 10) davon ausgeht, dass Softwareagenten im Auftrag und im Interesse von Menschen handeln und somit diese auch bei der Durchführung von Aufgaben in Organisationen unterstützen sollen, wurde bei der Definition des Begriffs Softwareagent bisher kein Bezug zur Diskussion um den Begriff des maschinellen Aufgabenträgers aus der deutschen Organisationstheorie geschaffen. Abschnitt 2.1.2 hat die verschiedenen Argumentationen dieser Diskussion sowie Kriterien aufgezeigt, unter welchen Bedingungen auch maschinelle Gebilde als Aufgabenträger in einer Organisation gelten können. Zentrale Voraussetzung zur Abgrenzung von Maschinen, die keine Aufgaben tragen, sind die selbstständige Entwicklung von Aktivitäten sowie die Verantwortung für deren Ausführung.

Kirn (1996b, S. 36 ff.) untersucht bereits die Übertragbarkeit des Konzepts des maschinellen Aufgabenträgers auf Softwareagenten, wie sie im Bereich der KI Verwendung finden und kommt zu dem Fazit, dass *„der Begriff des maschinellen Aufgabenträgers ... , wie ihn die betriebswirtschaftliche Organisationslehre geprägt hat, nicht hinreichend [ist], um die Herausforderungen abzudecken, die sich der Wirtschaftsinformatik durch das Ziel des produktiven Einsatzes kooperativ-intelligenter Agenten in betrieblichen Umgebungen stellen“* (Kirn, 1996b, S. 51). Die Rechtfertigung dieser Unterscheidung führt auf, dass intelligente Softwareagenten im Gegensatz zu maschinellen Aufgabenträgern der Organisationstheorie über individuelle Ressourcen verfügen können, die bei der Zuordnung zu einer Stelle zukünftige Handlungsmöglichkeiten einschränken (Kirn, 1996b, S. 46 f.). Dieser Argumentation kann jedoch nicht uneingeschränkt gefolgt werden: Zum einen verfügen durch die permanent steigende Ausstattung von Maschinen mit Informationstechnologie auch als nicht-intelligent zu bezeichnende Maschinen häufig über individuelle Ressourcen, unabhängig von der Bindung an eine bestimmte Stelle. Zum anderen sieht Grochla (1966) die Automation bzw. Automatisierung – die er einem Organisationsverständnis mit rein personellen Aufgabenträgern entgegensetzt – als deutlich weitergehend, als es die technischen Voraussetzungen sowohl bei der Entstehung seines als auch dieses Textes zulassen: Dabei wird versucht *„den Menschen in allen ihm bis dahin verbliebenen Funktionen der Aufgabenerfüllung zu ersetzen“* (Grochla, 1966, S. 30), so dass maschinelle Aufgabenträger auch sämtliche Kompetenzen in einer Organisation übernehmen müssen. Greift man diesen Aspekt auf, kann hingegen weiter in Automatisierung und Autonomisierung unterteilt werden (vgl. auch Abschnitt 2.1.2.5). Während die Automatisierung die vollständig automatische Ausführung nach vorgegebenen Regeln und Zielen vorsieht, erfordert die bereits von Grochla (1966) geforderte Ersetzung des Menschen auch die Generierung und Anpassung von Zielen und Regeln sowie der damit verbundenen Ressourcenzuteilung. Die Autonomisierung verfolgt ebendieses Ziel und fordert insbesondere anpassungsfähiges und lernendes Verhalten dieser Maschinen.⁴

⁴ zur Verknüpfung von Autonomie und Lernfähigkeit vgl. auch Abschnitt 2.1.1.1

Während aus Sicht der Multiagenten-Organisation Mitglieder als Ressource betrachtet werden können (vgl. Abschnitt 2.3.2.1.5 zum ressourcenbasierten Ansatz), bietet aus Sicht des Softwareagenten dieser der Multiagenten-Organisation hingegen Dienste an und benötigt für deren Bereitstellung Ressourcen. Softwareagenten besitzen nur einen beschränkten⁵ Zugriff auf Problemlösungskapazitäten (Bond und Gasser, 1988, S. 9), wozu ihnen mindestens der Zugriff auf eine Ressource „Rechenkapazität“ gesichert sein muss (Wooldridge, 2009, S. 66). Im Rahmen der VKI werden als Ressourcen eines Softwareagenten insbesondere – jedoch nicht ausschließlich – Hardwareressourcen verstanden: „*Some typical resources that are allocatable in DAI systems include processing time*“ (Bond und Gasser, 1988, S. 15). Weiss (1999) definiert Ressourcen im Glossar als „*Physical resources (processor, memory, etc.) and logical resources (channels, threads) that are used in the course of a computation*“ (Weiss, 1999, S. 602). Auch hier stehen die für Berechnungen notwendigen Ressourcen im Vordergrund.

Die dem Softwareagenten zur Verfügung stehenden Ressourcen lassen sich dabei in Bezug auf ihre Teilbarkeit unterscheiden (Malone und Crowston, 1994, S. 92 f., 112). Zwar ist es möglich, dass Softwareagenten teilbare Ressourcen mit anderen Softwareagenten teilen, diese Arbeit beschränkt sich allerdings auf Ressourcen, auf die der potentiell beitretende Softwareagent exklusiven Zugriff besitzt. Somit werden in dieser Arbeit durchaus neben nicht-teilbaren auch teilbare Ressourcen betrachtet, die jedoch zur Sicherstellung der Planbarkeit der Mitgliedschaft mit den daraus entstehenden Verpflichtungen zur Bereitstellung von Diensten einem exklusiven Zugriff durch den beitretenden Softwareagenten unterliegen. Über die Dienstbereitstellung an mehrere Multiagenten-Organisationen nutzen indirekt mehrere Akteure die Ressourcen des Softwareagenten, die Verwaltung dieser Aufteilung obliegt jedoch dem Softwareagenten selbst. Teilbare Ressourcen lassen sich weiter dahingehend unterscheiden, ob sich die jeweilige Nutzbarkeit bei einer Teilung verringert oder nicht: Während Hardwareressourcen bei einer Teilung jedem Teil nur in einem geringeren Umfang zur Verfügung stehen, lassen sich beispielsweise Informationen beliebig und ohne Einschränkung der Nutzbarkeit teilen. Zwar sind auch letztere für die Beitrittsentscheidung relevant, da Abhängigkeiten in Bezug auf die Dienstbereitstellung bestehen können, bilden aufgrund der uneingeschränkten Nutzbarkeit im Falle einer Teilung allerdings nur eine notwendige Voraussetzung. Kritisch bei der Beitrittsentscheidung – und somit im Folgenden betrachtet – sind hingegen diejenigen Ressourcen, deren Nutzbarkeit sich bei einer Teilung verringert und folglich die Möglichkeit Dienste bereitstellen zu können einschränken.

Eine Bezeichnung von Softwareagenten als maschinelle Aufgabenträger bei einem Einsatz in Organisationen ist aus den oben angeführten Gründen nicht grundsätzlich zu verneinen, die Heranziehung des Aufgabenträgerbegriffs zur Anforderungsdefinition an Softwareagenten bedarf hingegen einer weitergehenden Überprüfung der Kriterien (Kirn, 1996b, S. 48). Die nachfolgenden Abschnitte untersuchen daher die beiden wesentlichen Kriterien für Aufgabenträger (vgl. Abschnitt 2.1.2.4): (i) die selbstständige Entwicklung von Aktivitäten sowie (ii) das Tragen der Verantwortung für die Ausführung dieser.

⁵ vgl. auch Abschnitt 2.3.2.1.4 zur begrenzten Rationalität

Selbstständige Entwicklung von Aktivitäten. Kosiol (1969b) begründet seine Abgrenzung zwischen Menschen als Aufgabenträger und Maschinen als Arbeitsträger unter anderem dadurch, dass „*die Tätigkeit der Maschine als Arbeitsträger ... eine vom Menschen abgeleitete [sei], die durch die Konstruktions- und Programmierungs-(Ablauf-)Regeln vom Menschen in die Maschine hineingelegt wird*“ (Kosiol, 1969b, S. 234). Er adressiert somit die *Entwicklung* von Aktivitäten, die es ermöglicht, aus einer vorgegebenen Menge an Handlungsalternativen auszubrechen und neue zu generieren. Im Umkehrschluss ist es nach Kosiol folglich möglich, dass Maschinen bzw. Softwareagenten als Aufgabenträger bezeichnet werden könnten, wenn ihr Verhalten nicht vom Menschen vorgegeben ist. Hier lässt sich bereits der Bezug zu den in Abschnitt 2.1.1.1 untersuchten Kriterien *Autonomie* und *Lernfähigkeit* erkennen. Sind Softwareagenten in der Lage ihr Verhalten anhand von Erfahrungen auf neue Gegebenheiten anzupassen (Lernfähigkeit), ist dieses Verhalten nicht mehr vom Menschen vorgegeben, sondern wurde durch den Softwareagenten selbst neu- oder weiterentwickelt.

Gleichzeitig muss diese Entwicklung *selbstständig* durchgeführt werden und setzt somit eine gewisse Eigeninitiative des Aufgabenträgers voraus. In der Literatur zu Softwareagenten werden hierfür oftmals die Kriterien *Proaktivität* sowie *Zielorientierung* angeführt. Damit unmittelbar verknüpft hingegen ist das Merkmal der Zielorientierung (vgl. Abschnitt 2.1.1.4): Ein Softwareagent soll sein Handeln auf die Erfüllung von vorgegebenen oder selbst entwickelten Zielen ausrichten und handelt aktiv, um diese Ziele zu erreichen. Dieses Merkmal lässt sich übertragen auf Softwareagenten als Aufgabenträger, die Kosiol (1969b, S. 234) ebenfalls als „aktiv handelnd“ beschreibt.

Zentral ist somit, inwiefern Softwareagenten eigene (Teil-)Ziele aus den vorgegebenen oder selbst entwickelten Zielen ableiten können. Zwar können auch Multiagentensysteme, die ausschließlich aus rein reaktiven Agenten bestehen, (Teil-)Ziele entwickeln, die einzelnen Agenten jedoch nicht (vgl. Abschnitt 2.1.1.3). Insbesondere ist es ihnen nicht möglich, ein Teilziel des Beitritts zu einer Multiagenten-Organisation zu entwickeln, weshalb Beitrittsentscheidungen für rein reaktive Agenten im Allgemeinen keine Problemstellung darstellen. Für Softwareagenten im Sinne dieser Arbeit entsteht somit neben der Lernfähigkeit über die Entwicklung eigener (Teil-)Ziele eine weitere „Entfernung“ vom Entwickler des Softwareagenten (vgl. Abschnitt 2.1.1.4).

Verantwortung für die Ausführung von Aktivitäten. Wie in Abschnitt 2.1.2.5 bereits herausgestellt wurde, kann Verantwortlichkeit nicht von Maschinen übernommen werden, Verantwortung hingegen schon.⁶ Dies gilt entsprechend für Softwareagenten (Kirn, 1996b, S. 40). Kommen Softwareagenten in Organisationen zum Einsatz, können diese folglich ausschließlich Verantwortung für die Durchführung einer Aufgabe übernehmen, eine juristische Verantwortlichkeit kommt nur im Rahmen einer Fremdverantwortlichkeit in Betracht. Hierfür muss sich der Softwareagent im Eigentum einer natürlichen oder juristischen Person befinden, auf deren Rechnung der Softwareagent handelt und die hierfür betriebliche Ressourcen zur Verfügung stellt (Kirn und Müller-Hengstenberg, 2015, S. 132 ff.). Inwiefern eine juristische

⁶ zur Unterscheidung von Verantwortung und Verantwortlichkeit vgl. Abschnitt 2.1.2.2

Haftung des Eigentümers und somit eine Fremdverantwortlichkeit in Frage kommt, ist von der Gesetzeslage am Ort des Einsatzes, des Inverkehrbringens oder des Eigentümerwohnorts abhängig und insbesondere zur Zeit der Fertigstellung dieser Arbeit für die Bundesrepublik Deutschland nicht abschließend geklärt. Bereits durch diese juristischen Unklarheiten wird deutlich, dass eine Definition von Softwareagenten, die unabhängig von Rechtsräumen gilt, nicht an der Verantwortlichkeit, sondern nur an der (Durchführungs-)Verantwortung festgemacht werden kann.

Dem Begriff des (maschinellen) Aufgabenträgers könnte somit das Kriterium der Verantwortung zur Definition von Softwareagenten entnommen werden. Ein Softwareagent, der Mitglied einer Organisation ist und dort als Aufgabenträger dienen soll, müsste demnach Verantwortung für die Durchführung der Aufgabe übernehmen. Dieses Kriterium ist allerdings auf die Mitgliedschaft in einer Organisation beschränkt. Dies würde die Bezeichnung als Softwareagent von der Mitgliedschaft selbst abhängig und eine Betrachtung von Softwareagenten, die in keiner Organisation Mitglied sind, unmöglich machen. Insbesondere in Hinblick auf Fragestellungen der Beitrittsentscheidung ist diese Definition nicht zielführend. Eine Aufgabe kann jedoch nicht nur durch die Organisation selbst, sondern insbesondere durch den Eigentümer des Softwareagenten gestellt werden. Durch die bei der selbstständigen Entwicklung von Aktivitäten angesprochene Zielorientierung wird dem Softwareagenten bereits bei Instanziierung eine Aufgabe – die Erfüllung eben jenes Ziels – mitgegeben. Dabei kann es sich beispielsweise auch um eine Nutzenfunktion handeln, die es zu maximieren gilt. Durch die Zielorientierung ist der Softwareagent somit für die Durchführung dieser Aufgabe verantwortlich.

Aus der vorangegangenen Betrachtung der Anforderungen an maschinelle Aufgabenträger wird deutlich, dass insbesondere die *selbstständige Entwicklung von Aktivitäten* eine Verknüpfung mit den Kriterien der MAS-Literatur für Softwareagenten aus Abschnitt 2.1.1 zulässt. Hieraus lässt sich folgende Definition für Softwareagenten ableiten:

Definition 2 (Softwareagent) *Ein Softwareagent ist ein Softwaresystem, welches zielorientiert agiert, seine Wissensbasis einschließlich der darin enthaltenen Ziele durch Lernverfahren erweitern kann und somit über die Zeit eine gewisse Autonomie gegenüber seinem Entwickler erlangt. Softwareagenten können sich zu Gruppen zusammenschließen und Träger von Aufgaben in Organisationen werden.*

Aus dieser Definition wird deutlich, dass es sich bei Softwareagenten im Rahmen dieser Arbeit um Software handelt, die zwei wesentliche Eigenschaften besitzt und sich mit Hilfe derer von einzelnen Objekten einer objektorientierten Programmiersprache unterscheidet: (i) Zielorientierung und (ii) Lernfähigkeit.⁷ Rein reaktive Agenten fallen somit nicht unter die Definition von Softwareagenten dieser Arbeit (vgl. Abschnitt 2.1.1.3). Während Software durch das EVA-Prinzip bereits in der Lage ist über Ein- und Ausgabe mit der Umwelt zu

⁷ zur Abgrenzung von Softwareagenten zu Objekten einer objektorientierten Programmiersprache vgl. auch Franklin und Graesser (1997)

interagieren, bedeutet *Zielorientierung*, dass Softwareagenten ihr Handeln auf die Erfüllung von Zielen hin ausrichten. Diese können dem Softwareagenten sowohl vom Entwickler vorgegeben, als auch vom Softwareagenten selbst (weiter)entwickelt werden. Um diese Ziele zu erreichen, können Softwareagenten vorgegebene Verhaltensmuster anpassen oder neue entwickeln. Die *Lernfähigkeit* ermöglicht es Softwareagenten ein eigenes Verhalten zu entwickeln und somit ein gewisses Maß an Autonomie gegenüber dem Entwickler zu erreichen – insbesondere gegenüber dessen eingeschränkter Fähigkeit alle möglichen Umweltzustände bei der Programmierung vorherzusehen. Software, die die Kriterien der Definition 2 erfüllt und die somit im Rahmen dieser Arbeit als Softwareagent bezeichnet wird, ermöglicht die Übernahme von Aufgaben in Organisationen durch diese und somit die Erfüllung der Kriterien der Aufgabenträger-Eigenschaft im Sinne der organisationstheoretischen Betriebswirtschaftslehre.

2.2 Organisationsmodelle der verteilten künstlichen Intelligenz

Wie bereits zu Beginn dieses Kapitel erläutert, haben sich verschiedene Forschungsrichtungen innerhalb der VKI herausgebildet. Betrachtet man die Arbeit der einzelnen Forschungsrichtungen genauer, lassen sich verschiedene Typen von Organisationsmodellen unterscheiden. Diese unterteilen sich entsprechend den Forschungsrichtungen ebenfalls in *verteilte Problemlöser* (Abschnitt 2.2.2) und *Multiagentensysteme* (Abschnitt 2.2.3). Zunächst ist es jedoch notwendig, Grundmengen von Softwareagenten zu identifizieren, aus denen sich Gruppen von Softwareagenten zu einer Kooperation zusammenfinden können.

2.2.1 Grundmengen von Softwareagenten

Dieser Abschnitt analysiert die beiden Begriffe *Bekanntschaften* und *Multiagentengesellschaft*. Beide Begriffe beschreiben eine Menge von Softwareagenten, die als Grundlage zur Bildung von verschiedensten organisatorischen Gebilden dienen können.

2.2.1.1 Bekantschaften

Der Begriff der *Bekantschaften* (engl. „*acquaintances*“) bezieht sich stets auf die mikroperspektivische Sicht eines einzelnen Softwareagenten (Kirn, 1996a, S. 155). Bond und Gasser definieren Bekantschaften als „*the other actors known to the actor*“ (Bond und Gasser, 1988, S. 8), beschränken sich somit auf das Wissen eines Softwareagenten über die Existenz anderer.

Wooldridge et al. (2000, S. 297 f.) hingegen definieren bei der GAIA-Spezifikation ein *acquaintance model*, gehen dabei jedoch über das ausschließliche Wissen über die Existenz eines anderen Softwareagenten hinaus und fordern einen bestehenden Kommunikationsweg. Das Modell lässt offen, in welcher Form Kommunikation stattfindet, fordert jedoch, dass Kommunikation technisch möglich ist. Dabei handelt es sich zunächst um unidirektionale Kommunikationswege, die sich erst bei einer entsprechenden Konstellation zweier

Softwareagenten zu bidirektionalen Kommunikationswegen entwickeln können. Wooldridge et al. (2000) lassen dabei offen, ob sich im *acquaintance model* auch mehrstufige Kommunikationswege abbilden lassen.

Das Wissen um die Existenz eines anderen Softwareagenten alleine reicht demnach für die exakte Definition von Bekanntschaften nicht aus: Eine Möglichkeit der Interaktion muss gegeben sein, die sich jedoch auf die technische Möglichkeit beschränkt: Beispielsweise das Wissen über die IP-Adresse oder bei anderer als IP-basierter Kommunikation eine entsprechende Adressierungsmöglichkeit des Interaktionspartners sowie die technische Möglichkeit beider Akteure diese Kommunikationsschnittstelle zu nutzen. Somit hat jeder Softwareagent eine Menge an *Bekanntschaften*, mit denen er beispielsweise über Kooperationsbeziehungen verhandeln kann. Dabei ist neben dem direkten auch der indirekte Kommunikationsweg über andere Softwareagenten oder andere Zwischenstationen möglich, da das Ergebnis einer Kommunikation zwischen diesen zwei Softwareagenten sich nicht unterscheidet.

Setzt man voraus, dass sich die Mitgliedschaft eines Softwareagenten in einem Zusammenschluss nicht nur inherent bildet, sondern dieser sich für einen Beitritt explizit entscheidet, muss ihm dieser Zusammenschluss anderer Softwareagenten ebenfalls bekannt sein. Diese Zusammenschlüsse von Softwareagenten wie sie in den nachfolgenden Abschnitten betrachtet werden, können somit ebenfalls Teil der Bekanntschaften eines Softwareagent sein. Hieraus lässt sich die folgende Definition von Bekanntschaften ableiten:

Definition 3 (Bekanntschaften) *Bekanntschaften eines Softwareagenten A sind Mengen von anderen Softwareagenten und Zusammenschlüsse von Softwareagenten, von deren Existenz A weiß und mit denen A potentiell direkt oder indirekt interagieren kann.*

Diese Definition stellt in den Vordergrund, dass Interaktionen des Softwareagenten mit seinen Bekanntschaften potentiell möglich sein müssen. Dabei ist die technische Fähigkeit zur Kommunikation ausreichend und es muss keine tatsächliche Kommunikation stattfinden.

2.2.1.2 Multiagentengesellschaft

Der Begriff der *Multiagentengesellschaft* (engl. „*society of agents*“ oder „*agent society*“) wird wie der oben betrachtete Begriff des Softwareagenten in der Literatur ebenso unterschiedlich verwendet und es findet sich keine einheitlich anerkannte Definition. Zwar verzichten beispielsweise Huhns und Stephens (1999) auf eine exakte Definition, verwenden den Begriff der Multiagentengesellschaft allerdings als Überbegriff für Zusammenschlüsse einer Gruppe von Softwareagenten, in denen die einzelnen Softwareagenten Rollen besetzen (Huhns und Stephens, 1999, S. 112). Dieser Auffassung folgen Weigand und Dignum (2003, S. 222), die Multiagentengesellschaft als Synonym zum Begriff der Multiagenten-Organisation verwenden, in der ebenfalls Normen und Regeln existieren.

Kirn (1996a) definiert Multiagentengesellschaften als „eine Menge von Agenten ..., die – über ein- und mehrstufige Kommunikationsbeziehungen hinweg – in einem elektronischen Netzwerk Verbindungen unterhalten und diese grundsätzlich auch für die Durchführung

kooperativer Problemlösungsprozesse oder netzwerkweite Koordinationsaufgaben zur Verfügung stellen bzw. einsetzen können“ (Kirn, 1996a, S. 155). Auch diese Definition sieht eine Multiagentengesellschaft als einen engeren Verbund als den der Bekanntschaften, da erstere – abhängig von der Ausprägung der situativen Faktoren – dynamisch durch Rollen differenziert werden können und tatsächlich Kommunikationsbeziehungen unterhalten (Kirn, 1996a, S. 156).

Auch Müller (1993, S. 12 f.) sieht Multiagentengesellschaften als Überbegriff von verschiedensten Ausprägungsformen einer kooperierenden Menge an Softwareagenten, bei denen Rollendifferenzierungen ausgeprägt werden können, allerdings keine Voraussetzung sind. Dieses Verständnis des Begriffs sieht Multiagentengesellschaften als Obermenge verschiedener Ausprägungen, insbesondere der in den folgenden Abschnitten behandelten Multiagentensystemen (vgl. Abschnitt 2.2.3) und Multiagenten-Organisationen (vgl. Abschnitt 2.3.1.2). Im Gegensatz zu Bekanntschaften wird auch von Müller (1993) vorausgesetzt, dass zwischen Softwareagenten einer Multiagentengesellschaft Interaktion (beispielsweise Kommunikation oder Kooperation) stattfindet und die Bindung somit über den Status der *Bekanntschaft* und deren nur potentiell vorhandenen Möglichkeit der Kommunikation hinausgeht. Hieraus lässt sich für diese Arbeit eine Multiagentengesellschaft wie folgt definieren:

Definition 4 (Multiagentengesellschaft) *Eine Multiagentengesellschaft ist eine Menge von Softwareagenten, die direkt oder indirekt miteinander interagieren können.*

Multiagentengesellschaften stellen somit eine Obermenge an interagierenden Softwareagenten dar, die noch nichts über den Umfang oder die Dauerhaftigkeit dieser Interaktion aussagt. Hieraus werden in den nachfolgenden Abschnitten verschiedene Teilmengen näher betrachtet. Eine Multiagentengesellschaft ist hingegen keine Teilmenge der Bekanntschaften, da eine Multiagentengesellschaft durchaus Softwareagenten umfassen kann, die jeweils nicht Teil der Menge der Bekanntschaften des anderen Softwareagenten sind. Eine Interaktion kann demnach auch indirekt über „Dritte“ (andere Softwareagenten) realisiert sein.

2.2.2 Verteilte Problemlöser

Der Begriff der *verteilten Problemlöser* findet in der Literatur in zweierlei Hinsicht Verwendung: Zum einen wird er wie oben aufgezeigt zur Abgrenzung von verteilten Problemlösern zu Multiagentensystemen verwendet. Diese Unterscheidung wurde insbesondere durch Bond und Gasser (1988, S. 3) eingeführt und hat sich in der Literatur verfestigt. Zum anderen wird der Begriff verteilte Problemlöser ebenfalls verwendet, um eine bestimmte Klasse von verteilten Problemlösern zu beschreiben, die sich insbesondere von Blackboard-basierten Ansätzen sowie *kooperativen* verteilten Problemlösern abgrenzen (Kirn, 1996a, S. 116). Diese Unterscheidung ist insbesondere auf die frühen Entwicklungen in den 1980er Jahren im Bereich der VKI zurückzuführen, bei denen es zu parallelen Entwicklungen in verschiedenen Forschungsgruppen sowie regionale Unterschiede als auch eingeschränkte Kommunikationsmöglichkeiten gab. Insbesondere zählte zu diesen Kommunikationsmöglichkeiten die Mailingliste *DAI List Digest*, deren Ergebnisse in Kirn (1996a, S. 53 ff.)

eingehend analysiert werden. Die aufeinander aufbauende Entwicklung brachte die in den folgenden Abschnitten betrachteten Organisationsmodelle hervor, welche eine Abgrenzung von kooperativen verteilten Problemlösern von nicht-kooperativen beinhaltet, um die darin enthaltenen unterschiedlichen Konzepte zu untersuchen. Im Folgenden wird daher weiterhin der Begriff verteilte Problemlöser zur Abgrenzung zu Multiagentensystemen und bei verteilten Problemlösern, die sich von Blackboard-Systemen und kooperativen verteilten Problemlösern abgrenzen, der Term *nicht-kooperative verteilte Problemlöser* verwendet.⁸

Neben spezifischen Merkmalen, die in den folgenden Abschnitten aufgegriffen werden, besitzen alle drei Ausprägungsformen von verteilten Problemlösern auch gemeinsame Merkmale. Dabei kann unterschieden werden, welche Informationen bereits bei der Konzipierung und Implementierung eines verteilten Problemlösers dem Entwickler bekannt sein müssen und welche Entscheidungen ad-hoc zur Laufzeit getroffen werden. Im Gegensatz zu Multiagentensystemen sind bereits beim Entwurf eines verteilten Problemlösers Informationen über das zu lösende Problem bekannt:

Problemtyp. Der zu lösende Problemtyp selbst ist bekannt und wird dem verteilten Problemlöser vorgegeben. Ein Problemtyp kann beispielsweise das Problem des Handelsreisenden sein, welches es zu lösen gilt. Mit dem Problemtyp verbunden sind somit auch die vorgegebenen Nebenbedingungen (z.B. Spielregeln). Aus dieser Information ergeben sich die nachfolgenden detaillierten zur Verfügung stehenden Informationen. Aus der Vorgabe des zu lösenden Problems durch eine externe Instanz, ergibt sich, dass die beteiligten Softwareagenten bzw. Knoten des verteilten Problemlösers keinen Einfluss auf die Problemstellung selbst, sondern höchstens auf die zu wählende Lösungsalternative haben.

Problemkomplexität. Durch das Problem ist im Allgemeinen bereits bei der Entwicklung die Problemkomplexität bekannt. Hieraus ergibt sich beispielsweise, inwiefern das Problem analytisch gelöst werden kann oder ob die Anwendung von Heuristiken nötig ist. Um diese Entscheidung treffen zu können, sind insbesondere Informationen über den zur Lösung des Problems zur Verfügung stehenden Zeithorizont sowie die verfügbaren Ressourcen notwendig.

Problemstruktur und Lösungsansätze. Neben der Problemkomplexität sind durch das Wissen über das Problem in vielen Fällen Informationen über die Problemstruktur bekannt. Die Problemstruktur wiederum lässt Rückschlüsse auf mögliche Lösungsansätze zu. Insbesondere wird durch die Problemstruktur festgelegt, inwiefern sich das Problem in Teilprobleme unterteilen lässt. Grundsätzlich sind nur Probleme zur Lösung durch verteilte Problemlöser geeignet, die sich in einer irgendwie gearbeteten Form in Teilprobleme unterteilen lassen. Zwar ist es auch denkbar, dass ein Problem an mehrere Knoten zur Lösung übergeben wird und die durchschnittliche Lösungsgeschwindigkeit durch Verteilung der Wahrscheinlichkeit auf mehrere Knoten erhöht wird, jedoch wird diese Art der parallelen künstlichen Intelligenz zugeordnet,

⁸ zur weiteren Erläuterung vgl. Abschnitt 2.2.2.2

welche ausschließlich der geschwindigkeitsorientierten parallelen Verarbeitung dient, ohne konzeptionelle Vorteile zu bieten (Bond und Gasser, 1988, S. 3).

Benötigte Fähigkeiten. Aus der Problemstruktur und den Lösungsansätzen ergeben sich die benötigten Fähigkeiten der beteiligten Softwareagenten bzw. Knoten. Dabei können zur Lösung eines Problems unterschiedliche Fähigkeiten benötigt werden. Die notwendigen Fähigkeiten können insbesondere die Aufteilung des Problems in Teilprobleme, als auch die Lösung der Teilprobleme betreffen, so dass verschiedene Typen von Softwareagenten bzw. Knoten in den Problemlösungsprozess involviert sein können.

Verteilungsregeln. Die Zuordnung der einzelnen Teilprobleme zu den Softwareagenten bzw. Knoten, insbesondere in Abhängigkeit von deren Fähigkeiten, kann erst zur Laufzeit entschieden werden. Die Verteilungsregeln, nach welchen sich die Zuordnung richtet, hingegen, wird im Allgemeinen bereits bei der Konzipierung eines verteilten Problemlösers getroffen. Hierbei kann geregelt werden, welche Zuordnungsverfahren Verwendung finden sollen oder wie etwaige Verhandlungen zwischen verschiedenen Softwareagenten bzw. Knoten ablaufen.

Die vorangegangenen Informationen sind bei verteilten Problemlösern bereits bei der Konzipierung und Implementierung bekannt. Dabei lassen sich aus der Problembeschreibung bzw. dem Problemtyp dessen Komplexität und Struktur und hieraus wiederum mögliche Lösungsansätze und benötigte Fähigkeiten ableiten. Neben diesen sind zur Problemlösung jedoch noch weitere Informationen notwendig, die dem verteilten Problemlöser erst zur Laufzeit zur Verfügung stehen und aus denen sich Entscheidungsalternativen ergeben, die ad-hoc beantwortet werden:

Probleminstanz. Während der Problemtyp (z.B. „Spracherkennung“) bereits bei Konzipierung und Implementierung des verteilten Problemlösers bekannt ist, wird die Probleminstanz erst zur Laufzeit durch externe Vorgaben festgelegt. Die Lösung einer konkreten Probleminstanz wird dem verteilten Problemlöser als Ziel vorgegeben, die beteiligten Softwareagenten bzw. Knoten haben hierauf keinen Einfluss.

Zerlegung des Problems. Zur Entwicklungszeit des verteilten Problemlösers sind bereits die Problemstruktur und die damit verbundenen Lösungsansätze bekannt. Diese bestimmen im Allgemeinen, in welcher Art und Weise das Problem zerlegt werden kann. Die Zerlegung selbst hingegen ist abhängig von der konkreten Probleminstanz, die erst zur Laufzeit bekannt ist. Demnach kann die Zerlegung des Problems ebenfalls erst zur Laufzeit erfolgen. Dabei muss festgelegt sein, welche Knoten die Fähigkeit besitzen, das Problem zu zerlegen und welche Softwareagenten bzw. Knoten diese Zerlegung durchführen. Dies kann entweder bereits zur Zeit der Konzipierung erfolgen oder durch die beteiligten Softwareagenten bzw. Knoten zur Laufzeit ausgehandelt werden. Die Zerlegung selbst ist dabei stets abhängig von der gewählten Form der Implementierung (z.B. Sprache) und ist durch mögliche Abhängigkeiten zwischen den Teilproblemen nicht in jedem Fall redundanzfrei möglich (vgl. Abschnitte 2.2.2.2 und 2.2.2.3).

Zuweisung der Teilprobleme. Die Teilprobleme, die sich durch die Zerlegung des Problems ergeben, werden zur Laufzeit an die Mitglieder des verteilten Problemlösers verteilt. Die Verteilungslogik wird im Allgemeinen bereits bei der Konzipierung festgelegt, die tatsächliche Zuweisung erfolgt hingegen erst zur Laufzeit und kann beispielsweise zwischen den Softwareagenten bzw. Knoten anhand der vorhandenen Fähigkeiten ausgehandelt werden.

Die Softwareagenten bzw. Knoten eines verteilten Problemlösers, die an der Problemlösung mitarbeiten, stehen zur Laufzeit bereits fest. Es ist im Allgemeinen nicht vorgesehen, dass neue Softwareagenten bzw. Knoten einem verteilten Problemlöser beitreten, was sie insbesondere von Multiagentensystemen unterscheidet. Eine Beitrittsentscheidung von Softwareagenten zu einem oder mehreren verteilten Problemlösern wird im Allgemeinen nicht betrachtet, könnte jedoch durch den Entwickler insbesondere im Falle von kooperativen verteilten Problemlösern vorgesehen werden (vgl. auch Abschnitt 2.2.2.3). Organisationsmodelle aus dem Bereich der verteilten Problemlösern können hingegen für die Strukturierung von Multiagentensystemen Anwendung finden. In diesem Fall wird das Ziel jedoch nicht von einer dem verteilten Problemlöser externen Instanz vorgegeben, sondern bildet sich durch die Kooperation mehrerer Softwareagenten.

Bearbeitung der Teilprobleme. Die Bearbeitung der Teilprobleme kann ebenfalls erst zur Laufzeit erfolgen. Abhängig von der konkreten Implementierung müssen sich die Softwareagenten bzw. Knoten des verteilten Problemlösers zur Teilproblembearbeitung an unterschiedliche Vorgaben halten. So kann die Bearbeitungsweise unter Umständen exakt vorgegeben sein oder dem einzelnen Softwareagent bzw. Knoten Entscheidungsspielraum zugestanden werden. Auch kann die Bearbeitung eine weitere Zerlegung in Teilprobleme beinhalten, die eine erneute Zuweisung an andere Softwareagenten bzw. Knoten erfordert.

Zusammenführung der Teilprobleme. Nach Bearbeitung der Teilprobleme werden diese wieder zur übergeordneten Problemlösung zusammengeführt. Im Allgemeinen wird diese Zusammenführung von demjenigen Softwareagent bzw. Knoten des verteilten Problemlösers durchgeführt, der die Zerlegung des Problems vorgenommen hat und somit über Informationen über die Zusammenhänge zwischen den einzelnen Teilproblemen verfügt. In diesem Fall obliegt die Überwachung von Abhängigkeiten ausschließlich diesem Softwareagent bzw. Knoten. Alternativ werden in Abschnitt 2.2.2.3 kooperative verteilte Problemlöser eingeführt, die diese Abhängigkeit bereits bei der Bearbeitung der Teilprobleme betrachten.

Da die Menge an Softwareagenten oder Knoten im Allgemeinen bereits bei der Entwicklung eines verteilten Problemlösers feststeht, findet der Begriff *Mitgliedschaft* im Kontext von verteilten Problemlösern in der Literatur keine Anwendung. Im Hinblick auf Beitrittsentscheidungen von Softwareagenten, ergeben sich aus den vorangegangenen Überlegungen jedoch erste Erkenntnisse über die Mitgliedschaft in verteilten Problemlösern. In verteilten Problemlösern werden beispielsweise die oben angeführten Verteilungsregeln der zu

bearbeitenden Teilprobleme im Allgemeinen bereits bei der Entwicklung festgelegt. Dies bedeutet, dass die einzelnen Softwareagenten oder Knoten keinen direkten Einfluss auf die Problemverteilung haben und somit die Frage mit wem kooperiert werden soll ebenfalls nicht frei entscheiden können. Dementsprechend ist es dem Entwickler überlassen, ob Mitgliedschaften in mehreren verteilten Problemlösern vorgesehen werden. Abhängig von der jeweiligen Implementierung kann ein Softwareagent gleichzeitig Mitglied in verschiedenen verteilten Problemlösern sein. Dieser Aspekt wird jedoch in der Literatur im Allgemeinen nicht betrachtet.

2.2.2.1 Blackboard-Systeme

Frühe Vertreter⁹ der verteilten Problemlöser sind die Blackboard-Systeme, in denen so genannte Wissensquellen an der Lösung eines gestellten Problems zusammenarbeiten. Die Wissensquellen sind jeweils „Experten“ auf einem bestimmten Gebiet und können über einen globalen Zustandsraum (engl. *Blackboard*) – im Sinne einer gemeinsamen Datenbank – Informationen austauschen (Bond und Gasser, 1988, S. 5). Eine der ersten durchgängigen Implementierungen der Blackboard-Architektur ist das HEARSAY-II-System zur Erkennung von natürlicher Sprache (Erman et al., 1980). Der globale Zustandsraum ist bei HEARSAY-II in verschiedene Ebenen eingeteilt, die jeweils eine unterschiedliche grammatikalische Granularität aufweisen, z.B. Buchstaben, Silben, Wörter, Wortsequenzen und Sätze. Verschiedene Wissensquellen sind dabei auf unterschiedlichen Ebenen der Spracherkennung aktiv, sind somit auf die jeweiligen Abhängigkeiten zwischen diesen Ebenen spezialisiert.

Die in Abschnitt 2.2.2 angeführten vorab benötigten Informationen treffen ebenfalls auf Blackboard-Systeme zu. Darüber hinaus sind für den Entwickler jedoch weitere Informationen bereits bei der Implementierung notwendig, die jeweils am Beispiel HEARSAY-II im Folgenden veranschaulicht werden:

Struktur des globalen Zustandsraums. Gegenüber anderen verteilten Problemlösern basieren Blackboard-Systeme auf einem Informationsaustausch über einen globalen Zustandsraum. Dieser globale Zustandsraum ist für die Kooperation der Wissensquellen essentiell und die Struktur desselben entscheidend für die Problemlösungsfähigkeit. Die Struktur des globalen Zustandsraums ist bereits bei der Entwicklung des Blackboard-Systems festzulegen. Diese wird im Allgemeinen abhängig von der zu lösenden Aufgabe definiert – beispielsweise die Einteilung in die oben genannte grammatikalische Granularität im Falle von HEARSAY-II.

Wissensquellen. In einem Blackboard-System werden die aktiv handelnden Instanzen als Wissensquellen bezeichnet. Diese werden im Allgemeinen bereits bei Entwicklung des Systems festgelegt und auf die Struktur des globalen Zustandsraums abgestimmt. Diese beiden Komponenten sind voneinander abhängig, da entsprechende Experten in Form von Wissensquellen ihr Wissen unter Umständen nicht einsetzen können, falls die Struktur des globalen Zustandsraumes nicht passend gewählt wurde. Im

⁹ Die in Abschnitt 2.2.2.2 betrachteten Kontraktnetze entwickelten sich zeitlich parallel zu den hier vorgestellten Blackboard-Systemen.

Hearsay-II-System stehen neben der Struktur des globalen Zustandsraums auch die zur Lösung benötigten Wissensquellen bereits bei der Entwicklung fest.

Neben den voran genannten, bereits zur Entwicklungszeit nötigen Festlegungen, gibt es gegenüber anderen verteilten Problemlösern ebenfalls Unterschiede in Bezug auf die zur Laufzeit durchgeführten Entscheidungen. Diese lassen sich wie folgt charakterisieren:

Reihenfolge der Wissensquellen. Während die Wissensquellen selbst bereits bei der Entwicklung eines Blackboard-Systems festgelegt werden, wird die Reihenfolge, in der die einzelnen Wissensquellen aktiv werden, erst zur Laufzeit bestimmt. Diese Aufgabe übernimmt im HEARSAY-II-System der *Scheduler*, der hierdurch ebenfalls den Zugriff der Wissensquellen auf den globalen Zustandsraum mit Hilfe von Prioritäten steuert (Erman et al., 1980, S. 357 f.).

Aktivitäten der Wissensquellen. Die Durchführung von Aktionen auf dem globalen Zustandsraum durch die Wissensquellen wird zur Laufzeit durch die festgelegte Reihenfolge – z.B. durch einen *Scheduler* – initiiert. Die hierbei durchzuführenden Aktivitäten hingegen sind abhängig vom globalen Zustandsraum und können somit ebenfalls erst zur Laufzeit determiniert werden. Im Falle von HEARSAY-II können beispielsweise Wissensquellen Aktivitäten auf verschiedenen Ebenen des globalen Zustandsraums sowohl lesend als auch schreibend ausführen.

Als aktiv handelnde Instanzen innerhalb von Blackboard-Systemen, sind Wissensquellen dessen zentrale Bestandteile, welche bereits a priori bei der Systementwicklung festgelegt werden. Änderungen an den vorgesehenen Wissensquellen im Sinne von Ein- und Austritten sind im Konzept der Blackboard-Architektur im Allgemeinen nicht vorgesehen, jedoch haben bereits frühe Arbeiten hierin Potential erkannt: „*Greater utilization is afforded when the element is a free agent, able to contract services to many different tasks*“ (Fox, 1981, S. 77). So könnten zur Laufzeit neue Experten als Wissensquellen herangezogen und somit als neue „Mitglieder“ aufgenommen werden. Unklar bleibt in diesem Falle jedoch, welche Instanz innerhalb des Blackboard-Systems über die Aufnahme einer zusätzlichen Wissensquelle zu entscheiden hätte.

2.2.2.2 Nicht-kooperative verteilte Problemlöser

Der Begriff der *nicht-kooperativen verteilten Problemlöser* wird hier als Abgrenzung zu den *kooperativen verteilten Problemlösern* verwendet. Wie bereits in Abschnitt 2.2.2 angeführt, findet sich diese Abgrenzung in der Literatur im Allgemeinen nicht. Sie ist jedoch notwendig, um die konzeptionellen Unterschiede einer Reihe von verteilten Problemlösern zu Blackboard-Systemen und kooperativen verteilten Problemlösern aufzeigen zu können. Dabei bedeutet die hier untersuchte Variante der nicht-kooperativen verteilten Problemlöser nicht, dass keinerlei Kooperation zwischen den beteiligten Akteuren stattfindet, hier wird vielmehr auf eine Problemlösungsebene abgezielt. Das zu lösende Problem wird in Teilprobleme aufgeteilt, die von verschiedenen Knoten bearbeitet werden können. Die Vergabe von Teilproblemen an andere Knoten kann zwar bereits als Kooperation bezeichnet werden, wird hier jedoch als reine Auf- bzw. Verteilung angesehen. Der „nicht-kooperative“ Aspekt

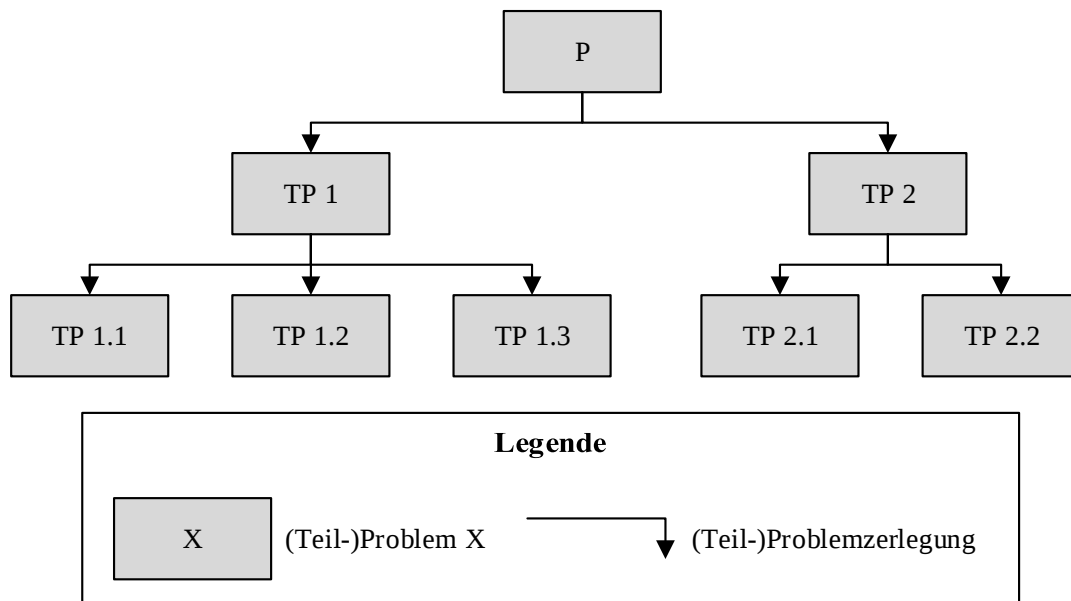


Abbildung 2.2: Problemzerlegung bei nicht-kooperativen verteilten Problemlösern

dieser verteilten Problemlöser lässt sich durch den Mangel an Abstimmung zwischen den unterschiedlichen Teilproblemen zurückführen: Die Knoten lösen die Teilprobleme unabhängig von anderen Teilproblemen des gleichen übergeordneten Problems. Abbildung 2.2 verdeutlicht die Struktur der Problemzerlegung.

Ein prominenter Vertreter der nicht-kooperativen verteilten Problemlöser ist das Kontrakt-netzprotokoll CNET (Smith, 1980), welches 2002 in abgewandelter Form als *FIPA Contract Net Interaction Protocol* standardisiert wurde. CNET stammt aus einer ähnlichen Zeit wie das oben angeführte HEARSAY-II und bezeichnet die Akteure des Systems als Knoten, greift jedoch ebenfalls den Begriff der *Wissensquellen* auf: *Distributed problem solving is the cooperative solution of problems by a decentralized and loosely coupled collection of knowledge-sources ..., located in a number of distinct processor nodes* (Smith, 1980, S. 1104). Der wesentliche Unterschied besteht jedoch in der Abwesenheit eines globalen Zustandsraums und Schedulers: „*there is neither global control nor global data storage*“ (Smith, 1980, S. 1104). Die Knoten im CNET-Protokoll bestehen aus einer bestimmten Anzahl an Wissensquellen und können Verträge mit anderen Knoten zur Bearbeitung von Teilproblemen eingehen. Die Knoten können in einem Vertrag zwei verschiedene Rollen einnehmen: (i) *Manager*, welcher (Teil-)Aufgaben zur Bearbeitung ausschreibt und (ii) *Contractor*, welcher einen Vertrag über die Bearbeitung einer (Teil-)Aufgabe eingegangen ist.¹⁰ Zwar ordnen Durfee et al. (1989, S. 68) das CNET-Protokoll den kooperativen verteilten Problemlösern zu, stellen dabei allerdings heraus, dass sich der kooperative Aspekt auf die Zerlegung der Problemstellung beschränkt, die so lange durchgeführt wird, bis ein einzelner Knoten das Teilproblem alleine lösen kann. Dieser Aspekt ist allerdings – wie oben ausgeführt – für das Verständnis dieser Arbeit von Kooperation bei verteilten Problemlösern nicht ausreichend. Dabei ist ebenfalls zu beachten, dass zur Zeit der Entstehung

¹⁰ Bei der Standardisierung im FIPA Contract Net Interaction Protocol wurden statt der Begriffe *Manager* und *Contractor* die Begriffe *Initiator* und *Participant* gewählt.

dieses Artikels 1989 in der VKI noch keine einheitlichen Begriffe Verwendung fanden (vgl. Abschnitt 2.2.2 sowie Kirn (1996a, S. 53 ff.)).

Wie für andere Typen verteilter Problemlöser gelten die in Abschnitt 2.2.2 aufgeführten Voraussetzungen für die Anwendung nicht-kooperativer verteilter Problemlöser. Zusätzlich muss bereits bei der Entwicklung der folgende Punkt bekannt sein:

Zerlegbarkeit in unabhängige Teilprobleme. Das zu lösende Problem muss in unabhängige Teilprobleme zerlegt werden können. Zwar kann ein Knoten, der als Contractor eine Teilaufgabe bearbeitet, zur weiteren Zerlegung gleichzeitig die Rolle eines Manager einnehmen, es findet jedoch keine Koordination der parallel bearbeiteten Teilaufgaben statt. Die Konsolidierung der einzelnen Teilaufgaben übernimmt der ausschreibende Manager-Knoten, Konflikte oder suboptimale Lösungen können folglich erst bei diesem Schritt zu Tage treten. Somit eignen sich nicht-kooperative verteilte Problemlöser insbesondere für Probleme, die sich in Teilprobleme zerlegen lassen, bei denen keine Interdependenzen be- oder entstehen. Diese Eigenschaft ist bereits bei der Entwicklung entscheidend und erfordert somit weitergehendes Wissen über das zu lösende Problem selbst. Die einzelnen Teilprobleme in Abbildung 2.2 sind nicht voneinander abhängig und eine Koordination der jeweils bearbeitenden Akteure ist nicht vorgesehen.

Die ad-hoc zur Laufzeit zu entscheidenden Punkte decken sich hingegen mit den in Abschnitt 2.2.2 aufgeführten übergreifenden Entscheidungsalternativen. Insbesondere die Zerlegung eines Problems in Teilprobleme durch die Rolle des Managers sowie die Zuweisung ebenjener Teilprobleme mit Hilfe einer Ausschreibung an bearbeitende Knoten (Contractors) sind die Entscheidungsalternativen der Knoten innerhalb von nicht-kooperativen verteilten Problemlösern.

In Bezug auf Beitrittsentscheidungen ergeben sich bei den nicht-kooperativen verteilten Problemlösern gegenüber den Blackboard-Systemen erste interessante Aspekte. In einem Kontraktnetz sind die Knoten beispielsweise nur „lose gekoppelt“: *„We use the term 'distributed' rather than 'parallel' to emphasize that the individual processors are loosely-coupled; ... such systems are highly modular, and hence offer considerable conceptual clarity and simplicity in their organization“* (Smith und Davis, 1978, S. 2). Diese lose Kopplung ermöglicht es, für die Bearbeitung bestimmter Aufgaben die Fähigkeiten und das Wissen der passenden Knoten heranzuziehen. Der Bindung zwischen Manager und Contractor mangelt es jedoch an Verbindlichkeit. Es sind keinerlei Sanktionierungsmöglichkeiten im Falle von nicht-bearbeiteten Aufgaben vorgesehen. Eine entsprechende Erweiterung wurde erst später durch Sandholm und Lesser (1995) vorgestellt (vgl. auch Abschnitt 2.4.3). Nach Smith und Davis steht diese lose Kopplung neben einer erhöhten Modularität auch für eine Fokussierung der Knoten auf die Problemlösung durch eine Minimierung der Kommunikation, die sich auf den Beitrittsprozess beschränkt. Der Mitgliedschaftsbegriff kann in diesem Zusammenhang auf zwei Ebenen verwendet werden: Zum einen können die beteiligten Knoten, die als Contractor an der Lösung eines Teilproblems arbeiten, zusammen als Mitglieder dieser Problem Instanz bezeichnet werden. Diese Auffassung

wird durch die Bezeichnung als *Participant* im FIPA Contract Net Interaction Protocol bestärkt. Zum anderen können die beteiligten Knoten als Mitglieder des nicht-kooperativen verteilten Problemlösers gesehen werden. Wie bei anderen Typen verteilter Problemlöser stehen dabei die Knoten bzw. Mitglieder im Allgemeinen bereits bei Entwicklung fest und eine Betrachtung der Beitrittsentscheidung ist somit nur in ersterem Fall möglich. Sieht der Entwickler hingegen bereits bei der Entwicklung des nicht-kooperativen verteilten Problemlösers die Mitgliedschaft eines Knoten in mehreren Systemen vor, gewinnt die Entscheidung des einzelnen Knotens für oder gegen eine Mitgliedschaft an Relevanz. Fraglich ist in diesem Fall jedoch, wie weit die Entscheidungskompetenz auf die Knoten übertragen werden kann. Wie bei anderen Typen verteilter Problemlöser steht die Lösung des von außen vorgegebenen Problems im Vordergrund. Individuelle Entscheidungen von Knoten könnten diesem Ziel zuwider laufen.

2.2.2.3 Kooperative verteilte Problemlöser

Smith und Davis (1981) untersuchen verschiedene Modelle von Kooperation bei verteilten Problemlösern und stellen nicht-kooperative verteilte Problemlöser ebenfalls den Blackboard-Systemen gegenüber. Dabei bezeichnen sie erstere als „*task-sharing*“ – im Sinne einer hierarchischen Top-Down Betrachtungsweise wie in Abbildung 2.2 dargestellt – und zweitere als „*result-sharing*“. Das Aufteilen von Problemen verursacht im Allgemeinen jedoch Abhängigkeiten zwischen den entstehenden Teilproblemen, was in den oben angeführten nicht-kooperativen verteilten Problemlösern nicht betrachtet wird: *„One approach to dealing with inconsistency is to not allow it in the first place, or at least to not consider it. For example, the Contract-Net protocol ... decomposed and distributed tasks in such a way that there was always a manager node to coordinate its contractors, who would in turn always pursue the task as expected“* (Durfee et al., 1989, S. 71). Das Austauschen von (Zwischen-)Ergebnissen der einzelnen Knoten, um zu einer dem übergeordneten Problem entsprechenden Lösung zu gelangen, kann hingegen als eine Variante des Bottom-Up-Ansatz verstanden werden und ermöglicht es, Abhängigkeiten zwischen Teilproblemen zu betrachten: *„Result-sharing is most useful in problem domains in which ... results achieved by one node influence or constrain those that can be achieved by another node“* (Smith und Davis, 1981, S. 68). Im Falle der Blackboard-Architektur ist jedoch eine zentrale Instanz nötig, die globales Wissen und die Koordination (Scheduler) bereitstellt. Kooperative verteilte Problemlöser kommen hingegen ohne eine solche zentrale Instanz aus, sind jedoch wie alle verteilten Problemlöser bereits bei der Entwicklung auf einen bestimmten Problemtyp ausgerichtet. Abbildung 2.3 ist an die vorangegangene Abbildung 2.2 angelehnt und zeigt die Abstimmung zwischen verschiedenen Knoten auf, die bei der Bearbeitung von Teilproblemen benötigt wird.

Kooperative verteilte Problemlöser vereinen somit zwei wesentliche Eigenschaften der vorangegangenen Typen verteilter Problemlöser: (i) Wie Blackboard-Systeme können die beteiligten Knoten an der Bearbeitung einer Teilaufgabe kooperieren und somit Abhängigkeiten zwischen verschiedenen Teilaufgaben bei der Bearbeitung berücksichtigen. (ii) Sie benötigen jedoch wie nicht-kooperative verteilte Problemlöser keine zentrale Instanz zur Koordination sowie zum Wissensaustausch. Erste Ansätze zur Kombination beider Ansätze unternahmen

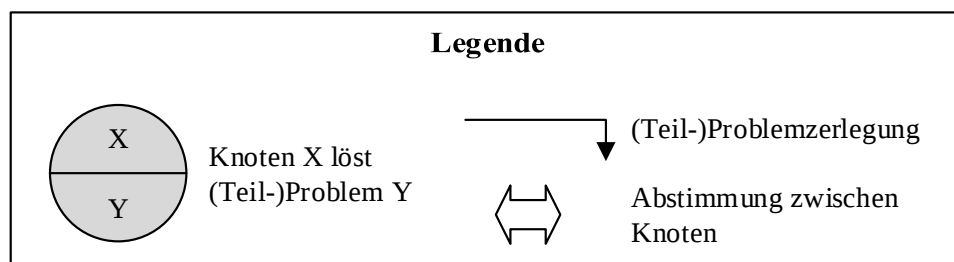
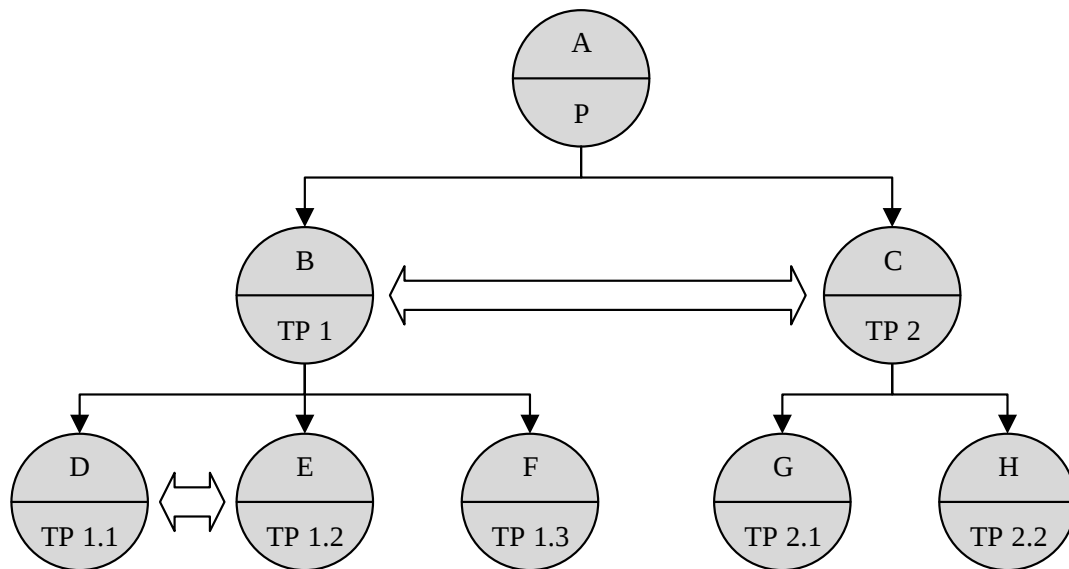


Abbildung 2.3: Abstimmungsbeispiel für kooperative verteilte Problemlöser

Lesser und Erman (1988) mit der Verteilung des oben vorgestellten Blackboard-Systems HEARSAY-II. Gegenüber Blackboard-Systemen und nicht-kooperativen verteilten Problemlösern ergeben sich hieraus weitere Aspekte, die im Falle der kooperativen verteilten Problemlöser bereits bei der Entwicklung bekannt sein müssen:

Zerlegbarkeit in Teilprobleme. Während nicht-kooperative verteilte Problemlöser nur für die Lösung von Problemen geeignet sind, die sich in unabhängige Teilprobleme zerlegen lassen, gilt diese Einschränkung für kooperative verteilte Problemlöser nicht. Letztere sind durch die Möglichkeit der Kooperation von mehreren Knoten in der Lage, sowohl Probleme mit abhängigen als auch mit unabhängigen Teilproblemen zu lösen. Die Voraussetzung liegt hierbei in der Zerlegbarkeit des Problems selbst: Bereits bei der Entwicklung eines kooperativen verteilten Problemlösers ist der Problemtyp und somit die Problemstruktur bekannt (vgl. Abschnitt 2.2.2). Ebenso ist es essentiell, dass sich aus dieser Problemstruktur die Zerlegbarkeit des Problems in Teilprobleme ergibt.

Kommunikationsfähigkeiten. Während Knoten bzw. Wissensquellen in Blackboard-Systemen nicht direkt, sondern nur über das Blackboard miteinander kommunizieren, müssen nicht-kooperative verteilte Problemlöser zur Problemzerlegung Teilaufgaben verteilen können – beispielsweise durch Ausschreibungen im CNET. Gegenüber diesen Typen sind die Anforderungen an die Kommunikationsfähigkeiten von kooperativen verteilten Problemlösern jedoch deutlich erweitert: Hier ist nicht nur die Teilaufgabe zuzuweisen, sondern die Knoten müssen in der Lage sein, bei der individuellen Teilproblembearbeitung sich mit anderen Knoten abzustimmen. Kommunikationsprotokolle zum Austausch von Zwischenergebnissen müssen somit bereits bei der Entwicklung von kooperativen verteilten Problemlösern vorgesehen sein. Diese enthalten nicht nur Informationen, wie die Kommunikation stattzufinden hat, sondern müssen auch vorsehen, in welchen Fällen sich ein Knoten mit welchem anderen Knoten in Verbindung setzt. Es sind somit auch Informationen über die Zerlegung des Problems und somit über die organisatorische Struktur nötig, damit sich ein Knoten mit den entsprechenden anderen Knoten in Verbindung setzen kann.

Aus der Notwendigkeit der erweiterten Kommunikationsfähigkeiten ergibt sich, dass auch verstärkt organisatorische Elemente bei der Konzipierung von kooperativen verteilten Problemlösern Beachtung finden: *„An organizational structure of a CDPS network is the pattern of information and control relationships that exist between the nodes, and the distribution of problemsolving capabilities among the nodes“* (Durfée et al., 1989, S. 72). Diese organisatorischen Elemente können entweder bereits bei der Entwicklung des kooperativen verteilten Problemlösers festgelegt oder erst zur Laufzeit zwischen den Knoten ausgehandelt werden. Essentiell in beiden Varianten ist jedoch die oben angeführte Kommunikationsfähigkeit der Knoten, die bereits bei der Entwicklung vorgesehen werden muss. Zur Laufzeit hingegen wird entschieden, wie die Kooperation zwischen den Knoten durchgeführt wird:

Kooperation der Knoten. Die Bearbeitung der Teilprobleme selbst wird in allen verteilten Problemlösern zur Laufzeit durchgeführt (vgl. Abschnitt 2.2.2). Im Falle von

kooperativen verteilten Problemlösern wird diese Bearbeitung jedoch überlagert von der durch die Abhängigkeit verschiedener Teilprobleme entstehenden Notwendigkeit zur Kooperation zwischen mehreren Knoten. Hierzu müssen die oben angeführten Kommunikationsmöglichkeiten gegeben und gegebenenfalls bestehende organisatorische Vorgaben den Knoten bekannt sein. Abhängig von der jeweiligen Implementierung können die Vorgaben dabei unterschiedlich restriktiv sein und den Knoten mehr oder weniger Handlungsspielraum ermöglichen, was wiederum sowohl die Rechenlast der einzelnen Knoten als auch den Kommunikationsbedarf beeinflusst.

Zwar werden Beitrittsverhandlungen bei kooperativen verteilten Problemlösern in der Literatur im Allgemeinen nicht betrachtet, grundsätzlich sind Ein- und Austritte jedoch auch in diesem Bereich möglich: *The Open Systems approach ... seems to represent an important conceptual framework for structuring large and complex CDPS networks made out of heterogeneous agents that can both passively tolerate and actively address inconsistencies* (Durfée et al., 1989, S. 72). Zwar handelt es sich bei dieser Auffassung um eine Entwicklung in Richtung der Multiagentensysteme (vgl. Abschnitt 2.2.3), sie zeigt jedoch auf, dass bereits für kooperative verteilte Problemlöser die Möglichkeit des Ein- und Austretens aufgegriffen wurde. Der Begriff „Beitritt“ kann sich zum einen auf den Beitritt zu dem kooperativen verteilten Problemlöser beziehen oder als Beitritt zu einer Gruppe von Softwareagenten bzw. Knoten verstanden werden, die sich zur Bearbeitung eines Teilproblems zusammenfindet. Letztere Variante hat jedoch im Allgemeinen nur einen kurzfristigen Charakter und würde ein Auflösen der Gruppe nach erfolgreicher Teilproblembearbeitung mit sich bringen. Einen Beitritt zum kooperativen verteilten Problemlöser hingegen könnte durchaus ein längerfristiger Charakter zugesprochen werden, insofern dieser durch den Entwickler vorgesehen und somit auch die Möglichkeit zur Mitgliedschaft in mehreren kooperativen verteilten Problemlösern bietet. Abschnitt 2.2.2 hat jedoch aufgezeigt, dass die Knoten verteilter Problemlöser im Allgemeinen bereits bei der Entwicklung festgelegt werden und somit zur Laufzeit kein Bei- oder Austritt vorgesehen ist. Dies unterscheidet sie von den sogenannten *Open Systems* (Bond und Gasser, 1988; Durfee et al., 1989), die im Abschnitt 2.2.3 zu Multiagentensystemen betrachtet werden. Den *Multiagentensystemen* werden in der Literatur teilweise auch sogenannte reaktive Multiagentensysteme untergeordnet. Dabei weisen die beteiligten „Agenten“ keinerlei deliberative Eigenschaften – beispielsweise durch ein Modell ihrer Umwelt – auf, fallen somit nicht unter den in dieser Arbeit verwendeten Begriff von Softwareagenten (vgl. Abschnitt 2.1 sowie Ferber und Drogoul (1992, S. 53)). Die reaktiven Multiagentensysteme weisen hingegen ähnliche Eigenschaften wie kooperative verteilte Problemlöser auf (vgl. Abschnitt 2.1.1.3) und lassen sich somit diesen zuordnen: Insofern es Ziele der einzelnen rein reaktiven Agenten gibt, sind diese bereits bei Entwicklung durch den Entwickler vorgegeben und sind im Allgemeinen nicht veränderlich. Sowohl die Zielvorgabe als auch die nicht vorhandene Lernfähigkeit schränkt den Einsatzbereich auf einen eng vorgegebenen Bereich ein. Die Knoten kooperativer verteilter Problemlöser sowie rein reaktive Agenten sind ausschließlich für die Bearbeitung einer vorab bestimmten Problemstellung ausgelegt, die entsprechenden Systeme sind somit den verteilten Problemlösern zuzuordnen. Dementsprechend ist oftmals auch die Art und Anzahl bereits

vorab festgelegt, was eine Veränderung der beteiligten Knoten bzw. rein reaktiven Agenten und somit einer Betrachtung von Beitrittsentscheidungen in diesen Systemen ausschließt. Multiagentensysteme im Sinne dieser Arbeit hingegen greifen den *Open Systems*-Begriff auf und werden im nachfolgenden Abschnitt 2.2.3 näher betrachtet.

2.2.3 Multiagentensysteme

Der Begriff *Multiagentensystem* findet in der Literatur unterschiedlichste Verwendung, wurde allerdings bereits früh in der verteilten künstlichen Intelligenz zu der in Abschnitt 2.2.2 betrachteten Kategorie der verteilten Problemlöser abgegrenzt: „*Multiagent (MA) systems research is concerned with coordinating intelligent behavior among a collection of (possibly pre-existing) autonomous intelligent 'agents' how they can coordinate their knowledge, goals, skills, and plans jointly to take action or to solve problems*“ (Bond und Gasser, 1988, S. 3). Ein wesentlicher Aspekt, der Multiagentensysteme von verteilten Problemlösern unterscheidet, wird hieraus deutlich: Softwareagenten existieren im Allgemeinen bereits vor einem Multiagentensystem, dem sie später zugehören. Bereits dieser Aspekt verdeutlicht, dass die Menge an Softwareagenten in einem Multiagentensystem nicht bei der Entwicklung vorgegeben ist, sondern die Softwareagenten zu deren Laufzeit diesem beitreten. Da die Mitgliedschaft in einem Multiagentensystem die Ressourcen eines Softwareagenten im Allgemeinen bindet, ist die Beitrittsentscheidung eines Softwareagenten zu einem Multiagentensystem somit relevant für dessen Verhalten.

In der Literatur wird häufig nur der Begriff Softwareagent explizit definiert und Multiagentensysteme implizit als eine Menge von diesen Softwareagenten angenommen. Dies wird beispielsweise bei dem sehr häufig zitierten Papier von Jennings (2000) deutlich, der zwar den Begriff *Multi-agent systems* in den Keywords auflistet, diesen im Text jedoch nicht verwendet. Das Papier fokussiert sich hingegen auf die Definition von Softwareagenten. Hieraus lässt sich ein erstes wesentliches Kriterium für Multiagentensysteme herleiten: Ein Multiagentensystem besteht aus zwei oder mehr Softwareagenten, die nach der Definition 2 einen gewissen Grad an Autonomie gegenüber ihrem Entwickler bzw. Lernfähigkeit aufweisen. Nach der Definition von Bond und Gasser können dabei die Ziele in Multiagentensystemen divergieren: „*The agents in a multiagent system may be working' toward a single global goal, or toward separate individual goals that interact*“ (Bond und Gasser, 1988, S. 3).

In der frühen Literatur zu verteilten Problemlösern wird bereits der Begriff *Open Systems* als Beschreibung eines Systems ohne zentrale Kontrolle aufgegriffen. Dabei zieht Hewitt (1986) Parallelen zu organisatorischen Problemstellungen und vergleicht offene Computersysteme mit schreibtischbasierten Arbeitsplätzen in Unternehmen. Letztere werden dabei durch folgende Eigenschaften charakterisiert: (i) Gleichzeitigkeit der Aktionen, (ii) Asynchronizität der Informationen, (iii) dezentrale Kontrolle, (iv) inkonsistente Informationen, (v) eingeschränkter Kreis an kooperierenden Mitgliedern sowie (vi) kontinuierliche Ausführung (Hewitt, 1986, S. 272 f.). Überträgt man diese Eigenschaften auf Multiagentensysteme, lässt sich der überwiegende Teil der vorangegangenen Aspekte als Folge von Punkt (iii) ableiten. Durch eine fehlende zentrale Kontrollinstanz finden unkoordinierte Aktionen gleichzeitig statt (i) und können zu inkonsistenten Informationen in den lokalen Wissensbasen der Soft-

wareagenten führen (iv). Insbesondere ist es den Softwareagenten nicht zwingend möglich, sämtliche anderen Softwareagenten und deren Fähigkeiten in einem Multiagentensystem zu kennen, was wiederum dazu führt, dass Softwareagenten nur mit einem ihnen bekannten Kreis an anderen Softwareagenten kooperieren können (v).

Die Offenheit von Multiagentensystemen stellt auch in späteren Arbeiten einen zentralen Aspekt für a priori unbekannte Zielsetzungen sowie Problemstellungen dar (Dignum und Padget, 2013, Anhalt, 2011, Petsch, 2006). Anhalt (2011) betrachtet die Offenheit von Systemen im Zusammenhang mit mobilen Softwareagenten und definiert diese als *„ein System der verteilten Datenverarbeitung, das hinsichtlich seiner Bestandteile Ausführungsplattformen und mobile Software-Agenten das Überschreiten bzw. Verändern von Systemgrenzen erlaubt“* (Anhalt, 2011, S. 25). Mit einem Fokus auf mobilen Softwareagenten, betrachtet diese Definition insbesondere die Migration¹¹ von mobilen Softwareagenten auf andere Systeme. Während in dieser Arbeit die speziellen Eigenschaften mobiler Softwareagenten nicht betrachtet werden, ist der Begriff der Offenheit stark mit dieser Sichtweise verknüpft. Zwar migrieren Softwareagenten bei der Beitrittsentscheidung zu einem Multiagentensystem im Sinne dieser Arbeit nicht auf andere Ausführungsplattformen, die Grenzen ebendieses Multiagentensystems hingegen verschieben sich und schließen nach einem erfolgten Beitritt den Softwareagenten mit ein.

Während die Eigenschaften von offenen Systemen – welchen weithin die Multiagentensysteme zugeordnet werden – in der Literatur meist nur oberflächlich behandelt und im Allgemeinen nur die Ergebnisse dieser Eigenschaft angenommen werden,¹² beschäftigt sich Petsch (2006) ausführlich mit dem Thema Offenheit von Multiagentensystemen und den sich daraus abzuleitenden Anforderungen an Multiagentensysteme. Dabei werden vier Perspektiven der Offenheit unterschieden:

Technische Offenheit. Damit Softwareagenten mit anderen Softwareagenten interagieren können, müssen die technischen Voraussetzungen hierfür erfüllt sein. Die technische Möglichkeit der Kommunikation ist eine Grundvoraussetzung damit eine Kommunikation zwischen den Softwareagenten innerhalb eines Multiagentensystems stattfinden kann, sowohl auf physikalischer als auch auf semantischer Ebene.¹³ Dies gilt entsprechend für beitretende Softwareagenten, die technisch in der Lage sein müssen, Beitrittsverhandlungen mit dem Multiagentensystem zu führen. Diese Voraussetzung stellt zunächst ausschließlich auf die technische Kommunikationsfähigkeit der Softwareagenten ab und entspricht daher einem Teil der in Abschnitt 2.2.1.1 eingeführten Anforderung an Bekanntschaften von Softwareagenten.

Technische Offenheit umfasst allerdings ebenfalls die Voraussetzung, dass ein Multiagentensystem selbst in der Lage sein muss, neu beitretende Softwareagenten aufzu-

¹¹ Unter *Migration* wird im Kontext mobiler Softwareagenten das *„Lösen und Wiederherstellen von Bindungen von einer Ausführungsplattform zu einer anderen“* (Anhalt, 2011, S. 12) verstanden.

¹² Huynh, Jennings und Shadbolt weisen beispielsweise Multiagentensystemen folgende Charakteristik bezüglich ihrer Offenheit zu: *„In an open MAS, agents can come and leave the system at anytime.“* (Huynh et al., 2006, S. 139)

¹³ Als Beispiel wählt Petsch (2006, S. 12 f.) das ISO/OSI-Schichtenmodell (Zimmermann, 1980). Technische Offenheit kann jedoch auch unabhängig von diesem Modell realisiert werden.

nehmen und technisch in die Kommunikationsbasis zu integrieren bzw. austretende entsprechend aus den vorhandenen Prozessen auszuschließen. Dabei handelt es sich um nach innen gerichtete Fähigkeiten des Multiagentensystems. Für Beitrittsverhandlungen sind jedoch auch nach außen gerichtete Fähigkeiten essentiell: Ist das Multiagentensystem für einen außenstehenden Softwareagenten nicht ansprechbar, können keine Beitrittsverhandlungen initiiert werden.

Ohne technische Offenheit können die nachfolgenden Perspektiven der Offenheit nicht realisiert werden. Die technische Offenheit ist somit auch Grundvoraussetzung für die anderen Perspektiven der Offenheit.

Systemische Offenheit. Aus der Systemtheorie wird die Perspektive der systemischen Offenheit entlehnt, welche die *„Wechselwirkung eines [Multiagenten-]Systems mit seiner Umwelt“* (Petsch, 2006, S. 134) betrachtet. Ein Aspekt hiervon ist die Möglichkeit, Elemente der Umgebung des Multiagentensystems aufzunehmen, beispielsweise Softwareagenten als neue Mitglieder einzugliedern. Während die technische Offenheit ausschließlich auf die technischen Möglichkeiten der Kommunikation und Integration eingeht und somit die Voraussetzung für die systemische Offenheit darstellt, fordert letztere, dass nicht nur die technische Möglichkeit besteht, sondern auch die Prozesse für deren Umsetzung existieren. Insbesondere im betrieblichen Einsatz können sich hieraus weitergehende Fragestellungen ergeben, die beispielsweise die Sicherheit als auch rechtliche Anforderungen wie Revisionssicherheit betreffen (Petsch, 2006, S. 135).

Soziale Offenheit. Die Perspektive der sozialen Offenheit zielt ähnlich den in Zusammenhang mit der Definition von Softwareagenten betrachteten Begriffs *sozial* (vgl. Abschnitt 2.1.1.2) auf deren Kommunikationsfähigkeit ab, grenzt sich dabei allerdings von der vorangegangenen Perspektive der technischen Offenheit wie folgt ab: *„Werden mittels der technischen Offenheit Voraussetzungen geschaffen, um die Kommunikation und Interaktion technisch zu realisieren (Interoperabilität), so werden bei der sozialen Offenheit Kommunikations- und Interaktionsprotokolle entworfen, die die semantische Klarheit der Kommunikation zwischen den Akteuren unterstützen“* (Petsch, 2006, S. 15). Es ist jedoch fraglich, inwiefern sich diese beiden Perspektiven überschneidungsfrei aufteilen lassen: Petsch (2006, S. 12 f.) nennt im Zuge der technischen Offenheit Protokolle auf den Schichten eins bis vier des ISO/OSI-Modells und für die soziale Offenheit das *FIPA Contract Net Interaction Protocol* (vgl. Abschnitt 2.2.2.2) als Beispiele. Bei beiden Beispielen handelt es sich um Protokolle, die Kommunikation zwischen verschiedenen Akteuren ermöglichen. Die Unterscheidung kann eher auf Ebene der Zielsetzung gesehen werden: Während die ersten vier Schichten des ISO/OSI-Modells auf die Nachrichtenübertragung abzielen, ermöglicht das FIPA Contract Net Interaction Protocol unabhängig von einem Übertragungsprotokoll die semantische Kommunikation zwischen den Akteuren.

Dementsprechend ist es für Beitrittsverhandlungen von Softwareagenten mit Multiagentensystemen entscheidend, dass die Verhandlungspartner aus der Perspektive

der sozialen Offenheit miteinander verhandeln können. Hierzu sind entsprechende Verhandlungsprotokolle auf semantischer Ebene notwendig.

Organisatorische Offenheit. Während die vorangegangenen Perspektiven die Interaktionen zwischen Softwareagenten und deren Voraussetzungen zum Kern haben, betrachtet die Perspektive der organisatorischen Offenheit insbesondere „*Probleme der Mitgliedschaft eines Agenten zu einem System und damit in Verbindung stehende organisatorische Strukturen*“ (Petsch, 2006, S. 16). Diese Perspektive ermöglicht es erst Multiagentensysteme mit betrieblichen Abläufen zu verknüpfen, indem diese nicht als rein technische, sondern durch diese Verknüpfung als sozio-technische Systeme angesehen werden. Erst die organisatorische Offenheit von Multiagentensystemen ermöglicht es, dass neue Mitglieder (zum Mitgliedschaftsbegriff vgl. Abschnitt 2.3.2) in einem Multiagentensystem aufgenommen werden. Hierdurch verändert sich automatisch auch ein Teil der organisatorischen Struktur, da neue Mitglieder Aufgaben übertragen bekommen und somit in die Prozesse des Multiagentensystems eingebunden werden. Durch die organisatorische Offenheit ist die Möglichkeit für die Mitgliedschaft von Softwareagenten in mehreren Multiagentensystemen inherent vorhanden, da ein Multiagentensystem die von einem Mitglied eingegangenen Verpflichtungen mit anderen Multiagentensystemen nicht abschließend begutachten kann.

Um Multiagentensysteme als offene Systeme zu bezeichnen, ist ein vertieftes Verständnis des Begriffs *Offenheit* notwendig. Durch die Unterteilung in technische, systemische, soziale und organisatorische Offenheit können Anforderungen an die Offenheit von Multiagentensystemen aus verschiedenen Perspektiven gestellt werden. Diese Anforderungen sind essentiell bei der Betrachtung von Beitrittsentscheidungen von Softwareagenten, die eine Offenheit auf allen vier Ebenen voraussetzen.

Gegenüber den in Abschnitt 2.2.2 betrachteten verteilten Problemlösern ergibt sich für Multiagentensysteme aus der fehlenden zentralen Kontrollinstanz ein wesentlicher Unterschied: Bei Betrachtung der Informationen, die bei der Entwicklung bzw. zur Laufzeit des Systems bekannt sein müssen, lässt sich kein Entwicklungszeitpunkt für das System festlegen. Durch die verteilte Kontrolle und im Allgemeinen auch Entwicklung, kann für den Entwicklungszeitpunkt nur auf denjenigen des Softwareagenten abgestellt werden. Das Multiagentensystem hingegen bildet sich erst zur Laufzeit verschiedener Softwareagenten. Bei der Entwicklung eines Softwareagenten ergeben sich hingegen zentrale Punkte, die zu diesem Zeitpunkt bereits bekannt sind:

Ziele. Softwareagenten bekommen im Allgemeinen bei der Entwicklung bereits Ziele vorgegeben. Dabei handelt es sich in der Regel um übergeordnete Ziele, wie beispielsweise die Maximierung des erzielten Gewinns ohne eine Vorgabe auf welche Art und Weise dieser Gewinn realisiert werden kann. Im Gegensatz zu verteilten Problemlösern können diese Ziele, mindestens jedoch die daraus abgeleiteten Teilziele durch den Softwareagenten zur Laufzeit verändert werden (Wooldridge, 2009, S. 23). Diese Zielorientierung ist bereits Grundvoraussetzung für die Betrachtung von Softwareagenten (vgl. Abschnitt 2.1.1.4).

Fähigkeiten. Jeder Softwareagent besitzt eine Menge von Fähigkeiten, die ihm die Lösung von Problemen alleine oder im Zusammenschluss ermöglichen. Zwar sind Softwareagenten stets in der Lage, in gewissem Umfang zu lernen (vgl. Abschnitt 2.1.1.1), dieser Lernfähigkeit sind im Allgemeinen jedoch Grenzen gesetzt. Stehen dem Softwareagenten bestimmte Sensoren oder Aktuatoren nicht zur Verfügung (z.B. Temperatursensor, Greifarm), so kann er bestimmte Umwelteinflüsse nicht wahrnehmen bzw. die Umwelt unter Umständen nicht in geeigneter Weise verändern. Die Fähigkeiten, die bei der Entwicklung einem Softwareagenten mitgegeben wurden, beeinflussen somit stets dessen Verhalten zur Laufzeit. Insbesondere wirken sich die vorhandenen Fähigkeiten auf die Notwendigkeit und den Ablauf von Zusammenschlüssen mit anderen Softwareagenten aus: Zum einen können fehlende Fähigkeiten eine Notwendigkeit für Zusammenschlüsse mit sich bringen, zum anderen können vorhandene oder fehlende Fähigkeiten Beitrittsverhandlungen zu bestehenden oder neuen Zusammenschlüssen beeinflussen.

Wissen. Während die voran genannten Fähigkeiten auf die Möglichkeiten eines Softwareagenten eingehen, benötigt er zur Ausführung ebenfalls Wissen über die Umwelt. Dieses Wissen kann zum Teil bereits bei der Entwicklung vorgegeben, zur Laufzeit jedoch wieder verworfen bzw. angepasst werden. Ebenfalls ist es möglich, bei der Entwicklung einem Softwareagenten bereits Informationen über andere Softwareagenten – die somit zur Menge seiner Bekanntschaften hinzugerechnet werden – bereitzustellen. Hierin können auch Informationen über die Fähigkeiten dieser Bekanntschaften enthalten sein. Dies erleichtert es dem Softwareagenten zur Laufzeit Zusammenschlüsse zur Lösung von Problemen einzugehen.

Auch die zur Laufzeit zu entscheidenden Parameter unterscheiden sich wesentlich von denen der verteilten Problemlöser. Insbesondere die Zuweisung der zu bearbeitenden Probleme folgt einem grundverschiedenen Ansatz:

Problem. Während verteilte Problemlöser zur Lösung a priori bekannter Probleme entwickelt werden, steht bei Softwareagenten die Zielsetzung jedes einzelnen Softwareagenten im Vordergrund. Ein zu lösendes Problem begegnet einem Softwareagenten somit erst zur Laufzeit. Zwar wird bereits bei der Entwicklung des Softwareagenten durch die Ausstattung mit entsprechenden Fähigkeiten eine Fokussierung auf mögliche Problemtypen durchgeführt, es können jedoch vor der Ausführung nicht abschließend sämtliche möglichen Problemstellungen und Umweltzustände betrachtet werden.

Zielanpassungen. Die oben erwähnten, bereits bei der Entwicklung des Softwareagenten mitgegebenen Ziele können durch den Softwareagenten zur Laufzeit abhängig von der jeweiligen Implementierung entweder komplett verändert oder in Teilziele aufgeteilt werden. Mindestens letztere können an sich verändernde Umweltgegebenheiten angepasst werden. Diese Zielanpassung führt ebenfalls dazu, dass die Mitgliedschaft in einem Multiagentensystem durch den Softwareagenten anders bewertet wird. Um dem Rechnung zu tragen, ist es notwendig, dass Softwareagenten Multiagentensysteme verlassen und anderen beitreten können. Eine Anpassung der Ziele hat somit direkten

Einfluss auf die Beitrittsentscheidung von Softwareagenten zu Zusammenschlüssen mit anderen Softwareagenten.

Zusammenschlüsse. Aus den Problemstellungen, denen sich ein Softwareagent gegenüber sieht, und seinen gewählten (Teil-)Zielen ergeben sich unter Umständen Möglichkeiten der Kooperation mit anderen Softwareagenten. Zur Laufzeit ist es dem Softwareagenten somit möglich, Zusammenschlüsse mit anderen Softwareagenten – aus dem Kreis seiner Bekanntschaften – einzugehen. Dabei können neue Zusammenschlüsse gegründet, bestehenden beigetreten oder aus einem Zusammenschluss ausgetreten werden.

Kooperation der Softwareagenten. Mitglieder eines Zusammenschlusses kooperieren im Allgemeinen mit anderen Mitgliedern. Die Entscheidung, welche Softwareagenten eines Multiagentensystems mit anderen kooperieren, findet erst zur Laufzeit statt und kann im Allgemeinen nicht bei der Entwicklung geplant werden. Die Art der Kooperation beeinflusst insbesondere die Bindung der Softwareagenten an den Zusammenschluss: Ist ein Softwareagent beispielsweise zur Bereitstellung von Diensten verpflichtet, schränkt das die ihm für andere Aufgaben zur Verfügung stehenden Ressourcen ein. Die bestehenden Kooperationen zwischen Softwareagenten haben somit einen zentralen Einfluss auf die Beitrittsentscheidung.

Neues Wissen. Das Wissen eines Softwareagenten kann zur Laufzeit durch die vorhandene Lernfähigkeit (vgl. Abschnitt 2.1.1.1) erweitert oder angepasst werden. Neben dem Wissen über die Umwelt, fällt hierunter auch die Menge an Bekanntschaften. Diese kann durch den Softwareagenten zur Laufzeit ebenfalls erweitert oder angepasst werden. Durch neue Bekanntschaften stehen einem Softwareagenten zur Laufzeit mehr Möglichkeiten für Zusammenschlüsse zur Verfügung, nicht mehr verfügbare Bekanntschaften hingegen finden bei der Aktualisierung von Zusammenschlüssen Berücksichtigung. Eine größere Anzahl an Bekanntschaften bringt auch neue Kombinationsmöglichkeiten für Zusammenschlüsse mit sich, zwischen denen der Softwareagent abwägen muss und steigert wiederum die Komplexität von Beitrittsentscheidungen.

Nach Wooldridge (2009, S. 224) besteht ein Multiagentensystem aus Softwareagenten, die sich zu Kooperationsverbünden mit organisationsähnlich festgelegten Regeln und Normen zusammenschließen. Hierzu müssen die Softwareagenten untereinander interagieren können und bilden auf dieser Grundlage das Multiagentensystem. Organisationen im Sinne der Organisationstheorie sind allerdings auf Dauerhaftigkeit ausgelegt (vgl. Abschnitt 2.3.1.1), während bei Multiagentensystemen im Allgemeinen eine höchstmögliche Flexibilität im Vordergrund steht (Kirn, 2006, S. 53 ff.). Multiagentensysteme können bereits durch diese Eigenschaften nicht alle der in Abschnitt 2.3.1.1 angeführten Anforderungen an Organisationen erfüllen und stellen somit einen eigenständigen Typus eines Zusammenschlusses von Softwareagenten dar. Im Rahmen dieser Arbeit wird der Begriff des Multiagentensystems abgegrenzt gegenüber den später behandelten Multiagenten-Organisationen. Hieraus leitet sich folgende Definition ab, die an Kirn (1996a, S.8) angelehnt ist:

Definition 5 (Multiagentensystem) *Ein Multiagentensystem ist ein Zusammenschluss von mehreren Softwareagenten, die sie sich aus ihrer lokalen Perspektive von der Zusammenarbeit einen individuellen oder globalen Vorteil versprechen. Das Multiagentensystem ist ein offenes System, in das Mitglieder ein- und wieder austreten können.*

Gegenüber der in Abschnitt 2.2.1.1 betrachteten Menge an Bekanntschaften eines Softwareagenten, findet in einem Multiagentensystem Kooperation zwischen Softwareagenten statt. Dies setzt allerdings nicht voraus, dass alle Softwareagenten eines Multiagentensystems in jeder Menge an Bekanntschaften der jeweiligen Mitglieder vorhanden sein müssen.

2.3 Organisationstheoretische Ansätze für Multiagenten-Organisationen

Aus dem vorangegangenen Abschnitt 2.2 wurde bereits deutlich, dass in Systemen der VKI verschiedene Organisationsmodelle zum Einsatz kommen, die Einfluss auf die Betrachtung von Beitrittsentscheidungen von Softwareagenten nehmen. Dabei liegt es nahe, Ansätze der betriebswirtschaftlichen Organisationstheorie zu betrachten, um mögliche Rückschlüsse auf Multiagenten-Organisationen ziehen zu können. Dieser Abschnitt untersucht daher die Begriffe *Organisation* (Abschnitt 2.3.1) und *Mitgliedschaft* (Abschnitt 2.3.2) aus organisationstheoretischer Perspektive, um jeweils Zusammenhänge zu Multiagenten-Organisationen herauszuarbeiten.

2.3.1 Der Organisationsbegriff

Dieser Abschnitt untersucht zunächst den Organisationsbegriff der Betriebswirtschaftslehre und stellt anschließend die Verbindung zu Multiagenten-Organisationen her.

2.3.1.1 Der Organisationsbegriff der Betriebswirtschaftslehre

Um unterschiedlichen Fragestellungen nachzugehen und beantworten zu können, haben sich in der Organisationstheorie verschiedene Formen von Organisationsverständnissen herausgebildet. Eine wesentliche Unterscheidung findet dabei zwischen dem instrumentellen und institutionellen Organisationsbegriff statt (Schreyögg, 1996, S. 5 ff.). Dabei sieht Picot (2005, S. 24 f.) den instrumentellen Organisationsbegriff als „*Mittel zum Zweck der Lösung des Organisationsproblems*“ und den institutionellen Organisationsbegriff als Interpretation der Organisation als soziales System. Diese unterschiedlichen Sichten lassen sich durch die unterschiedliche Verwendung des Terminus „Organisation“ zur Beschreibung zum einen der Tätigkeit des Organisierens, zum anderen des Ergebnisses ebendieser Tätigkeit erklären (Hoffmann, 1980, S. 1428). Sind Softwareagenten in eine Organisation involviert, entstehen abhängig vom jeweiligen Organisationsverständnis unterschiedliche Fragestellungen, die mit einer Beitrittsentscheidung des Softwareagenten einhergehen. Dieser Abschnitt geht vertiefend auf die Unterscheidung dieser beiden Organisationsbegriffe ein und zeigt die Zusammenhänge zur Begriffsfindung für Multiagenten-Organisationen auf.

Nach dem instrumentellen Organisationsverständnis wird „*als Organisation sowohl die Tätigkeit im Sinne des Organisierens als auch das Ergebnis dieser Bemühungen ... bezeichnet*“

(Kosiol, 1962, S. 15). Auch Hill et al. (1994, S. 17) verstehen unter Organisation eine Menge an Maßnahmen zur zielgerichteten Beeinflussung von sozialen Systemen und der den Mitgliedern zugeordneten Aktivitäten. Diese Auffassung des Organisationsbegriffs wurde lange in der deutschen Betriebswirtschaftslehre geteilt, wobei sich nach Schreyögg (1996, S. 5) allerdings zwei weitere Strömungen innerhalb der instrumentellen Sichtweise unterteilen lassen:

Der funktionale Organisationsbegriff. Nach dem funktionalen Organisationsverständnis wird die Organisation aus der Perspektive der Unternehmensführung als Aufgabe zur Zweckerfüllung der Unternehmung betrachtet (Schreyögg, 1996, S. 5). Die Zweckerfüllung wird somit nicht durch die Organisation selbst bestimmt, sondern von außen vorgegeben. Teilweise wird der funktionale Organisationsbegriff dem institutionellen und instrumentellen gegenübergestellt, er weist jedoch einige Gemeinsamkeiten mit letzterem auf und wird daher hier als Spezialfall dessen aufgefasst (vgl. Schreyögg (1996, S. 5)). Dies wird auch dadurch deutlich, dass die Auffassung von Gutenberg (1983) dem funktionalen Organisationsverständnis zugeordnet wird, er selbst jedoch vom „*instrumentalen Charakter*“ (Gutenberg, 1983, S. 236) der Organisation spricht. Die Organisation ist dabei strikt abgegrenzt und umfasst ausschließlich umsetzende Tätigkeiten, die der Planung nachgeordnet sind: *„Während Planung den Entwurf einer Ordnung bedeutet, nach der sich der gesamtbetriebliche Prozess vollziehen soll, stellt Organisation den Vollzug, die Realisierung dieser Ordnung dar“* (Gutenberg, 1983, S. 235).

Der konfigurative Organisationsbegriff. Während das funktionale Verständnis die Organisation als der Planung nachgelagert und somit untergeordnet sieht, versteht der konfigurative Organisationsbegriff hierunter die *„dauerhafte Strukturierung von Arbeitsprozessen“* (Schreyögg, 1996, S. 7). Innerhalb des instrumentellen Organisationsbegriffs bildet diese somit einen Gegenpol zum funktionalen Verständnis, was Kosiol als ein Hauptvertreter deutlich macht: *„Die Organisation wird hier deshalb folgerichtig der (laufenden) Disposition vorgeordnet; sie schafft den Rahmen, innerhalb dessen dann die dispositiven Anordnungen getroffen werden können“* (Kosiol, 1962, S. 20). Wie Kosiol sieht auch Grochla die Organisation als *„menschliche Gestaltungshandlungen, die durch das Merkmal der Strukturierung gekennzeichnet sind“* (Grochla, 1966, S. 73).

Beiden Strömungen innerhalb des instrumentellen Organisationsverständnisses ist gemein, dass sie aus einer Führungsperspektive (z.B. Unternehmensführung) Maßnahmen ergreifen, die soziale Systeme und deren Mitglieder zielgerichtet beeinflussen sollen. Die Betrachtung von Entitäten, die nicht Mitglieder eines sozialen Systems bzw. potentielle Mitglieder sind (zur Mitgliedschaft vgl. Abschnitt 2.3.2), fällt somit nicht in den Gegenstandsbereich des instrumentellen Organisationsbegriffs. Dies wird insbesondere dadurch deutlich, dass Kosiol die Organisation als starre Struktur sieht (Schreyögg, 1996, S. 8). Hieraus ergibt sich ein wesentliches Problem bei der Verwendung des instrumentellen Organisationsbegriffs zur Berücksichtigung von Beitrittsentscheidungen von Softwareagenten: Der instrumentelle Organisationsbegriff setzt eine vorgegebene Menge an zu organisierenden Mitgliedern voraus und betrachtet somit nicht potentiell neue Mitglieder oder deren Bei-

trittsentscheidung, unabhängig davon, ob es sich dabei um menschliche oder artifizielle potentielle Mitglieder handelt.

Schreyögg (2004b, S. 969) kritisiert die nicht mit Argumenten unterlegte Stabilitätsorientierung des instrumentellen Organisationsverständnisses und reklamiert die Notwendigkeit einer Erweiterung über die Perspektive der Organisation als Führungsinstrument hinaus. Diese sieht er im institutionellen Organisationsbegriff, der „den Blickwinkel auf das gesamte System, auf die Institution [lenkt]“ (Schreyögg, 2004b, S. 969). Dieser Blickwinkel ermöglicht insbesondere die Betrachtung von zunächst außerhalb der Organisation befindlichen potentiellen Mitgliedern und somit auch von deren Beitrittsentscheidung. Der institutionelle Organisationsbegriff versteht „Organisation“ nicht als Tätigkeit einer führungsgetriebenen Strukturierung, sondern als Institution, die unabhängig von einer Führungsebene existiert. Aber auch innerhalb der Vertreter des institutionellen Organisationsbegriffs gibt es keine eindeutige Definition. In der Literatur wird die Organisation im institutionellen Verständnis jedoch meist mit Hilfe von Merkmalen definiert. Um diese Merkmale weiter zu präzisieren, werden diese zunächst zu drei zentralen Merkmalen zusammengeführt, welche im Folgenden charakterisiert werden:

Soziales System. Der institutionelle Organisationsbegriff fasst eine Organisation als soziales System bzw. soziales Gebilde auf (Kieser und Walgenbach, 2007, S. 6). Dieses Merkmal lässt eine eindeutige Unterscheidung des Verständnisses von Organisationen als Instrument oder Institution zu. Hill et al. (1994) grenzen die beiden Begriffe Organisation und soziales System aus Sicht des instrumentellen Organisationsbegriffs ab und setzen voraus, dass sich die Organisationslehre *„mit Strukturierungsproblemen einer bestimmten Klasse von sozialen Systemen, die ... (weil der institutionelle Organisationsbegriff nicht verwendet werden soll) als 'organisierte soziale Systeme' umschrieben seien“* (Hill et al., 1994, S. 20), beschäftigt.

Fraglich ist, inwiefern Softwareagenten in einem solchen sozialen System betrachtet werden können oder ob diese eher nicht als leistungserbringender Teil der Organisation, sondern ähnlich anderer Maschinen als Ressourcen angesehen werden. Dies wird insbesondere dadurch deutlich, dass die organisationstheoretische Betriebswirtschaftslehre den Begriff des sozio-technischen Systems zur Abgrenzung von rein sozialen Systemen verwendet (vgl. beispielsweise Heidhardt (1980, S. 2077 f.) sowie zur Bedeutung des Begriffs *sozial* im Kontext von Softwareagenten Abschnitt 2.1.1.2). Wenn Softwareagenten in Organisationen thematisiert werden, muss die Bedingung des Bezugs zum Menschen herausgestellt werden. Hier stellt sich die Frage, ob die Softwareagenten zu diesem sozialen System gehören oder ob sie wie Sachmittel nur Ressourcen darstellen, auf die Mitglieder des sozialen Systems zugreifen können. Dieser Aspekt unterscheidet insbesondere eine Organisation im betriebswirtschaftlichen Sinn von einer Multiagenten-Organisation: Während erstere ein soziales System darstellt, welches ausschließlich menschliche Akteure als deren Mitglieder betrachtet, wird der Begriff des sozialen bei weiterem erweitert interpretiert und umfasst somit alle an einer Gesellschaft (hier somit der Multiagenten-Organisation) teilnehmenden Akteure, insbesondere auch Softwareagenten. Der zweiten, offeneren Interpretation

eines sozialen Systems folgt bereits Irle (1969, S. 1505), der den Begriff *sozial* auf die Interaktion zweier oder mehrerer Akteure bezieht und somit offen lässt, auch andere Arten als menschliche Akteure mit einzubeziehen.

Das soziale System Organisation weist dabei „*beständige Grenzen*“ (Schreyögg, 2004b, S. 970) auf, die es ermöglichen, Mitglieder von Nichtmitgliedern zu unterscheiden (vgl. Abschnitt 2.3.2 zur Mitgliedschaft). *Beständig* bedeutet dabei nicht, dass sich diese Grenzen nicht verschieben können. Im Gegenteil, ermöglicht eine Definition der Zugehörigkeit zu einer Organisation erst den Ein- und Austritt und somit die notwendige Adaptivität einer Organisation (Reichwald, 2004, S. 1004). Dies wird ebenfalls durch die Definition von Luhmann bekräftigt, der Organisation als „*eine Art der Bildung sozialer Systeme, in die man durch Entscheidung eintreten und aus denen man durch Entscheidung wieder austreten kann*“ (Luhmann, 1987, S. 41) versteht. Luhmann (1984, S. 16) baut seine Auffassung auf die allgemeine Systemtheorie auf und fasst neben Interaktionen und Gesellschaften Organisationen als Konkretisierung von sozialen Systemen auf.

Die Existenz von Grenzen einer Multiagenten-Organisation ermöglicht somit auch erst die Unterscheidung von Softwareagenten, die Mitglied einer Multiagenten-Organisation sind von denen, die es nicht sind. Folglich kann auch der Beitritt eines Softwareagenten zu einer Multiagenten-Organisation nur betrachtet werden, wenn klare Grenzen existieren und der Softwareagent eine Definition dieser Grenzen und den damit verbundenen Kriterien einer Mitgliedschaft verwendet. Zwar ist es denkbar, dass die Auffassung eines Softwareagenten und einer Multiagenten-Organisation, ob ersterer Mitglied von zweiterer ist, nicht deckungsgleich sind, in dieser Arbeit wird jedoch davon ausgegangen, dass beide Parteien ein einheitliches Verständnis mitbringen.

Zweckorientierung. Das Merkmal der Zweck- oder auch Zielorientierung findet sowohl in Zusammenhang mit dem institutionellen als auch mit dem instrumentellen Organisationsbegriff Verwendung. Während bei der Organisation als Instrument übergeordnete Ziele durch das soziale System (beispielsweise durch das Unternehmen) bereits vorgegeben sind und insbesondere zur Bewertung von Alternativen herangezogen werden (Hill et al., 1994, S. 143 f.), ist nach Auffassung einer Organisation als Institution das Unternehmen selbst eine Organisation und gibt sich somit selbst Ziele vor (Kieser und Walgenbach, 2007, S. 9). Das wiederum bedeutet, dass ein Softwareagent, der einer Multiagenten-Organisation beitrifft, durch diesen Beitritt bereits direkten oder indirekten Einfluss auf die Ziele der Multiagenten-Organisation im institutionellen Verständnis erhält. Unabhängig von der Beitrittsentscheidung des einzelnen Softwareagenten wägt die Multiagenten-Organisation somit ebenfalls den Beitritt neuer Mitglieder ab. Bei einem Beitritt können sich folglich in Abhängigkeit des Einflusses jedes einzelnen Mitglieds auch die Ziele der Organisation selbst verändern.

Grochla unterscheidet „*organisatorische Systeme*“ – als Abgrenzung zur Organisation als Instrument – „*von anderen Strukturen ... durch spezifische Zwecksetzung: Es handelt sich um Systeme zur Erfüllung von Daueraufgaben*“ (Grochla, 1966, S. 73).

Kieser und Walgenbach (2007, S. 7 f.) stellen ebenfalls auf die Dauerhaftigkeit dieser Organisationsziele ab und betonen die Unterscheidung zwischen Zielen der Organisation und deren Mitglieder (vgl. Abschnitt 2.3.2). Es ist durchaus möglich, dass mehrere Mitglieder gegenläufige Ziele verfolgen und diese sich von den Zielen der Organisation unterscheiden oder diesen widersprechen (Schreyögg, 2004b, S. 969 f.). Ein Ziel der Organisationsleitung ist es somit oftmals, Anreize für Organisationsmitglieder zu schaffen, den Organisationszielen zu folgen. Dies kann beispielsweise erreicht werden, indem es gelingt, dass Organisationsmitglieder die Ziele der Organisation als eigene Ziele übernehmen (z.B. durch Gewinnbeteiligungen bei Unternehmen). Ziele werden somit in verschiedenen Kontexten verwendet und erfüllen nach Bea (2004, S. 1677) drei unterschiedliche Funktionen: (i) die Entscheidungsfunktion zur Bewertung von Alternativen, (ii) die Koordinationsfunktion zur Integration von Teilaktivitäten sowie die (iii) Motivationsfunktion zur Motivation von Mitarbeitern. Diese drei Funktionen sind auch beim Beitritt von Softwareagenten zu Multiagenten-Organisationen zu betrachten: Die Ziele des beitretenden Softwareagenten sind im Allgemeinen nicht deckungsgleich zu denen der Multiagenten-Organisation. Vielmehr muss letztere bereits bei der Beitrittsverhandlung auf die Ausgestaltung des Beitritts im Sinne ihrer eigenen Ziele achten. Insbesondere die Motivationsfunktion, beispielsweise durch leistungsabhängige Kompensationszahlungen, beeinflusst die folgende Zusammenarbeit und Stabilität der Mitgliedschaft selbst sowie der Interaktion innerhalb der Multiagenten-Organisation.

Formale Struktur Mit Hilfe der formalen Struktur soll das Verhalten der Mitglieder in Richtung der Organisationsziele ausgerichtet werden (Kieser und Walgenbach, 2007, S. 6) und stellt somit die Verknüpfung des Merkmals Zweckorientierung auf verschiedenen Ebenen dar: Aus Organisationszielen, die Entscheidungsfunktionen erfüllen, können beispielsweise auf Abteilungsebene Ziele mit Koordinationsfunktion und auf Mitarbeiterebene Ziele mit Motivationsfunktion herausgebildet werden. Dieses Beispiel stellt nochmals den zentralen Aspekt der Motivationsfunktion im Kontext von Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen heraus: Die Motivationsfunktion muss bereits bei Beitritt beachtet werden und aus Sicht der Organisation auf deren übergeordnetes Ziel hin ausgerichtet werden. Diese Motivationsfunktion findet sich dann ebenfalls in der formalen Struktur der Organisation wieder. Das Merkmal der formalen Struktur findet sowohl in Bezug zur Organisation als Instrument sowie als Institution Verwendung. Mitglieder einer Organisation müssen sich den Regeln bzw. der formalen Struktur unterwerfen, um ihr Verhalten für andere Mitglieder in gewissen Grenzen vorhersagbar zu gestalten (Luhmann, 1987, S. 41). Dieses Verhältnis zwischen verschiedenen Mitgliedern führt zu einer weiteren Aufgabe der formalen Struktur: Die Arbeitsteilung bildet die Basis der organisationalen Wertschöpfung und wird durch die formale Struktur manifestiert (Schreyögg, 1996, S. 10). Erst durch die Arbeitsteilung können Organisationen effizient arbeiten. So beteiligen sich auch Softwareagenten an der Arbeitsteilung einer Multiagenten-Organisation, beispielsweise durch die Bereitstellung von Diensten.

Diese drei Merkmale bilden die Grundlage für das institutionelle Organisationsverständnis, wobei insbesondere die Charakterisierung einer Organisation als soziales System den Begriff vom instrumentellen Verständnis abgrenzt. Dabei wird das Institut Organisation als offenes System verstanden, deren Mitglieder wechseln können und die Mitglieder somit nicht als feste zur Verfügung stehende Menge angesehen werden. Erst aus diesem Begriffsverständnis heraus wird es möglich, die Beitrittsentscheidungen der einzelnen potentiellen Mitglieder zu betrachten. Auch wird die Perspektive zu der Möglichkeit einer Mitgliedschaft in mehreren Organisationen geöffnet (Kieser und Walgenbach, 2007, S. 3). Dabei taucht vermehrt die Frage auf, welche Kriterien an die Mitgliedschaft in einer Organisation zu stellen sind bzw. wann ein Akteur als Mitglied einer Organisation bezeichnet werden kann. Auf den Aspekt der Mitgliedschaft aus der Perspektive verschiedener Organisationstheorien geht Abschnitt 2.3.2.1 vertiefend ein.

2.3.1.2 Multiagenten-Organisationen

Für den Begriff *Organisation* findet sich im Kontext von Multiagentensystemen – ähnlich dessen Verwendung im Forschungsfeld der Organisationstheorie (vgl. Abschnitt 2.3.1.1) – keine einheitliche Definition. Insbesondere ist zunächst unklar, ob es sich hierbei um ein Instrument oder eine Institution handelt. Bereits Bond und Gasser beschäftigen sich mit *Organisation* im Kontext von Softwareagenten und legen den Fokus insbesondere auf die Reglementierung und dadurch Einhaltung von erwartetem Verhalten, betrachten diese somit aus einer instrumentellen Perspektive: „*An organization can provide a framework of constraints and expectations about the behavior of agents (e.g., a set of "roles") that focuses the decision making and action of particular agents*“ (Bond und Gasser, 1988, S. 20). Eine Definition von Multiagenten-Organisationen setzt durch die uneinheitlich und unklar verwendeten Begriffe jedoch voraus, dass zunächst auch Ansätze betrachtet werden, die nicht direkt den Begriff der *Organisation* verwenden, aber konzeptionelle Ähnlichkeiten aufweisen. Hierzu untersucht der nachfolgende Abschnitt die Verwendung der Begriffe *Kongregation*, *Koalition* sowie *Team* und grenzt diese zu Multiagenten-Organisationen ab.

Bereits Malone (1987) untersucht verschiedene Koordinationsstrukturen in sozialen Organisationen in Hinblick auf deren Informationsverarbeitung ohne dabei direkt Softwareagenten zu erwähnen. Die Informationsverarbeitung selbst ist hingegen eine Kerneigenschaft informationstechnischer Systeme und somit auch von Softwareagenten in Organisationen. So widmet sich Malone den Analogien zwischen sozialen Organisationsformen und Computerprogrammen sowie deren Übertragung auf verteilte Systeme: „*Observations of, and in some cases detailed theories about, human organizations can ... be used to design highly parallel computer systems*“ (Malone, 1990, S. 57). Bereits zuvor hatte Fox (1981) begonnen, Ansätze aus der Organisationstheorie auf verteilte Computersysteme zu übertragen, beschränkt sich dabei jedoch auf Organisationsstrukturen. So konstatieren auch Bond und Gasser (1988, S. 20), dass die bis dahin veröffentlichten Ansätze zur Verknüpfung von Organisationstheorie und verteilter künstlicher Intelligenz sich ausschließlich auf strukturelle Modelle von Organisationen fokussieren.

Seit dem haben sich verschiedene Forschungsstränge mit jeweils anderem Fokus herausgebildet. Gasser (2001), Ferber (2001, S. 113 ff.), Ferber et al. (2004) und Kirn und Gasser (1998) stellen Ansätze dar, die versuchen, bekanntes Wissen über Organisationen aus der Organisationstheorie auf ähnliche Strukturen im Kontext von Zusammenschlüssen von Softwareagenten zu übertragen. Diese Forschungsansätze greifen verschiedene Entwicklungslinien der Organisationstheorie auf und beziehen sich beispielsweise auf Cyert und March (1963) und Mintzberg (1979).

Horling und Lesser (2004) geben hingegen eine Übersicht über verschiedene Organisationstypen und -strukturen in Multiagentensystemen und schließen dabei ebenfalls Ansätze mit ein, die nicht explizit den Term *Multiagenten-Organisation* verwenden. Die Autoren geben einen Überblick über ein inhaltlich breites Feld und verdeutlichen damit den Mangel an einer einheitlichen Begriffsdefinition bzw. Abgrenzung im Bereich der Multiagenten-Organisationen. Dabei finden sich neben dem Term *Multiagenten-Organisation* in der Literatur weitere Begriffe, die in einem gewissen Zusammenhang zu dem hier vorgestellten Begriff stehen. Dabei kann es sich um angrenzende Teilgebiete oder auch um Synonyme zu dem hier vorgestellten Gegenstand handeln. Dabei fallen drei Begriffe auf, die in der Literatur für die Beschreibung von Zusammenschlüssen von Softwareagenten zu finden sind und deren Bezug bzw. deren Abgrenzung zum Begriff der Multiagenten-Organisation nachfolgend erläutert wird.

Kongregation. Der Begriff der Kongregation findet sich im Vergleich zu den anderen beiden Begriffen Koalition und Team nur selten in der Literatur. Insbesondere Brooks und Durfee (2003) verwenden den Begriff *Kongregation*, um eine Art Zusammenschluss von Softwareagenten zu beschreiben, die ähnliche Fähigkeiten besitzen. Dieses Vorgehen dient zur Reduktion der Suchkosten in großen Multiagentensystemen. Hierfür müssen die Softwareagenten in der Lage sein, Kongregationen anderen Softwareagenten zu beschreiben und so den Zusammenschluss zu ermöglichen. Zur Komplexitätsreduktion verwenden Brooks und Durfee (2003, S.152) spezielle *Labeleragenten*, die eine Einteilung vornehmen.

Das Konzept der Kongregation ist auf die erleichterte Auffindung von anderen Softwareagenten ausgelegt und beschäftigt sich nicht direkt mit Fragen der Zusammenarbeit selbst. Im Zusammenhang mit Multiagenten-Organisationen können Kongregationen als Vorstufe Verwendung finden, die dem Auffinden potentieller Mitglieder bzw. Multiagenten-Organisationen dient. Ähnlich dem sozialen Gebilde Organisation können Kongregationen zur Strukturierung von Arbeitsteilung genutzt werden. Dabei sind letztere jedoch durch den Zusammenschluss von Softwareagenten mit gleichen Fähigkeiten in ihrem Tätigkeitsfeld beschränkt, während Organisationen hingegen gerade durch den Zusammenschluss von Akteuren mit unterschiedlichen Fähigkeiten eine breiter gefächerte Arbeitsteilung ermöglichen.

Koalition. Koalitionen und insbesondere die hierfür benötigte Koalitionsformation sind Forschungsbereiche der kooperativen Spieltheorie, die die Fragestellung betrachten, welche Akteure zusammenarbeiten sollten und wie diese Kooperation ausgestaltet wird. In der Multiagenten-Forschung finden sich zahlreiche Ansätze, die Verfahren

aus der Spieltheorie übernehmen und in Multiagentensystemen einsetzen. Wooldridge (2009, S. 272) beispielsweise nutzt ebenfalls Ansätze aus der Spieltheorie, um Koalitionen zu formen. Dabei wird der Gesamtprozess in drei Phasen unterteilt: (1) Koalitionsformation, wobei festgelegt wird, welche Softwareagenten miteinander arbeiten, (2) Teamformation, wobei sich die Softwareagenten auf die Art und den Umfang der Zusammenarbeit verständigen und (3) die Durchführung der koordinierten kooperativen Aktionen. Dabei fällt auf, dass in der Phase der Koalitionsformation keine Vereinbarungen über die Aufgabenverteilung innerhalb der Koalition getroffen werden, sondern dies der darauffolgenden Teamformationsphase vorbehalten ist. Da ein Softwareagent eine Koalition jeder Zeit wieder verlassen kann, ist der Untersuchungsgegenstand dieser Arbeiten, welche Koalitionen stabil sind und die Softwareagenten somit keinen Anreiz zum Austritt haben. Dabei ist es fraglich, inwiefern diese Phasen unabhängig voneinander betrachtet werden können, insbesondere wenn ein Softwareagent Mitglied in mehreren Koalitionen sein kann. Da der Softwareagent durch die Mitgliedschaft selbst in Phase 1 zunächst keine weiteren Verpflichtungen hat, hat er unter Umständen keinerlei Nachteile bei Mitgliedschaften in beliebig vielen Koalitionen. Die Formation von Koalitionen kann somit nicht unabhängig von den damit zusammenhängenden Rechten und Pflichten jedes einzelnen Softwareagenten durchgeführt werden.

Ebenfalls Ansätze aus der kooperativen Spieltheorie verfolgen Klusch und Gerber (2002), die dabei nicht explizit einen – wie oben beschriebenen – mehrstufigen Prozess voraussetzen. Die Autoren entwickeln einen Koalitionsformationsalgorithmus für stochastische kooperative Spiele, in denen nur stochastische anstatt deterministische Auszahlungen gegeben sind und somit eher für eine Anwendung bei Softwareagenten geeignet sind. Der Algorithmus wird stets lokal bei den Softwareagenten ausgeführt und unterstützt somit auch die verteilten Informationen und Interessen im Kontext von Softwareagenten. Für eine Menge an Softwareagenten muss dieser Algorithmus allerdings allen beteiligten Softwareagenten bekannt sein und diese müssen sich auf dessen Anwendung in einem vorgelagerten Schritt geeinigt haben. Erst dann ist die Ausführung des Algorithmus möglich.

Diese Ansätze setzen somit voraus, dass es eine zentrale Stelle gibt, die ein System zur Koalitionsformation vorgibt oder alle potentiellen Softwareagenten sich auf ein entsprechendes Verfahren geeinigt haben und im Moment der Koalitionsbildung bereits zur Verfügung stehen. Ein nachträglicher Beitritt oder ein Ausscheiden von Softwareagenten ist in aller Regel nicht vorgesehen. Ein erster dezentraler Ansatz wird bereits von Ketchpel (1993, S. 75 ff.) beschrieben, der bereits zu Beginn der Arbeit die Offenheit der betrachteten Multiagentensysteme betont. Trotzdem ist auch bei diesem Ansatz ein einheitliches Verfahren und die zeitgleiche Durchführung der Koalitionsverhandlungen Voraussetzung.

Team. Neben der Koalitions- ist die Teamformation eine in der Literatur häufig betrachtete Forschungsrichtung. Wooldridge (2009, S. 271 f.) sieht ein Team als eine Koalition mit weiter ausgeprägten Verträgen (siehe oben), geht auf den Begriff selbst und

die genaue Unterscheidung zu einer Koalition allerdings nicht weiter ein. Jennings (1995, S. 206 ff.) verwendet ebenfalls den Begriff Team und setzt dabei voraus, dass sämtliche Teammitglieder ein gemeinsames Ziel verfolgen. Diese Voraussetzung lässt sich jedoch nicht allgemein auf Multiagenten-Organisationen übertragen. Es ist durchaus möglich, dass verschiedene Mitglieder keine gemeinsamen Ziele haben und die Adressierung dieser Zielfdivergenz bildet einen zentralen Punkt bei der Betrachtung von Multiagenten-Organisationen. Insbesondere durch die Ausrichtung einer Multiagenten-Organisation auf Dauerhaftigkeit, können sich über diesen vergleichsweise langen Zeitraum die Ziele der Multiagenten-Organisation ändern und bereits dadurch zu Konflikten mit den Zielen der einzelnen Mitglieder führen. Es wird somit nicht angenommen, dass jeder Softwareagent das übergeordnete Ziel der Multiagenten-Organisation selbst verfolgt, vielmehr muss diese ihre Mitglieder mit entsprechenden Verträgen zu einem Handeln bewegen, dass dem Wohle der Multiagenten-Organisation dient. Dies kann dazu führen, dass das Ziel der Organisation unter Umständen zu einem Ziel deren Mitglieder wird.

Beavers und Hexmoor (2001) grenzen Koalitionen und Teams über den Einfluss der individuellen über Nutzenfunktionen repräsentierte Ziele der beteiligten Softwareagenten am Ergebnis der Koalition bzw. des Teams ab: Die Mitglieder eines Teams verfolgen die Lösung eines bestimmten Problems, wozu diese auch zwischenzeitliche Einschnitte beim erreichten Nutzen in Kauf nehmen. In einer Koalition hingegen steht nach Beavers und Hexmoor (2001, S. 575) der langfristige Nutzen und dessen Stabilität im Vordergrund. Auch Beavers und Hexmoor (2001, S. 575) setzen für die Bildung eines Teams das Vorhandensein mindestens eines gemeinsamen Ziels voraus, was den Begriff von Multiagenten-Organisationen abgrenzt.

Beitrittsentscheidungen werden bei der Teamformation nicht betrachtet, da diese Ansätze von einer beständigen Menge an Teammitgliedern ausgehen (Griffiths und Luck, 2003, S. 17). Auch dieser Punkt widerspricht der Dauerhaftigkeit von Multiagenten-Organisationen, da die Menge an Mitgliedern nicht sich verändernden Bedingungen angepasst werden kann.

Definitionen im Rahmen der VKI zielen ähnlich der Definitionen von Organisationen im betriebswirtschaftlichen Verständnis insbesondere auf eine gewisse Dauerhaftigkeit ab, so erzielen bereits Durfee et al. (1987) mit organisatorischen Strukturen eine auf Langfristigkeit bzw. Dauerhaftigkeit ausgerichtete Netzwerkkoordination in verteilten Systemen. Auch Kirn (1996a) sieht insbesondere die Dauerhaftigkeit als Unterscheidungsmerkmal zu Multiagentensystemen: *„Gegenüber 'nicht organisierten' Agentensystemen zeichnen sich Multiagenten-Organisationen vor allem durch ihre Dauerhaftigkeit, die Existenz formalisierter Strukturen, ihre Zweck- und Zielgerichtetheit sowie durch eine gewisse wechselseitige Unabhängigkeit von Agent und Agentenorganisation aus“* (Kirn, 1996a, S. 159). Zusammen mit der Abgrenzung zu verwandten Begriffen aus dem Bereich der Multiagentensysteme, lässt sich mittels der Organisationstheorie und VKI das Verständnis von Multiagenten-Organisationen im Rahmen dieser Arbeit herausstellen: Im Rahmen dieser Arbeit wird unter einer Multiagenten-Organisation ein Zusammenschluss von mehreren Softwareagenten ver-

standen, die durch vertraglich bindende Regelungen im Rahmen dieses Zusammenschlusses besondere Rechte und Pflichten haben (vgl. hierzu Abschnitt 2.3.2.2). Dieser Zusammenschluss muss auf Langfristigkeit ausgelegt und darf nicht ohne Konsequenzen kündbar sein. Er darf somit nicht ausschließlich der Erfüllung eines kurzfristigen Ziels dienen. Multiagenten-Organisationen sind somit ein spezieller Typ von Multiagentengesellschaften, die sich insbesondere durch die Dauerhaftigkeit und der hierdurch erreichten Stabilität von anderen Zusammenschlüssen von Softwareagenten unterscheiden. Diese Eigenschaften lassen sich in der folgenden Definition zusammenfassen:

Definition 6 (Multiagenten-Organisation) *Eine Multiagenten-Organisation ist ein dauerhafter Zusammenschluss von mehreren unabhängigen Softwareagenten, die jeweils ihre individuellen Ziele verfolgen und durch vertragliche Regelungen zur Mitwirkung an der Erfüllung eines vorgegebenen Organisationsziels motiviert werden.*

2.3.2 Mitgliedschaft

Die einschlägige Literatur zu Organisationstheorien bietet keine einheitliche Definition zur Mitgliedschaft in Organisationen. Stattdessen unterscheiden sich die Definitionen in wesentlichen Punkten und so sieht beispielsweise Luhmann (2000, S. 90) die Arbeitnehmer einer Organisation nicht als Mitglieder an, während March und Simon (1958, S. 109) bzw. Cyert und March (1963, S. 27) selbst Kunden und Lieferanten eines Unternehmens zu den Mitgliedern dieser Organisation zählen. Abgesehen von diesen beiden existieren in der Literatur eine Vielzahl unterschiedlicher Auffassungen, die den Kreis der Mitglieder einer Organisation zwischen diesen beiden konträren Positionen sehen. Dabei bezieht sich die organisationstheoretische Literatur ausschließlich auf die Mitgliedschaft von Menschen in Organisationen und betrachtet nicht, inwiefern auch Softwareagenten Mitglied einer Organisation sein können. Grundsätzlich ist anzumerken, dass dem Mitgliedschaftsbegriff selbst überwiegend von Autoren mit institutionellem Organisationsverständnis nachgegangen wird. Da in der Literatur zum instrumentellen Organisationsbegriff das „Organisieren“ einer vorgegebenen Menge an Akteuren, Ressourcen, etc. im Fokus steht, stellt sich hier nur zu Beginn die Frage, über welche (personellen) Ressourcen verfügt werden kann, was im Allgemeinen durch Verträge geregelt ist.

Neben der Unterscheidung in Mitglieder und Nicht-Mitglieder einer Organisation, wird in diesem Zusammenhang auch der Begriff der Organisationsgrenze zur Unterscheidung von Organisation und Umwelt genannt (Bea und Göbel, 2002, S. 6). Nach Schreyögg ist die „Grenze zwischen Organisation und Umwelt ... absichtsvoll hergestellt und weist ein gewisses Maß an Stabilität auf“ (Schreyögg, 2008, S. 9). Erst die Stabilität dieser Grenzziehung ermöglicht eindeutig identifizierbare und ebenfalls stabile Mitgliedschaften (Schreyögg, 2008, S. 9). Reichwald (2004, S. 998 ff.) untersucht die Definition von Organisationsgrenzen aus den Perspektiven (i) der Transaktionskostentheorie, (ii) der Verfügungsrechtetheorie, (iii) des Ressourcenabhängigkeitsansatzs, sowie (iv) der Ressourcentheorie. Dabei werden die Organisationsgrenzen in Abhängigkeit des jeweiligen Fokus der gewählten Perspektive definiert: So kann beispielsweise für die Transaktionskostentheorie „die Organisationsgrenze

... als Trennung zwischen der Organisation als Träger der Leistungserstellung und dem umgebenden Marktsystem gesehen werden“ (Reichwald, 2004, S. 1001), für die Verfügungsrechtetheorie hingegen „als Bündel von Property Rights über mehrere Güter, die sich im Besitz einer Institution befinden“ (Reichwald, 2004, S. 1001) definiert werden. Insbesondere die Abhängigkeit einer Organisation von ihren (Personal-)Ressourcen lässt eine Grenzziehung im Sinne des Mitgliedschaftsbegriffs zu. Pfeffer und Salancik (1978) sehen hierfür den Einfluss der Organisation als wesentliches Kriterium: „*The boundary is where the discretion of the organization to control an activity is less than the discretion of another organization or individual to control that activity*“ (Pfeffer und Salancik, 1978, S. 32). Die Abwägung der Einflussnahme einer Organisation bezüglich derer einer anderen Organisation auf den gleichen Akteur kann hingegen nur als Kriterium für die Grenze der Organisation dienen, da Einflussnahme stets von der individuellen Entscheidung des Akteurs abhängig ist. Scott (1981, S. 247 ff.) sieht das Kriterium der Verhaltenskontrolle als wichtigen Indikator zur Bestimmung von Organisationsgrenzen, fordert allerdings noch eindeutige normative Kriterien. Diese Kriterien an die Bestimmung der Organisationsgrenzen können als Grundlage einer eindeutigen Definition der Mitgliedschaft in Organisationen dienen. Auf diese Weise kann festgelegt werden, ab wann ein potentielles Mitglied einer Organisation zu einem tatsächlichen Mitglied dieser wird, zunächst unabhängig davon, ob es sich hierbei um einen Menschen oder einen Softwareagenten handelt.

Die oben bereits angeführten Verträge im Kontext des instrumentellen Organisationsbegriffs können hier als normatives Kriterium der Mitgliedschaft dienen und finden bereits im Rahmen der Organisation als Institution Verwendung. So definieren Kieser und Walgenbach Mitgliedschaft als „*eine juristische Beziehung zwischen einer Organisation und einem Individuum, die durch beidseitige Anerkennung zustande kommt*“ (Kieser und Walgenbach, 2007, S. 15). Verträge zwischen mehreren Parteien beinhalten im schuldrechtlichen Verständnis eine oder mehrere Pflichten zwischen Vertragsparteien (§241 BGB). In einem Arbeitsvertrag, der eine Person an eine Organisation bindet, verpflichtet sich der Mitarbeiter unter anderem zur Erbringung von Arbeitsstunden, während sich die Organisation insbesondere zur Zahlung von Gehalt verpflichtet. Für dieses Kriterium der Mitgliedschaft und somit dem Beitritt zu einer Organisation ist die interne Struktur dieser Organisation zunächst nicht relevant. Im Zusammenhang mit dem Beitritt von Softwareagenten zu (Multiagenten-)Organisationen ist dabei anzumerken, dass zunächst die Möglichkeit für Softwareagenten existieren muss, bindende Verträge mit einer (Multiagenten-)Organisation einzugehen. Da es sich bei Softwareagenten im juristischen Sinn weder um eine natürliche noch juristische Person handelt, können diese Verträge nicht gegen den Softwareagenten selbst wirken. Hingegen muss der Softwareagent im Auftrag einer natürlichen oder juristischen Person handeln, die somit alle durch den Softwareagenten geschlossenen Verträge für und gegen sich anrechnen lassen muss. In dieser Arbeit wird angenommen, dass ein Softwareagent durch eine natürliche oder juristische Person in Verkehr gebracht wird, die sich über die Handlungsweise des Softwareagenten bewusst ist und gegen die die vom Softwareagenten geschlossenen Verträge wirken. Abhängig vom Kriterium der Mitgliedschaft kann es durch diese Konstellation dazu kommen, dass durch den Beitritt eines Softwareagenten zu einer (Multiagenten-)Organisation, auch die natürliche oder juristische Person,

die den Softwareagenten in Verkehr gebracht hat, Mitglieder ebendieser wird. Unabhängig vom Ergebnis der Beurteilung der Mitgliedschaftsbeziehung betrachtet diese Arbeit Entscheidungen von Softwareagenten zum Beitritt zu (Multiagenten-)Organisationen, ebenfalls unabhängig von etwaigen weiteren Folgen wie dem faktischen weiteren Beitritt einer natürlichen Person. Ähnlich der *Verantwortlichkeit* bei der Betrachtung von Softwareagenten als maschinelle Aufgabenträger in Abschnitt 2.1.3 ist die Vertragsfähigkeit abhängig von nationalen Gesetzen. Eine allgemeine abschließende Beurteilung der weiteren Folgen des Beitritts eines Softwareagenten für andere Akteure ist daher im Rahmen dieser Arbeit nicht möglich und wird im weiteren nicht betrachtet.

2.3.2.1 Mitgliedschaft in der Organisationstheorie

Der Begriff der Mitglieder von Organisationen wird in der Literatur zur Organisationstheorie häufig verwendet, Mitgliedschaft selbst jedoch nur in wenigen Fällen definiert. Für die verschiedenen Ansätze der Organisationstheorie bilden dabei unterschiedliche Kriterien die Grundlage für die Definition von Mitgliedschaft, die oftmals nur implizit angenommen wird. Um jedoch auf die Bedeutung des Mitgliedschaftsbegriffs verschiedener Organisationstheorien eingehen zu können, werden zunächst das *Organisationsverständnis* sowie die betrachteten *Fragestellungen* dieser Theorien untersucht. Die nachfolgenden Abschnitte gehen auf Organisationstheorien verschiedener Epochen ein, beginnend mit dem Bürokratie-Ansatz von Max Weber. Anschließend werden weitere Organisationstheorien betrachtet, die einen direkten Bezug zur Betrachtung von Einzelentscheidungen mit sich bringen.

2.3.2.1.1 Bürokratie-Ansatz von Max Weber

Der Bürokratieansatz nach Weber (1922) untersucht das Entstehen von Organisationen sowie deren Abläufe unter Berücksichtigung der *Bürokratie* als Herrschaftsform. Schreyögg (2004c) schreibt zur Unterscheidung zu anderen Ansätzen: *„Im Gegensatz zu den anderen klassischen Ansätzen ist Weber nicht daran gelegen, Prinzipien zur Optimierung betrieblicher Organisation zu entwickeln; sein Denkansatz ist explikativ, er will das Aufkommen und das Funktionieren großer Organisationen ... mit dem Idealtypus der Bürokratie als technisch gesehen rationalste Form der Herrschaftsausübung verständlich machen“* (Schreyögg, 2004c, S. 1071). Den Begriff Mitglied verwendet Weber (1922) dabei in zahlreichen Zusammenhängen, ohne ihn weiter zu definieren. Aus den weiteren Ausführungen wird jedoch deutlich, dass Weber (1922) Mitglieder hauptsächlich als Anordnungsempfänger auffasst, unter der *„Vorstellung ... daß also der typische legale Herr: der 'Vorgesetzte', indem er anordnet und mithin befiehlt ... [und] daß ... der Gehorchende nur als Genossen und nur 'dem Recht' gehorcht. Als Vereinsgenosse, Gemeindegensosse, Kirchenmitglied, im Staat: Bürger“* (Weber, 1922, S. 125). Mitglieder in einer solchen Herrschaftsform Organisation haben somit feste Zuständigkeiten und sind an Weisungen anderer gebunden. Dabei wird die Bürokratie als Herrschaftsform angesehen, die von jeglichen persönlichen Interessen der Beteiligten abstrahiert. Hieraus ergibt sich eine Gehorsamkeit für erhaltene Befehle, die *„ohne Rücksicht auf die eigene Ansicht über den Wert oder Unwert des Befehls als solchen“* (Weber, 1922, S. 123) erfolgen soll. Dieses Idealbild eines bürokratischen Befehlsempfängers wiederum ermöglicht die Austauschbarkeit der einzelnen Mitglieder. Stellen in

einer Organisation können somit unabhängig vom umsetzenden Mitglied geschaffen und strukturiert werden. Diese Annahme einer absolut stringenten hierarchischen Strukturierung ist allerdings nicht vereinbar mit Überlegungen, dass Individuen Mitglied in mehreren Organisationen sein können (Kieser und Walgenbach, 2007, S. 40). Auch die Annahme einer vollständig entpersonalisierten Durchführung übertragener Aufgaben, ohne die Einbringung eigener Interessen der Mitglieder, schränkt die Aussagekraft des Bürokratie-Ansatzes ein. Nichtsdestotrotz lassen sich für den Mitgliedschaftsbegriff erste Aspekte aufgreifen, welche in Tabelle 2.1 zusammenfassend dargestellt sind.

Tabelle 2.1: Bürokratie-Ansatz, angelehnt an Kieser und Kubicek (1992, S. 34)

Organisations- verständnis	Organisation als Herrschaftsform
Fragestellung	Wie ist die Funktionsweise von organisierter Herrschaft durch Bürokratie in Unternehmen zu erklären?
Bedeutung für den Mitgliedschaftsbegriff	Mitglieder haben feste Zuständigkeiten und sind in der Struktur austauschbar

2.3.2.1.2 Anreiz-Beitrags-Theorie

Die Anreiz-Beitrags-Theorie wird den verhaltenswissenschaftlichen Ansätzen der entscheidungsorientierten Organisationstheorie zugeordnet und untersucht, wie Organisationsmitglieder durch Anreize zum Leisten von Beiträgen bewegt werden können (Osterloh, 2004, S. 224 ff.). Dabei wird die Organisation als System aus miteinander kooperierender Mitglieder, die Menge an Mitgliedern im Gegensatz zu vorangegangenen Ansätzen hingegen nicht mehr als gegeben, sondern als variable Größe betrachtet. Individuen können einer Organisation somit beitreten und hierdurch zum Mitglied dieser Organisation werden sowie durch Austritt diesen Status wieder verlieren: *„the connection of any member with an organization is necessarily intermittent, and there is frequent substitution of persons“* (Barnard, 1938, S. 70). Mit der Beitrittsentscheidung selbst setzen sich insbesondere March und Simon (1958) auseinander (vgl. Abschnitt 2.4.1). Dabei fokussieren sie ihre Ausführungen zwar auf Mitarbeiter eines Unternehmens, wählen den Mitgliedschaftsbegriff jedoch deutlich weiter und zählen neben den Mitarbeitern insbesondere Investoren, Zulieferer, Händler und Kunden hinzu (March und Simon, 1958, S. 109). Die Notwendigkeit zur Beschränkung auf Mitarbeiter in Unternehmen begründen March und Simon (1958) durch die vorhandenen Vorarbeiten: *„Taken too literally, this conception of organizations incorporates almost any knowledge about human behavior as a part of organization theory. ... Labor mobility has been studied at some length by both economists and social psychologists.“* (March und Simon, 1958, S. 109). Die Erweiterung des Ansatzes auf andere Typen von Mitgliedern ist jedoch von March und Simon (1958, S. 125 ff.) ebenfalls vorgesehen und wird in Analogie zu Arbeitsverträgen betrachtet. Zwar schlagen March und Simon (1958, S. 109) die Verwendung einer funktionalen Definition von Mitgliedschaft gegenüber der gebräuchlichen rechtlichen Abgrenzung vor, explizieren die Kriterien jedoch nicht weiter, sondern stellen ein allgemeines Modell für die Beitritts- bzw. Austrittsentscheidung als

Verhältnis der Anreize durch die Organisation zu den Beiträgen des Mitglieds vor (March und Simon, 1958, S. 112 f.). Hieraus lässt sich ableiten, dass Mitglieder im Sinne der Anreiz-Beitrags-Theorie solche Akteure sind, die durch Anreize von einer Organisation zum Leisten von Beiträgen zu ebendieser Organisation bewegt werden. Mitglieder leisten ihre Beiträge demnach auch nur, wenn der individuelle Nutzen für die erhaltenen Anreize den Nutzen der zu leistenden Beiträge übersteigen. Tabelle 2.2 fasst die Kernaspekte der Anreiz-Beitrags-Theorie zum Mitgliedschaftsbegriff zusammen.

Tabelle 2.2: Anreiz-Beitrags-Theorie

Organisationsverständnis	Organisation als System von kooperierenden Mitgliedern
Fragestellung	Wie können Mitglieder durch Anreize zum Leisten von Beiträgen bewegt werden?
Bedeutung für den Mitgliedschaftsbegriff	Mitglieder stehen der Organisation nur in dem Maß zur Verfügung, wie ausreichende Anreize geschaffen werden

2.3.2.1.3 Entscheidungslogisch-mathematische Ansätze

Die entscheidungslogisch-mathematischen Ansätze der organisatorischen Entscheidungsfor-schung sind durch den Einsatz formaler mathematischer Modelle in den Wirtschafts- und Sozialwissenschaften entstanden (Schreyögg, 2004c, S. 1077). Ziel ist die Optimierung von organisatorischen Gestaltungsentscheidungen mit Hilfe mathematischer Modelle und stehen somit in enger Relation zu den Verfahren des Operations Research. Dabei lassen sich sowohl Arbeiten identifizieren, die vertikale Interdependenzbeziehungen zwischen verschiedenen Hierarchieebenen der Aufbauorganisation betrachten, als auch entsprechende Arbeiten, die den Fokus auf die horizontalen Interdependenzen auf der gleichen Hierarchieebene legen.

Effizienz-Konzept

Die Literatur zu entscheidungslogisch-mathematischen Ansätzen orientiert sich an der präskriptiven Entscheidungslehre (Schreyögg, 2004c, S. 1077 f.). Die Team-Theorie nach Marschak (1955) bildet hierfür die ersten Ansätze in der Organisationstheorie. Dabei versucht die Team-Theorie Verhaltens- und Kommunikationsregeln derart zu optimieren, dass unter Berücksichtigung dieser horizontalen Interdependenzen die Gesamtkosten minimiert werden. Marschak (1955) nimmt dabei an, dass die involvierten Akteure dem übergeordneten Ziel des Teams folgen, ihre eigenen Ziele somit vollständig zurückstellen. Dieser Annahme kann jedoch für die Betrachtung dieser Arbeit nicht gefolgt werden, da das Verfolgen individueller Ziele als ein wesentliches Merkmal von Softwareagenten angesehen wird. Auch die Mitwirkung an der Erfüllung eines Organisationsziels ist stets beeinflusst durch die individuellen Ziele der Akteure. Als Weiterentwicklung der Arbeiten von Marschak (1955) bzw. Marschak und Radner (1972) kann das Effizienz-Konzept nach Frese et al. (2012) angesehen werden. Dieses befasst sich ebenfalls mit horizontalen Interdependenzbeziehungen einer Hierarchieebene, beispielsweise zwischen verschiedenen Abteilungen und berücksichtigt dabei im Gegensatz zur Team-Theorie die individuelle Zielstellung der einzelnen Akteure.

Die Unterscheidung zwischen *Entscheidungen* sowie deren *Realisation* ist ein wesentliches Merkmal des entscheidungsorientierten Konzepts der Organisationsgestaltung nach Frese et al. (2012) und geht zurück auf Grochla (1966, S. 98 ff.). Einer Mitgliedschaft geht im Allgemeinen ein Beitritt sowie die damit zusammenhängende Beitrittsentscheidung voraus. Abschnitt 2.4.1 geht detaillierter auf diese Beitrittsentscheidung aus organisations-theoretischer Sicht ein. Für den Mitgliedschaftsbegriff leitet sich hieraus ab, dass zuvor die Entscheidung zugunsten eines Beitritts, sowohl des beitretenden Softwareagenten als auch der entsprechenden Multiagenten-Organisation, sowie die Realisation des Beitritts selbst vorausgegangen ist.

Das Effizienz-Konzept geht davon aus, dass Mitglieder einer Organisation ebenfalls Mitglieder von einer oder mehreren Teileinheiten der Organisation sind, beispielsweise Abteilungen. Diese beiden Dimensionen der Mitgliedschaft unterscheiden sich jedoch grundlegend: Während die Mitgliedschaft in einem Unternehmen beispielsweise durch einen Arbeitsvertrag – und somit zwei übereinstimmende Willenserklärungen – manifestiert werden kann, kann die Zuordnung zu unternehmensinternen Teileinheiten verschiedenste Abstimmungsformen annehmen und im Zweifel hierarchisch – ohne Einwilligung des Betroffenen – bestimmt werden. Im Effizienz-Konzept nach Frese et al. (2012) werden Interdependenzen betrachtet, die durch die Mitgliedschaft in mehreren organisatorischen Teileinheiten entstehen. Mehrfachmitgliedschaften in verschiedenen Organisationen stehen nicht im Fokus des Effizienz-Konzepts, jedoch sind wesentliche Merkmale der Interdependenzen bei Mitgliedschaft in mehreren organisatorischen Teileinheiten auch auf entstehenden Interdependenzen bei Mitgliedschaft in verschiedenen Organisationen übertragbar. In beiden Fällen erlangt das Mitglied im Allgemeinen eine Entscheidungsbefugnis über ein abgegrenztes Entscheidungsfeld (vgl. Abschnitt 2.4.1), welche sich unter Umständen überschneiden können. Ist ein Softwareagent beispielsweise Mitglied in zwei Multiagenten-Organisationen und nutzt dieselben Ressourcen zur Bereitstellung von Diensten in beiden Multiagenten-Organisationen, überschneiden sich die Entscheidungsfelder mindestens in der Verwendung der eigenen Ressourcen. Interdependenzen zwischen Mitgliedschaften in organisatorischen Teileinheiten unterscheiden sich jedoch in einem wesentlichen Punkt von Interdependenzen zwischen Mitgliedschaften in Organisationen: Während erstere durch eine höhere hierarchische Ebene, beispielsweise mit Hilfe organisatorischer Regeln, aufgelöst oder gesteuert werden können, existiert für die Interdependenzen zwischen Mitgliedschaften in Organisationen im Allgemeinen keine solche Instanz. Die Mitgliedschaft ist demnach ausschließlich der gesetzgebenden Instanz untergeordnet, die Grenzen setzt, aber im Allgemeinen nicht aktiv in den Abstimmungsprozess einschreitet. Die Auflösung oder Steuerung der Interdependenzen, die durch die Mitgliedschaft eines Softwareagenten in mehreren Multiagenten-Organisationen entstehen, ist somit im Wesentlichen dem Softwareagenten selbst überlassen. Somit kann der Softwareagent die Verantwortung zur Abstimmung verschiedener Mitgliedschaften nicht an eine höhere Instanz abtreten. Die juristische Verantwortlichkeit hingegen kann vom Softwareagenten weiterhin nicht übernommen werden (vgl. Abschnitte 2.1.2.5 und 2.3.2). Tabelle 2.3 fasst die für den Mitgliedschaftsbegriff wesentlichen Aspekte zusammen.

Tabelle 2.3: Effizienz-Konzept

Organisationsverständnis	Organisation als Menge interagierender organisationaler Teileinheiten
Fragestellung	Wie sind Entscheidungskompetenzen auszugestalten unter Berücksichtigung von Kommunikations- und Autonomiekosten?
Bedeutung für den Mitgliedschaftsbegriff	Organisationsmitglieder sind ebenfalls Mitglied in organisationalen Teileinheiten

Delegationswert-Konzept

Das Delegationswert-Konzept geht auf Laux (1979) zurück und optimiert implizite Verhaltensnormen einer Organisation, die durch entsprechende explizite Verhaltensnormen ergänzt werden können. Der Fokus liegt auf vertikalen Interdependenzbeziehungen, die durch eine hierarchische Organisationsstruktur entstehen (Osterloh, 2004, S. 223 f.). Die dabei betrachtete Entscheidungssituation wird durch verschiedene Merkmale charakterisiert. Diese wurden im Laufe der Entwicklung des Delegationswert-Konzepts auf Basis der ersten Fassung nach Laux (1979, S. 26 ff.) weiter präzisiert und nach Laux (2005b, S. 214 ff.) zu insgesamt zehn Merkmalen überführt. Einzelne dieser Merkmale geben weiteren Aufschluss über das Mitgliedschaftsverständnis des Delegationswert-Konzepts und werden im Folgenden näher betrachtet. Der Delegationswert wird dabei definiert als *„die Differenz aus (a) dem Gewinnerwartungswert, der bei Delegation vor Abzug der Delegationskosten erzielt wird, und (b) dem Gewinnerwartungswert bei Entscheidung durch die Instanz ohne Informationsbeschaffung“* (Laux, 1979, S. 41).

Die *Instanz* entscheidet im Delegationswert-Konzept über eine Delegation einer Entscheidung an eine untergeordnete Ebene. Eine Delegation der Entscheidung wird erwogen, falls die Instanz annimmt, dass der (potentielle) Entscheidungsträger mehr Informationen über Auswirkungen von Handlungsalternativen besitzt, wie beispielsweise Wahrscheinlichkeitsverteilungen (Merkmal 6). Die Instanz optimiert nach dem Delegationswert-Konzept ausschließlich ihren eigenen Gewinnerwartungswert (Merkmal 2) und wählt im Falle einer eigenen Entscheidung – folglich die Ablehnung einer Delegation – diejenige Handlungsalternative, die den höchsten Gewinnerwartungswert verspricht (Merkmal 5). Im Falle einer Delegation hingegen optimiert der Empfänger der Entscheidungsbefugnis die Entscheidung bezüglich seiner eigenen Zielfunktion (Merkmal 8). Für die Mitgliedschaft in Organisationen bedeutet dies, dass die jeweiligen Mitglieder – die Entscheidungsträger innerhalb einer Organisation – selbstständig auf Basis einer Zielfunktion Entscheidungen treffen können. Damit sich die Entscheidungsträger jedoch an den Zielen der Instanz orientieren, kann die Instanz versuchen, die Entscheidung durch entsprechende Vorgaben zu beeinflussen (Merkmal 9). Beispielsweise kann durch eine Gewinnbeteiligung oder Erfolgsprämie der Entscheidungsträger in Richtung der vorgegebenen Zielfunktion beeinflusst werden. Für den Mitgliedschaftsbegriff kann somit abgeleitet werden, dass Mitglieder im Allgemeinen innerhalb ihrer Entscheidungsbefugnis selbstständig entscheiden und dabei ihren eigenen Zielfunktionen folgen, jedoch durch hierarchisch übergeordnete Instanzen zielorientiert

beeinflusst werden können. Durch eine Beeinflussung mit entsprechenden Anreizsystemen kann die Zielfunktion des Mitglieds an diejenige der Organisation angenähert bzw. angepasst werden. Tabelle 2.4 fasst die für den Mitgliedschaftsbegriff wesentlichen Aspekte des Delegationswert-Konzepts zusammen.

Tabelle 2.4: Delegationswert-Konzept

Organisationsverständnis	Organisation als Menge an individuellen Entscheidungsträgern
Fragestellung	Wie sind Entscheidungskompetenzen zu verteilen?
Bedeutung für den Mitgliedschaftsbegriff	Mitglieder treffen eigenständig Entscheidungen, die von der Organisation zielorientiert gesteuert werden können

2.3.2.1.4 Neue Institutionenökonomik

Institutionenökonomische Konzepte stehen in enger Verbindung zu den entscheidungslogisch-mathematischen Ansätzen und werden von Osterloh (2004, S. 223 f.) ebenfalls den entscheidungslogischen Ansätzen zugeordnet. Sie bilden allerdings eine Weiterentwicklung der neoklassischen Organisationstheorie, indem sie versuchen, den Anwendungsbereich zu erweitern (Richter und Furubotn, 2010, S. 2 f.). Der Begriff der Institution wird dabei in der Literatur nicht einheitlich verwendet, jedoch häufig als eine Art Regelsystem aufgefasst: „... ein System miteinander verknüpfter, formgebunderer (formaler) und formungebunderer (informeller) Regeln (Normen) einschließlich der Vorkehrungen zu deren Durchsetzung“ (Richter und Furubotn, 2010, S. 7). Neben dieser Art Regelsystem lässt Erlei et al. (2007, S. 22) auch die alternative Interpretation als Vertrag oder Vertragssystem zu – abermals inklusive der dazugehörigen Durchsetzungsmechanismen. Regeln und deren Einhaltung, gegebenenfalls durch entsprechende Durchsetzungsmechanismen, sind somit ein wesentliches Merkmal, welches die Neue Institutionenökonomik von der neoklassischen Auffassung unterscheidet. Regeln, die durch eine vertragliche Bindung auf Grundlage von gesetzlichen Rahmenbedingungen entstehen, können beispielsweise bei Nichteinhaltung durch staatliches Eingreifen sanktioniert werden. Während Institutionen ausschließlich ein Regelsystem darstellen, versteht die Neue Institutionenökonomik unter *Organisationen* den Zusammenschluss von Institutionen mit einer personellen und sachlichen Dimension (Erlei et al., 2007, S. 22). Mitgliedschaft ist demnach nur in einer Organisation und nicht in einer Institution im Sinne der Neuen Institutionenökonomik möglich, da dem Regelsystem Institution die personelle Dimension fehlt. Gleichzeitig beeinflussen Institutionen das Zielsystem der Mitglieder einer Organisation, da sie durch Regeln Handlungsalternativen einschränken. Die Organisation wird in der Neuen Institutionenökonomik demnach nicht als Instrument, sondern als soziales System betrachtet (vgl. Abschnitt 2.3.1).

Im Folgenden werden weitere Annahmen der Neuen Institutionenökonomik untersucht, die zwar auch in anderen organisationstheoretischen Bereichen Anwendung finden, allerdings das Mitgliedschaftsverständnis der Neuen Institutionenökonomik in besonderem Maße beeinflussen.

Methodologischer Individualismus Gegenüber vorangegangenen Ansätzen der Organisationstheorie, sieht die Neue Institutionenökonomik die Organisation nicht mehr als ein losgelöst von ihren Mitgliedern handelndes Subjekt, sondern betrachtet das Verhalten der einzelnen Individuen als Beitrag zur Organisation (Richter und Furubotn, 2010, S. 3). Dabei haben die individuellen Präferenzen und Ziele der beteiligten Individuen Einfluss auf die Entstehung, Entwicklung und Handlungen von Organisationen. Dieses Bild der Mitglieder einer Organisation deckt sich mit dem Organisationsverständnis als soziales System, dessen Erklärungsschwerpunkt auf der Interaktion von Individuen und den daraus entstehenden Kooperationsstrukturen liegt. Softwareagenten unterscheiden sich von konventioneller Software unter anderem durch Lernfähigkeit, welche ihnen zu einem von ihrem vorgegebenen Wissen unabhängigen Handeln verhilft. Für die Mitgliedschaft von Softwareagenten in Multiagenten-Organisationen ergibt sich hieraus, dass diese ebenfalls nicht zwingend den Zielen der Multiagenten-Organisation, sondern primär ihren eigenen Interessen folgen. Aus Sicht der Multiagenten-Organisation müssen die Mitglieder demnach durch Anreizsysteme motiviert werden, den übergeordneten Zielen zuzuarbeiten. Diese Unabhängigkeit der einzelnen Individuen ist Voraussetzung, damit Mitglieder überhaupt eigenständige Beitrittsentscheidungen treffen können. Erst durch das Vorhandensein individueller Ziele können sich Kooperationsstrukturen wie Organisationen bilden, durch deren Mitgliedschaft sich die Subjekte einen Mehrwert versprechen.

Begrenzte Rationalität Die Nähe der Neuen Institutionenökonomik zu neoklassischen Ansätzen wird nochmals durch die Annahme der begrenzten Rationalität der Individuen deutlich. March und Simon (1958, S. 157 ff.) als Vertreter der Anreizbeitrags-Theorie werden im Allgemeinen den neoklassischen Ansätzen zugeordnet und grenzen sich von vorangehenden Ansätzen insbesondere durch die Annahme begrenzter Rationalität ab. Die Rationalität des Individuums ist dabei in der Form begrenzt, als ihm im Allgemeinen nicht sämtliche für eine Entscheidung relevanten Handlungsalternativen sowie deren Auswirkungen bekannt sind. Aufgrund der beschränkten Ressourcenkapazität, die zur Alternativen- und Auswirkungsevaluation eingesetzt werden kann, begnügen sich Individuen meist auch mit nicht optimalen Lösungsalternativen, die dem Entscheider zufriedenstellend erscheinen. Diese Einschränkungen gelten sowohl für Mitglieder in Organisationen als auch Softwareagenten in Multiagenten-Organisationen. Beiden stehen nur begrenzte Ressourcen zur Problemlösung und damit Entscheidungsfindung zur Verfügung, weshalb beide nur begrenzt rational handeln können. Sämtliche Entscheidungen, die innerhalb der (Multiagenten-)Organisation getroffen werden, sind demnach durch die begrenzte Rationalität der entscheidenden Mitglieder betroffen und müssen bei der Strukturierung der Auf- und Ablauforganisation Berücksichtigung finden. Die begrenzte Rationalität der Individuen wirkt sich hingegen nicht nur bei organisationsinternen Entscheidungen, sondern auch auf die vorangegangenen Beitrittsentscheidungen aus. Abschnitt 2.4.1 geht näher auf die Auswirkungen der begrenzten Rationalität auf Beitrittsentscheidungen ein.

Aus diesen beiden zentralen Annahmen der Neuen Institutionenökonomik lassen sich Rückschlüsse auf den verwendeten Mitgliedschaftsbegriff ziehen. Diese sind in der Tabelle 2.5 zusammengefasst.

Tabelle 2.5: Übergreifende Annahmen der Neuen Institutionenökonomik

Organisationsverständnis	Organisation als Verknüpfung von Institutionen (Regelsystemen) und Mitgliedern
Fragestellung	Welche Bedeutung haben Institutionen (Regelsysteme) für Wirtschaftsprozesse?
Bedeutung für den Mitgliedschaftsbegriff	Mitglieder verfolgen eigene Ziele und sind begrenzt rational

Transaktionskostentheorie

Bereits Coase (1937) stellt fest, dass Transaktionen im Sinne von marktbasierten Tauschgeschäften nicht ohne Nebenkosten durchzuführen sind. Ein Teil dieser Transaktionskosten wird durch die begrenzte Rationalität (siehe oben) und den sich hieraus ergebenden Aufwendungen (z.B. Suchkosten) verursacht (March und Simon, 1958, S. 190 ff.). Williamson (1985) unterscheidet dabei in ex-ante und ex-post entstehende Transaktionskosten: „*[ex ante transaction costs] are the costs of drafting, negotiating, and safeguarding an agreement*“ (Williamson, 1985, S. 20). Transaktionskosten, die hingegen nach Vertragsabschluss – beispielsweise durch unvollständige Verträge – entstehen, werden als *ex-post* bezeichnet. Die Transaktionskostentheorie betrachtet eine Organisation als Menge von Austauschbeziehungen: „*A firm ... [has] a role to play in the economic system if it were possible for transactions to be organized within the firm at less cost than would be incurred if the same transactions were carried out through the market*“ (Coase, 1991, S. 48). Neben den nach außen gerichteten Transaktionen mit anderen Organisationen oder Personen, finden Transaktionen demnach auch innerhalb einer Organisation statt. Dabei sind sowohl Transaktionen zwischen der Organisation und ihren Mitgliedern als auch zwischen verschiedenen Mitgliedern möglich. Transaktionen zwischen der Organisation und ihren Mitgliedern umfassen beispielsweise Arbeitsverträge und die damit verbundenen Leistungen sowie die Bereitstellung von Ressourcen. Zwischen den Mitgliedern wiederum sichern Transaktionen die Zusammenarbeit innerhalb der Organisation und werden teilweise durch explizite organisatorische Regeln vorgeschrieben, aber auch durch bi- oder multilaterale informelle Regelungen zwischen den Mitgliedern ergänzt. Die Mitgliedschaft ist somit stets an solche Transaktionen gebunden, die mit der zugehörigen Organisation bzw. mit anderen Mitgliedern ebenjener Organisation durchgeführt werden. Die Transaktionen wiederum verursachen Transaktionskosten, sowohl für die Organisation als auch für die einzelnen Mitglieder. Diese Transaktionskosten entstehen nicht nur während der Mitgliedschaft in einer Organisation, sondern insbesondere auch beim Beitritt in diese. Potentielle Mitglieder müssen unter anderem Suchkosten aufwenden, um die Organisation selbst sowie Informationen über die Organisation zu beschaffen und in die Beitrittsentscheidung einzubringen. Dies gilt gleichermaßen für Softwareagenten im Rahmen von Beitrittsverhandlungen in

Multiagenten-Organisationen. Tabelle 2.6 fasst die für den Mitgliedschaftsbegriff relevanten Aspekte der Transaktionskostentheorie zusammen.

Tabelle 2.6: Transaktionskostentheorie

Organisationsverständnis	Organisation als Menge von Austauschbeziehungen
Fragestellung	Wie sind Austauschbeziehungen zwischen verschiedenen Parteien unter Berücksichtigung der verbundenen Kosten auszugestalten?
Bedeutung für den Mitgliedschaftsbegriff	Mitglieder stehen in Austauschbeziehungen mit anderen Mitgliedern sowie der Organisation selbst, was Transaktionskosten verursacht

Verfügungsrechtetheorie

Die Verfügungsrechtetheorie versteht unter Verfügungsrechten eine aus dem juristischen Eigentumsrecht abgeleitete Form des ausschließlichen Zugriffsrechts auf ein materielles oder immaterielles Gut. Richter und Furubotn (2010, S. 95) verweisen hierzu insbesondere auf die juristischen Definitionen des § 903 BGB sowie Art. 14 GG. Fraglich ist dabei, inwiefern Verfügungsrechte tatsächlich mit dem Eigentumsrecht als vielmehr mit dem Besitz (§ 854 BGB) über das betreffende Gut gleichzusetzen sind. Neben diesen rechtlich gesicherten absoluten und relativen Verfügungsrechten¹⁴ existieren jedoch auch andere Verfügungsrechte, denen keine rechtliche Basis zugrunde liegt, wie beispielsweise solche, die durch soziale Bindungen entstehen. Ungeachtet der Art an Verfügungsrechten schränken diese durch ihre Ausschließlichkeit automatisch die Verfügungsrechte anderer über dasselbe Gut ein und können so zu unerwünschten externen Effekten führen (Demsetz, 1976, S. 348). In Bezug auf die Mitgliedschaft in Organisationen lassen sich zwei mögliche Interpretationsformen identifizieren: (i) Die Eigentümer (Aktionäre, Gesellschafter, etc.) der Organisation besitzen Verfügungsrechte über sämtliche im Eigentum der Organisation stehenden materiellen und immateriellen Dinge und treten diese an die Mitglieder der Organisation auf Basis eines schuldrechtlichen Vertrags ab. (ii) Die Organisation besitzt Verfügungsrechte, die die Freiheit der Mitglieder einschränken, beispielsweise könnten der Organisation Verfügungsrechte über die Arbeitszeit ihrer Mitglieder zugesprochen werden. Erstere Variante wäre im Rahmen der Verfügungsrechtetheorie nur möglich, wenn Verfügungsrechte durch Besitz und nicht durch Eigentum entstehen. Nur dann können die Verfügungsrechte an einem Gut der Organisation temporär zur Erfüllung der organisationalen Aufgaben an ein Mitglied übertragen werden. Zusammen mit eigenen Verfügungsrechten, die ein Mitglied in die Organisation einbringt, kann das Mitglied seine Aufgabe innerhalb der Organisation erfüllen. Diese Übertragung von Verfügungsrechten kann somit als Kriterium für Mitgliedschaft in einer Organisation angesehen werden. Die zweite Variante hingegen sieht eine Abtretung der Verfügungsrechte der Mitglieder vor (z.B. der ihnen zur Verfügung stehenden Zeit) und betrachtet somit

¹⁴ Unter absoluten Verfügungsrechten werden solche Verfügungsrechte verstanden, „die von jedermann zu beachten sind“ (Richter und Furubotn, 2010, S. 95 f.), während relative Verfügungsrechte durch Verträge zwischen mehreren Akteuren entstehen und somit auch nur zwischen diesen gelten.

eine andere Dimension als erstere Variante, setzt jedoch voraus, dass die Allokation der Zeit von der Organisation bzw. Organisationsleitung übernommen wird und die einzelnen Mitglieder ausschließlich umsetzende Tätigkeiten ausführen. Diese Interpretation wird der betrieblichen Realität zumindest in dieser Allgemeinheit nicht gerecht, weshalb die Verfügungsrechte im Rahmen einer organisationalen Mitgliedschaft hier als Zugriffsrechte auf Ressourcen der Organisation verstanden werden. Tabelle 2.7 zeigt die Bedeutung der Verfügungsrechtetheorie für den Mitgliedschaftsbegriff zusammenfassend auf.

Tabelle 2.7: Verfügungsrechtetheorie

Organisationsverständnis	Organisation als Struktur von Verfügungsrechten
Fragestellung	Wie entstehen Verfügungsrechte-Strukturen und welche Konsequenzen haben diese für die verbundenen Subjekte?
Bedeutung für den Mitgliedschaftsbegriff	Mitglieder besitzen Verfügungsrechte, die ihre Handlungen beeinflussen

Prinzipal-Agenten-Theorie und weitere Vertragstheorien

Im Rahmen der Neuen Institutionenökonomik wird in der Literatur oftmals die Prinzipal-Agenten-Theorie der Transaktionskosten- sowie Verfügungsrechtetheorie gegenübergestellt (Schreyögg, 2004c, S. 1078 f.). Die Prinzipal-Agenten-Theorie deckt jedoch nur einen – wenn auch den in der Literatur am weitesten vertretenen – Teil der vertragstheoretischen Konzepte der Neuen Institutionenökonomik ab. Während die Prinzipal-Agenten-Theorie bei der Vertragsgestaltung von rechtlich durchsetzbaren Vereinbarungen ausgeht, betrachtet die *Theorie sich selbst durchsetzender oder impliziter Verträge* auch ebensolche Beziehungen, die ohne rechtliche Regelungen auskommen, und die *Theorie relationaler Verträge* die Bedeutung impliziter und somit nicht rechtsverbindlicher Regelungen zwischen den Vertragsparteien (Richter und Furubotn, 2010, S. 181 ff.).

Nach den oben angeführten allgemeinen Annahmen der Neuen Institutionenökonomik ist ein Überwachungs- und Durchsetzungssystem notwendig, um die Vertragseinhaltung der Vertragsparteien zu garantieren (Richter und Furubotn, 2010, S. 7). Insbesondere für die Prinzipal-Agenten-Theorie ist dieser Aspekt essentiell, da ohne ein entsprechendes System die Parteien im Zweifel Vertragsbestandteile nicht vertragskonform ausführen. Das Überwachungs- und Durchsetzungssystem inkludiert Möglichkeiten, um abweichendes Verhalten zu sanktionieren. Dabei ist fraglich, inwiefern diese Sanktionierungsmöglichkeiten auch auf Softwareagenten angewandt werden können. Softwareagenten sind zunächst keine rechtsfähigen Subjekte und können demnach keine Verträge auf eigene Rechnung abschließen.

Zwar würde die Anwendung von sich selbst durchsetzenden bzw. impliziten Verträgen die rechtliche Problematik beiseite schieben und somit den fokussierten Blick auf die Vertragsgestaltung selbst ermöglichen, das Kriterium der Dauerhaftigkeit von Multiagenten-Organisationen bringt jedoch stets auch die Forderung nach stabilen Vertragsverhältnissen

mit sich. Dies lässt sich im Allgemeinen nur mit expliziten Verträgen sowie einem entsprechenden Überwachungs- und Durchsetzungssystem gewährleisten. Im Rahmen der Vertragstheorien wird dieser Aspekt im wesentlichen von der Prinzipal-Agenten-Theorie abgedeckt.

Die Prinzipal-Agenten-Theorie versteht eine Organisation als Menge vertraglich gebundener Personen: „*An organization is the nexus of contracts, written and unwritten, among owners of factors of production and customers*“ (Fama und Jensen, 1983, S. 302). Dabei gehen Fama und Jensen (1983) davon aus, dass ein wesentliches Ziel von Organisationen das „Überleben“ ist und setzen somit implizit die Dauerhaftigkeit von Organisationen voraus. Verträge mit Mitgliedern der Organisation müssen demnach so gestaltet werden, dass diese durch Anreize zur Mitwirkung am Ziel des Überlebens der Organisation ermutigt werden. Im Rahmen der Prinzipal-Agenten-Theorie finden sämtliche Verträge im Kontext der Wertschöpfung einer Organisation Beachtung, nicht nur Arbeitsverträge, sondern insbesondere auch jene Verträge mit Zulieferern, Kunden, Kreditgebern, etc. (Jensen und Meckling, 1976, S. 310). Der Mitgliedschaftsbegriff findet dabei zunächst keine Verwendung, jedoch liegt in Anbetracht der Fokussierung auf das langfristige Überleben einer Organisation der Schluss nahe, dass nur solche Verträge für die Abgrenzung von Mitgliedern zu Nicht-Mitgliedern betrachtet werden, welche auf Dauerhaftigkeit ausgelegt sind. Für Organisationen sind dies insbesondere – jedoch nicht ausschließlich – jene Verträge, die zwischen der Organisation selbst als Vertreter der Eigentümer ebenjener (z.B. Aktionäre, Gesellschafter) sowie den verantwortlichen und ausführenden Mitarbeitern dieser Organisation geschlossen wurden. Tabelle 2.8 fasst die für den Mitgliedschaftsbegriff wesentlichen Aspekte der Prinzipal-Agenten-Theorie zusammen.

Tabelle 2.8: Prinzipal-Agenten-Theorie

Organisationsverständnis	Organisation als Menge vertraglich gebundener Personen
Fragestellung	Wie sind Verträge unter Berücksichtigung asymmetrisch verteilter Informationen auszugestalten?
Bedeutung für den Mitgliedschaftsbegriff	Mitglieder sind durch einen langfristig wirkenden Vertrag gebunden

Die Überführung dieser Aspekte der Mitgliedschaft auf Softwareagenten in Multiagenten-Organisationen setzt zunächst voraus, dass Softwareagenten rechtlich wirksame Verträge abschließen können. Um die auf Dauerhaftigkeit angelegte Multiagenten-Organisation zu unterstützen und deren Stabilität zu wahren, sind in Hinblick auf die Prinzipal-Agenten-Theorie die Verträge mit Softwareagenten ebenso auf Dauerhaftigkeit auszulegen.

2.3.2.1.5 Ressourcenbasierter Ansatz

Der ressourcenbasierte Ansatz entwickelte sich bereits früh als Erklärung von Wettbewerbsvorteilen von Unternehmen (Bresser, 2004, S. 1269). Dabei sieht der ressourcenbasierte Ansatz Ressourcen als den wesentlichen Wettbewerbsfaktor von Organisationen (Barney, 1991). Die Entwicklung einer ersten Grundlage für die Forschung im Bereich der Ressourcen

als Wettbewerbsvorteile im strategischen Management gelang hingegen erst Wernerfelt (1984). Er definiert die Ressourcen einer Organisation wie folgt: „*a firm's resources at a given time could be defined as those (tangible and intangible) assets which are tied semipermanently to the firm*“ (Wernerfelt, 1984, S. 172). Als Beispiel für Ressourcen führt Wernerfelt (1984) „*employment of skilled personnel*“ auf, welche nach der vorangegangenen Definition nur semipermanent an die Organisation gebunden sind. Die Mitgliedschaft eines Arbeitnehmers in einer Organisation ist somit nicht endgültig und Änderungen im Status dieser Mitgliedschaft sind nicht ausgeschlossen. Der Begriff „semipermanent“ kann folglich in Hinblick auf die Bindung der Ressourcen an die Organisation verstanden werden als eine auf Dauerhaftigkeit ausgelegte, jedoch grundsätzlich veränderliche Beziehung. Dies impliziert ebenfalls für den Mitgliedschaftsbegriff als Bindung der Organisationsmitglieder an die Organisation ebenjenen dauerhaften Charakter, deren Veränderung jedoch nicht ausgeschlossen ist. Somit lässt der ressourcenbasierte Ansatz das Hinzutreten potentieller Mitglieder sowie das Austreten bestehender Mitglieder zu.

Während Wernerfelt (1984) eine weitere Klassifizierung unterlässt, geht Barney (1991, S. 101) detaillierter auf die Unterscheidung verschiedener Ressourcentypen ein: (i) *Physikalische Ressourcen* einschließlich sämtlicher physikalischer Anlagegüter und Rohmaterialien, (ii) *Human Kapital Ressourcen* als Menge an Fähigkeiten und Verbindungen zwischen den einzelnen Mitgliedern der Organisation sowie (iii) *Organisationskapital Ressourcen*, die die formale Struktur einschließlich formeller und informeller Koordinationsmechanismen beinhalten. Bei dieser Unterscheidung bleibt offen, in welche Kategorie immaterielle Anlagegüter wie beispielsweise Patente oder Software einzuordnen sind. Auch ohne physisch greifbar zu sein, teilen sich solche Anlagegüter im Gegensatz zu anderen immateriellen Ressourcen, wie beispielsweise Wissen oder Fähigkeiten, wesentliche Eigenschaften mit den physikalischen Ressourcen (Hall, 1992, S. 136). Eine Unterteilung von immateriellen Ressourcen in die Kategorien Anlagegüter und Fähigkeiten ist somit notwendig, um diese in den oben angeführten Ressourcentypen zuordnen zu können. Dabei werden immaterielle Ressourcen in Form von Anlagegütern den physikalischen und Fähigkeiten den Human Kapital Ressourcen zugeordnet. Eine Einordnung von Softwareagenten scheint hingegen schwierig, da durch die Lernfähigkeit eine einseitige Verfügungsgewalt durch die Multiagenten-Organisation nicht gegeben ist. Obwohl es sich bei Softwareagenten zunächst um Software handelt und somit als immaterielle Anlagegüter den physikalischen Ressourcen zugeordnet werden müssten, ist durch die in Abschnitt 2.1 eingeführte Abgrenzung zu konventioneller Software ebenfalls eine Abgrenzung erforderlich. Die Lernfähigkeit und somit eingeschränkte Beeinflussbarkeit hingegen weist ähnliche Merkmale auf, wie man sie den Human Kapital Ressourcen zuordnen würde. Für diese Zuordnung sprechen ebenfalls die ähnlich zu den Human Kapital Ressourcen ablaufende Aufnahme- und Austrittsprozesse von Softwareagenten, welche erst nach erfolgreichen Verhandlungen zu einem Beitritt bereit sind. Softwareagenten bringen ähnlich den Human Ressourcen spezielle Fähigkeiten in eine Multiagenten-Organisation ein und sind somit unter Umständen für diese nicht ohne weiteres ersetzbar. Tabelle 2.9 fasst die für den Mitgliedschaftsbegriff wesentlichen Aspekte zusammen.

Tabelle 2.9: Ressourcenbasierter Ansatz

Organisationsverständnis	Organisation als Sammlung von materiellen und immateriellen Ressourcen
Fragestellung	Welche Wettbewerbsvorteile entstehen durch das Vorhandensein organisationsspezifischer Ressourcen?
Bedeutung für den Mitgliedschaftsbegriff	Mitglieder bringen Ressourcen mit in die Organisation ein oder stellen für die Organisation selbst Ressourcen dar

2.3.2.1.6 Fazit

Die vorangegangenen Ausführungen haben gezeigt, dass die verschiedenen Ansätze der Organisationstheorie mit unterschiedlichen Erklärungsschwerpunkten eine Vielzahl von Aspekten mit sich bringen, die Mitglieder in Organisationen charakterisieren. Dieser Abschnitt fasst die wesentlichen Punkte zusammen und stellt die Bezüge zu Beitrittsentscheidungen potentieller Mitglieder her. Auf die Beitrittsentscheidung selbst wird in Abschnitt 2.4 näher eingegangen.

Abbildung 2.4 verdeutlicht die Zusammenhänge zwischen Organisationsmitgliedern und der Organisation als Entity-Relationship-Diagramm in IDEF1X Notation. Aus der vorangegangenen Betrachtung lassen sich hieraus folgende Beziehungen herausstellen:

Vertragliche Bindung. Im Kern der Beziehung steht die vertragliche Bindung zwischen dem Organisationsmitglied und der Organisation (siehe bidirektionale Assoziation (A) in Abbildung 2.4). Diesen Aspekt stellen gleich mehrere Organisationstheorien heraus. Insbesondere die Neue Institutionenökonomik (Abschnitt 2.3.2.1.4) mit den dazugehörigen Vertragstheorien sehen den Vertrag als Grundlage für die Organisationsmitgliedschaft. Aber auch die Anreiz-Beitrags-Theorie (Abschnitt 2.3.2.1.2), wie auch bereits der Bürokratie-Ansatz von Max Weber (Abschnitt 2.3.2.1.1) gehen in den überwiegenden Fällen von einer vertraglichen Grundlage aus.

Transaktionen. Auf Basis der vertraglichen Bindung werden zwischen dem Organisationsmitglied und der Organisation Transaktionen durchgeführt (B). Als Grundlage hierzu dient die Transaktionskostentheorie (Abschnitt 2.3.2.1.4), ebenfalls als Teil der Neuen Institutionenökonomik. Dabei wird allerdings auch deutlich, dass Transaktionen nicht nur zwischen dem Organisationsmitglied und der Organisation, sondern auch zwischen verschiedenen Organisationsmitgliedern durchgeführt werden (C). Bei letzteren Transaktionen steht wiederum keine bilaterale vertragliche Vereinbarung zwischen den Organisationsmitgliedern zur Verfügung. Diese werden hingegen auf Basis der organisatorischen Regeln ausgeführt – nach dem Bürokratie-Ansatz möglichst ohne Handlungsspielraum der Individuen.

Beiträge und Anreize. Der Organisation ist im Allgemeinen daran gelegen, dass das Organisationsmitglied Beiträge zu dieser leistet (D). Um dieses Ziel zu erreichen, muss die Organisation dem Mitglied entsprechende Anreize setzen (E). Diese beiden Aspekte lassen sich insbesondere aus der Anreiz-Beitrags-Theorie ableiten und finden

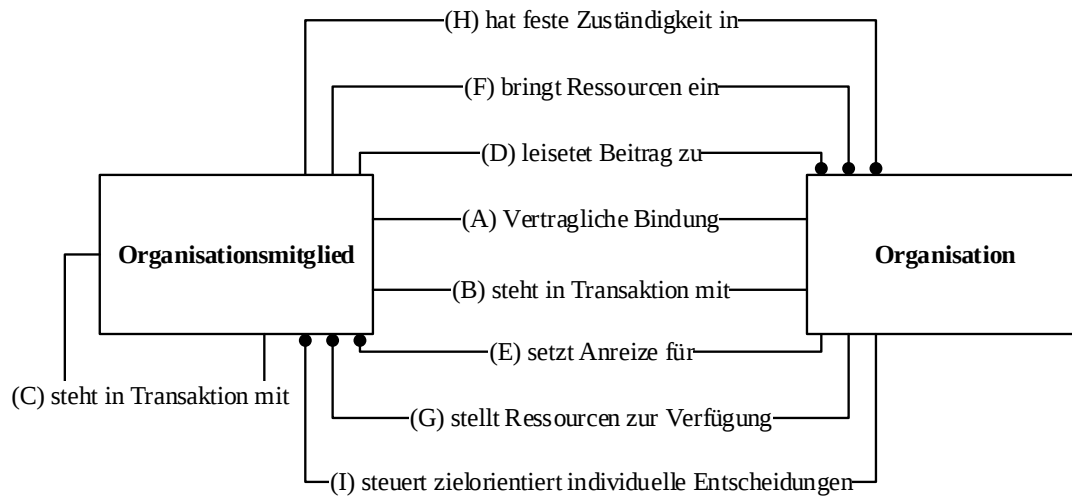


Abbildung 2.4: ER-Modell der Beziehung zwischen Organisation und deren Mitgliedern¹⁵

in der Regel ebenfalls bei der vertraglichen Gestaltung Berücksichtigung. Aber auch ohne vertragliche Bindung lassen sich Anreize zum Leisten von Beiträgen nutzen, dabei stellen allerdings Verträge durch ihre Verbindlichkeit und die zwingend vorangegangenen übereinstimmenden Willenserklärungen eine besondere Art der Fixierung von Anreiz-Beitrags-Bedingungen dar. Sowohl Anreize als auch Beiträge lassen sich auch als Transaktionen auffassen, die zwischen Organisation und Organisationsmitglied durchgeführt werden. Hierdurch wird die enge Verknüpfung der vertraglichen Bindung, der Transaktionen sowie der Anreiz-Beitrags-Konstellation deutlich.

Ressourcennutzung. Mitglieder in Organisationen bringen Ressourcen in Organisationen ein (F), beispielsweise indem sie letzterer ihre Arbeitszeit als Ressource zur Verfügung stellen. Umgekehrt ist es hingegen oftmals nötig, dass die Organisation ihren Mitgliedern Ressourcen zur Verfügung stellt (G), beispielsweise in Form von Arbeitsräumen, -material oder -gerätschaften. Zur Betrachtung dieser Perspektive eignet sich insbesondere der Ressourcen-basierte Ansatz (Abschnitt 2.3.2.1.5). Inwiefern die Ressourcennutzung in einem Vertrag zwischen Mitglied und Organisation explizit definiert wird, unterscheidet sich je nach Konstellation und Vertrag deutlich. Während beispielsweise in einem Arbeitsvertrag die Arbeitszeit als zur Verfügung zu stellende Ressource des Mitglieds in aller Regel konkret genannt wird, ist die Organisation selbst im Allgemeinen an einem reibungslosen Ablauf und somit an der Bereitstellung der entsprechenden Ressourcen für das Mitglied interessiert.

Zuständigkeiten. Mit einer Stelle in einer Organisation sind feste Zuständigkeiten verbunden, welche dem stellenbesetzenden Mitglied übertragen werden (H). Diese Verknüpfung der Mitglieder mit Zuständigkeiten wird in zahlreichen organisationstheoretischen Ansätzen explizit oder implizit angenommen. Der Bürokratie-Ansatz sticht durch personenunabhängige Stellenbesetzung und die fest vorgegebenen Abläufe hier besonders hervor, aber auch insbesondere die entscheidungslogisch-mathematischen Ansätze

¹⁵ Legende vgl. Abbildung 2.1

(Abschnitt 2.3.2.1.3) setzen für die innerorganisatorische Ablaufoptimierung feste Zuständigkeiten und somit vorhersehbares Verhalten voraus.

Entscheidungssteuerung. Mitglieder haben in Organisationen zwar feste Zuständigkeiten, folgen allerdings trotzdem ihren eigenen Zielen, was insbesondere durch die Annahmen der Ansätze der Neuen Institutionenökonomik deutlich wird. Damit die Mitglieder im Sinne der Organisation handeln, ist es nötig, dass diese die individuellen Entscheidungen zielorientiert steuert (I). Das Delegationswert-Konzept (Abschnitt 2.3.2.1.3) liefert hierzu erste Ansätze, die aber auch wieder in der Neuen Institutionenökonomik und Anreiz-Beitrags-Theorie aufgegriffen werden. Die Prinzipal-Agenten-Theorie (Abschnitt 2.3.2.1.4) sowie die Gegenüberstellung von Anreizen und Beiträgen zielen auf eine Überbrückung der gegebenenfalls divergierenden Ziele der Organisation und der Ziele ihrer Mitglieder ab, bilden somit eine Entscheidungssteuerung im Sinne der organisatorischen Ziele.

2.3.2.2 Mitgliedschaft in Multiagenten-Organisationen

Eine zentrale Frage bei der Analyse des Beitritts eines Softwareagenten zu einer Multiagenten-Organisation wirft der Mitgliedschaftsbegriff selbst auf. Zunächst muss geklärt werden, ab wann ein Softwareagent als Mitglied einer Multiagenten-Organisation angesehen werden kann. Die in Abschnitt 2.3.1.2 aufgegriffene Literatur zu Multiagenten-Organisationen nimmt implizit an, dass die Mitglieder einer Multiagenten-Organisation klar von den Nicht-Mitgliedern abgegrenzt werden können. Die Arbeiten verzichten jedoch auf eine exakte Definition der Mitgliedschaft, die als Unterscheidungsmerkmal dienen könnte. Dies hängt insbesondere mit der häufig gewählten Top-Down Perspektive zusammen, mit Hilfe derer die Eigenschaft der Mitgliedschaft für jeden Softwareagent durch eine zentral koordinierende Einheit festgelegt wird.

Demgegenüber gibt es nur wenige Arbeiten, die Kriterien für die Mitgliedschaft von Softwareagenten in Multiagenten-Organisationen festlegen. Ein Kriterium stellt hierbei die Verbindlichkeit der Interaktionsbeziehung des teilnehmenden Softwareagenten mit der Multiagenten-Organisation bzw. deren anderer Mitglieder dar, die beispielsweise durch die Anerkennung eines Regelwerkes zustande kommt: *„Das durch formale Prozesse konstituierte Regelwerk der Multiagenten-Organisation muss gegenüber den in diese eintretenden Agenten als verbindlich erklärt werden. Die Anerkennung dieses Regelwerks stellt eine notwendige Voraussetzung für den Eintritt eines Agenten in ein organisiertes Agentensystem dar und bildet damit ein wesentliches Mitgliedschaftskriterium“* (Kirn, 1996a, S. 215). Voraussetzung für die hierbei geforderte Verbindlichkeit der Anerkennung ist somit deren Umsetzbarkeit im Kontext von Softwareagenten. Die bloße Anerkennung eines Regelwerkes hingegen bildet nur eine Voraussetzung für die Mitgliedschaft und konstituiert nicht bereits die Mitgliedschaft selbst. Andernfalls wären einseitige Beitrittsentscheidungen durch ausschließliche Anerkennung eines Regelwerkes möglich. Die Mitgliedschaft muss hingegen von beiden Seiten – dem potentiellen Mitglied sowie der Multiagenten-Organisation – anerkannt werden.

Die beidseitige Anerkennung von Regeln deutet bereits auf die Verwendung von Verträgen hin. Wie in Abschnitt 2.3.2.1 aufgezeigt, findet dieses Kriterium auch im Rahmen der

Organisationstheorie Verwendung und bildet die Grundlage für die weiteren Handlungen: Transaktionen können zwischen den Mitgliedern durchgeführt werden, die Verträge setzen Anreize zur Erbringung von Beiträgen, die Mitglieder können Ressourcen der Multiagenten-Organisation nutzen und bringen selbst Ressourcen ein. Gleichzeitig regeln diese Verträge auch Zuständigkeiten innerhalb der Multiagenten-Organisation und ermöglichen – auch durch das Setzen von Anreizen – eine Steuerung der Einzelentscheidungen.

Verträge zwischen mehreren Parteien beinhalten im schuldrechtlichen Verständnis ein oder mehrere Pflichten zwischen Vertragsparteien (§241 BGB), dies gilt unabhängig davon, ob eine Person selbst handelt oder hierbei durch einen Softwareagenten repräsentiert wird. Durfee et al. (1989) gehen davon aus, dass Verträge im Gegensatz zu Organisationen stets nur für ein bestimmtes Projekt oder einen Auftrag geschlossen werden und damit durch eine gewisse Kurzfristigkeit gekennzeichnet sind: *„contracts represent temporary alliances (as in a construction project, where the relationship ends when the structure is built) while an organization is more permanent (as in a corporation, where the roles of the president and vice-president are stable for long periods of time)“* (Durfee et al., 1989, S. 72). Diese Sichtweise unterschlägt jedoch, dass die im Beispiel genannten Mitglieder von Organisationen ebenfalls durch (Arbeits-)Verträge an die Organisation gebunden sind. Verträgen allgemein eine kurzfristige Absicht zu unterstellen, ist somit nicht zutreffend. Das Ziel von Multiagenten-Organisationen gegenüber Multiagentensystemen ist langfristige Stabilität. Das *Vorhandensein von Verträgen* alleine ist kein ausreichendes Kriterium, da diese Verträge auch ausschließlich einmalige Leistung umfassen können und somit nicht auf die langfristige Stabilität von Multiagenten-Organisationen ausgerichtet sind. Daher sind zwei Nebenbedingungen für das Kriterium der Mitgliedschaft zu erfüllen: (i) Der betreffende Vertrag muss langfristig ausgelegt sein und (ii) es müssen beiderseitige Ausstiegskosten vorhanden sein, sodass Verträge ein Mindestmaß an Bindungswirkung zwischen dem Softwareagenten und der Multiagenten-Organisation erfüllen. Hieraus lässt sich folgende Definition ableiten:

Definition 7 (Mitgliedschaft in Multiagenten-Organisationen) *Softwareagent A ist Mitglied in einer Multiagenten-Organisation B, wenn zwischen A und B ein auf Langfristigkeit ausgelegter Vertrag besteht, mit dem sich A dem Regelwerk von B unterwirft und das Auflösen oder die Nichteinhaltung des Vertrags sowohl für A als auch für B mit expliziten oder impliziten Kosten verbunden ist.*

Legt man, wie oben aufgezeigt, die Open Systems Definition zugrunde, ist klar, dass Softwareagenten in der Lage sind, mit mehreren Multiagenten-Organisationen langfristig bindende Verträge einzugehen und somit auch in diesen Mitglied zu sein. Wie auch beim sozialen Gebilde Organisation (vgl. Abschnitt 2.3.2) kann eine Mehrfachmitgliedschaft von Softwareagenten zu Zielkonflikten oder zu Konflikten bei der Beachtung unterschiedlicher Regelwerke führen. Diese Problematik wird im Kontext von Softwareagenten insbesondere bei der Koalitionsformation betrachtet (Horling und Lesser, 2004, S. 288). Abschnitt 2.4.4 geht näher auf diese Problemstellung ein.

2.4 Beitrittsentscheidung

Die Beitrittsentscheidung von Softwareagenten zu Multiagenten-Organisationen ist die wesentliche Problemstellung dieser Arbeit. Zur Analyse des Stands der Forschung werden zunächst Beitrittsentscheidungen aus organisatorischer Sicht betrachtet (Abschnitt 2.4.1). Die Verknüpfung von Beitrittsentscheidungen zu Multiagenten-Organisationen durch einen Vertrag (Abschnitt 2.4.2) mit Nebenbedingungen zur Ausdifferenzierung von Angeboten (Abschnitt 2.4.3) bildet die Grundlage für die Verhandlung zwischen Softwareagenten und Multiagenten-Organisationen. Die sich aus möglichen Mehrfachmitgliedschaften ergebende Problematik der Ressourcenallokation des Softwareagenten wird in Abschnitt 2.4.4 betrachtet. Abschnitt 2.4.5 fasst abschließend die bestehende Forschungslücke zusammen.

2.4.1 Beitrittsentscheidung aus organisationstheoretischer Sicht

Wie in Abschnitt 2.3.2.1 aufgezeigt, definieren nur wenige Organisationstheorien explizit den Begriff der Mitgliedschaft, nehmen allerdings aus unterschiedlicher Perspektive Einfluss auf ihn. Die Beitrittsentscheidung wiederum wird nur von der Anreiz-Beitrags-Theorie (vgl. Abschnitt 2.3.2.1.2) detailliert untersucht. Die voran betrachteten Perspektiven zur Mitgliedschaft haben jedoch auch ohne eine explizite Erwähnung der Beitrittsentscheidung Einfluss auf diese. Unter *Beitritt* wird im Zusammenhang mit Mitgliedschaft das „Mitglied werden“ verstanden, d.h. das Eintreten der Mitgliedschaftskriterien und somit die Verschiebung der Grenze einer Organisation zum Einschluss des beitretenden Akteurs.

Zur *Entscheidung* hingegen existieren in der betriebswirtschaftlichen Literatur unzählige Perspektiven und aus Sicht der Organisationstheorie liegt es nahe, die entscheidungsorientierte Organisationstheorie zur Lösung organisatorischer Entscheidungsprobleme zu betrachten. Zur Formalisierung des Entscheidungsproblems unterscheiden Frese et al. (2012, S. 93 ff.) zwischen *Entscheidungen* sowie der *Realisation* ebenjener Entscheidungen. Letztere ist ausschließlich der „Vollzug einer Handlung“ (Frese et al., 2012, S. 94) und hat keinerlei Spielraum, da die Handlung bereits durch die vorangegangene Entscheidung determiniert ist. Diese Unterscheidung kann ebenfalls für den Beitritt zu einer Multiagenten-Organisation übertragen werden: Auch in diesem Fall ist zwischen der Entscheidung für einen Beitritt sowie der tatsächlichen Durchführung des Beitritts zu unterscheiden. Die Entscheidung über einen Beitritt kann beispielsweise durch einen Softwareagenten unabhängig von der Entscheidung der Multiagenten-Organisation getroffen werden, die Realisation dieser Entscheidung ist hingegen abhängig von der entsprechenden Entscheidung der Multiagenten-Organisation, die einem Beitritt ebenfalls zustimmen muss (vgl. auch Abschnitt 2.3.2.1.3).

Dies verdeutlicht bereits die Abhängigkeiten, die zwischen verschiedenen Entscheidungen sowohl des gleichen Entscheidungsträgers als auch mehrerer Entscheidungsträger entstehen können. Frese et al. (2012, S. 112 ff.) unterscheiden dabei zwei Formen von Entscheidungsinterdependenzen:

Sequenzielle Verknüpfung von Realisationsprozessen. Verschiedene Realisationsprozesse können miteinander sequenziell verknüpft werden, so dass Abhängigkeiten bezüglich der einzelnen Realisationen entstehen. Abhängigkeiten zwischen

Realisationsprozessen ziehen automatisch Abhängigkeiten der dazugehörigen Entscheidungen mit sich, da diese „*Entscheidungen einer Einheit die interne Umwelt einer anderen Einheit zielrelevant verändern*“ (Frese et al., 2012, S. 114). Diese voneinander abhängigen Entscheidungen können somit in Bezug auf die Realisation der Entscheidungen nicht losgelöst voneinander betrachtet werden.

Sequenzielle Verknüpfungen von Realisationsprozessen können bei innerorganisatorischen Abläufen auftreten, wenn beispielsweise die Vertriebsabteilung von der Produktion abhängt. Entsprechende Abhängigkeiten können allerdings auch bei Abläufen einer Multiagenten-Organisation mit ihrer Umwelt entstehen. Im Falle eines potentiellen Beitritts eines Softwareagenten muss sowohl der Softwareagent als auch die Multiagenten-Organisation zu der Entscheidung kommen, dass ein Beitritt vorteilhaft ist. Einseitige Entscheidungen würden an der sequenziellen Verknüpfung der nachgelagerten Realisationsprozesse scheitern, deren Aufnahme nur durch einvernehmlichen Beitritt gelingen kann.

Überschneidung von Entscheidungsfeldern. Entscheidungsfelder entsprechen dem Ausgangszustand, auf dessen Basis Entscheidungen getroffen werden können und lassen sich in Ressourcen und Umwelt unterteilen (Frese et al., 2012, S. 94 f.). Dabei bilden Ressourcen den Bereich eines Entscheidungsfeldes, über den der Entscheidungsträger verfügen kann, während die Umwelt den entsprechenden Bereich darstellt, der dem Entscheidungsträger nicht zur unmittelbaren Disposition zur Verfügung steht, durch seine Entscheidung jedoch beeinflusst werden kann. Haben zwei Entscheidungsträger überlappende Entscheidungsfelder, führen Entscheidungen des einen Entscheidungsträgers zur Veränderung der Umwelt des anderen und es entstehen Interdependenzen zwischen den jeweiligen Entscheidungen. Kirsch (1971, S. 70 f.) hingegen verwendet für diese Form der Entscheidungsinterdependenzen den Begriff der *interindividuellen Konflikte*. Dabei wird jedoch bereits implizit von einem Konflikt – im Sinne von „*Konkurrenz oder Inkompatibilität der Zielerreichung*“ (Kirsch, 1971, S. 71) – der abhängigen Entscheidungsträger ausgegangen. Entscheidungsinterdependenzen müssen jedoch nicht zwangsläufig in konfliktären Situationen bestehen. Eine Überschneidung von Entscheidungsfeldern kann hingegen auch eintreten, falls die beteiligten Entscheidungsträger unabhängig voneinander die gleiche Entscheidung auswählen würden.

Die Überschneidung von Entscheidungsfeldern tritt ebenfalls bei der Beitrittsentscheidung von Softwareagenten zu Multiagenten-Organisationen auf. Ist ein Softwareagent bereits Mitglied in einer Multiagenten-Organisation sind die ihm zur Verfügung stehenden Ressourcen in einem bestimmten Maße bereits an diese Multiagenten-Organisation gebunden – diese Bindung kann sowohl in quantitativer (es ist ein bestimmter Anteil der Ressourcen gebunden) oder temporaler (die Ressourcen sind zu bestimmten Zeiten gebunden) Dimension bestehen. Eine Beitrittsentscheidung des Softwareagenten in Bezug zu einer anderen Multiagenten-Organisation, deren Mitgliedschaft mindestens einen Teil der gleichen Ressourcen beanspruchen würde, ist somit stets abhängig von den bereits bestehenden Mitgliedschaften in anderen Multiagenten-Organisationen.

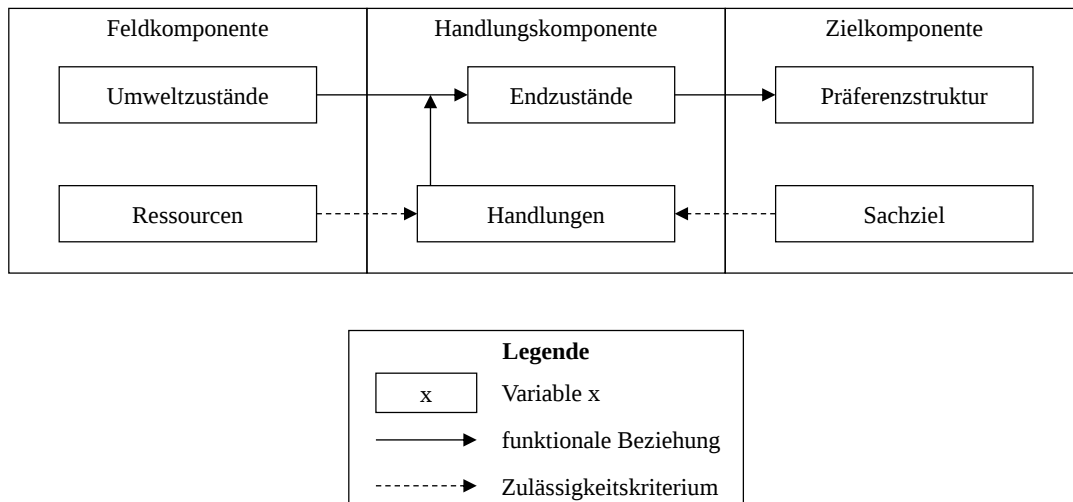


Abbildung 2.5: Entscheidungsmodell in Organisationen nach Frese et al. (2012, S. 97)

Insbesondere die zur Disposition des Entscheidungsträgers stehenden Ressourcen der Entscheidungsfelder – Entscheidungsfeld „Beitritt Multiagenten-Organisation A“ sowie Entscheidungsfeld „Beitritt Multiagenten-Organisation B“ – überschneiden sich, weshalb losgelöste Einzelentscheidungen im Allgemeinen nicht zu einem optimalen Ergebnis führen würden. Es bestehen somit Entscheidungsinterdependenzen zwischen Beitrittsentscheidungen, die Auswirkungen auf die gleichen Ressourcen eines Softwareagenten mit sich bringen. Sowohl vergangene Beitrittsentscheidungen bzw. bestehende Mitgliedschaften in Multiagenten-Organisationen beeinflussen eine akut anstehende Beitrittsentscheidung, wie auch diese den Spielraum für zukünftige Beitrittsentscheidungen beeinflusst.

Die präskriptive Entscheidungstheorie unterteilt im Allgemeinen Entscheidungsprobleme in vier Komponenten: (i) Handlungsalternativen, (ii) Umwelteinflüsse, (iii) Konsequenzen sowie (iv) Ziele und Präferenzen (Eisenführ et al., 2010, S. 20). Für den Kontext von Organisationen erweitern Frese et al. (2012) dieses Modell um die Variablen *Ressourcen* und *Sachziel*. Abbildung 2.5 stellt das Entscheidungsmodell in Organisationen nach Frese et al. (2012, S. 97) dar. Die Erweiterung um die genannten beiden Variablen ist jedoch nicht zwingend notwendig, um auch organisatorische Entscheidungsprozesse mit Hilfe des präskriptiven Entscheidungsmodells nach Eisenführ et al. (2010, S. 20 ff.) abzubilden. Die Variable *Ressourcen* schränkt dabei den möglichen Handlungsspielraum des Entscheidungsträgers ein, ebenso die durch die organisatorische Struktur vorgegebene Variable des Sachziels. Im präskriptiven Entscheidungsmodell nach Eisenführ et al. (2010) lassen sich beide Variablen implizit durch Nebenbedingungen der Handlungen – folglich ebenso als Einschränkung des Handlungsspielraums – darstellen, eine explizite Repräsentation der Ressourcenknappheit sowie der organisatorischen Sachziele erfolgt hingegen nicht. Der optionale Charakter dieser beiden Variablen wird bereits dadurch deutlich, dass diese beiden Variablen in Abbildung 2.5 ausschließlich durch die Verknüpfung „Zulässigkeitskriterium“ verbunden sind. Dabei ist anzumerken, dass diese beiden Variablen keine abschließende Menge an Zulässigkeitskriterien in Organisationen bilden. Der Handlungsspielraum wird ebenso durch Regelungen –

organisatorische und gesetzliche – als auch durch die Fähigkeiten des Entscheidungsträgers eingeschränkt. Aus dem Entscheidungsmodell nach Frese et al. (2012) lässt sich vielmehr ableiten, dass im organisatorischen Kontext Handlungen sowohl durch Variablen des Entscheidungsfeldes – d.h. des „Zustand[s] der Realität im Handlungszeitpunkt“ (Frese et al., 2012, S. 94) – als auch durch solche der Zielkomponente – insbesondere durch Sachziele im Sinne von angestrebten Endzuständen – beeinflusst werden.

Die Analyse des Gleichgewichts zwischen den durch die Organisationen geschaffenen Anreize und den durch ihre Mitglieder geleisteten Beiträge bildet das zentrale Forschungsfeld der verhaltenswissenschaftlichen Entscheidungstheorie. Dabei differenzieren March und Simon (1958) klar zwischen der Entscheidung eines einzelnen Organisationsmitglieds, ob und wie den vorgegebenen Organisationsregeln gefolgt wird, und der Entscheidung des potentiellen Organisationsmitglieds einer Organisation beizutreten bzw. der Entscheidung eines Organisationsmitglieds eine Organisation zu verlassen. Die Analyse und Beeinflussung von Beitrittsentscheidungen bildet somit einen Teilbereich der Anreiz-Beitrags-Theorie. Sowohl Anreize als auch Beiträge werden als messbare Leistungen verstanden, die unabhängig von den Nutzenfunktionen der Individuen sind. Sämtliche Anreize und Beiträge werden zu einem Nutzen mittels einer individuellen Nutzenfunktion aggregiert, für die March und Simon (1958, S. 106) drei zentrale Annahmen treffen:

- (i.) Die Nutzenfunktion ändert sich nur langsam.
- (ii.) Die Nutzenfunktion verläuft monoton in Bezug auf Anreize bzw. Beiträge.
- (iii.) Die Nutzenfunktionen verschiedener Individuen verlaufen sehr ähnlich.

Zwar konzentrieren sich March und Simon (1958, S. 108) auf die Beziehung zwischen Arbeitgebern und Arbeitnehmern, ziehen die Grenzen einer Organisation jedoch trotzdem sehr weit, sodass dieser auch die Investoren, Lieferanten und Kunden eines Unternehmens zuzurechnen sind. Die oben angeführte Annahme (iii.) ist unter diesem Gesichtspunkt kritisch zu hinterfragen: Mit einem weiter gefassten Mitgliedschaftsbegriff wächst die Wahrscheinlichkeit, bei verschiedenen Mitgliedern unterschiedlich ausgeprägte Nutzenfunktionen vorzufinden.

Eine Anwendung der Anreiz-Beitrags-Theorie auf Softwareagenten legt der ähnliche konzeptionelle Aufbau eines Menschen in der Organisationstheorie sowie eines Softwareagenten aus technischer Sicht nahe. Beide agieren mit ihrer *Umwelt* und haben einen *Internal State*, der insbesondere von vergangenen Handlungen und Ereignissen abhängt (March und Simon, 1958, S. 28; Georgeff und Ingrand, 1989). Weitere Ähnlichkeiten sind bei der Verhaltensstruktur von Softwareagenten sowie menschlichen Entscheidern, wie beispielsweise Schachspielern oder Managern zu erkennen: Wenn die Möglichkeit vorhanden ist, eine Entscheidungsalternative ausgiebig zu evaluieren, können viele Einflussfaktoren mit eingerechnet werden (deliberatives Verhalten). Falls diese Möglichkeit nicht besteht und eine kurzfristige Entscheidung getroffen werden muss, stehen feste Regeln zur Verfügung (beim Menschen die Intuition), die eine einfache Entscheidungsfindung ermöglichen (March und Simon, 1958, S. 12).¹⁶

¹⁶ vgl. Abschnitt 2.1.1.3

Während für March und Simon (1958) eine Organisation innerhalb eines gemeinsamen Kulturkreises der Individuen entsteht, ist dies für Multiagenten-Organisationen, deren Mitglieder potentiell in verschiedenen Kulturkreisen entwickelt wurden und somit auf unterschiedliche Weise Prägungen erfahren haben können, nicht zwingend zutreffend. Dieses Argument spricht ebenfalls dafür, dass sich bei Anwendung der Anreiz-Beitrags-Theorie nach March und Simon (1958) auf Zusammenschlüsse von Softwareagenten die Annahme (iii.) nur bedingt halten lässt. Die Reduzierung dieser Annahme wirkt in beide Richtungen: Sowohl der potentiell beitretende Softwareagent sieht sich bei verschiedenen Multiagenten-Organisationen unterschiedlichen Nutzenfunktionen gegenüber, als auch die Multiagenten-Organisation, die bei den Beitrittsverhandlungen zahlreiche Verläufe von Nutzenfunktionen potentiell beitretender Softwareagenten betrachten muss.

2.4.2 Beitritt zu einer Multiagenten-Organisation durch Vertrag

Nach Definition 7 ist für die Mitgliedschaft in Multiagenten-Organisationen eine vertragliche Bindung notwendig. Für Softwareagenten bietet das Kontrakt-Netz-Protokoll eine Möglichkeit, Verträge zwischen Softwareagenten zu schließen. Dabei schreibt ein *Manager* ein Teilproblem aus, auf das die *Contractors* bieten können (Smith, 1980). Ein Softwareagent kann jedoch auch gleichzeitig Manager und Contractor sein. Das ursprüngliche Protokoll wurde zur Lösung verteilter Probleme entwickelt, die Top-Down in Teilprobleme heruntergebrochen werden können (vgl. Abschnitt 2.2.2.2). Divergierende Interessen einzelner Softwareagenten können hierdurch nur eingeschränkt dargestellt werden, da unter anderem keine bindenden Verträge geschlossen werden, sondern diese durch Nichterfüllung ohne Strafzahlung beendet werden können, was eine Anwendung in dieser Form für Mitgliedschaftsverträge in Multiagenten-Organisationen ausschließt.

Das Kontrakt-Netz-Protokoll wurde bis heute für verschiedene Bereiche erweitert (z.B. Sandholm (1993), Karänke und Kirn (2010)) und 2002 von der *Foundation for Intelligent Physical Agents* standardisiert (FIPA, 2002). Hier werden statt *Manager* und *Contractor*, die Begriffe *Initiator* und *Participant* verwendet und das ursprüngliche Protokoll um Absagen und Bestätigungen erweitert. Hierdurch wird eine schnellere Weiterverhandlung und ebenso die Einführung vertraglicher Nebenbedingungen ermöglicht. Der nachfolgende Abschnitt 2.4.3 geht tiefer auf die Bedeutung dieser Nebenbedingungen ein.

Einen Ansatz zur Modellierung von Multiagenten-Organisationen bietet Dignum (2004): In der Modellierungssprache OperA werden die Beziehungen zwischen Softwareagenten und Multiagenten-Organisationen durch Verträge fixiert. Van Riemsdijk et al. (2009) greifen dieses Konzept auf und teilen den Lebenszyklus einer Mitgliedschaft in drei Phasen: (i) Organisationsbeitritt, (ii) Rollenausübung in der Organisation sowie (iii) Verlassen der Organisation. Im Fokus dieser Arbeit steht zunächst nur die erste Phase dieses Lebenszyklus. Bei der Beitrittsentscheidung muss der Softwareagent jedoch ebenfalls die erwarteten Konsequenzen der darauffolgenden Phasen in die Kalkulation mit einbeziehen. Voraussetzung für die Verhandlung ist ein gemeinsames Verständnis über den zu verhandelnden Gegenstand, insbesondere die durch den Softwareagenten auszuführende Rolle (Dastani et al., 2003). Der Begriff der Rolle wird dabei der Organisationstheorie entlehnt

und auf Softwareagenten übertragen. Die Auslegung dieser Rolle eines Softwareagenten in einer Multiagenten-Organisation kann verschiedenste Aspekte umfassen, insbesondere kann hierunter die Verpflichtung zur Erbringung eines Dienstes verstanden werden: *„The role is an abstract representation of an agent function, service or identification within a group.“* (Ferber und Gutknecht, 1998, S. 130).

Bereits in Abschnitt 2.2.3 wurde die Offenheit von Multiagentensystemen anhand der Definition von Petsch (2006) betrachtet. Dabei nutzt Petsch (2006) vertragsähnliche Konstrukte zur Fixierung von Rechten und Pflichten von Mitgliedern eines Multiagentensystems. Petsch (2006, S. 176 f.) thematisiert die Problematik, inwiefern es Softwareagenten aus rechtlicher Sicht überhaupt möglich ist, rechtlich verbindliche Willenserklärungen abzugeben, um Verträge einzugehen. Während Petsch (2006) diesen Aspekt aus seiner Arbeit ausklammert, gehen Müller-Hengstenberg und Kirn (2016) hierauf vertiefend ein. Da es sich bei Softwareagenten nicht um natürliche oder juristische Personen im Sinne des bürgerlichen Gesetzbuches handelt, können Softwareagenten zumindest zur Zeit der Entstehung dieser Arbeit in der Bundesrepublik Deutschland keine Vertragspartner werden. Softwareagenten *„sind stets im Eigentum und im Besitz einer natürlichen oder juristischen Person und stellen für diese ein Investitionsgut dar, dessen Entwicklung und Einsatz betriebliche Ressourcen verzehrt und deshalb Erträge liefern“* (Kirn und Müller-Hengstenberg, 2015, S. 60), weshalb ein Softwareagent unter Umständen Willenserklärungen abgeben kann, die gegen seinen Eigentümer bzw. Besitzer wirken. Dabei wird in der juristischen Literatur die überwiegende Auffassung vertreten, dass der Rechner nur die zuvor programmierten Befehle ausführt und der Nutzer sich bei einer weitgehenden Automatisierung des daraus resultierenden Risikos bewusst sein muss (Müller-Hengstenberg und Kirn, 2016, S. 140). Umstrittener hingegen ist die Frage, inwiefern durch Softwareagenten übermittelte Willenserklärungen durch den Eigentümer bzw. Besitzer angefochten werden können. Die Erörterung dieser Frage ist jedoch nicht Teil dieser Arbeit. Vielmehr ist für die Beitrittsentscheidung eines Softwareagenten zunächst nur die Rechtsverbindlichkeit der abgegebenen Willenserklärung und somit des zustande gekommenen Vertrags relevant. Ohne diese sind im Allgemeinen Sanktionen bei Pflichtverletzungen schwierig bis überhaupt nicht durchzusetzen, da Alternativen, wie Reputationsmechanismen (Petsch, 2006, S. 176) oder Sanktions-Richtlinien (Martinez et al., 2010) nur greifen, wenn der Softwareagent zum Zeitpunkt der Pflichtverletzung Vorteile aus der Mitgliedschaft bezieht.

2.4.3 Vertragliche Nebenbedingungen

In vielen Fällen ist die vertragliche Regelung der Dienstleistung an mindestens eine Bedingung geknüpft, die nicht Hauptgegenstand des Vertrags werden soll und hier als Nebenbedingung bezeichnet wird. Dies können beispielsweise Dienstgüteanforderungen oder auch Vertragsstrafen sein, die bei Nichterfüllung oder vorzeitiger Beendigung des Vertrags zu entrichten sind.

Zwar ist es im oben vorgestellten Kontraktnetz-Protokoll nach Smith (1980) nicht möglich, Verträge explizit vorzeitig zu beenden, jedoch hat das Nichterfüllen eines Vertrags keine nachteiligen Konsequenzen, was einem Vertragsausstieg gleichkommt. Sandholm und Lesser

(1995) erweitern das oben vorgestellte Kontraktnetz-Protokoll und eröffnen den Softwareagenten einen größeren Handlungsspielraum. Verträge sind weder ohne Konsequenzen noch überhaupt nicht kündbar, sondern jeder Vertragspartner kann, unter Zahlung der vereinbarten Vertragsstrafe, den Vertrag vorzeitig beenden. Diese *Leveled Commitments* bieten somit die Möglichkeit, Verträge, die unter Umständen ex-ante als sinnvoll bewertet wurden, sich allerdings ex-post als unvorteilhaft erwiesen haben, zu beenden. Abhängig von der jeweiligen Konstellation können hiervon sogar alle Vertragsparteien profitieren und es bieten sich somit Chancen, eine höhere Gesamtwohlfahrt zu erwirtschaften. Andersson und Sandholm (1998) untersuchen das vorgestellte Prinzip der *Leveled Commitments* auf die Auswirkungen im Gesamtsystem. Die Autoren betrachten in diesem Papier nicht-strategische Softwareagenten, untersuchen in einer späteren Arbeit allerdings auch strategisch handelnde Softwareagenten. Während im ursprünglichen Kontrakt-Netz-Protokoll (Smith, 1980, S. 1108) Softwareagenten beim Abschluss von mehreren Verträgen diese sequenziell abarbeiten müssen, lassen Andersson und Sandholm (2001, S. 616 ff.) auch mehrere Verträge parallel zu. Somit ist es Softwareagenten möglich, Mitglied in mehreren Gruppierungen zu sein, einschließlich der sich hieraus ergebenden Rechte und Pflichten. Das Erfüllen der eingegangenen Verträge zur Verhinderung drohender Vertragsstrafen obliegt in diesen Fällen der Mehrfachmitgliedschaft den einzelnen Softwareagenten.

Aus dem erweiterten Entscheidungsraum über das vorzeitige Auflösen von geschlossenen Verträgen entstehen neue Problemstellungen für den einzelnen Softwareagenten. Sandholm und Lesser (2002) geben einen Überblick über die spieltheoretische Betrachtung strategischer Decommitting-Entscheidungen. Statt einen Vertrag sofort aufzulösen, falls dieser als nicht vorteilhaft eingestuft wird, bietet sich dem Softwareagenten aus strategischer Sicht die Möglichkeit, auf die Kündigung der anderen Vertragspartei zu spekulieren und somit keine Vertragsstrafe zahlen zu müssen bzw. selbst sogar die Vertragsstrafe zu erhalten. In der Regel existiert keine dominante Strategie und das Nash-Gleichgewicht wird erreicht, indem von beiden Vertragsparteien die „beste“ Antwort-Strategie auf das „beste“ Verhalten des Vertragspartners gewählt wird. Als Beispiel wird von den Autoren die Luftfahrtindustrie genannt, in denen das Wissen über Nutzen und Kosten eines Flugabschnitts statistisch ermittelt werden kann und die Stornierung von Verträgen bereits einkalkuliert wird (Sandholm und Lesser, 2002, S. 93). Bereits aus diesen Ansätzen kann vermutet werden, dass Ansätze, die in der Luftfahrtindustrie angewandt werden, in gleicher oder angepasster Form einen Mehrwert bei der Beitrittsentscheidung von Softwareagenten haben könnten. Abschnitt 3.1 geht vertiefend auf die Anwendbarkeit der als „Revenue Management“ bekannten Ansätze auf die Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen ein.

Eine Möglichkeit der abstrakten Integration von vertraglichen Nebenbedingungen ist die Einführung von Dienstgüteklassen, die sich in eine für alle betrachteten Softwareagenten gleiche Präferenzordnung einfügen. Es wird beispielsweise angenommen, dass eine höhere Verfügbarkeit stets vor einer niedrigeren, sowie eine geringere Antwortzeit stets vor einer höheren präferiert wird. Bei einer ansteigenden Zahl an Nebenbedingungen – beispielsweise durch eine hohe Anzahl an SLA-Eigenschaften – steigt jedoch die Anzahl an potentiellen Vertragsgestaltungen sehr schnell an, so dass Verhandlungen unter Umständen

einen entsprechend langen Zeitraum einnehmen. Die Verhandlungen selbst konsumieren stets Ressourcen des potentiell beitretenden Softwareagenten als auch der entsprechenden Multiagenten-Organisation. Dementsprechend ist stets zwischen einem „optimalen“ Verhandlungsergebnis und den hierfür zu investierenden Verhandlungskosten – insbesondere in Form von Such- und Kommunikationskosten – abzuwägen (vgl. auch „*Begrenzte Rationalität*“ in Abschnitt 2.3.2.1.4).

2.4.4 Ressourcenallokation und Mehrfachmitgliedschaft

Um einer Multiagenten-Organisation Dienste bereitstellen zu können, stehen Softwareagenten Ressourcen zur Verfügung. Bei der Dienstbereitstellung werden diese Ressourcen genutzt und stehen somit für andere Dienste nicht mehr zur Verfügung. Abhängig von den zur Verfügung stehenden Ressourcen sowie dem Ressourcenbedarf eines einzelnen Dienstes, können unterschiedliche Dienste angeboten werden. Werden die Dienste durch die Multiagenten-Organisationen dauerhaft nachgefragt, kann der Softwareagent den Ressourcenbedarf eindeutig berechnen und die Dienstbereitstellung entsprechend anpassen.

Die Ressourcenauslastung kann allerdings auch abhängig von der Nutzung des bereitgestellten Dienstes sein: Wenn ein durch einen Softwareagenten bereitgestellter Dienst (z.B. die Lösung einer rechenintensiven Aufgabe) durch die Multiagenten-Organisation nicht genutzt wird, werden die hierfür notwendigen Ressourcen beim Softwareagenten ebenfalls nicht genutzt und stehen für andere Dienste zur Verfügung. Insbesondere wenn Dienste nur selten nachgefragt werden, müssen Softwareagenten abwägen, wieviele Verpflichtungen sie eingehen, um allen mit hoher Wahrscheinlichkeit nachkommen zu können. Dieses Problem verschärft sich, wenn Softwareagenten in mehreren Multiagenten-Organisationen Mitglied sind und dort ihre Dienste zur Verfügung stellen. Zusätzlich können unterschiedliche Regelwerke der verschiedenen Multiagenten-Organisationen durch Regelkonflikte zu Problemen bei Mehrfachmitgliedschaften führen. Diese Problemstellung wurde bereits von Kirn (1996a, S. 114 f.) aufgezeigt und im Kontext von Koalitionsformationen für Softwareagenten ebenfalls adressiert (Horling und Lesser, 2004, S. 288). Lösungsansätze für Mehrfachmitgliedschaften in Multiagenten-Organisationen nach dem Verständnis dieser Arbeit stehen jedoch nicht zur Verfügung. Hierzu müssen Softwareagenten in der Lage sein, ihren voraussichtlichen Ressourcenbedarf beim Beitritt zu einer Multiagenten-Organisation abzuschätzen und zusammen mit bestehenden Verpflichtungen in anderen Multiagenten-Organisationen zu bewerten.

Auch im Rahmen der betriebswirtschaftlichen Organisationstheorie werden Mitgliedschaften in mehreren Organisationen betrachtet (vgl. Abschnitt 2.3.2.1). Kieser und Walgenbach thematisieren Ressourcen, die nicht nur einer Organisation zur Verfügung gestellt werden und somit die „*Individuen gleichzeitig Mitglied in mehreren Organisationen*“ (Kieser und Walgenbach, 2007, S. 3) sind. Schreyögg sieht eine organisatorische Mitgliedschaft daher „*in der Regel nur [als] eine Teilmitgliedschaft ..., [da] nur ein Teil der Handlungen der fraglichen Organisation gilt (Partialinklusion), andere Handlungen gelten anderen Organisationen oder freien Zwecken*“ (Schreyögg, 2008, S. 9). Analogien der von betriebswirtschaftlichen Organisationen und Multiagenten-Organisationen finden sich beispielsweise durch Scott (1981, S.

246 ff.), der Organisationen als „*offene Systeme*“ bezeichnet, ebenso wie Bond und Gasser (1988, S. 7 f.) mit diesem Charakteristikum Systeme mit Softwareagenten beschreiben.¹⁷ Zwar verwenden die Autoren jeweils die gleichen Begriffe, beziehen den Begriffsteil „*offen*“ jedoch auf verschiedene Gegebenheiten. Während Scott die Organisation selbst als offen und zwar „*auf den Austausch mit anderen Systemen angewiesen*“ (Scott, 1981, S. 246) sieht, verwenden Bond und Gasser den Begriff, um Systeme zu beschreiben, in denen es keine zentrale Entscheidungen oder Wissensrepräsentation gibt (Bond und Gasser, 1988, S. 3).

2.4.5 Fazit und Forschungslücke

In den vorangegangenen Abschnitten wurde der Stand der Forschung zum Thema Beitrittsentscheidungen zu Multiagenten-Organisationen sowie die hierfür notwendigen Begriffe und Konzepte beschrieben sowie eingehend analysiert. Sowohl in der Organisationstheorie als auch der VKI wird das Thema der Beitrittsentscheidungen nur in wenigen Arbeiten betrachtet, obwohl der Term *Mitgliedschaft* in der Organisationstheorie weite Verbreitung findet und im Allgemeinen eine vorangegangene Beitrittsentscheidung voraussetzt. In der VKI hingegen wird in der Forschungsrichtung der Multiagentensysteme von so genannten *Open Systems* ausgegangen (vgl. Abschnitt 2.2.3) und somit ein Beitritt zu Multiagentensystemen und Multiagenten-Organisationen vorausgesetzt. Die Beitrittsentscheidung jedes einzelnen Softwareagenten hingegen wird nur in wenigen Arbeiten betrachtet, während andere Arbeiten zentrale Kontrollinstanzen mit einheitlichen Verfahren voraussetzen und somit dem Gedanken von *Open Systems* zuwiderlaufen (vgl. auch Abschnitt 2.3.1.2). Insbesondere der Bereich der Koalitions-, Team- und Kongregationsformation wird in der Literatur ausschließlich aus Top-Down-Perspektive betrachtet, wobei im Allgemeinen eine zentrale Kontrollinstanz notwendig ist, um ein einheitliches Verfahren durchzusetzen. Dabei wird von einer vorgegebenen Menge an Softwareagenten ausgegangen, ein Beitritt von neuen Softwareagenten ist nicht vorgesehen.

Die in Definition 7 aufgegriffene Forderung der vertraglichen Regelung der Mitgliedschaft wird in der Literatur unter anderem durch *Leveled Commitments* Rechnung getragen (vgl. Abschnitt 2.4.3). Durch das Einbeziehen vertraglicher Nebenbedingungen wird die Komplexität der Verhandlung erweitert und kann zur individuellen Gestaltung von Mitgliedschaftsbeziehungen genutzt werden. Bei der Beitrittsentscheidung von Softwareagenten zu Multiagenten-Organisationen sind diese vertraglichen Nebenbedingungen zu berücksichtigen. Dabei ist in der Literatur derzeit offen, wie diese Entscheidungen für den Softwareagenten nutzenmaximierend auszugestalten sind.

Bei jeder Beitrittsentscheidung sind die bisher eingegangenen Verpflichtungen in anderen Multiagenten-Organisationen zu berücksichtigen (vgl. auch Thema *Offenheit* in Abschnitt 2.2.3). Insbesondere diese potentiellen Mehrfachmitgliedschaften stellen eine besondere Herausforderung für die Ausgestaltung eines Entscheidungsverfahrens für Beitritte zu Multiagenten-Organisationen dar.

¹⁷ zum Begriff der Offenheit im Forschungsfeld der verteilten künstlichen Intelligenz vgl. Abschnitt 2.2.3

3. Verfahrensentwurf

Die Beitrittsentscheidung von Softwareagenten zu Gruppierungen mit anderen Softwareagenten ist stets abhängig von den Beitrittsbedingungen als auch den bereits bestehenden Verpflichtungen und den somit verfügbaren Ressourcen des potentiell beitretenden Softwareagenten. Die durch eine Mitgliedschaft gebundenen Ressourcen sind insbesondere bei Beitrittsentscheidungen zu Multiagenten-Organisationen relevant, da diese nach Definition 6 einen dauerhaften Zusammenschluss bilden und somit die dem potentiell beitretenden Softwareagenten zur Verfügung stehenden Ressourcen ebenfalls für einen a priori unbekannten Zeitraum binden.

Revenue Management hat sich als eine Teildisziplin des *Operations Research* seit den 1970er Jahren entwickelt. Insbesondere stehen dabei die Umsatz- oder Gewinnmaximierung im Fokus, unter Berücksichtigung begrenzter zur Verfügung stehender Ressourcen, der bisherigen Verpflichtungen sowie der zukünftig erwarteten Nachfrage (Meffert, 1995, S. 464). Die Ansätze des Revenue Management lassen sich dabei zunächst in Kapazitäts-basierte und Preis-basierte Ansätze unterteilen. Während erstere eine Maximierung des erwarteten Umsatzes bzw. Gewinns durch die Steuerung der zur Verfügung gestellten Kapazität erzielen, erreichen zweitere dieses Ziel durch die Anpassung des Preises des angebotenen Produkts oder der angebotenen Dienstleistung. Wie der Abbildung 3.1 zu entnehmen ist, können auch Auktionen dem Preis-basierten Revenue Management zugeordnet werden.

Im Gegensatz zu anderen Verfahren zur Steuerung von Vertragsbeziehungen werden den Kapazitäts-basierten Revenue Management-Ansätzen einige methodische Unterschiede zugeschrieben: Gegenüber Auktionen bieten Kapazitäts-basierte Revenue Management Ansätze die Möglichkeit Ressourcen unabhängig von Verfahren Dritter und von der Anzahl der beteiligten Akteure zu allokalieren. Auktionen sind zwar auch bei nur zwei verhandelnden Akteuren möglich, verletzen dann allerdings das in der Literatur im Allgemeinen geforderte Kriterium der Anreizkompatibilität. Während sequenziell ablaufende Verhandlungen wechselseitige Angebotserstellungen und damit auch eine partielle Preisgabe der eigenen Präferenzordnung erfordern, deren Komplexität insbesondere bei einer hohen Anzahl von Nebenbedingungen stark steigt, ermöglichen Kapazitäts-basierte Revenue Management

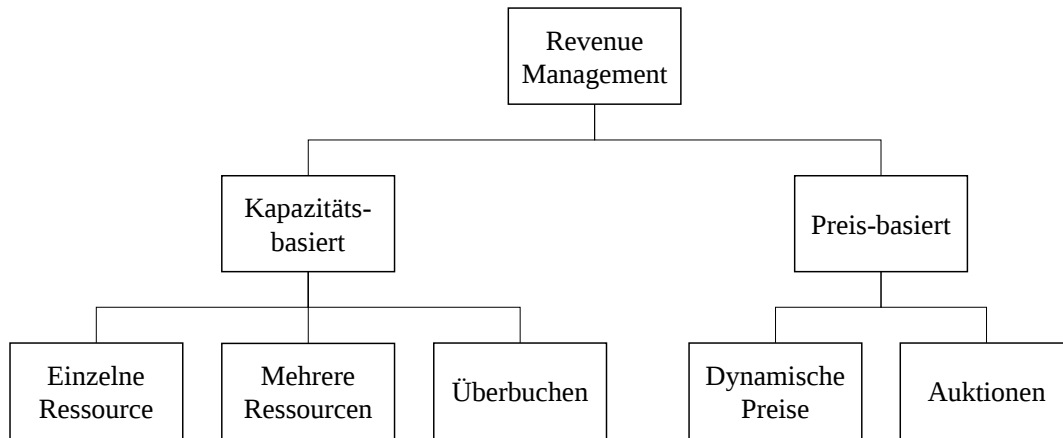


Abbildung 3.1: Revenue Management Ansätze (Talluri und van Ryzin, 2004)

Ansätze im Allgemeinen eine Abwicklung in wenigen Schritten durch eine dementsprechend aufwendige Vorberechnung. Dieser Abschnitt betrachtet die Anwendung von Revenue Management auf die Beitrittsentscheidung von Softwareagenten zu Multiagenten-Organisationen und passt bestehende Verfahren auf die entsprechenden Besonderheiten an.

Zur Entwicklung eines Revenue Management-basierten Verfahrens zur Beitrittsentscheidung von Softwareagenten wird zunächst der Sachverhalt mit Hilfe mathematischer Konstrukte modelliert. Die entsprechende Darstellung des zu lösenden Problems stellt einen wesentlichen Beitrag zur Lösung selbst dar. Simon (1996) beschreibt diesen Sachverhalt wie folgt: „*Solving a problem simply means representing it so as to make the solution transparent*“ (Simon, 1996, S. 132). Dies trifft insbesondere auf die Entwicklung eines Revenue Management-basierten Verfahrens für Beitrittsentscheidungen von Softwareagenten zu, da sich die Darstellung des Problems von bisherigen Herangehensweisen unterscheidet. Die Modellierung ist somit bereits ein wesentlicher Bestandteil des Artefakts und dessen Entwicklung somit auch Teil des Verfahrensentwurfs.

Abschnitt 3.1 analysiert die Anwendbarkeit bzw. Übertragbarkeit von Revenue Management Ansätzen auf Beitrittsentscheidungen von Softwareagenten und geht dabei auch auf andere Anwendungsbereiche von Revenue Management ein. Abschnitt 3.2 modelliert die Beitrittsentscheidung zur Anwendung von Revenue Management Ansätzen, Abschnitt 3.3 modelliert die Prognose a priori unbekannter Parameter. Abschnitt 3.4 überträgt Ansätze des Revenue Management zur Anwendung auf die modellierten Beitrittsentscheidungen von Softwareagenten. Abschließend geht Abschnitt 3.5 auf das *Überbuchen* als erweitertes Verfahren des Revenue Management ein.

3.1 Anwendbarkeit von Revenue Management auf Beitrittsentscheidungen von Softwareagenten

Die Anreiz-Beitrags-Theorie nimmt an, dass sich die Nutzenfunktionen der (potentiellen) Organisationsmitglieder sehr ähnlich sind, was im Kontext von Multiagenten-Organisationen jedoch in dieser Allgemeinheit nicht gültig ist (vgl. Abschnitt 2.4.1). Rückt man von dieser

Annahme ab, beeinträchtigt dies zwar die Anwendung der Anreiz-Beitrags-Theorie selbst, bietet allerdings anderweitige Gestaltungsmöglichkeiten. Das Vorhandensein unterschiedlicher Nutzenfunktionen ermöglicht die Anwendung diskriminierender Ansätze mit denen je nach Ausgestaltung der Nutzen aller involvierten Parteien erhöht werden kann. Dabei wird im Allgemeinen versucht, einen Vorteil aus der unterschiedlichen Zahlungsbereitschaft der Individuen für ein Gut bzw. einen Dienst bei verschiedenen Nutzenfunktionen zu ziehen. Entsprechende Ansätze finden sich verstärkt im *Revenue Management* als Teilbereich des *Operation Research*. Typische Anwendungsfelder des Revenue Management sind Fluggesellschaften, Hotels sowie Autovermietungen, deren Produkte bzw. Dienste eine wesentliche Eigenschaft gemein haben: Wird ein Produkt bis zu seinem Verfallsdatum verkauft oder ein Dienst zu einem bestimmten Zeitpunkt nicht erbracht, können die für dessen Produktion bzw. Erbringung notwendigen Ressourcen nicht zu einem späteren Zeitpunkt eingesetzt werden, sind folglich nicht oder nur eingeschränkt lagerbar (Talluri und van Ryzin, 2004).

Im Gegensatz zu Menschen in einer Organisation können Softwareagenten als technisches System quantifizierbare Diskriminierungskriterien wie die in Abschnitt 2.4.3 vorgestellten vertraglichen Nebenbedingungen effizienter umsetzen. So können beispielsweise durch Dienstgütevereinbarungen Antwortzeiten, Verfügbarkeiten, etc. definiert und durch den Softwareagenten umgesetzt werden. Tritt ein Softwareagent einer Multiagenten-Organisation bei, können hierbei ebenfalls vertragliche Nebenbedingungen in verschiedenen Ausprägungen vereinbart werden, welche das Verhalten des Softwareagenten in Bezug auf dessen Bereitstellung seiner angebotenen Dienste an die Multiagenten-Organisation beeinflussen.

Im Zeitpunkt der Beitrittsentscheidung ist es möglich, dass der Softwareagent bereits Mitglied in mindestens einer anderen Multiagenten-Organisation ist. Bei der Beitrittsentscheidung müssen somit bestehende Mitgliedschaftsbeziehungen zu anderen Multiagenten-Organisationen in die Entscheidungsfindung mit einbezogen werden. Durch diese Mehrfachmitgliedschaften in mehreren Multiagenten-Organisationen entstehen Szenarien mit mehrstufigen Wertschöpfungssystemen.

3.1.1 Revenue Management und dessen Anwendungen

Die Ursprünge der Ansätze, die unter der Bezeichnung *Revenue Management* bekannt sind, werden in der Literatur oftmals der amerikanischen Luftfahrtindustrie zugeschrieben, die in den 1970er Jahren durch verschiedene Maßnahmen versuchte, ihren Umsatz und Gewinn zu steigern (Chiang et al., 2007, S. 98). Durch die damalige Gesetzgebung waren die Luftfahrtgesellschaften in ihren Preisgestaltungsmaßnahmen eingeschränkt und mussten zur Umsatz- und Gewinnsteigerung somit auf neue Verfahren setzen. Tatsächlich sind allerdings einige der Verfahren des Revenue Management nicht neu und wurden schon vor der spezifischen Problemstellung der Luftfahrtindustrie eingesetzt: Dies wird insbesondere durch die Übersicht in Abbildung 3.1 deutlich, die auch Auktionen den Preis-basierten Revenue Management Ansätzen zuschreibt. Auch bei anderen Ansätzen gab es bereits Vorläufer: Preisbildungsmechanismen auf Märkten haben schon vor der Luftfahrtindustrie erkannt, dass Produkte – wie auch Dienstleistungen – ein Verfallsdatum haben, welches in die Preisbildung einfließen kann. Während Kunden auf einem Markt morgens den

vollen Preis bezahlen, bieten Händler im Tagesverlauf oftmals Preisnachlässe an, falls sich abzeichnet, dass nicht die gesamte Ware zum vorgesehenen Preis verkauft werden kann (Dynamische Preise). Alternativ können beispielsweise Produkte deren Haltbarkeit eingeschränkt ist (z.B. Fallobst) günstiger angeboten werden (Kapazitäts-basierter Ansatz).

Diese als Revenue Management¹⁸ bezeichneten Ansätze wurden jedoch im Kontext der amerikanischen Luftfahrtindustrie erstmals formalisiert und auf dieser formalen Basis weiterentwickelt. Diese Formalisierung ermöglichte jedoch auch die Übertragung auf andere Anwendungsdomänen: Insbesondere Hotels und Kraftfahrzeugvermietungen machen in größerem Umfang Gebrauch dieser Ansätze (Chiang et al., 2007, S. 101). Ein Vorteil des Revenue Management ist dabei die leichte Anwendbarkeit auf bestehende Preisbildungsverfahren. Im Allgemeinen ist keine Änderung der Preisstruktur nötig, sondern es wird ausschließlich die Art der Entscheidungsfindung beeinflusst: *„RM does not involve radically changing the structure of pricing and sales practices; rather, it is a matter of making more intelligent decisions“* (Talluri und van Ryzin, 2004, S. 11).

In den oben angeführten Kategorien des Revenue Management ist es für Preis-basierte Ansätze charakteristisch, dass der Preis für das Produkt, die Dienstleistung oder auch durch Buchungsklassen differenzierte Ausprägung hiervon durch dynamische Anpassung variiert oder mit Hilfe von Auktionen bestimmt wird. Kapazitäts-basierte Ansätze zeichnen sich hingegen dadurch aus, dass Preise für Produkte, Dienstleistungen oder entsprechende Buchungsklassen a priori bestimmt werden und zur Ausführungszeit hingegen die Verfügbarkeit des Angebots durch kapazitätssteuernde Maßnahmen determiniert wird. Unter Buchungsklassen werden dabei Ausprägungen eines Produkts bzw. einer Dienstleistung verstanden, deren Kernbestandteil identisch ist (beispielsweise die Beförderung von A nach B mit einem Flugzeug), deren Nebenbedingungen sich jedoch unterscheiden (Buchungsklasse 1 ermöglicht kostenloses umbuchen, Buchungsklasse 2 erhebt ggf. Umbuchungsgebühren). Dabei wird davon ausgegangen, dass es Kunden gibt, die bereit sind, einen Aufpreis für die höherpreisige Buchungsklasse zu zahlen. Die Zuordnung der Kunden zu den angebotenen Buchungsklassen und somit der Zahlungsbereitschaft zu den unterschiedlich bepreisten Produkten bzw. Diensten findet dabei durch den Kunden selbst statt und wird somit der Preisdifferenzierung zweiten Grades zugeordnet (Diller, 2008, S. 228 f.). Es besteht stets die Gefahr, dass Kunden eine Buchungsklasse wählen, die unterhalb ihrer Zahlungsbereitschaft liegt und dem Anbieter somit ein nicht realisiertes Umsatzpotential entgeht. Die Kapazitäts-basierten Ansätze des Revenue Management setzen an diesem Punkt an und versuchen Antworten auf das Entscheidungsproblem der Kapazitätssteuerung zu finden.

Kern der Revenue Management Ansätze sind dabei stets die Abhängigkeiten zwischen den verschiedenen Dimensionen, die auf die Wertschätzung des Kunden für das Produkt bzw. für die Dienstleistung einwirken. Abbildung 3.2 zeigt die drei wesentlichen Dimensionen (i) *Kunde*, (ii) *Produkt* bzw. *Dienst* sowie (iii) *Zeit*, welche auf die vierte Dimension *Wertschätzung* einwirken (Talluri und van Ryzin, 2004, S. 11). Diese Abhängigkeiten sind verantwortlich dafür, dass das Entscheidungsproblem, für welchen Kunden für welchen Dienst zu welchem

¹⁸ In der Literatur wird teilweise auch der Begriff *Yield Management* verwendet. Die beiden Begriffe Revenue Management und Yield Management werden in dieser Arbeit synonym verwendet.

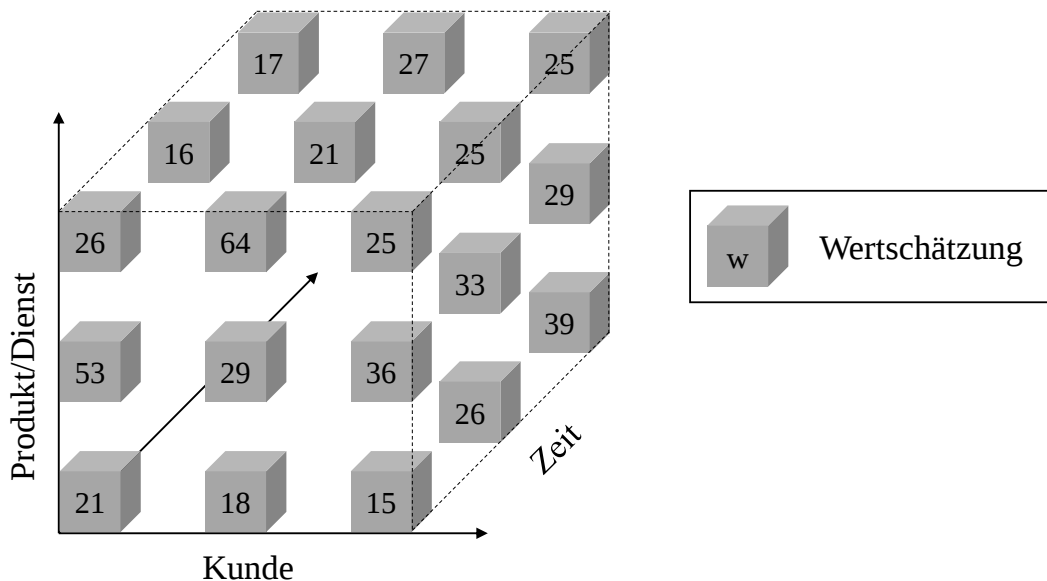


Abbildung 3.2: Vierdimensionale Darstellung der Produkt- bzw. Dienstnachfrage (Talluri und van Ryzin, 2004, S. 11)

Zeitpunkt welcher Preis (Preis-basierte Ansätze) bzw. welche Kapazität (Kapazitäts-basierte Ansätze) festgesetzt werden, entscheidend an Komplexität zunimmt (Talluri und van Ryzin, 2004, S. 12 f.). Um die verschiedenen Einflussfaktoren der Anwendungsdomänen einheitlich darzustellen, haben Weatherford und Bodily (1992) eine entsprechende Taxonomie entwickelt, auf die in Abschnitt 3.2.1 zur Problemklassifizierung zurückgegriffen wird.

3.1.2 Voraussetzungen zur Anwendung von Revenue Management-basierten Ansätzen

Für die Anwendung von Revenue Management-basierten Ansätzen werden in der Literatur verschiedene Voraussetzungen genannt, die erforderlich sind, damit die Anwendung dieser Ansätze einen positiven Nutzen für den Anwender stiftet. Die Ausprägungen dieser Voraussetzungen unterscheiden sich zwar abhängig von der jeweiligen Anwendungsdomäne, jedoch können Kriterien identifiziert werden, die eine domänenübergreifende Analyse ermöglichen. Angelehnt an Talluri und van Ryzin (2004, S. 13 ff.) können hierzu fünf Voraussetzungen zusammengefasst werden: (i) *Heterogenität der Kunden*, (ii) *Nachfrageschwankungen und Unsicherheiten*, (iii) *Inflexibilität der Produktion*, (iv) *Preis ist kein Qualitätsmerkmal* sowie (v) *Informationstechnische Infrastruktur*. Diese Voraussetzungen werden in den folgenden Abschnitten erläutert und im Kontext von Beitrittsentscheidungen von Softwareagenten analysiert.

3.1.2.1 Heterogenität der Kunden

Die Ansätze des Revenue Management nutzen die Heterogenität der adressierten Kunden aus, um einen möglichst hohen Umsatz zu generieren. Diese Heterogenität kann verschiedene Aspekte umfassen: Im Allgemeinen wird davon ausgegangen, dass Kunden unterschiedliche Präferenzen bezüglich der angebotenen Produkte bzw. Dienste sowie der damit verknüpften vertraglichen Nebenbedingungen aufweisen. Jedoch können neben den Produkt- bzw.

Diensteigenschaften noch weitere Aspekte in die Präferenzen der Kunden einfließen. In klassischen Revenue Management Domänen wie der Luftfahrt- oder der Hotelindustrie fließen auch zeitliche Komponenten mit ein: Je nach Einschätzung der Kundenpräferenzen können auf diese Weise sehr kurzfristige Buchungen teurer oder günstiger abgegeben werden. Diese unterschiedlichen Wertschätzungen der verschiedenen Aspekte werden genutzt, um eine Preisdiskriminierung durchzusetzen und einen möglichst hohen Anteil der Zahlungsbereitschaft abzuschöpfen. Dazu werden die Kunden im Allgemeinen in Segmente unterteilt, so dass sich die Präferenzen von Kunden eines Segments möglichst ähneln und Produkte bzw. Dienste mit den dazugehörigen vertraglichen Nebenbedingungen auf die Kunden dieses Segments zugeschnitten. Der Verzicht auf die Heterogenität der Kunden würde eine essentielle Dimension des Diagramms in Abbildung 3.2 entfernen: Die Dimension Kunde hätte keinen Einfluss auf die Zahlungsbereitschaft und somit ebenfalls nicht auf den Preis.

Für Softwareagenten bestehen Kunden aus Dienstabnehmern, die sich im Rahmen dieser Arbeit aus anderen Softwareagenten, Multiagentensystemen sowie insbesondere Multiagenten-Organisationen zusammensetzen. Softwareagenten können diesen Kunden Dienste zur Verfügung stellen und erhalten für diese Leistung eine Kompensation. Da sich im Allgemeinen nur durch die Mitgliedschaft in einer Multiagenten-Organisation eine langfristige Bindung des Softwareagenten und damit auch dessen Ressourcen zur Dienstbereitstellung besteht, stellen diese „Kunden“ für den Softwareagenten das Segment dar, deren Bindung sich umfangreich auf dessen Flexibilität auswirkt und eine Beitrittsentscheidung entsprechend abgewogen werden muss. Da für einen Softwareagenten in Frage kommende Multiagenten-Organisationen aus verschiedensten Kulturkreisen entstammen und somit mit hoher Wahrscheinlichkeit auch unterschiedliche Präferenzen aufweisen, stehen auch Softwareagenten heterogenen „Kunden“ gegenüber. Auch Multiagenten-Organisationen lassen sich aus Sicht eines potentiell beitretenden Softwareagenten in verschiedene Segmente unterteilen, die einzelne Aspekte der vertraglichen Nebenbedingungen eines Dienstes differenziert bewerten. So hängt bereits das Kriterium Antwortzeit einer Dienstgütevereinbarung, welches je nach Vereinbarung die Flexibilität des Softwareagenten mehr oder weniger stark beeinträchtigt, von der weiteren Verwendung in der Multiagenten-Organisation ab. Die Heterogenität der Diensteannehmer kann im Falle von Softwareagenten somit angenommen werden und bildet eine der Grundvoraussetzungen für Ansätze des Revenue Management.

3.1.2.2 Nachfrageschwankungen und Unsicherheiten

In Situationen, in denen die Nachfrage nach einem Produkt bzw. einem Dienst perfekt vorhersehbar ist, können Entscheidungen auf dieser Grundlage im Allgemeinen optimal berechnet werden. Die Entscheidungstheorie bezeichnet solche Situationen als Entscheidungen unter Sicherheit (Laux, 2005a, S. 22). Da auf die Auswirkungen einer Entscheidung meist auch Variablen einwirken, die vom Entscheider nicht überblickt oder eingeschätzt werden können, sind diese Situationen in der Realität kaum anzutreffen. Häufig müssen hingegen Entscheidungen aus Sicht des Entscheiders unter Unsicherheit oder Ungewissheit getroffen werden. In klassischen Revenue Management Anwendungsbereichen ist dies insbesondere die Nachfrage nach dem angebotenen Produkt bzw. Dienst. Diese kann aufgrund vielfältiger

Faktoren, beispielsweise zeitlicher (Tageszeit, Wochentag, Jahreszeit, etc.) oder wirtschaftlicher (Kaufkraft, Konjunktur, etc.) Art, schwanken und sich somit auf den zu erwartenden Umsatz auswirken (Pechtl, 2005, S. 252). Die Variabilität der Nachfrage ist eine wesentliche Voraussetzung für den Einsatz von Revenue Management Ansätzen. Zur Nachfrage zählen dabei sowohl die Nachfrage im engeren Sinne in Form der Buchungshäufigkeit, aber auch die Nachfrage nach dem Abruf des Produkts bzw. Dienstleistung nach erfolgter Buchung. Letzteres ist beispielsweise aus der Luftfahrtindustrie sowie dem Hotelgewerbe bekannt: Trotz Buchung werden manche Ressourcen vom Kunden nicht abgerufen (der Flug bzw. die Übernachtung nicht angetreten) und können nicht zur Generierung von Umsatz beitragen.

Die Nachfrage bei der Beitrittsentscheidung von Softwareagenten entsteht durch die Anfrage durch Multiagenten-Organisationen nach der Bereitstellung von Diensten. Dabei handelt es sich um die oben bezeichnete Nachfrage im engeren Sinne zum Eingehen einer beidseitigen Verpflichtung. Diese Nachfrage nach Diensten des Softwareagenten ist diesem a priori nicht bekannt und jede Änderung der Nachfrage erscheint ihm als nicht beeinflussbare Nachfrageschwankung. Eine Beitrittsentscheidung zu einer Multiagenten-Organisation ist folglich ebenfalls eine Entscheidung unter Unsicherheit bzw. Ungewissheit. Allerdings ist die Ressourcennutzung des Softwareagenten nicht ausschließlich von diesen Beitrittsentscheidungen abhängig, sondern analog zu den oben genannten Beispielen von der Nachfrage nach dem Abruf des Dienstes. Es ist durchaus möglich, dass ein Softwareagent zwar durch den Beitritt zur Bereitstellung eines Dienstes verpflichtet ist, dieser Dienst jedoch nie durch die Multiagenten-Organisation abgerufen wird. Ressourcen, die der Softwareagent ausschließlich zu diesen Zwecken reserviert hat, verbleiben ungenutzt und stünden somit für andere Dienste zur Verfügung (vgl. auch Abschnitte 3.1.3.3 und 3.5). Softwareagenten stehen bei einer Beitrittsentscheidung zu Multiagenten-Organisationen demnach mehreren Nachfrageschwankungen gegenüber, was den Einsatz von Revenue Management Ansätzen befürwortet.

3.1.2.3 Inflexibilität der Produktion

Ist ein Anbieter von Produkten oder Diensten in der Lage, die Produktion bzw. die Ressourcenkapazität an die Nachfrage mit einem zu vernachlässigenden Aufwand anzupassen, ist die Komplexität seiner Entscheidungsfindung deutlich verringert: Er produziert genau so viel, bzw. stellt exakt die Ressourcenkapazität zur Verfügung, die die Nachfrage erfordert (Talluri und van Ryzin, 2004, S. 14). In den traditionellen Revenue Management Disziplinen ist jedoch genau das Gegenteil der Fall: Die Ressourcen, die zur Produkt- oder Dienstbereitstellung zur Verfügung stehen, sind in Bezug auf eine Anfrage fix und können nur langfristig und meist in großen Schritten erhöht oder verringert werden (Pechtl, 2005, S. 251). Am Beispiel der Luftfahrtindustrie sind die zur Verfügung stehenden Sitzplätze auf einem Flugabschnitt nicht variabel, die Kapazität kann nur durch langfristig geplante Anpassungen, wie dem Einsatz eines weiteren Flugzeuges oder dem Streichen eines Flugabschnitts, erhöht bzw. verringert werden. Auch Hotels können ihre zur Verfügung stehenden Zimmerkapazitäten nur langfristig durch An- oder Umbauten ändern. Diese Inflexibilität der vorhandenen Ressourcen durch vorgegebene Kapazitäten erhöht sich weiter durch mangelnde Lagerfähigkeit von Diensten. Dienste sowie die hierfür benötigten Ressourcen

lassen sich nicht lagern: Ein Hotel kann das in einer Nacht leerstehende Zimmer nicht in der darauffolgenden Nacht zusätzlich verkaufen, der Umsatzverlust, der durch diesen Leerstand verursacht wird, kann nicht in späteren Zeitperioden wieder gewonnen werden. Ebenso sind bereits zustande gekommene Verträge nicht ohne kurzfristig anfallende Kosten, z.B. durch Vertragsstrafen, kündbar. Andernfalls könnten Dienstanbieter Verträge mit einem geringeren zu erwartenden Gewinn zugunsten neuer Anfragen mit entsprechend höheren Gewinnerwartungswerten auflösen. Entsprechend langfristig ausgelegte Verträge sind demnach essentiell für die Stabilität der vertraglichen Bindung. Dementsprechend ist ein Überwachungs- und Durchsetzungssystem nötig, wie es auch in der Neuen Institutionenökonomik gefordert wird, um diese sich nicht selbst durchsetzenden oder impliziten Verträge überwachen sowie durchsetzen zu können (vgl. Abschnitt 2.3.2.1.4). Die Kombination dieser verschiedenen Dimensionen der Inflexibilität der Produktion ist ein wesentlicher Faktor für den Einsatz von Revenue Management-basierten Ansätzen.

Diese Inflexibilität der zur Verfügung stehenden Ressourcen findet sich auch bei der Dienstbereitstellung von Softwareagenten. Zum einen stellen Softwareagenten einer Multiagenten-Organisation analog den traditionellen Revenue Management Domänen ebenfalls Dienste bereit, was eine mögliche Lagerung der angebotenen Leistung bereits ausschließt, unabhängig von der Art des angebotenen Dienstes. Zum anderen sind die vorhandenen Ressourcen zur Dienstbereitstellung im Allgemeinen ebenfalls begrenzt, stehen somit nicht in unbeschränktem Umfang zur Verfügung. Die Art der Ressourcen eines Softwareagenten kann sich abhängig vom Einsatzzweck stark unterscheiden und kann auch physische Ressourcen umfassen. Allen Softwareagenten ist jedoch gemein, dass ihre nativen Rechen- und Speicherkapazitäten beschränkt sind, die im Allgemeinen ebenfalls zur Dienstbereitstellung benötigt werden. Insbesondere rechen- oder speicherintensive Dienste beanspruchen die ausführungsrelevanten Ressourcen des Softwareagenten, wie Prozessor, Haupt- und Festplattenspeicher, in besonderem Maße. Der Softwareagent wird im Allgemeinen jedoch nicht in der Lage sein, die Menge der zur Verfügung stehenden ausführungsrelevanten Ressourcen zu erhöhen, da hierzu physische Eingriffe in die Rechnerarchitektur oder die Migration auf andere Systeme¹⁹ nötig wären. Aus Sicht des Softwareagenten können die zur Verfügung stehenden Ressourcen somit als vorgegeben betrachtet werden, was ebenfalls Einflüsse auf die Beitrittsentscheidung zu Multiagenten-Organisationen hat und die Komplexität der Entscheidungsfindung erhöht: Der Beitritt zu einer Multiagenten-Organisation bindet die Ressourcen des Softwareagenten, sodass diese nicht mehr für andere Aufgaben zur Verfügung stehen. Ressourcen hingegen, die in einer Zeitperiode nicht genutzt werden, verursachen ein ungenutztes Potential, welches nicht mehr in nachfolgenden Zeitperioden erwirtschaftet werden kann.

3.1.2.4 Preis ist kein Qualitätsmerkmal

Während der Preis für ein Produkt oder einen Dienst im Allgemeinen nur die Bewertung dessen darstellt und nicht Teil des Produktes bzw. des Dienstes selbst ist, kann es bei bestimmten Produkten und Diensten dazu kommen, dass der Preis als Qualitätsmerkmal

¹⁹ zu mobilen Agenten vgl. Anhalt (2011)

angesehen wird (Gabor und Granger, 1966, S.43 ff.). Dies wiederum führt dazu, dass ein Konsument höherpreisige Produkte bzw. Dienste niederpreisigen vorzieht, selbst wenn sich die restlichen Merkmale eines Produkts oder eines Dienstes nicht unterscheiden. Zum einen kann es sich dabei um Luxusartikel handeln, bei denen ein hoher Preis die Menge der interessierten Käufer einschränkt und ihnen somit eine gewisse Exklusivität sichert. Zum anderen können mangelnde Informationen über die wahren Qualitätseigenschaften eines Produkts bzw. eines Dienstes einen interessierten Käufer zur Annahme verleiten, dass ein höherpreisiges Produkt bzw. ein höherpreisiger Dienst eine in irgendeiner Form besser geartete Qualität bereitstellt als sein niederpreisiges Äquivalent. Da sich in diesen Situationen eine Änderung des Preises ebenfalls auf die wahrgenommene Qualität auswirkt, sind Revenue Management-basierte Ansätze nicht für derartige Produkte bzw. Dienste geeignet (Talluri und van Ryzin, 2004, S. 14 f.). In den traditionellen Revenue Management Domänen ist die Qualität des angebotenen Dienstes durch andere Faktoren als den Preis bestimmbar, beispielsweise durch die Einteilung in Economy- und Business-Klassen oder die Sterneklassifizierung im Hotelgewerbe.

Bei den Beitrittsverhandlungen von Softwareagenten zu Multiagenten-Organisationen ist ebenfalls davon auszugehen, dass die Multiagenten-Organisation den Preis, d.h. die Kompensation, die ein Softwareagent für die verpflichtende Bereitstellung eines Dienstes erhält, nicht als Qualitätsmerkmal gesehen wird. Im Gegenteil zu anderen Diensten können die von Softwareagenten bereitgestellten Dienste durch Dienstgütekriterien konkretisiert und daher für den Dienstabnehmer – die Multiagenten-Organisation – bewertbar gemacht werden. Die Multiagenten-Organisation hat somit keine Ungewissheit über die Art der Bereitstellung und sieht die Kompensation somit losgelöst von der Diensterbringung selbst. Die Kompensation ist somit kein Qualitätsmerkmal und die Multiagenten-Organisation ist bestrebt diese Kompensation zu minimieren, um ihren eigenen Nutzen zu maximieren. Damit ist diese Voraussetzung zur Anwendung von Revenue Management-basierten Ansätzen für die Beitrittsentscheidung von Softwareagenten zu Multiagenten-Organisationen erfüllt.

3.1.2.5 Informationstechnische Infrastruktur

Ansätze aus dem Revenue Management benötigen umfassende Informationen über die Nachfrage nach angebotenen Produkten bzw. Diensten und die implementierten Systeme sollten Entscheidungen über Angebote und Preise möglichst automatisiert einfließen lassen. Eine informationstechnische Infrastruktur zur Datenerhebung über eingehende Anfragen, Ressourcenauslastungen, etc. ist somit Grundvoraussetzung für die Anwendung dieses Verfahrens (Talluri und van Ryzin, 2004, S. 15).

Im Falle von Beitrittsverhandlungen von Softwareagenten mit Multiagenten-Organisationen erfolgen Anfragen und Angebote über angebotene Dienste mittels informationstechnischer Kommunikation, sodass die benötigte Infrastruktur für die Verarbeitung entsprechender Informationen per se gegeben ist. Es liegt somit am Softwareagenten selbst, inwiefern er diese Informationen speichert und auswertet. Wie für die Dienstbereitstellung selbst, müssen für die Anwendung des in dieser Arbeit entwickelten Verfahrens dem Softwareagenten zur Verfügung stehende Ressourcen bereitgestellt werden: Zur Speicherung von

Nachfragestatistiken muss der Softwareagent Speicherplatz vorhalten, für die Anwendung des Verfahrens selbst wird Prozessorleistung benötigt und Hauptspeicher belegt. In dieser Arbeit wird davon ausgegangen, dass Softwareagenten für die Anwendung des entwickelten Verfahrens Zugriff auf eigenständige Ressourcen besitzen, die überschneidungsfrei mit den Ressourcen zur Bereitstellung von Diensten sind.

3.1.3 Voraussetzungen zur Anwendung erweiterter Verfahren des Revenue Management

Während Abschnitt 3.1.2 die Voraussetzungen für die allgemeine Anwendung von Revenue Management Ansätzen für Beitrittsentscheidungen von Softwareagenten untersucht hat, existieren Revenue Management Ansätze, die weitergehende Möglichkeiten der Umsatz- bzw. Gewinnmaximierung nutzen. Hierzu lassen sich in der Literatur drei wesentliche Voraussetzungen identifizieren: (i) *Flexibilität der Ressourcennutzung*, (ii) *Abhängigkeiten zwischen angebotenen Diensten* sowie (iii) *Überbuchen von Ressourcen*. Diese Voraussetzungen werden analog des vorangegangenen Abschnitts in den nachfolgenden Abschnitten erläutert und im Kontext von Beitrittsentscheidungen von Softwareagenten analysiert.

3.1.3.1 Flexibilität der Ressourcennutzung

Ressourcen in Revenue Management Domänen sind nicht flexibel bezüglich der Produktion von Produkten bzw. der Bereitstellung von Diensten. Wie oben aufgezeigt, können die zur Verfügung stehenden Ressourcen weder auf eine beobachtete Nachfrage angepasst, noch für spätere Zeitperioden eingelagert werden. Dieser mangelnden Flexibilität bei der Produktion von Produkten bzw. Bereitstellung von Diensten kann jedoch eine gewisse Flexibilität bei der Ressourcennutzung gegenüberstehen: Abhängig von der Domäne können Ressourcen zur Produktion von verschiedenen Produkten bzw. zur Bereitstellung verschiedener Dienste genutzt werden (Netessine und Shumsky, 2002, S. 35). Beispielsweise können Fluggesellschaften den gleichen Sitzplatz in einem Flugzeug zu verschiedenen Konditionen anbieten, die auf privat oder geschäftlich Reisende ausgerichtet sind. In diesem Beispiel wird die gleiche Ressource – der Sitzplatz – genutzt, um unterschiedliche Dienste anzubieten. In einer detaillierten Betrachtung verändert sich die zentrale Diensterbringung – die Beförderung von A nach B auf einem Sitzplatz mit gewissem Standard – nicht, sondern es werden nur die vertraglichen Nebenbedingungen, wie Zusatzleistungen oder Stornierungskosten, variiert. Wesentlich ist jedoch, dass die Kombination aus Hauptleistung des Vertrages und den vertraglichen Nebenbedingungen es der Fluggesellschaft ermöglicht, eigene Preiskategorien zu erstellen und somit als eigene Dienste zu klassifizieren.

Diese Flexibilität der Ressourcennutzung findet sich oftmals auch bei Softwareagenten. Abhängig von der Art der angebotenen Dienste eines Softwareagenten, können die zur Verfügung stehenden Ressourcen ebenfalls zur Bereitstellung unterschiedlicher Dienste genutzt werden. Verfügt ein Softwareagent ausschließlich über Ressourcen in Form von Prozessorleistung, Hauptspeicher sowie Festplattenspeicher, kann er diese Ressourcen zur Bereitstellung vielfältiger Dienste nutzen. Da hierbei nicht nur eine Flexibilität bezüglich der vertraglichen Nebenbedingungen, sondern sogar bezüglich der Hauptleistung gegeben ist, stehen einem Softwareagenten, der mehrere unterschiedliche Dienste anbietet

zunächst eine zusätzliche Dimension bei der Entscheidungsfindung zur Verfügung. Die Betrachtung von Kombinationen aus Hauptleistung des Vertrages sowie den vertraglichen Nebenbedingungen als eigenen Dienst relativiert dies hingegen, sodass die Komplexität der Entscheidungsfindung hierdurch nicht steigt. Dem Softwareagenten steht allerdings bei der Beitrittsverhandlung mit Multiagenten-Organisationen stets die Möglichkeit zur Verfügung, Dienste in verschiedenen Dienstgüteklassen anzubieten.

3.1.3.2 Abhängigkeiten zwischen angebotenen Diensten

In klassischen Revenue Management Disziplinen können durch verschiedene Faktoren zwei Arten von Abhängigkeiten zwischen den angebotenen Diensten entstehen: Erstens, können Ressourcen in bestimmten Situationen zur Bereitstellung verschiedener Dienste genutzt werden. Beispielsweise kann ein Sitz in einem Flugzeug genutzt werden, um verschiedene Kombinationen aus Dienst – die Beförderung auf diesem Sitz von A nach B – und vertraglicher Nebenbedingungen – Stornierungskosten, etc. – anzubieten (vgl. Abschnitt 3.1.3.1). Zweitens, können Abhängigkeiten zwischen mehreren angebotenen Diensten auftreten, die unterschiedliche Ressourcen nutzen. Bei einer Anfrage, die mehrere unterschiedliche Flugabschnitte A-B sowie B-C umfasst, entsteht eine Abhängigkeit zwischen dem angebotenen Dienst der Beförderung von A nach B sowie dem angebotenen Dienst der Beförderung von B nach C. Beide Flugabschnitte verwenden unterschiedliche Ressourcen – Sitzplätze in zwei Flugzeugen – müssen zur Beantwortung der Anfrage jedoch zusammen betrachtet werden. Diese Abhängigkeit der Ressourcen kann auch eine zeitliche Komponente mit sich führen: Eine Anfrage für eine einzelne Hotelübernachtung beeinflusst ebenfalls nachfolgende Anfragen, sowohl für dieses einzelne Datum, als auch sämtliche längeren Anfragen, die dieses Datum mit einschließen. In diesem Fall kann jedes Datum als einzelne Ressource aufgefasst werden, welche in Abhängigkeit zu den benachbarten Ressourcen steht.

Im Fall von Beitrittsverhandlungen von Softwareagenten mit Multiagenten-Organisationen treten potentiell beide Arten von Abhängigkeiten auf. Wie in Abschnitt 3.1.3.1 aufgeführt, können Softwareagenten ihre Ressourcen zur Bereitstellung unterschiedlicher Dienste bzw. Kombinationen aus Dienst sowie Dienstgüte nutzen. Die Verpflichtung zur Bereitstellung eines Dienstes im Rahmen eines Beitritts zu einer Multiagenten-Organisation beeinflusst somit auch die Möglichkeit des Softwareagenten zur Bereitstellung anderer Dienste, die auf die gleichen Ressourcen zugreifen. Gleichzeitig besteht ebenfalls eine Abhängigkeit zwischen verschiedenen Zeitabschnitten: Verpflichtet sich ein Softwareagent einer Multiagenten-Organisation Dienste unter Einsatz seiner verfügbaren Ressourcen bereitzustellen, beeinflusst dies seine zukünftigen Handlungsalternativen. Jeder Beitritt bzw. jede Verpflichtung zur Bereitstellung von Diensten nutzt Ressourcen ab dem Moment des Beitritts bzw. der Verpflichtung und schränkt somit die Möglichkeiten ein, anderen Multiagenten-Organisationen in darauf folgenden Zeitperioden beizutreten. Diese Abhängigkeiten zwischen verschiedenen Diensten ermöglicht es, erweiterte Ansätze des Revenue Management, wie beispielsweise *Network-Capacity-Control*, einzusetzen.

3.1.3.3 Überbuchen von Ressourcen

Werden mehr Verpflichtungen zur Lieferung von Produkten oder Bereitstellung von Diensten eingegangen als die zur Verfügung stehenden Ressourcen es erlauben würden, spricht man im Kontext von Revenue Management von *Überbuchung* (engl. *overbooking*). Dieses Phänomen tritt in zahlreichen Domänen auf, auch unabhängig vom Revenue Management: An Finanzbörsen können Händler durch so genannte Leerverkäufe Wertpapiere verkaufen, die sich nicht in ihrem Besitz befinden, und spekulieren darauf, dass der Kurs bis zum vereinbarten Liefertermin fällt. In Revenue Management-dominierten Domänen hingegen spekuliert der Anbieter darauf, dass nicht alle eingegangenen Verpflichtungen erfüllt werden müssen, da ein Teil der Kunden die gebuchte Leistung nicht abrufen. Insbesondere Hotels und Mietwagenfirmen gehen im Allgemeinen mehr Verpflichtungen in einer Buchungsklasse (Zimmer- oder Mietwagenkategorie) ein, als sie erfüllen könnten und weichen bei unerwartet hohem Abruf der Leistung auf andere Buchungsklassen (höhere oder niedrigere Kategorie) aus. Im Zweifel müssen Ressourcen von anderen Anbietern eingekauft oder eine entsprechend vereinbarte Vertragsstrafe gezahlt werden. Sowohl den Revenue Management-verbundenen als auch den übrigen Arten des Überbuchens ist gemein, dass der Liefertermin des Produkts bzw. der Bereitstellungstermin für den Dienst in der Zukunft liegt. Nur in dieser Situation kann es für den Anbieter vorteilhaft sein, höhere Verbindlichkeiten einzugehen, als er im Moment des Vertragsabschlusses erfüllen könnte.

Während bestimmte Ansätze versuchen, das zufällige Überbuchen (engl. *overcommitment*) bereits durch ein Ressourcenallokationsprotokoll zu verhindern (Karänke, 2014), wird in anderen Ansätzen im Kontext von informationstechnologischbasierten Diensten das Prinzip des Überbuchens bereits angewandt: Im Bereich von Cloud Computing Anbietern werden bei Infrastructure-as-a-Service (IaaS) insbesondere Prozessor- und Netzwerkressourcen zur Dienstleistung verwendet. Da nur ein verhältnismäßig kleiner Teil der Dienstnehmer die ihnen zur Verfügung gestellten Dienste in vollem Umfang nutzt, würde ohne eine Überbuchung der Ressourcen stets ein Großteil ungenutzt verbleiben. Das Überbuchen der vorhandenen Ressourcen kann in diesem Bereich die Auslastung der Ressourcen signifikant steigern und den Umsatz entsprechend erhöhen (Urgaonkar et al., 2009). Da Softwareagenten ähnlich einem Cloud-Rechenzentrum auf informationstechnische Ressourcen wie Prozessorleistung, Haupt- sowie Festplattenspeicher zurückgreifen und mit diesen Ressourcen Dienste anbieten, lässt sich das Prinzip des Überbuchens ebenfalls auf die Beitrittsverhandlungen von Softwareagenten zu Multiagenten-Organisationen übertragen. Bei Beitritt eines Softwareagenten zu einer Multiagenten-Organisation geht der Softwareagent eine Verpflichtung zur Dienstbereitstellung – auch in zukünftigen Zeitperioden – ein, diese Verpflichtung hat somit Auswirkungen auf die ihm in Zukunft zur Verfügung stehenden Ressourcen. Ein Dienst kann beispielsweise in der Lösung einer rechenintensiven Aufgabe bestehen. Allerdings wird diese Aufgabe dem Softwareagenten unter Umständen nur bei Bedarf gestellt, sodass auch die Ressourcen des Softwareagenten nur in diesen Zeitabschnitten zur Bereitstellung dieses Dienstes genutzt werden. In der verbleibenden Zeit werden durch die Mitgliedschaft in dieser Multiagenten-Organisation und der damit zusammenhängenden Bereitstellung des Dienstes die Ressourcen des Softwareagenten nicht weiter beansprucht.

In dieser Zeit könnte der Softwareagent seine Ressourcen zur Bereitstellung anderer Dienste nutzen, setzt sich dabei jedoch der Gefahr aus, dass durch eine weitere Dienstbereitstellung seine Fähigkeit eingeschränkt wird, den ersteren Verpflichtungen nachzukommen. Bei der Planung der zukünftig zur Verfügung stehenden Ressourcen eines Softwareagenten steht dieser zwei Dimensionen an Unsicherheiten gegenüber: Zum einen kann der bereitgestellte Dienst zu einem gewissen Zeitpunkt durch die Multiagenten-Organisation nicht eingefordert werden. Zum anderen könnte die Mitgliedschaft in dieser Multiagenten-Organisation bereits zu einem davor liegenden Zeitpunkt beendet worden sein. Beide Dimensionen ermöglichen es einem Softwareagenten, mehr Verpflichtungen zur Dienstbereitstellung einzugehen, als seine Ressourcen bei einer gleichzeitigen Erfüllung zulassen würden. Dabei liegt der Fokus auf der Gleichzeitigkeit: Zeitliche Verschiebungen zwischen verschiedenen Dienstabrufen ermöglichen eine wahrscheinlichkeitsbasierte Kalkulation des erwarteten Ressourcenbedarfs. Bei der Entscheidung, wieviele Ressourcen zur Dienstbereitstellung eingeplant werden können, spielt insbesondere die vereinbarte Dienstgüte – die vertraglichen Nebenbedingungen – eine entscheidende Rolle.

3.1.4 Zusammenfassung zur Anwendbarkeit von Revenue Management

In den vorangegangenen Abschnitten wurden sowohl die grundlegenden Voraussetzungen zur Anwendung Revenue Management-basierter Ansätze auf Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen (Abschnitt 3.1.2) als auch die entsprechenden Voraussetzungen zur Anwendung erweiterter Verfahren des Revenue Management (Abschnitt 3.1.3) untersucht. Tabelle 3.1 fasst die Ergebnisse von Abschnitt 3.1.2 zusammen und verdeutlicht, dass die Voraussetzung zur Anwendung Revenue Management-basierter Ansätze auf Beitrittsentscheidungen von Softwareagenten erfüllt sind. Ebenso konnte Abschnitt 3.1.3 aufzeigen, dass nicht nur die Grundvoraussetzungen erfüllt sind, sondern im Kontext von Softwareagenten auch die Anwendung erweiterter Verfahren des Revenue Management möglich sind. Tabelle 3.2 gibt einen Überblick über die Voraussetzungen zur Umsetzung dieser erweiterten Verfahren.

Die vorangegangene Analyse der Voraussetzungen hat ebenfalls aufgezeigt, dass wesentliche Unterschiede zwischen Domänen, in denen Revenue Management bereits Anwendung findet, und Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen bestehen. Während bei ersteren im Allgemeinen Anfragen für Dienste betrachtet werden, deren Anfang und Ende und somit auch der potentielle Ressourcenverbrauch bei Vertragsabschluss feststehen, binden sich Softwareagenten beim Beitritt zu einer Multiagenten-Organisation auf unbestimmte Zeit. Die Mitgliedschaft in einer Multiagenten-Organisation beeinflusst somit potentiell beliebig viele Zeitabschnitte nach dem Vertragsabschluss. Dieser Sachverhalt erfordert die Anpassung der im Revenue Management verwendeten Modelle und Verfahren auf Verträge mit unbestimmter Laufzeit.

Tabelle 3.1: Erfüllung der Voraussetzungen zur Anwendung Revenue Management-basierter Ansätze

Voraussetzungen	Entsprechung bei Beitrittsentscheidungen von Softwareagenten
Heterogenität der Kunden	Die Nachfrage nach Diensten wird durch Multiagenten-Organisationen generiert, die aus Sicht des einzelnen Softwareagenten heterogene Präferenzen aufweisen.
Nachfrageschwankungen und Unsicherheiten	Die Nachfrage nach angebotenen Diensten ist Softwareagenten a priori nicht bekannt. Ebenso ist Mitgliedern einer Multiagenten-Organisation a priori nicht bekannt, in welchem Umfang die zu Verfügung gestellten Dienste abgerufen werden.
Inflexibilität der Produktion	Die zur Verfügung stehenden Ressourcen von Softwareagenten werden als gegeben betrachtet und können vom Softwareagenten kurzfristig nicht erhöht werden.
Preis ist kein Qualitätsmerkmal	Die Kompensation, die ein Softwareagent für die Mitgliedschaft in einer Multiagenten-Organisation erhält, ist kein Qualitätsmerkmal. Qualitätsmerkmale eines Dienstes werden im Rahmen von vertraglichen Nebenbedingungen in Dienstgütekriterien festgehalten.
Informationstechnische Infrastruktur	Softwareagenten nutzen einen festen Teil der ihnen zur Verfügung stehenden Ressourcen zur Speicherung von Anfragen und zur Anwendung des in dieser Arbeit entwickelten Verfahrens.

Tabelle 3.2: Erfüllung der Voraussetzungen zur Anwendung erweiterter Verfahren des Revenue Management

Voraussetzungen	Entsprechung bei Beitrittsentscheidungen von Softwareagenten
Flexibilität der Ressourcennutzung	Softwareagenten können ihre internen Ressourcen zur Bereitstellung verschiedener Dienste in unterschiedlichen Dienstgüteklassen nutzen.
Abhängigkeiten zwischen angebotenen Diensten	Verpflichtungen, die ein Softwareagent im Rahmen einer Mitgliedschaft in einer Multiagenten-Organisation eingeht, bilden sowohl zeitliche Abhängigkeiten zwischen verschiedenen Zeitperioden, als auch Abhängigkeiten zwischen unterschiedlichen Diensten durch den Zugriff auf die gleichen Ressourcen.
Überbuchen von Ressourcen	Da in einer Multiagenten-Organisation im Allgemeinen nicht zu jedem Zeitpunkt jeder Dienst benötigt wird, ist es Softwareagenten möglich mehr Verpflichtungen einzugehen als gleichzeitig erfüllt werden könnten.

3.2 Revenue Management-basiertes Modell der Beitrittsentscheidung

Dieser Abschnitt modelliert die Beitrittsentscheidung von Softwareagenten zu Multiagenten-Organisationen. Dazu wird das zu Grunde liegende Problem zunächst klassifiziert (Abschnitt 3.2.1) und anschließend der Zeithorizont (Abschnitt 3.2.2), Dienste und Ressourcen (Abschnitt 3.2.3), Anfragen von Multiagenten-Organisationen (Abschnitt 3.2.4), Dienstbereitstellungsangebote (Abschnitt 3.2.5) sowie Mitgliedschaftsverträge (Abschnitt 3.2.6) modelliert.

3.2.1 Problemklassifizierung

Softwareagenten stehen bei einem potentiellen Beitritt zu einer Multiagenten-Organisation vor der Entscheidung, ob und in welcher Form, d.h. unter welchen vertraglichen Nebenbedingungen, sie der Multiagenten-Organisation beitreten würden. Abschnitt 3.1 hat aufgezeigt, dass die Voraussetzungen zur Anwendung von Ansätzen des Revenue Management für die Beitrittsentscheidung von Softwareagenten zu Multiagenten-Organisationen gegeben sind. Zur Klassifizierung von Problemstellungen des Revenue Management haben Weatherford und Bodily (1992) eine Taxonomie entwickelt, die auf Basis verschiedener Elemente und deren Ausprägungsformen eine Einordnung und Gegenüberstellung von Problemen ermöglicht, die die oben genannten Voraussetzungen erfüllen. Aufgrund des gegenüber anderen Revenue Management Domänen anderen Zeithorizonts, sind Teile dieser Taxonomie auf das Problem der Beitrittsentscheidungen nicht anwendbar. Beispielsweise verändert sich die Zahlungsbereitschaft bezüglich der Kompensation einer Multiagenten-Organisation nicht zwingend über die Zeit, die eines Kunden von Fluggesellschaften oder Hotels im Allgemeinen hingegen schon, da für kurzfristige Buchungen meist eine höhere Zahlungsbereitschaft besteht. Die Taxonomie muss deshalb für die vorliegende Problemstellung angepasst werden. Hieraus ergeben sich die folgenden Elemente:

1. **Ressourcen.** Die zur Verfügung stehenden Ressourcen können entweder in diskreten Schritten oder in kontinuierlicher Weise genutzt werden. Ein Hotel hat beispielsweise nur eine ganzzahlige Anzahl an Zimmern zur Verfügung und kann nur ganze Zimmer vergeben. Bei den beschränkten Ressourcen von Softwareagenten, die zur Bereitstellung von Diensten genutzt werden, handelt es sich insbesondere um informationstechnische Ressourcen, wie Prozessor, Haupt- und Festplattenspeicher. Zwar kann beispielsweise Speicher auf der kleinsten Granularitätsebene ebenfalls nur in ganze Bits unterteilt werden, da sich sowohl für Festplatten- als auch Hauptspeicher mittlerweile die Angabe der Kapazität mindestens in Millionen Byte durchgesetzt hat, kann hier sowohl bei Kapazität als auch bei der tatsächlichen Nutzung von einem kontinuierlichem Ressourcentyp ausgegangen werden. Gleiches gilt für die zunächst in ganze Prozessoroperationen unterteilbare Rechenkapazität bzw. Prozessornutzung.
2. **Kapazität.** Die zur Verfügung stehenden Ressourcen sind bei den meisten Problemen des Revenue Management limitiert und begründen dadurch bereits einen Teil der nach Abschnitt 3.1.2.3 vorausgesetzten Inflexibilität der Produktion. Es gibt hingegen

Domänen, die in gewissem Maße über unbegrenzte Ressourcen verfügen und eine Anwendbarkeit von Revenue Management Ansätzen zulassen: Bei der Bereitstellung von Software über digitale Kanäle ist beispielsweise der Ressourcenverbrauch durch jede einzelne Transaktion so klein, dass dieser vernachlässigt werden kann. Bei der Dienstbereitstellung durch Softwareagenten an Multiagenten-Organisationen ist dies nicht der Fall. Softwareagenten verfügen über begrenzte Ressourcen, deren Kapazität im Allgemeinen nicht ohne Einschränkungen erhöht oder vermindert werden kann.

3. **Preis/Kompensation.** In Revenue Management Domänen wird meist angenommen, dass der Preis für die angebotenen Produkte oder Dienste bei der Entscheidung über eine Angebotserstellung bereits feststehen. Diese Annahme ist nicht inherent mit dem jeweiligen Entscheidungsprobleme in der Domäne verknüpft, sondern ist auf eine Trennung dieser beiden Fragestellungen zurückzuführen: Zunächst wird der Preis für ein Produkt bzw. einen Dienst festgesetzt und anschließend erst über dessen Verfügbarkeit entschieden. Grundsätzlich kann dieses Verfahren auch in umgekehrter Reihenfolge durchgeführt oder sowohl Preis und Verfügbarkeit gleichzeitig bestimmt werden (Weatherford und Bodily, 1992, S. 835). Im Falle eines Beitritts von Softwareagenten zu Multiagenten-Organisationen kommen alle drei der vorangenannten Ansätze in Betracht. Diese Arbeit beschäftigt sich mit der Beitrittsentscheidung selbst und geht somit davon aus, dass ein Softwareagent die Kompensationsleistung, zu der er bereit ist, einen Dienst einer Multiagenten-Organisation dauerhaft bereitzustellen, bereits festgelegt hat, und folgt damit den verbreiteten Ansätzen des Revenue Management. Die Berechnung dieser Kompensation ist nicht Gegenstand dieser Arbeit und bedarf eines vorgelagerten Schrittes.
4. **Vertragslaufzeit.** Die Vertragslaufzeit ist kein Kriterium in der Taxonomie von Weatherford und Bodily (1992), jedoch kann sie sich in verschiedenen Anwendungsbereichen wesentlich unterscheiden und dabei Einfluss auf die Problemstellung selbst nehmen. Die Taxonomie muss daher um dieses Element erweitert werden. Dabei kann das Element *Vertragslaufzeit* zwei Ausprägungen annehmen: (i) Die Vertragslaufzeit kann bei Vertragsabschluss *bekannt* sein, wie es in den traditionellen Revenue Management Domänen der Fall ist. Bei Vertragsabschluss ist bekannt, wann der bereitgestellte Dienst – ein Einzelflug, eine Kombination von Flugabschnitten, mehrere Übernachtungen in einem Hotel – beginnt und wann die Bereitstellung wieder endet. (ii) Bei einer unbestimmten Vertragslaufzeit kann hingegen die tatsächliche Vertragslaufzeit bei Vertragsabschluss *unbekannt* sein. Dies ist beispielsweise bei der Bereitstellung von IT-Diensten im Rahmen von Cloud Computing der Fall, kommt allerdings auch bei der Mitgliedschaft eines Softwareagenten in einer Multiagenten Organisation zum tragen. Ein beitretender Softwareagent hat keine Sicherheit über die Dauer der Mitgliedschaft, sondern wird bei einem Beitritt zunächst auf unbestimmte Zeit Mitglied der Multiagenten-Organisation.
5. **Vergünstigte Buchungsklassen.** Abhängig von der Heterogenität der Dienstanfrager (vgl. Abschnitt 3.1.2.1), kann die Anzahl der angebotenen vergünstigten Buchungsklassen variieren. Dabei werden nur die Anzahl an Buchungsklassen betrach-

tet, die gegenüber einer Referenzbuchungsklasse (engl. *full fare* oder *full price*) einen vergünstigten Tarif anbieten. Hierbei ist insbesondere die Unterscheidung zwischen einer oder mehr als einer vergünstigten Buchungsklasse von Bedeutung. Der Taxonomie von Weatherford und Bodily (1992) fehlt die Möglichkeit, keine vergünstigten Buchungsklassen zu betrachten, jedoch kann diese Ausprägungen beispielsweise zur Anwendung von Überbuchungsverfahren ohne Produkt- oder Dienstdifferenzierung durchaus in Betracht kommen. Softwareagenten, die in Beitrittsverhandlungen mit einer Multiagenten-Organisation eintreten, haben keine Beschränkungen bezüglich der zu vereinbarenden vertraglichen Nebenbedingungen, beispielsweise in Form von Dienstgütevereinbarungen. Abhängig von der individuellen Zielfunktion des Softwareagenten können hier beliebige Buchungs- bzw. Dienstgüteklassen definiert werden. Diese Arbeit setzt voraus, dass ein Softwareagent bei der Beitrittsentscheidung, das heißt bei Angebotserstellung, bereits eine Unterteilung der angebotenen Dienste in verschiedene Dienstgüteklassen vorgenommen hat. Eine problem inherente Beschränkung der Anzahl dieser Dienstgüteklassen ist hingegen nicht gegeben.

6. **Nachfrage.** Die Nachfrage, der sich ein Produkt- oder Dienstanbieter gegenüber sieht, kann deterministisch oder stochastisch sein. Dabei kann es auch dazu kommen, dass sich die Einordnung der Nachfrage bei einzelnen Buchungsklassen unterscheidet, so dass eine Buchungsklasse durch den Anbieter als deterministisch und eine andere als stochastisch wahrgenommen wird. Dabei handelt es sich stets um die Wahrnehmung durch den Anbieter. Wenn die Nachfrage nach einem Produkt oder einem Dienst einer eindeutigen Gesetzmäßigkeit folgt, diese Gesetzmäßigkeit dem Anbieter jedoch nicht bekannt ist, wird die Nachfrage durch den Anbieter subjektiv als stochastisch eingeschätzt. Softwareagenten, die Dienste anbieten, sehen sich im Allgemeinen ebenfalls einer stochastischen Nachfrage gegenüber: Wann und in welchem Umfang ein Dienst nachgefragt wird, ist dem Softwareagenten nicht bekannt, er kann mittels geeigneter Vorhersageverfahren nur versuchen, eine entsprechende Nachfrage zu prognostizieren. Dies gilt für alle Dienstgüteklassen, für deren Bereitstellung sich ein Softwareagent entscheidet.
7. **Dienstabbruf.** Bei einer vertraglichen Verpflichtung zur Erbringung eines Dienstes zu einem Zeitpunkt, der bei Vertragsabschluss in der Zukunft liegt, können sich die Präferenzen des Dienstabnehmers bis zum vertraglichen Zeitpunkt ändern, so dass dieser eine Erbringung des Dienstes unter Umständen nicht mehr wünscht. Das bedeutet, dass der Dienstbringer seinen angebotenen Dienst nicht mehr erbringen muss und seine hierfür vorgesehenen Ressourcen ebenfalls nicht hierfür benötigt. Bei Vertragsabschluss muss dabei festgelegt werden, wie die Verpflichtung des Dienstabnehmers – die Zahlung des Preises bzw. einer Kompensation – in diesem Falle zu behandeln ist. In den klassischen Revenue Management Domänen unterscheiden sich die Zahlungsverpflichtungen im Allgemeinen in Abhängigkeit von der gewählten Buchungsklasse und werden als Stornierungskosten bezeichnet. So werden in der höchstpreisigen Buchungsklasse meist keine Stornierungskosten fällig, während in der niedrigstpreisigen die Stornierungskosten dem Preis entsprechen. Softwareagenten,

die einer Multiagenten-Organisation beigetreten sind, sehen sich hingegen nicht nur der Frage gegenüber, *ob* der zur Verfügung gestellte Dienst nachgefragt wird, sondern auch *wann* dies geschieht. Während der Flug einer Fluggesellschaft oder ein gebuchtes Hotelzimmer fest terminiert sind, ist der Abruf der zur Verfügung gestellten Dienste eines Softwareagenten für diesen im Allgemeinen nicht deterministisch. Es ist zwar möglich, dass der bereitgestellte Dienst ununterbrochen oder auch zu festen Zeiten von der Multiagenten-Organisation abgerufen wird, in einigen Konstellationen wird der Dienst jedoch zu nicht vorbestimmten Zeiten ad-hoc abgerufen.

8. **Gruppenbuchungen.** In klassischen Revenue Management Domänen ist es möglich, dass mehrere Individuen gemeinsam das gleiche Produkt bzw. den gleichen Dienst nachfragen und diese nur ein Angebot akzeptieren, dass alle Individuen berücksichtigt. Diese Gruppenanfragen bieten dem Anbieter die Möglichkeit, eine größere Anzahl an Produkten bzw. Diensten abzusetzen, die andererseits eine entsprechend große Menge seiner Ressourcen belegt. Hier ist die Frage, in welcher Buchungsklasse die Anfrage akzeptiert wird, insbesondere falls bei einer entsprechenden Anzahl sequenziell eintreffender Anfragen unterschiedlich bepreiste Buchungsklassen angeboten würden. Diese Arbeit untersucht das Entscheidungsproblem beim Beitritt einzelner Softwareagenten, so dass weder Gruppen von Softwareagenten als potentielle Mitglieder noch Gruppen von Multiagenten-Organisationen, die gemeinsam Dienste nachfragen, betrachtet werden. Fragt jedoch eine Multiagenten-Organisation gleichzeitig mehrere Dienste eines Softwareagenten an, weist diese Anfrage Analogien zu Gruppenanfragen von mehreren Individuen auf. Wie in Gruppenanfragen in traditionellen Revenue Management Domänen ist der Beitritt von der Verfügbarkeit mehrerer Dienste abhängig. Ebenso wird ein Beitritt, der die Bereitstellung mehrerer Dienste umfasst, einen entsprechend großen Teil der Ressourcen des Softwareagent in Anspruch nehmen.
9. **Buchungsklassenwechsel.** Unter dem Begriff Buchungsklassenwechsel (engl. *diversion*) wird im Revenue Management die Bereitschaft von Kunden verstanden, eine niederpreisige Buchungsklasse zu wählen, obwohl ihre Zahlungsbereitschaft die Wahl einer höherpreisigen Buchungsklasse erlauben würde. Beispielsweise wären Geschäftsreisende unter Umständen dazu bereit, eine andere als die höchstpreisige Buchungsklasse bei einer Flugbuchung zu wählen, wenn diese zur Verfügung stehen würde (Weatherford und Bodily, 1992, S. 836). Oft ergibt sich allerdings diese Wahlmöglichkeit nicht, da beispielsweise für kurzfristige Buchungen nur noch die höchstpreisige Buchungsklasse angeboten wird. Neben den Stornierungsbedingungen diskriminiert die Fluggesellschaft zusätzlich nach dem Kriterium Zeit. Ist jedoch ein Geschäftsreisender, dessen Zahlungsbereitschaft auch höherpreisige Buchungsklassen zulassen würde, bereit, sämtliche Einschränkungen günstigerer Buchungsklassen in Kauf zu nehmen – eine frühzeitige Buchung sowie eingeschränkte Stornierungsmöglichkeiten – kann die Fluggesellschaft den Geschäftsreisenden nicht mehr von einem privat Reisenden unterscheiden und die Diskriminierungskriterien verfehlen ihr Ziel. Die Auswahl dieser Kriterien ist somit entscheidend für den Erfolg von Revenue Management Ansätzen. Dies gilt gleichfalls für die Beitrittsentscheidung

von Softwareagenten zu Multiagenten-Organisationen. Ein Softwareagent, der einer Multiagenten-Organisation Dienste anbietet, kann diese Dienste mit unterschiedlichen vertraglichen Nebenbedingungen anbieten. Dabei trifft er im Allgemeinen die Annahme, dass bestimmte vertragliche Nebenbedingungen für die dienstabnehmende Multiagenten-Organisation einen höheren Wert darstellen als andere. Dies können Bestimmungen über die Mitgliedschaftsbeendigung wie auch Dienstgütekriterien sein. Wäre eine Multiagenten-Organisation bereit, für die Bereitstellung eines Dienstes im Rahmen der Mitgliedschaft eines Softwareagenten eine gewisse Kompensation zu zahlen, entspricht dies der Zahlungsbereitschaft der Multiagenten-Organisation. Bietet der Softwareagent seine Dienste sowohl mit 95% als auch mit 99% Verfügbarkeit an, nimmt er implizit an, dass er für die höhere Verfügbarkeit eine entsprechend höhere Kompensation erhalten kann. Ist dieses Kriterium jedoch für die Multiagenten-Organisation nicht relevant, ist es als Diskriminierungskriterium falsch gewählt und die Multiagenten-Organisation wird die vertraglichen Nebenbedingungen mit einer niedrigeren Kompensation wählen, obwohl sie auch zur Zahlung höherer Kompensationen bereit wäre. Die Wahl der „richtigen“ Diskriminierungskriterien ist somit für den Erfolg von Revenue Management Ansätze essentiell, muss jedoch auf den konkreten Anwendungsfall – den angebotenen Dienst des Softwareagenten – angepasst werden.

10. **Netzwerkeffekte.** Der Begriff *Netzwerkeffekte* fasst die in Abschnitt 3.1.3.2 als Voraussetzungen für die Anwendung erweiterter Verfahren des Revenue Management zusammen. Während in einfachen Szenarien angenommen wird, dass nur ein einzelnes angebotenes Produkt bzw. ein einzeln angebotener Dienst betrachtet wird, stellen manche Problemstellungen die Betrachtung von Abhängigkeiten zwischen verschiedenen Produkten bzw. Diensten voraus. Die Beschränkung auf ein einzelnes Produkt oder einen einzelnen Dienst ist dabei meist nicht problemimmanent, sondern resultiert in einer Vereinfachung des tatsächlichen Problems. Hotels können beispielsweise nur jedes einzelne Zimmer je Nacht betrachten, vernachlässigen damit allerdings die Abhängigkeiten, die zum einen durch Buchungsanfragen entstehen, die länger als eine Nacht andauern oder die mehr als ein Zimmer benötigen. Gleiches gilt für die Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen: Es wäre zwar möglich, einzelne Ressourcen und Zeitabschnitte gesondert zu betrachten, dies würde jedoch die Tatsache vernachlässigen, dass sowohl zwischen verschiedenen Diensten als auch zwischen verschiedenen Zeitabschnitten Abhängigkeiten bestehen. Die Mitgliedschaft in einer Multiagenten-Organisation und somit die durch die Dienstbereitstellung verbundene Auslastung der zur Verfügung stehenden Ressourcen beeinflusst die Möglichkeit des Softwareagenten, Mitgliedschaften in anderen Multiagenten-Organisationen einzugehen. Bei den Beitrittsverhandlungen sind somit stets zukünftige mögliche Mitgliedschaften mit einzubeziehen.
11. **Vertragswidriges Verhalten.** Insbesondere durch Nutzung von den in Abschnitt 3.1.3.3 beschriebenen Möglichkeiten der Überbuchung von Ressourcen, kann es dazu kommen, dass ein Softwareagent seinen Verpflichtungen gegenüber einer Multiagenten-

Organisation zur Bereitstellung von Diensten nicht vertragskonform nachkommen kann. Diese Form des vertragswidrigen Verhaltens wird im englischen als „*Overcommitment*“ bezeichnet. Während andere Ansätze versuchen, Overcommitment durch eine entsprechende Gestaltung des Allokationsmechanismus vollständig zu verhindern (Karänke, 2014), wird in Revenue Management-basierten Ansätzen dieses Risiko zur Auslastungssteigerung in Kauf genommen. Abhängig von der jeweiligen Problemstellung können jedoch unterschiedliche Strafen bei Missachtung der vertraglichen Verpflichtungen die Folge sein. Fluggesellschaften oder Hotels, die einen Flug oder ein Zimmer nicht vertragsgemäß anbieten, müssen dem Kunden entweder einen Ersatz anbieten – ggf. auch bei einer anderen Fluggesellschaft bzw. in einem anderen Hotel – oder den Preis zuzüglich etwaiger Schadensersatzforderungen zurückerstatten. Softwareagenten, die einer Verpflichtung im Zuge ihrer Mitgliedschaft nicht nachkommen, sehen sich meist einer Verletzung der zugesicherten Dienstgüteeigenschaften gegenüber. Diese umfassen beispielsweise Kriterien wie Reaktionszeit oder Zuverlässigkeit. Die Beschaffung zusätzlicher Ressourcen kommt im Allgemeinen nicht in Betracht, da der Mangel an Ressourcen meist erst im Moment der Bereitstellungsverpflichtung entsteht und dadurch nur ein geringer zeitlicher Spielraum verbleibt. Da ein Ersatz durch den Einkauf von anderen Diensten somit nicht in Frage kommt, verbleibt die Möglichkeit der Rückerstattung der erhaltenen Kompensationszahlungen. Der hierfür zu betrachtende Zeitraum hingegen muss in den vertraglichen Nebenbedingungen verankert werden, was einer entsprechenden Strafzahlung gleichkommt. Ebenfalls in Frage kommt der Ausschluss aus der Multiagenten-Organisation, u.U. verknüpft mit einer entsprechenden Strafzahlung bzw. Einbehaltung der Kompensation. Dies ist abhängig von der Domäne der Anwendung sowie den ausgehandelten vertraglichen Nebenbedingungen.

12. **Entscheidungsregel.** Um eine Anfrage eines Kunden beantworten zu können, sind Entscheidungsregeln nötig. Dies gilt sowohl für traditionelle Revenue Management Domänen als auch für Beitrittsentscheidungen von Softwareagenten. Insbesondere in der Luftfahrtindustrie haben sich hierbei verschiedene Typen von Entscheidungsregeln herausgebildet, die in ihrer Komplexität einen Teil des Gesamtproblems adressieren. Da in diesem speziellen Fall jeder Flugabschnitt als separater Dienst betrachtet werden kann, kommen für die Freigabe einzelner Buchungsklassen einfache Kriterien, wie verbleibende Zeitdauer bis zum Abflug oder verfügbare Restkapazität, in Frage. Da nur ein Flug, der zu einem bestimmten Zeitpunkt startet betrachtet wird, kann einmalig eine entsprechende Entscheidungsregel aufgestellt werden. Betrachtet man hingegen die tatsächliche Nachfrage nach verschiedenen Flugabschnitten sowie deren Abhängigkeiten untereinander, so steigt die Komplexität des Problems. Eine ähnliche Komplexität kann bei Beitrittsentscheidungen von Softwareagenten beobachtet werden, da sich die Nachfrage von Multiagenten-Organisationen nach den Diensten des Softwareagenten mit der Zeit verändern kann und dieser sein Verhalten sowie die Bewertung der verbleibenden Ressourcen dynamisch anpassen muss. Die Entscheidungsregel selbst ist nicht originär ein Merkmal des Problems, sondern bildet eine geeignete Lösung für dieses. Durch die Veränderlichkeit von Einflussfaktoren,

die von außen auf Softwareagenten einwirken – z.B. weitere Beitrittsanfragen und Dienstbereitstellung – bedürfen die Entscheidungsregeln für Beitrittsentscheidungen von Softwareagenten einer über den Zeitverlauf dynamischen Anpassung.

Tabelle 3.3 stellt die Einordnung des Problems der Beitrittsentscheidung von Softwareagenten in Multiagenten-Organisationen den Revenue Management Domänen Luftfahrtindustrie und Hotelgewerbe gegenüber.

Tabelle 3.3: Gegenüberstellung von Revenue Management Domänen mittels Taxonomie, angelehnt an Weatherford und Bodily (1992, S. 835)

Element	Luftfahrt	Hotel	Multiagenten-Organisation
Ressource	Diskret	Diskret	Kontinuierlich
Kapazität	Unveränderlich	Unveränderlich	Unveränderlich
Preis/Kompensation	Festgesetzt	Festgesetzt	Festgesetzt
Vertragslaufzeit	Bekannt	Bekannt	Unbekannt
Vergünstigte	k	k	k
Buchungsklassen			
Nachfrage	Stochastisch	Stochastisch	Stochastisch
Dienstabruf	Unsicher	Unsicher	Unsicher
Gruppenbuchungen	Ja	Ja	Ja
Buchungsklassenwechsel	Möglich	Möglich	Möglich
Netzwerkeffekte	Ja	Ja	Ja
Vertragswidriges Verhalten	Umbuchung	Umbuchung	Vertragsstrafe
Entscheidungsregel	Dynamisch	Dynamisch	Dynamisch

Im traditionellen Revenue Management wird davon ausgegangen, dass Angebots- sowie Kunden-Entscheidungen einmalig getroffen werden und sich ausschließlich in einem entsprechend abgegrenzten zeitlichen Rahmen auswirken. Diese Annahme vernachlässigt unter anderem die starke Kundenbindung im Dienstleistungssektor durch die erst a-posteriori mögliche Qualitätsbeurteilung von Dienstleistungen (Bruhn, 2014, S. 22 ff.). V. Martens (2008) berücksichtigt den individuellen Kundenwert bei der Kapazitätssteuerung in einem konzeptionellen Modell und untersucht dabei sowohl die strategische, taktische und operative Ebene. Die Berücksichtigung des Kundenwerts bildet dabei einen erster Schritt in die Richtung langfristiger – und somit auch organisatorischer – Bindungen zwischen Anbietern und Abnehmern von Diensten. Trotz dieser Erweiterung bleiben jedoch grundsätzliche Unterschiede: Während trotz der Berücksichtigung des Kundenwerts der Dienstanbieter bei jeder Anfrage neu entscheiden kann, verpflichten sich Softwareagenten beim Beitritt in eine Multiagenten-Organisation zur – im Allgemeinen bedarfsorientierten – Dienstbereitstellung während der gesamten Mitgliedschaft. Die Beitrittsentscheidungen von Softwareagenten entfalten somit im Gegensatz zu einmaligen (Kauf-)Entscheidungen für beide beteiligten Parteien in einem a priori nicht bestimmten Zeitraum Wirkung. Der betrachtete Zeithorizont ist somit ein wesentliches Modellierungselement bei der Beitrittsentscheidung (vgl. Abschnitt 3.2.2).

3.2.2 Zeithorizont

In traditionellen Revenue Management Domänen wird auf der Zeitachse im Allgemeinen die Differenz zwischen Anfrage- bzw. Vertragsabschlusszeitpunkt und dem tatsächlichen Dienstbringungszeitpunkt besonders betrachtet (vgl. Abschnitte 3.1.1 und 3.2.1). Zusätzlich kann die Dauer der Dienstbringung mit in die Angebotsentscheidungen einbezogen werden: Dies ist insbesondere bei Flügen über mehrere Abschnitte, Hotelbuchungen über mehrere Nächte oder Vermietungen von Kraftfahrzeugen über mehrere Tage relevant. Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen zeichnen sich hingegen dadurch aus, dass diese im Allgemeinen nicht gezielt für einen in der Zukunft liegenden zeitlichen Abschnitt mit a priori bekannter Dauer abgeschlossen werden. Im Gegenteil ist die Mitgliedschaft im Sinne der Definition 7 auf Langfristigkeit ausgelegt. Softwareagenten sind in der Lage, auch kurzfristige Dienste mit einem festen oder absehbaren Zeitfenster anzubieten, beispielsweise die Übersetzung eines Buches. Die Ressourcen zur Erledigung dieser einmaligen Aufgabe können exakt geplant und den anderweitigen Verpflichtungen angepasst werden. Durch das Fehlen einer langfristigen Bindung handelt es sich hierbei allerdings nicht um eine Mitgliedschaft in einer Multiagenten-Organisation. Die Mitgliedschaft könnte hingegen durch die dauerhafte Besetzung einer Stelle (vgl. Abschnitt 2.1.2.3) als „Übersetzer“ in ebenjener Multiagenten-Organisation begründet werden.

Zahlreiche Ansätze des Revenue Management betrachten zunächst ein einzelnes Produkt bzw. einen einzelnen Dienst, beispielsweise einen Flugabschnitt zwischen zwei Lokationen. Es existieren bereits Ansätze, die berücksichtigen, dass auch während die Dienstleistung erbracht wird, Anfragen für andere Zeitabschnitte eintreffen können, z.B. für Hotels (Pak, 2005, S. 55 ff.) sowie für Cloud Computing (Anandasivam und Premm, 2009). Diese Ansätze betrachten jedoch keine Anfragen zur Dienstleistungserbringung, deren Dauer a priori unbekannt ist. Eine Anwendung von Revenue Management auf Beitrittsentscheidungen von Softwareagenten erfordert somit die Anpassung dieser grundlegenden Komponente: Die Mitgliedschaft eines Softwareagenten in einer Multiagenten-Organisation wird im Allgemeinen auf unbestimmte Zeit geschlossen (vgl. Abschnitt 2.3.2.2) und ermöglicht somit stabile organisatorische Strukturen.

Der betrachtete Zeithorizont für einen Revenue Management-basierten Ansatz umfasst somit sämtliche in der Zukunft liegende Zeitabschnitte sowie die in der Vergangenheit liegenden Zeitabschnitte mit Wirkung bis zur Gegenwart bzw. bis in die Zukunft. Entscheidungsregeln für Beitrittsentscheidungen zu Multiagenten-Organisationen bedürfen folglich einer umfassenden Berücksichtigung bestehender Verpflichtungen in anderen Multiagenten-Organisationen und deren Auswirkungen bzw. Interdependenzen zwischen diesen.²⁰

Die Zeit wird in diskreten Zeitschritten gemessen und definiert als $t \in \mathbb{N}$. Für den Verfahrensentwurf ist die tatsächliche Länge einer Zeiteinheit nicht von Bedeutung, sondern wird für den jeweiligen Anwendungsfall individuell festgelegt. Sind für bestimmte Anwendungsfälle sehr zeitnahe Reaktionen bzw. Entscheidungen nötig, kann die Länge einer Zeiteinheit beliebig klein definiert werden.

²⁰ zu organisatorischen Interdependenzen von Beitrittsentscheidungen vgl. Abschnitt 2.4.1

3.2.3 Dienste und Ressourcen

Softwareagenten in einer Multiagenten-Organisation stellen dieser Dienste zur Verfügung (vgl. Abschnitt 2.4). Welche Dienste der Softwareagent der Multiagenten-Organisation bereitzustellen hat, wird bereits bei den Beitrittsverhandlungen festgelegt. Dem Softwareagenten stehen dabei verschiedene Dienste der Menge $S = \{s_1, \dots, s_{|S|}\}$ zur Verfügung, die im Rahmen dieser Arbeit als unveränderlich angenommen wird. Diese Unveränderlichkeit stellt jedoch keine Beschränkung der Allgemeinheit dar: Sollte der Softwareagent – beispielsweise durch Lernverfahren – neu entwickelte Dienste anbieten können, kann die Menge S erweitert und das folgende, hierauf aufbauende Verfahren neu initiiert werden.

Zur Bereitstellung der Dienste stehen dem Softwareagenten Ressourcen in begrenztem Maße zur Verfügung, z.B. Rechenkapazität, Speicherplatz, Hauptspeicher. Die dem Softwareagenten zur Verfügung stehenden Gesamtkapazitäten werden in der Menge $R = \{r_1, \dots, r_{|R|}\}$ abgebildet. Zu einem beliebigen Zeitpunkt t stehen dem Softwareagenten im Allgemeinen jedoch nur ein Teil dieser Gesamtkapazität zur Verfügung. Die Kapazität der Ressourcen im Zeitpunkt t wird durch den Vektor $\mathbf{x}_t$ dargestellt:

$$\mathbf{x}_t = \begin{pmatrix} x_1^t \\ \vdots \\ x_{|R|}^t \end{pmatrix}.$$

Die angebotenen Dienste können durch den Softwareagenten in verschiedenen Dienstgüteklassen zur Verfügung gestellt werden. Diese Dienstgüteklassen sind somit vertragliche Nebenbedingungen der beidseitigen Beitrittsentscheidung (vgl. Abschnitt 2.4.3). Dem Softwareagenten stehen hierfür $|B|$ Dienstgüteklassen der Menge $B = \{b_1, \dots, b_{|B|}\}$ zur Verfügung. Die Dienstgüteklassen sind dabei nach der erforderlichen Kompensation absteigend sortiert, sodass die Klasse b_1 die höchste und $b_{|B|}$ die niedrigste Kompensation erfordert. Dienstgüteklassen können entsprechend gängiger SLAs unterschiedliche Qualitätsmerkmale aufnehmen: Hierzu zählen insbesondere die Reaktions- oder Antwortzeit sowie Verfügbarkeit, die jeweils Einfluss auf die benötigten Ressourcen des Softwareagenten haben. Dienste in unterschiedlichen Dienstgüteklassen können somit unterschiedliche Ressourcenverbräuche aufweisen. Die Modellierung der Verknüpfung von Diensterbringung mit dem zugehörigen Ressourcenverbrauch erfordert somit die Einbeziehung der verwendeten Dienstgüteklasse. Hieraus ergibt sich die $|S| \times |R|$ Matrix $\mathbf{M}_b$ für die Ressourcennutzung je Dienst und Dienstgüteklasse:

$$\mathbf{M}_b = \left(m_{s,r}^b \right) = \begin{pmatrix} m_{1,1}^b & \cdots & m_{1,|R|}^b \\ \vdots & \ddots & \vdots \\ m_{|S|,1}^b & \cdots & m_{|S|,|R|}^b \end{pmatrix}.$$

Die Matrix $\mathbf{M}_b$ macht deutlich, dass dieselben Ressourcen zur Bereitstellung verschiedener Dienste genutzt werden können. Der Nutzen (in Form von Kompensationszahlungen) für den bereitstellenden Softwareagenten kann sich jedoch beträchtlich unterscheiden. Es ist

somit möglich, dass es für den Softwareagenten beispielsweise vorteilhafter ist, Dienst s_1 statt s_2 bereitzustellen, da beide die gleiche Menge an Ressourcen konsumieren, ersterer jedoch eine höhere Kompensation erwirtschaftet. Dies ermöglicht neue und in traditionellen Revenue Management Disziplinen nicht umsetzbare Möglichkeiten der Differenzierung durch Ressourcenflexibilität (vgl. Abschnitt 3.1.3.1): Während ein einzelner Sitzplatz auf einem Flugabschnitt zwar in verschiedenen Buchungs- bzw. Dienstgüteklassen angeboten werden kann, ist ein Transfer dieser einzelnen Ressource auf andere Flugabschnitte nicht möglich. Durch das Angebot unterschiedlicher Dienste – und somit nicht nur die Differenzierung der Nebenbedingungen – können Revenue Management Ansätze angewandt werden, selbst wenn die Voraussetzung der „Heterogenität der Kunden“ (vgl. Abschnitt 3.1.2.1) nicht oder nur eingeschränkt erfüllt wird: Haben alle in Frage kommenden Multiagenten-Organisationen die gleichen Nutzenfunktionen, so kann sich trotzdem der Bedarf an Diensten unterscheiden, da beispielsweise ein Dienst in einer Multiagenten-Organisation bereits durch einen anderen Softwareagenten abgedeckt wird. Revenue Management Ansätze können in diesem Fall dazu dienen, abzuwägen, welcher Dienst bei Beitrittsverhandlungen angeboten werden soll. Softwareagenten können eine Differenzierung somit auf zwei unterschiedlichen Dimensionen durchführen: (i) verschiedene Dienste, die dieselben Ressourcen nutzen und (ii) gleiche Dienste in verschiedenen Dienstgüteklassen. Das zu entwickelnde Verfahren kann somit auf beiden Dimensionen aufbauen, wobei jede einzelne bereits zur Anwendung von Revenue Management Ansätzen geeignet ist.

3.2.4 Anfragen von Multiagenten-Organisationen

Die Anfrage einer Multiagenten-Organisation zur Dienstbereitstellung eines Softwareagenten ist die Entscheidung ersterer den Realisationsprozess des Beitritts zu starten (vgl. Abschnitt 2.4.1). Die Anfrage stellt hingegen keine Entscheidung der Multiagenten-Organisation für einen Beitritt des Softwareagenten dar, da sie in diesem Moment noch über keine Informationen zu den Bedingungen des Beitritts verfügt. Die Multiagenten-Organisation stellt hingegen den Bedarf für einen oder mehrere Dienste fest, für deren Bereitstellung Angebote eingefordert werden. Wird nicht nur ein einzelner Dienst angefragt, sondern ein Bündel von Diensten, so ist davon auszugehen, dass auch nur Angebote zur Bereitstellung aller in diesem Bündel enthaltenen Dienste von der Multiagenten-Organisation akzeptiert werden. Ohne Beschränkung der Allgemeinheit wird in dieser Arbeit davon ausgegangen, dass in jedem Zeitabschnitt maximal eine Anfrage eingeht. Diese Annahme findet sich in zahlreichen Revenue Management-Ansätzen (vgl. beispielsweise Talluri und van Ryzin (1998, S. 1579)) und kann durch eine entsprechend kurze Gestaltung der Zeiteinheiten vernachlässigt werden (Talluri und van Ryzin, 2004, S. 87). Eine Anfrage zum Zeitpunkt t wird durch den Vektor $\mathbf{r}_t$ abgebildet:

$$\mathbf{r}_t = \begin{pmatrix} r_1^t \\ \vdots \\ r_{|S|}^t \end{pmatrix}.$$

Dieser Anfragevektor ist nicht abhängig von Dienstgüteklassen, da zum Zeitpunkt der Anfrage alle Dienstgüteklassen für die Multiagenten-Organisation potentiell interessant sind. Welche Dienstgüteklassen vom Softwareagenten angeboten werden, liegt in dessen Entscheidungsspielraum und ist erst bei Angebotserstellung von Interesse. Die Entscheidung über die Akzeptanz von Buchungsklassen und deren Auswahl erfolgt hingegen nach Erhalt der Angebote durch die Multiagenten-Organisation. Die Anfrage bezieht sich jedoch stets auf die zur Bereitstellung angeforderten Dienste. Welche Ressourcen der Softwareagent in welchem Umfang zur Bereitstellung dieser Dienste nutzt, ist der Multiagenten-Organisationen hingegen im Allgemeinen nicht bekannt.

3.2.5 Dienstbereitstellungsangebote

Nach dem Erhalt einer Anfrage $\mathbf{r}_t$ zum Zeitpunkt t steht es dem angefragten Softwareagenten frei, der Multiagenten-Organisation ein Angebot für den Beitritt zu unterbreiten. Das Angebot kann dabei als Willenserklärung des Softwareagenten analog zu den nach BGB vorgesehenen Willenserklärungen von natürlichen Personen angesehen werden.²¹ Die Erstellung eines Angebots ist somit bereits die positiv ausgefallene Entscheidung zum Beitritt zur anfragenden Multiagenten-Organisation aus Sicht des Softwareagenten und somit ein weiterer Schritt des Realisationsprozesses (vgl. Abschnitt 2.4.1). Diese einseitige Beitrittsentscheidung wird durch den binären Vektor $\bar{\mathbf{d}}_t$ repräsentiert:

$$\bar{\mathbf{d}}_t = \begin{pmatrix} \bar{d}_1^t \\ \vdots \\ \bar{d}_{|B|}^t \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix}$$

wobei $\bar{\mathbf{d}}_t$ auf die Menge $\mathcal{D}(\mathbf{x}_t) = \left\{ \bar{\mathbf{d}}_t \in \{0, 1\}^{|B|} : \bar{d}_b^t \mathbf{M}_b^\top \mathbf{r}_t \leq \mathbf{x}_t \ \forall b \in B \right\}$ aufgrund der zur Verfügung stehenden Ressourcenkapazität des Softwareagenten beschränkt ist.

Teil des Angebotes zur Dienstbereitstellung ist die hierfür eingeforderte Kompensation. Diese kann grundsätzlich als wiederkehrender Fixpreis oder nutzenbasiert gezahlt werden. Untersuchungen im Bereich der Software-Dienstbereitstellung durch Unternehmen haben aufgezeigt, dass die Aufwendungen für die Abrechnung und Kontrolle der Nutzung bei einer nutzenbasierten Entlohnung die Erträge übersteigen (Bala und Carr, 2010). Analog zu Mitgliedschaften von menschlichen Organisationsmitgliedern wird hier angenommen, dass eine feste Kompensation in jeder Zeiteinheit für die Bereitstellung eines entsprechenden Dienstes gezahlt wird. Die Kompensation für die Bereitstellung jeweils einer Einheit der Dienste wird repräsentiert durch den Vektor $\mathbf{p}_b$ für Dienstgüteklasse b :

$$\mathbf{p}_b = \begin{pmatrix} p_1^b \\ \vdots \\ p_{|S|}^b \end{pmatrix}.$$

²¹ Zur Ausführlichen Betrachtung der Rechtsfähigkeit von Softwareagenten vgl. auch Müller-Hengstenberg und Kirn (2016) sowie Sorge (2005)

Die Kompensation multipliziert sich um die Menge der Anfrage durch die Multiagenten-Organisation. Hat eine Multiagenten-Organisation beispielsweise einen erhöhten Bedarf nach Dienst s_1 und fragt hiervon fünf Einheiten an, so beträgt die Kompensation $5p_1^b$ in Dienstgütekategorie b . Hieraus lässt sich der Ressourcenverbrauch zur Dienstbereitstellung abhängig der angebotenen Dienstgütekategorie b wie folgt berechnen:

$$\mathbf{M}_b^\top \mathbf{r}_t = \begin{pmatrix} m_{1,1}^b & \cdots & m_{|S|,1}^b \\ \vdots & \ddots & \vdots \\ m_{1,|R|}^b & \cdots & m_{|S|,|R|}^b \end{pmatrix} \begin{pmatrix} r_1^t \\ \vdots \\ r_{|S|}^t \end{pmatrix}.$$

Das Ergebnis ist ein Vektor der Dimension $|R|$ mit dem Ressourcenverbrauch jeder einzelnen Ressource. Für den Softwareagenten ist dieser Ressourcenverbrauch in Verbindung mit der erzielten Kompensation wesentlicher Bestandteil der Beitrittsentscheidung, da diese Ressourcen gebunden sind und nicht mehr für die Dienstbereitstellung in anderen Multiagenten-Organisationen zur Verfügung stehen. Dabei wird davon ausgegangen, dass der Softwareagent sich über den potenziellen Wert seiner angebotenen Dienstleistungen für Multiagenten-Organisationen bewusst ist und den Vektor $\mathbf{p}_b$ bereits vorab definiert hat. Der Ansatz dieser Arbeit hat nicht die Festlegung von Preisen bzw. Kompensationen zum Kern, sondern die Ausnutzung von Potentialen bei der Beitrittsentscheidung unter Berücksichtigung bestehender Richtwerten für Kompensationsleistungen.

3.2.6 Mitgliedschaftsverträge

Hat ein Softwareagent einer Multiagenten-Organisation eines oder mehrere Angebote zum Beitritt erstellt und diese einem durch eine entsprechende Willenserklärung zugestimmt, ist ein Vertrag zwischen beiden Parteien zustande gekommen, der die Mitgliedschaft des Softwareagenten in der Multiagenten-Organisation konstituiert (vgl. Abschnitt 2.4.2). Der binäre Vektor für erfolgreiche (nach positiver Rückmeldung) Beitrittsentscheidungen und somit abgeschlossene Verträge $\mathbf{d}_t$ in Dienstgütekategorie b in t ist analog zur einseitigen Beitrittsentscheidung $\bar{\mathbf{d}}_t$ aufgebaut, enthält jedoch höchstens ein Mal den Wert 1, da die abgegebenen Angebote als nicht-disjunkte Alternativen aufzufassen sind:

$$\mathbf{d}_t = \begin{pmatrix} d_1^t \\ \vdots \\ d_{|B|}^t \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix}$$

mit $d_b^t \in \{0, 1\} \forall b$ und $\sum_{b=1}^{|B|} d_b^t \leq 1$.

Die abgeschlossenen Verträge des Softwareagenten und somit die Mitgliedschaften in Multiagenten-Organisationen werden in der $|S| \times |B|$ Matrix $\mathbf{C}_t$ im Zeitpunkt t für sämtliche Dienste und Dienstgüteklassen wie folgt definiert:

$$\mathbf{C}_t = (c_{s,b}^t) = \begin{pmatrix} c_{1,1}^t & \cdots & c_{1,|B|}^t \\ \vdots & \ddots & \vdots \\ c_{|S|,1}^t & \cdots & c_{|S|,|B|}^t \end{pmatrix}$$

mit dem b -ten Spaltenvektor $\mathbf{c}_b^t$ der Matrix $\mathbf{C}_t$.

Die eingegangenen Mitgliedschaften des Softwareagenten basieren entsprechend den Anfragen $\mathbf{r}_t$ der Multiagenten-Organisationen auf den bereitzustellenden Diensten. Der Ressourcenverbrauch für diese Bereitstellung ergibt sich erst aus den internen Strukturen des Softwareagenten und kann sich durchaus zwischen mehreren Softwareagenten unterscheiden. Aus den Verpflichtungen des Softwareagenten, die sich zum Zeitpunkt t aus der Matrix $\mathbf{C}_t$ ergeben, lässt sich der Gesamtressourcenverbrauch des Softwareagenten berechnen:

$$\sum_{b=1}^{|B|} \mathbf{M}_b^\top \mathbf{c}_b^t = \sum_{b=1}^{|B|} \begin{pmatrix} m_{1,1}^b & \cdots & m_{|S|,1}^b \\ \vdots & \ddots & \vdots \\ m_{1,|R|}^b & \cdots & m_{|S|,|R|}^b \end{pmatrix} \begin{pmatrix} c_{1,b}^t \\ \vdots \\ c_{|S|,b}^t \end{pmatrix}.$$

Dieser Gesamtressourcenverbrauch ist Grundlage für weitere Beitrittsentscheidungen, da er die noch frei verfügbaren Kapazitäten bestimmt. Für den nachfolgenden Zeitabschnitt verändert sich die Matrix $\mathbf{C}_t$ wie folgt:

$$\mathbf{C}_{t+1} = \mathbf{C}_t + \mathbf{r}_t \mathbf{d}_t^\top = \begin{pmatrix} c_{1,1}^t & \cdots & c_{1,|B|}^t \\ \vdots & \ddots & \vdots \\ c_{|S|,1}^t & \cdots & c_{|S|,|B|}^t \end{pmatrix} + \begin{pmatrix} r_1^t d_1^t & \cdots & r_1^t d_{|B|}^t \\ \vdots & \ddots & \vdots \\ r_{|S|}^t d_1^t & \cdots & r_{|S|}^t d_{|B|}^t \end{pmatrix}.$$

3.3 Prognosemodell

Die in Abschnitt 3.1 erläuterten Parallelen von konventionellen Revenue Management Ansätzen und den Beitrittsentscheidungen von Softwareagenten basieren auf dem Prinzip der Anfragenabweisung zum Vorteil potentieller zukünftiger Anfragen. Das hier entwickelte Verfahren ermöglicht es Softwareagenten, Anfragen von Multiagenten-Organisationen auf ihre zukünftigen Auswirkungen hin zu beurteilen, um somit die aktuelle Anfrage eingehend bewerten zu können. Um diese in der Zukunft liegenden Auswirkungen betrachten zu können, sind Informationen über ebenfalls in der Zukunft liegende Ereignisse nötig, die einen Einfluss auf die derzeitige Beitrittsentscheidung haben. Hierzu zählen insbesondere die nicht durch den Softwareagenten selbst beeinflussbaren Ereignisse, die (potentielle) Mitgliedschaften und einen dadurch veränderten Ressourcenverbrauch nach sich ziehen. Änderungen der Mitgliedschaft eines Softwareagenten können dabei in zwei Richtungen erfolgen: Zum einen kann der Softwareagent weiteren Multiagenten-Organisationen beitreten und zum anderen kann eine Multiagenten-Organisation, in der er Mitglied ist, ebendiese

Mitgliedschaft aufkündigen. Eine Prognose sowohl der zukünftigen Anfragen für potentielle Mitgliedschaften als auch der zukünftigen Kündigung von Mitgliedschaften durch die Gegenseite ist somit essentiell, um deren Auswirkungen in der Zukunft und somit auf die aktuelle Beitrittsentscheidung beurteilen zu können.

Während in traditionellen Revenue Management Domänen das Prognoseproblem sehr konkret bekannt ist (z.B. Anzahl Fluggäste pro Flugabschnitt, Anzahl Übernachtungsgäste), kann die Beitrittsentscheidung von Softwareagenten zu Multiagenten-Organisationen in vielen verschiedenen Domänen mit sehr unterschiedlichen Gegebenheiten angewandt werden. Das hier vorgestellte Prognosemodell kann somit nur eine rudimentäre Basis bilden, die je nach Anwendung beliebig weit ausgestaltet werden kann. Je nach Anwendung können somit verschiedene Faktoren mit einbezogen werden, insbesondere eine zeitliche Unterscheidung nach Tageszeit, Wochentage oder Jahreszeit.

Das hier vorgestellte Prognoseverfahren basiert auf der Idee von Littlewood (1972), welches jedoch grundlegender Anpassung für Beitrittsentscheidungen von Softwareagenten bedarf. Während Littlewood (1972) auf die Prognose einer einzelnen in der Zukunft liegenden Dienstleistung abzielt, finden Dienstleistungen im Rahmen von Mitgliedschaften in Multiagenten-Organisationen ebenso wie etwaige Kündigungen ebenjener kontinuierlich statt. Aus Mangel an weiteren Informationen liegt es nahe, vergangene Ereignisse zur Vorhersage zukünftiger heranzuziehen. Hierzu wird die Annahme getroffen, dass aus vergangenen Ereignissen auf mögliche zukünftige geschlossen werden kann.

Es wird angenommen, dass die erwarteten Kündigungen einer Exponentialverteilung folgen mit einer erwarteten durchschnittlichen Vertragsdauer λ_t im Zeitpunkt t . Da die Mitgliedschaft in einer Multiagenten-Organisation auf Langfristigkeit ausgelegt ist (vgl. Abschnitt 2.3.2.2), ist die tatsächliche Dauer der Mitgliedschaft a priori nicht bekannt. Zur Bestimmung von λ_t wird der Durchschnitt der bis zum Zeitpunkt t bereits gekündigten Mitgliedschaften herangezogen. Hieraus ergibt sich der Erwartungswert zum Zeitpunkt t für zukünftige Zeitabschnitte $\hat{t}$ für Ressource x_r^t :

$$E_t(x_r^t, \hat{t}) = x_r^t \left(1 - e^{-\lambda_t(\hat{t}-t)} \right).$$

Neben den verringernden Auswirkungen der Ressourcennutzung durch Kündigungen, sind ebenfalls die Nachfragen nach neuen Mitgliedschaften für eine zu treffende Beitrittsentscheidung von Interesse. Bei der Nachfrage nach einem Dienst können dabei entweder die bisher angefragten oder die tatsächlich eingegangenen Mitgliedschaften herangezogen werden. Da jedoch die Zahl der angefragten die der tatsächlich eingegangenen Mitgliedschaften deutlich übersteigen kann und für zukünftige Entscheidungen jedoch auch in der Vergangenheit abgelehnte Beitritte relevant sein können, ist es zielführend, bei der Berechnung der Nachfrage die erhaltenen Anfragen heranzuziehen. Hierbei gilt es jedoch zu beachten, dass nicht jede Anfrage einer Multiagenten-Organisation nur aufgrund der Ablehnung des angefragten Softwareagenten nicht zustande kommt, sondern dies auch in einem nicht adäquaten Angebot des Softwareagenten begründet sein kann. Es besteht somit die Gefahr, dass die zukünftige Nachfrage nach bestimmten Diensten überschätzt wird

und somit zu viele Anfragen abgelehnt werden. Ein Korrekturfaktor sollte diese Überschätzung korrigieren, was durch eine entsprechende Anpassung bei der Bid-Price-Berechnung durchgeführt wird (vgl. Abschnitt 3.4.3).

Wie bei den Kündigungen auch, wird angenommen, dass aus den vergangenen Nachfragen auf zukünftige Ereignisse geschlossen werden kann. Dabei wird ebenfalls angenommen, dass die Nachfrage nach Diensten einer Normalverteilung mit Mittelwert μ_b^t und Standardabweichung σ_b^t zum Zeitpunkt t folgt. Die Nachfrage zum Zeitpunkt t wird für jeden Dienst s und jede Dienstgütekategorie b separat erfasst und durch $D_{s,b}^t$ repräsentiert. $D_{s,b}^t$ spiegelt die Nachfrage nach Diensten des Softwareagenten wider. Für die Entscheidung über einen Beitritt hingegen ist nicht die Nachfrage nach Diensten, sondern jene nach den Ressourcen des Softwareagenten relevant. Nur die Ressourcen des Softwareagenten beschränken die Bereitstellung von Diensten durch deren begrenzte Verfügbarkeit. Die erwartete Nachfrage in t nach Ressource r mit Nachfrage $D_{s,b}^t$ nach Dienst s in Dienstgütekategorie b ergibt sich somit wie folgt:

$$D_t^r(t, \hat{t}) = \sum_{\bar{t}=t}^{\hat{t}} \sum_{s=1}^{|S|} \sum_{b=1}^{|B|} m_{s,r}^b D_{s,b}^{\bar{t}}$$

mit der durch Faltung entstehenden Verteilung $\mathcal{N}(\mu_t^r, \sigma_t^r)$ von $D_t^r(t, \hat{t})$. Die erwartete Nachfrage nach Ressourcen in einem zukünftigen Zeitraum $\hat{t} - t$ ist Basis für Beitrittsentscheidungen von Softwareagenten für das in dieser Arbeit entwickelte Verfahren. Die nachfolgenden Abschnitte nutzen diese Prognose, um dem Softwareagenten eine möglichst nutzenstiftende Beitrittsentscheidung zu ermöglichen.

3.4 Kapazitätssteuerung bei Beitrittsentscheidungen

Der Beitritt in eine Multiagenten-Organisation hat für den beitretenden Softwareagenten weitgehende Konsequenzen. Die in der Multiagenten-Organisation gebundenen Ressourcen stehen ihm nicht mehr für andere Aktivitäten zur Verfügung und lassen sich durch den langfristigen Charakter der Mitgliedschaft nicht ohne weiteres wieder freigeben. Die Entscheidung für einen solchen Beitritt ist somit sorgfältig abzuwägen. Für Entscheidungsmodelle von Softwareagenten haben Nissen und Sengupta (2006, S. 147) drei Phasen identifiziert, aus der sich die in Abschnitt 2.4.1 beschriebene Entscheidung zusammensetzt:

1. Zunächst wird eine *universelle Menge* gesucht, in der sämtliche angebotenen Dienste enthalten sind. Diese Menge an grundsätzlich möglichen angebotenen Diensten wird hier bereits als gegeben angenommen. Durch die in Abschnitt 3.2.3 definierte Menge S sind bereits sämtliche durch den Softwareagenten anzubietenden Dienste definiert.
2. Als Teilmenge der universellen Menge gilt es, eine *Alternativenmenge* herauszubilden, welche abhängig von den aktuellen Umständen angeboten werden kann. Hierzu zählen insbesondere die aktuell frei verfügbaren Ressourcen, die eine Mitgliedschaft in einer weiteren Multiagenten-Organisation ermöglichen. Durch einen Vergleich der zur Verfügung stehenden Restressourcen $\mathbf{x}_t$ und den für einen Beitritt benötigten

Ressourcen $\mathbf{M}_b^\top \mathbf{r}_t$ kann direkt die in Frage kommende Alternativenmenge ermittelt werden (vgl. Abschnitte 3.2.3 und 3.2.5).

3. Vor der anschließenden Realisationsphase gilt es, eine Auswahl aus der Alternativenmenge zu treffen, die für den Softwareagenten den höchsten Nutzen verspricht. Diese dritte Phase ist Kern des nachfolgend entwickelten Verfahrens. Auf Basis einer a priori getroffenen Alternativenmenge, die bereits Informationen zu verfügbaren Ressourcen sowie der wirtschaftlichen Bewertung in Form von Kompensationszahlungen enthält, wählt das Verfahren diejenige Alternative aus, die unter Berücksichtigung vergangener Lernprozesse den höchsten zu erwartenden Nutzen erzielt.

Als Basis dient das in Abschnitt 3.2 vorgestellte Modell der Beitrittsentscheidung sowie die prognostizierten Werte für zukünftige Ereignisse (vgl. Abschnitt 3.3). Hierauf aufbauend wird zunächst der Kern des Entscheidungsverfahrens sowie das Bid-Price-Verfahren entwickelt, um abschließend auf die Berechnung der Bid-Prices einzugehen.

3.4.1 Entscheidungsverfahren

Nach Erhalt einer Mitgliedschaftsanfrage $\mathbf{r}_t$, muss der angefragte Softwareagent die Entscheidung treffen, ob er ein oder mehrere Angebote für einen Beitritt abgibt. Hat ein Softwareagent zur Zeit der Anfrage ungenutzte Ressourcen, wird zunächst jeder Beitritt, der eine positive Kompensation bietet, dem Softwareagenten einen positiven Nutzen stiften. Dabei gilt es jedoch zu beachten, dass die Ressourcen, die der Softwareagent durch diesen Beitritt bindet auch für mögliche nachfolgende Anfragen nicht mehr zur Verfügung stehen. Würde der Softwareagent bei gleichem Ressourceneinsatz in einer nachfolgenden Anfrage eine wesentlich höhere Kompensation erzielen, könnte es vorteilhaft sein, erstere auch bei vorhandenen Ressourcen abzulehnen, um die zweite höher kompensierte Anfrage zu bedienen. Während beispielsweise eine Fluggesellschaft eine abgelehnte Anfrage durch eine später eintreffende verlustfrei kompensieren kann, ist dies für den Softwareagent nicht zwingend möglich: Lehnt ein Softwareagent trotz ungenutzter Ressourcen eine Anfrage einer Multiagenten-Organisation ab, da er auf eine höherwertige Anfrage in der Zukunft spekuliert, entsteht ihm durch die weiterhin ungenutzten Ressourcen ein entgangener Gewinn. Fraglich ist, inwiefern dieser entgangene Gewinn tatsächlich durch spätere Anfragen überkompensiert werden kann, damit sich die Ablehnung für den Softwareagenten rückblickend als vorteilhaft herausstellt. Da die zukünftige Nachfrage allerdings nicht a priori bekannt ist, hilft das in Abschnitt 3.3 vorgestellte Prognosemodell entsprechende Erwartungswerte zu berechnen.

Basis der Beitrittsentscheidung und somit der Annahme oder Ablehnung von Anfragen sind stets die verfügbaren Ressourcen, die der Softwareagent zum Zeitpunkt t nicht gebunden und somit nicht gewinnerwirtschaftend eingesetzt hat. Aus den zum Zeitpunkt t zur Verfügung stehenden Ressourcen $\mathbf{x}_t$ lassen sich die freien Kapazitäten des nachfolgenden Zeitabschnitts wie folgt berechnen:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \sum_{b=1}^{|B|} \left(d_b^t \mathbf{M}_b^\top \mathbf{r}_t \right).$$

Die frei verfügbaren Ressourcen $\mathbf{x}_t$ stellen für den Softwareagenten einen gewissen Wert dar, da nur freie Ressourcen ihm den Beitritt in weitere Multiagenten-Organisationen ermöglichen. Es wird angenommen, dass eine Funktion $V_t(\mathbf{x})$ existiert, die die Wertschätzung des Softwareagenten zum Zeitpunkt t für die ihm zur freien Verfügung stehende Ressourcenkapazität $\mathbf{x}_t$ abbildet. Die tatsächliche Umsetzung der Funktion $V_t(\mathbf{x})$ kann dabei variieren und ist insbesondere abhängig von der Risikoeinstellung des Softwareagenten, da ausschließlich zukünftige Ereignisse auf die Wertbildung Einfluss nehmen. Ungeachtet dessen ermöglicht sie die Bewertung der verfügbaren Restkapazität, welche essentiell zur Beantwortung einer Anfrage $\mathbf{r}_t$ ist. Um eine für den Softwareagenten optimale Entscheidung treffen zu können, sollte diese stets die Bellman-Gleichung erfüllen (Bellman, 1957, S. 83 ff.). Auf die Beitrittsentscheidung von Softwareagenten übertragen lautet diese:

$$V_t(\mathbf{x}_t) = E \left[\max_{\vec{\mathbf{d}}_t \in \mathcal{D}(\mathbf{x}_t)} \sum_{b=1}^{|B|} \vec{d}_b^t \mathbf{p}_b^\top \mathbf{r}_t + V_{t+1}(\mathbf{x}_{t+1}) \right].$$

Dabei existiert im Falle von Beitrittsentscheidungen von Softwareagenten im Gegensatz zu anderen Revenue Management Domänen durch die a priori unbekannte Vertragsdauer keine Randbedingung für $V_t(\mathbf{x})$ bei Erreichen eines bestimmten Zeitpunkts. Aus der angepassten Bellman-Gleichung wiederum lässt sich ableiten, dass eine optimale Entscheidung folgender Regel genügt:

$$\vec{d}_b^t = \begin{cases} 1 & \text{falls } \mathbf{p}_b^\top \mathbf{r}_t \geq V_{t+1}(\mathbf{x}_t) - V_{t+1}(\mathbf{x}_t - \mathbf{M}_b^\top \mathbf{r}_t) \text{ und } \mathbf{M}_b^\top \mathbf{r}_t \leq \mathbf{x}_t \\ 0 & \text{sonst.} \end{cases} \quad (3.1)$$

Gleichung 3.1 beschreibt die intuitiv gültige Entscheidungsregel einer Multiagenten-Organisation nur dann beizutreten, wenn die aus einem Beitritt entstehende Kompensation die Opportunitätskosten übersteigt (Talluri und van Ryzin, 2004, S. 88 f.).

3.4.2 Bid-Price Verfahren

Beitrittsentscheidungen eines Softwareagenten und die daraus entstehenden Mitgliedschaften binden im Allgemeinen mehrere Ressourcentypen (Prozessorleistung, Speicherplatz, etc.) über mehrere Zeiteinheiten, insbesondere in der Zukunft liegende. Im Revenue Management werden Problemstellungen, die mehrere Ressourcentypen oder -mengen gleichzeitig betrachten, den Netzwerk-Problemen zugeordnet, da bei der Angebotserstellung für eine Dienstbereitstellung die bestehenden Abhängigkeiten zwischen den Ressourcen mit betrachtet werden (vgl. Abschnitt 3.2.1). Einen verbreiteten Ansatz stellt dabei der insbesondere von Williamson (1992) entwickelte Bid-Price-Ansatz dar. Der *Bid-Price*²² ist eine Art Reservationspreis für jede Ressource, die zur Beurteilung jeder einzelnen Dienstgütekategorie dient. Dabei werden nur diejenigen Dienstgütekategorien angeboten, die einen höheren Umsatz als der Reservationspreis generieren. Während im Falle einer einzelnen Ressource alle möglichen Kombinationen aus Restkapazität und Zeitkomponente vorberechnet werden

²² Da es sich bei der Bezeichnung *Bid-Price* um einen in der Revenue Management Literatur etablierten Fachbegriff handelt, wird hier ebenfalls der englische Begriff verwendet.

können, ist dies bei komplexeren Problemen im Allgemeinen nicht möglich: Um sich verändernden Restkapazitäten und Abweichungen von der erwarteten Nachfrage anzupassen, muss dieser Reservationspreis regelmäßig neu berechnet werden. Die indirekte Bepreisung jeder einzelnen Ressource ermöglicht es, Dienste, die mehrere Ressourcen – Typen oder Einheiten – benötigen, einfach zu bewerten, indem die entsprechenden Reservationspreise der Ressourcen summiert werden. Durch die Einfachheit der Bewertung mit vorberechneten Bid-Prices, können auch komplexere Anfragen zeitnah beantwortet werden.

Bid-Price Verfahren sind nicht zwingend optimal, weisen jedoch bei komplexen Problemen ein asymptotisch optimales Verhalten auf (Talluri und van Ryzin, 2004, S. 120). Bid-Price Verfahren haben gegenüber anderen Verfahren den Vorteil, dass eine Bestimmung des Bid-Price nur für jede Ressource notwendig ist und nicht für jedes Produkt bzw. jeden Dienst. Da eine Ressource – sowie Kombinationen aus Ressourcen – zur Bereitstellung einer Vielzahl von Diensten genutzt werden kann, kann die Komplexität des Problems dadurch beträchtlich reduziert werden (vgl. Abschnitt 3.1.3.1 sowie Talluri und van Ryzin (2004, S. 93)).

Das Bid-Price-Verfahren geht davon aus, dass die in Gleichung 3.1 festgelegte Differenz der Valuierungsfunktion $V_{t+1}(\mathbf{x}_t) - V_{t+1}(\mathbf{x}_t - \mathbf{M}_b^T \mathbf{r}_t)$ berechnet werden kann und somit ein Gradient $\nabla V_t(\mathbf{x})$ existiert. Dieser Gradient lässt sich interpretieren als der Reservationspreis, aus welchem wiederum die Bid-Prices abgeleitet werden können. Aufgrund der Komplexität der Bewertung der freien Ressourcenkapazitäten ist dieser Übergang jedoch im Allgemeinen nur approximativ möglich (vgl. Abschnitt 3.4.3). Die Bid-Prices werden in einem $|R|$ -dimensionalen Vektor dargestellt:

$$\boldsymbol{\pi}_t = \begin{pmatrix} \pi_1^t \\ \vdots \\ \pi_{|R|}^t \end{pmatrix}.$$

Die Bid-Prices π_r^t stellen einen Reservationspreis für jede Ressource r zum Zeitpunkt t dar. Jede Anfrage einer Multiaagenten-Organisation nach der Bereitstellung einer bestimmten Menge an Diensten kann somit durch Umrechnung der Dienstbereitstellung zum relevanten Ressourcenverbrauch auf ihre Vorteilhaftigkeit gegenüber den berechneten Bid-Prices überprüft werden (vgl. Abschnitt 3.2.5). Die einseitige Entscheidung des Softwareagenten $\bar{d}_b^t$ stellt bereits dessen Beitrittsentscheidung dar, obwohl diese noch von der Zustimmung der Multiagenten-Organisation abhängig ist. Da es sich bei der Angebotsabgabe des Softwareagenten um ein bindendes Angebot handelt, muss die Entscheidung bereits zu diesem Zeitpunkt fallen. Hieraus ergibt sich die binäre Ergebnismenge für jede Dienstgüteklasse, die nur eine Zu- oder Absage für eine Anfrage zulässt. Unter der Annahme, dass oben genannter Gradient $\nabla V_t(\mathbf{x})$ existiert, lässt sich Gleichung 3.1 zur Anwendung der Bid-Prices umformen: Die Entscheidung, ob der angefragte Softwareagent unter den

Bedingungen der Dienstgütekategorie b die Anfrage $\mathbf{r}_t$ einer Multiagenten-Organisation zum Zeitpunkt t akzeptiert, lautet somit:

$$\bar{d}_b^t = \begin{cases} 1 & \text{falls } \mathbf{p}_b^\top \mathbf{r}_t \geq (\mathbf{M}_b^\top \mathbf{r}_t)^\top \boldsymbol{\pi}_t \text{ und } \mathbf{M}_b^\top \mathbf{r}_t \leq \mathbf{x}_t \\ 0 & \text{sonst.} \end{cases} \quad (3.2)$$

Die vorliegenden Werte des Vektors $\boldsymbol{\pi}_t$ sind somit Voraussetzung für diese Entscheidung. Diesen zu berechnen ist nicht-trivial und im Allgemeinen nur heuristisch möglich. Der nachfolgende Abschnitt geht näher auf die Berechnung der Bid-Prices ein.

3.4.3 Bid-Price Berechnung

Die Berechnung des Bid-Price-Vektors $\boldsymbol{\pi}_t$ bedingt die Ausprägung einer Nutzenfunktion und somit der Möglichkeit einer Bewertung der frei verfügbaren Ressourcen. Gleichung 3.1 in Verbindung mit Gleichung 3.2 machen deutlich, dass der Bid-Price von der Bewertung der freien Ressourcen vor und nach einem potentiellen Beitritt abhängig ist. Da Revenue Management-basierte Ansätze stets auf Prognosemodellen aufbauen und somit nie Entscheidungen unter Sicherheit treffen können, ist die Entscheidung hierbei stets auch von der Risikobereitschaft des Akteurs abhängig (Chen und Alexander, 2010, S. 266 ff.). Die konkrete Berechnung des Bid-Price im vorliegenden Fall ist somit ebenfalls von der Risikobereitschaft des Softwareagenten abhängig. Im Folgenden wird davon ausgegangen, dass es sich bei dem Softwareagenten um einen Risiko-neutralen Entscheider handelt. Einer möglichen Risiko-aversen oder gar Risiko-freudigen Einstellung wird dabei durch den im Laufe dieses Abschnitts eingeführten Korrekturfaktor β_t^r Rechnung getragen.

Die Vorteile des Bid-Price-Ansatzes zur einfachen Entscheidung komplexer Anfragen erfordern die Bestimmung des Bid-Price π_r^t für jede Ressource r . Eine Erweiterung des *Expected Marginal Seat Revenue (EMSR)* Ansatzes zur Bestimmung des Bid-Price für ein Ein-Ressourcen-Problem stellt das auf Williamson (1992, S. 73 ff.) zurückgehende *Prorated Expected Marginal Seat Revenue (PEMSR)* dar. Während EMSR nur eines von vielen möglichen Verfahren zur Approximation des Bid-Price einer einzelnen Ressource darstellt, ist die „Prorated“-Erweiterung universell einsetzbar. Sie erlaubt die Erweiterung jedes Ein-Ressourcen-Lösungsansatzes auf Probleme mit mehreren Ressourcentypen. Kern des Ansatzes ist dabei die Aufteilung des Beitrags jeder einzelnen Ressource zu einem angebotenen Dienst mit Hilfe eines Gewichtungsvektors $\boldsymbol{\alpha}$. Dies ermöglicht es, jede Ressource einzeln zu bewerten und im Falle von Beitrittsentscheidungen von Softwareagenten um eine kontinuierliche zeitliche Perspektive zu erweitern. Dabei wird der Beitrag jeder Ressource zur Gesamtkompensation herangezogen, um die in der Komplexität reduzierten Ein-Ressourcen-Probleme mit dieser anteiligen Kompensation lösen zu können.

Um den Beitrag jeder Ressource bei der Dienstbereitstellung bestimmen zu können, muss die Kompensation, welche durch einen Beitritt und die damit zusammenhängende Dienstbereitstellung erwirtschaftet wird, auf die Ressourceneinheiten jedes Ressourcentyps aufgeteilt werden. Gegenüber Problemen, die in der Revenue Management Literatur betrachtet werden, ist die Komplexität dieser Aufteilung im Falle von Beitrittsentscheidungen von

Softwareagenten jedoch weiter erhöht: Zur Dienstbereitstellung werden jeweils beliebige Mengen von verschiedenen Ressourcentypen benötigt (z.B. 5 Gleitkommaoperationen pro Sekunde an Rechenleistung, 10 GB Speicherplatz). Für die weitere Betrachtung ist diese Aufteilung notwendig, jedoch nur approximativ mit Hilfe einer Heuristik möglich. Um einer Multiagenten-Organisation beizutreten, die die Bereitstellung des Dienstes s in der Dienstgütekategorie b fordert, benötigt ein Softwareagent die Ressourcen $m_{s,1}^b, \dots, m_{s,|R|}^b$. Beim Einsatz dieser Ressourcen wiederum erwirtschaftet er die Kompensation p_s^b . Die anteilige imaginäre Kompensation, die dabei durch jede einzelne Ressource r erzielt wird, wird mit $\hat{p}_{s,r}^b$ bezeichnet. Ein intuitiver Ansatz teilt die Kompensation des gesamten Dienstes gleichmäßig auf jeden Ressourcentyp und jede Ressourcenmenge auf, wobei jeweils

$$\frac{p_s^b}{\sum_{r=1}^{|R|} m_{s,r}^b}$$

erwirtschaftet würden. Dieser Ansatz würde jedoch vernachlässigen, dass die verschiedenen Ressourcen in unterschiedlicher Weise vorhanden sind und nachgefragt werden, insbesondere auch für verschiedene Dienste. Analog zum PEMSR-Ansatz wird hier der Vektor $\alpha_t = (\alpha_1^t, \dots, \alpha_{|R|}^t)^\top$ zur Anpassung der Aufteilung der Kompensation auf die einzelnen Ressourcen verwendet (Talluri und van Ryzin, 2004, S. 103). Der Vektor α_t zielt dabei insbesondere auf die Berücksichtigung von Dienstanfragen ab, welche verstärkt bestimmte Ressourcentypen nachfragen und folglich in diesem Bereich zu Ressourcen-Engpässen führen können. Dienste hingegen, welche überwiegend Ressourcen nutzen, die überproportional stark verfügbar sind, werden eher angeboten. Erweitert man den an Talluri und van Ryzin (2004, S. 102 f.) angelehnten Ansatz um eine Gewichtung nach α_t hat jede Ressource r somit einen imaginären Wert für Dienst s in Dienstgütekategorie b von

$$\hat{p}_{s,r}^b = \frac{\alpha_r^t}{m_{s,r}^b \sum_{k=1}^{|R|} \alpha_k^t} p_s^b.$$

Im Gegensatz zu anderen Revenue Management Domänen ist es wahrscheinlich, dass Softwareagenten zur Bereitstellung eines einzelnen Dienstes nicht nur mehrere Ressourcentypen, sondern auch mehrere Einheiten einer Ressource benötigen. Die Zuweisung des Beitrags einer einzelnen Ressourceneinheit zur erwirtschafteten Gesamtkompensation für einen Beitritt erfordert daher neben den Ressourcentypen auch die Berücksichtigung der Mengen einer einzelnen Ressource. Die verfügbare Menge $\mathbf{x}_t$ verändert sich im Allgemeinen durch Ein- und Austritte über die Zeit. Bereits Williamson (1992, S. 251) stellt fest, dass für Fluggesellschaften nicht-veränderliche Gewichtungswerte als nicht sehr robust einzustufen sind. Der Gewichtungsvektor α_t sollte daher über die Zeit angepasst werden. Hierzu stehen iterative Verfahren als Erweiterung von PEMSR zur Verfügung (Talluri und van Ryzin, 2004, S. 108 ff.), um α flexibel anpassen zu können. Durch die hohe Komplexität der Beitrittsentscheidung in Bezug auf Diensttyp und Dienstgüte auf der Nachfrageseite, liegt es nahe, den Wert einer Ressourceneinheit entsprechend der Nachfrage über alle Dienste sowie der verbleibenden Restkapazität der Ressourcen zu bestimmen und somit den Gewichtungsvektor für Ressource t zum Zeitpunkt t festzulegen auf $\alpha_r^t = \frac{1}{x_r^t} \forall r$.

In traditionellen Revenue Management Domänen finden sich meist diskret abgestufte Angebotsmengen sowie eins-zu-eins Beziehungen zwischen angebotenen Dienst und benötigten Ressourcen. So benötigt beispielsweise die Erbringung einer Beförderung per Flugzeug exakt einen Sitzplatz im Flugzeug, die Bereitstellung einer Übernachtungsmöglichkeit genau ein Hotelzimmer. Die in Abschnitt 3.2.3 eingeführte Matrix $\mathbf{M}_b$ zur Ressourcennutzung bei einer Mitgliedschaft in einer Multiagenten-Organisation lässt hingegen jede mögliche Kombination, auch in Abhängigkeit der Dienstgüteklasse b zu. Übliche im Revenue Management anzutreffenden Verfahren zur Bestimmung des Wertes $V_t(\mathbf{x}_t)$ der verbleibenden Ressourcen durch Formulierung als deterministisches lineares oder stochastische nicht-lineares Programm sind im Falle von Softwareagenten aufgrund der gestiegenen Komplexität nicht zielführend. Zur Bestimmung des Bid-Price π_r^t für eine Ressource r zum Zeitpunkt t wird daher das Modell nach Littlewood (1972) mit zwei Dienstgüteklassen herangezogen und auf die Bedürfnisse von Beitrittsentscheidungen von Softwareagenten erweitert.

Ausschlaggebend für die Bestimmung des Bid-Price ist dabei die Wahrscheinlichkeit, dass die zu erwartenden Anfragen durch Multiagenten-Organisationen die verfügbaren Ressourcen übersteigt. Nur, falls dieser Fall eintritt, sind Auswirkungen auf die aktuelle Beitrittsentscheidung zu erwarten. Hat der Softwareagent hingegen ausreichend Ressourcen zur Verfügung, um allen zu erwartenden Anfragen nachkommen zu können, steht einer – wirtschaftlich positiv bewertenden – Entscheidung für einen Beitritt nichts entgegen. Die Wertermittlung für den Bid-Price π_r^t für Ressource r zum Zeitpunkt t kann somit nur auf dem Verhältnis der Wahrscheinlichkeiten der zu erwartenden Anfragen bzw. Beitritte und den unter Berücksichtigung der anstehenden Austritte verfügbaren Ressourcen erfolgen:

$$\pi_r^t = \max_{s,b} \left\{ \hat{p}_{s,r}^b \right\} P(D_t^r(t, t + \delta_t) > E_t(x_r, t + \delta_t))$$

mit der erwarteten Nachfrage $D_t^r(t, t + \delta_t)$ im Zeitraum δ_t^r für Ressource r und der erwarteten Restkapazität auf Basis der aktuellen Mitgliedschaften $E_t(x_r, t + \delta_t^r)$ (zur Berechnung vgl. Abschnitt 3.3). Als Basis für den Bid-Price wird der höchste nach der oben getätigten Aufteilung auf die Ressourcen ermittelte Preis $\max_{s,b} \{ \hat{p}_{s,r}^b \}$ herangezogen, der laut dem Gewichtungsvektor α_t im Verhältnis zum Ressourcenverbrauch die höchste Kompensation verspricht. Je höher die Wahrscheinlichkeit, dass die Nachfrage nicht durch die Ressourcenkapazitäten gedeckt werden kann, desto höher der Bid-Price und desto mehr Anfragen werden durch den Softwareagenten abgelehnt.

Durch die Einführung des Zeitraums δ_t^r wird ein weiterer Unterschied zu anderen Revenue Management Domänen deutlich: Durch den sofortigen Beginn einer Mitgliedschaft und der dadurch unmittelbar realisierten Kompensation, ist die Spekulation auf eine höherwertige Dienstgüteklasse oder einen höherwertigen Dienst mit der weiteren Unsicherheit verbunden, ob die bis zu diesem Zeitpunkt nicht erwirtschaftete Kompensation durch die höhere Entlohnung noch realisiert werden kann. Auch wenn die Mitgliedschaft in Multiagenten-Organisationen grundsätzlich auf Langfristigkeit ausgelegt ist, finden Ein- und Austritte statt und die Mitgliedschaft ist somit im Allgemeinen auch zeitlich begrenzt (wenn auch a priori nicht bekannt). Geht man von einer durchschnittlichen Dauer λ^t einer Mitgliedschaft

in einer Multiagenten-Organisation aus, so muss eine höherwertige Anfrage einer anderen Multiagenten-Organisation nach einer abgelehnten niederwertigen Anfrage in einem bestimmten Zeitraum eintreffen, um den entgangenen Gewinn überkompensieren zu können. Da jede Kombination aus Dienst und Dienstgütekategorie ein anderes Verhältnis aufweist, kann dieser Zeitraum jedoch nur approximativ bestimmt werden. Zur Approximation des Zeitraums δ_t^r wird das Verhältnis der teuersten und günstigsten Dienstgütekategorie für jeden Dienst herangezogen und der Durchschnitt gebildet:

$$\delta_t^r := \beta_t^r \frac{\lambda^t}{|S|} \sum_{s=1}^{|S|} \left(1 - \frac{\hat{p}_{s,r}^{|B|}}{\hat{p}_{s,r}^1} \right)$$

mit dem Korrekturfaktor β_t^r . Dieser Korrekturfaktor bildet zum einen die Risiko-Einstellung des Softwareagenten ab und kann mittels lernender Verfahren kontinuierlich angepasst werden.

Für die Berechnung des Bid-Price-Vektors π_t und der Annahme einer normalverteilten Nachfrage (vgl. 3.3) ergibt sich für jede Ressource r :

$$\pi_r^t = \max_{s,b} \left\{ \hat{p}_{s,r}^b \right\} \left(1 - \frac{1}{\sigma_r \sqrt{2\pi}} \int_{-\infty}^{E_t(x_r, t + \delta_t)} e^{-\frac{1}{2} \left(\frac{x - \mu_r}{\sigma_r} \right)^2} dx \right).$$

3.5 Überbuchen bei Beitrittsentscheidungen von Softwareagenten

Das Überbuchen von Ressourcen ist den erweiterten Verfahren des Revenue Management zuzuordnen (siehe Abschnitt 3.1.3.3). Der Begriff *Überbuchen* stammt aus der Luftfahrtindustrie, welche das Eingehen von verbindlichen Verträgen als Buchung bezeichnet und daran angelehnt das Eingehen von mehr verbindlichen Verträgen als es die zur Verfügung stehenden Ressourcen zulassen als „Überbuchen“. Die bisher adressierten Ansätze im Revenue Management beschäftigen sich hauptsächlich mit der Frage, wie Preise und verfügbare Kapazitäten allokiert werden sollen, um einen möglichst hohen Gewinn bzw. Umsatz zu generieren. Überbuchen hingegen zielt auf die möglichst hohe Auslastung vorhandener Kapazität ab. Die Zielfunktionen unterscheiden sich somit in ihren Zielgrößen Auslastung und Gewinn, die jedoch durch den automatisch auftretenden „entgangenen Gewinn“ bei nicht vollständig ausgelasteten Kapazitäten zusammenhängen.

Grundvoraussetzung zur Anwendung des Überbuchens ist eine Wahrscheinlichkeit größer Null, dass ein Teil der zugesicherten Dienste nicht durch Vertragspartner eingefordert werden. Dies ist dem Dienstanbieter a priori nicht bekannt, weshalb beim Überbuchen somit stets eine Abwägungsentscheidung verschiedener Chancen bzw. Risiken zu treffen ist: *„On a planning level, overbooking involves controlling the level of reservations to balance the potential risks of denied service against the rewards of increased sales“* (Talluri und van Ryzin, 2004, S. 130). Das Risiko besteht beim Eingehen von Verbindlichkeiten, für die keine Ressourcen zur Verfügung stehen, stets darin, dass der Dienst doch durch den Vertragspartner eingefordert wird. Kann im Falle von Fluggesellschaften oder Autovermietungen einer Reservierung nicht

nachgekommen werden, stehen dem Kunden im Allgemeinen eine Entschädigungszahlung oder eine Ersatzleistung (z.B. durch einen anderen Dienstleister) zu. Beide Varianten verursachen Kosten beim Dienstleistungsanbieter, da er entweder die Entschädigung zahlen oder die Dienstleistung bei einem Konkurrenten einkaufen muss, welche im Regelfall teurer als die eigene Dienstleistung ist.

Insbesondere wenn ein Softwareagent Mitglied in mehreren Multiagenten-Organisationen ist (vgl. Abschnitt 2.4.4), können zu hohe Verbindlichkeiten dazu führen, dass dieser die zugesagten Dienste nicht bereitstellen kann, falls ihm im Augenblick der Anforderung zur Dienstbereitstellung die Ressourcen fehlen. In diesem Fall stehen ihm ebenfalls die Möglichkeiten offen, die Dienstleistung nicht bzw. später als vereinbart zu erbringen und eine entsprechende Vertragsstrafe zahlen zu müssen oder einen weiteren Softwareagenten mit der Dienstbereitstellung zu beauftragen und diesem eine entsprechende Kompensation zu zahlen. In beiden Fällen entstehen dem Softwareagenten zusätzliche Aufwendungen, die die Wirtschaftlichkeit seiner Mitgliedschaft beeinträchtigen. Eine Entscheidung, mehr Verpflichtungen einzugehen, als die Ressourcen des Softwareagenten durchgängig leisten können (Überbuchen) ist somit sorgfältig abzuwägen. Im Folgenden wird davon ausgegangen, dass ein Softwareagent einen monetären Schaden erleidet, wenn er seinen Verpflichtungen nicht nachkommen kann. Dabei wird abstrahiert, ob es sich dabei um eine Vertragsstrafe, erhöhte Aufwendungen für einen Drittanbieter oder gar implizite Reputationsschäden handelt (vgl. Abschnitt 2.4.3). Die jeweiligen Ausprägungen der Schadensart ist dabei abhängig vom jeweiligen Kontext sowie der Vertragsgestaltung der Mitgliedschaftsbeziehung.

Im Revenue Management wird zwischen dem kurzfristigen Nicht-Abrufen der Dienstleistung (engl. *no-show*) und der vorzeitigen Auflösung des Vertrags (engl. *cancellation*) unterschieden (Talluri und van Ryzin, 2004, S. 138). Im Falle von Mitgliedschaften von Softwareagenten in Multiagenten-Organisationen lassen sich diesen beiden Begriffe ebenfalls zwei mögliche Änderungen bezüglich der Dienstnutzung zuordnen: Zum einen ist es möglich, dass eine Multiagenten-Organisation die Mitgliedschaftsbeziehung zum dienstbereitstellenden Softwareagenten aufkündigt (Cancellation). Zum anderen kann es sein, dass die Multiagenten-Organisation die ihr zugesicherten Dienste nicht nutzt, der bereitstellende Softwareagent seine hierfür reservierten Ressourcen somit ebenfalls nicht benötigt. Einige Unterschiede müssen bei dieser Zuordnung jedoch beachtet werden: Während im Falle einer Cancellation die Fluggesellschaft im Allgemeinen die Möglichkeit hat, den entgangenen Gewinn durch einen erneuten Verkauf an einen Dritten zu kompensieren, ist diese Möglichkeit im Falle von Softwareagenten abhängig von der Vertragsgestaltung. Es sind dabei auch sofort wirksame Austritte denkbar, die dann einem No-Show gleich kämen und direkt einen entsprechenden entgangenen Gewinn verursachen würden. Ebenfalls unterscheidet sich die Erstattung bei nicht angeforderten Diensten: Bei Fluggesellschaften wird abhängig von der Dienstgütekategorie bei Nicht-Nutzung (Cancellation) Geld zurück erstattet. Werden Dienste, die ein Softwareagent einer Multiagenten-Organisation zur Verfügung stellt, nicht genutzt, ist dies nicht in der Verantwortung des Softwareagenten, eine Erstattung muss im Allgemeinen somit nicht betrachtet werden. Ausnahmen hiervon bilden die in Abschnitt 3.2.5 erwähnten Verträge mit nutzenbasierter Entlohnung.

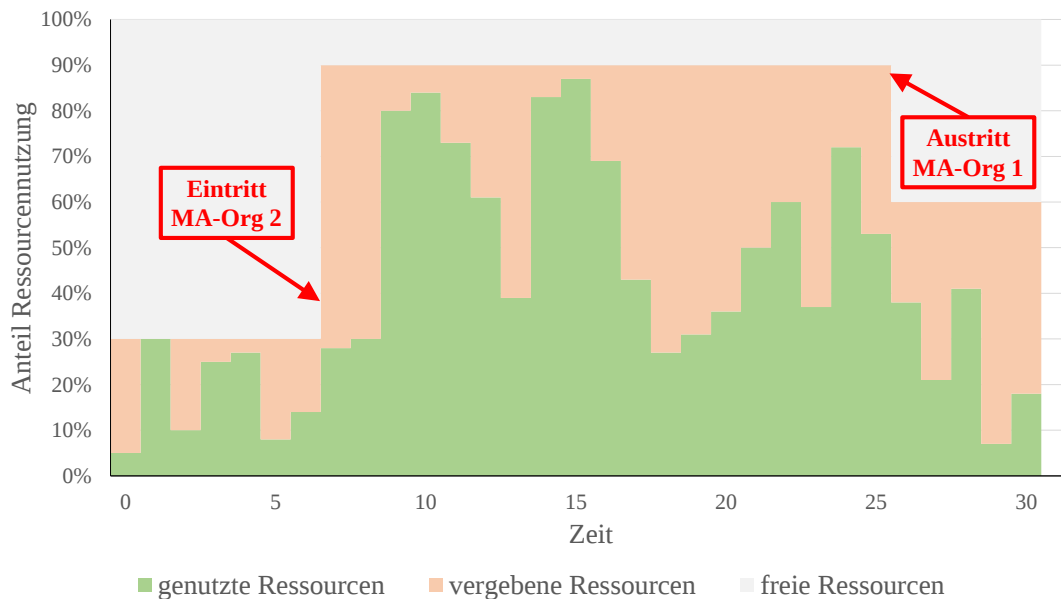


Abbildung 3.3: Beispiel zum Verlauf der freien, vergebenen und genutzten Ressourcen bei Mitgliedschaften in mehreren Multiagenten-Organisationen

Abbildung 3.3 veranschaulicht die beiden möglichen Kategorien von nicht-genutzten Ressourcen. Während ein gewisser Anteil der Ressourcen nicht in Multiagenten-Organisationen gebunden ist (grau), kann der gebundene Teil in genutzte (grün) und vergebenen Ressourcen (rot) unterteilt werden. Die freien Ressourcen stehen dem Softwareagenten für weitere Mitgliedschaften zur Verfügung. Die vergebenen Ressourcen hingegen sind notwendig, um bereits bestehende Verpflichtungen zur Bereitstellung von Diensten nachzukommen, die im jeweiligen Zeitpunkt jedoch nicht durch die Multiagenten-Organisation abgerufen werden. Das Überbuchen der Ressourcen eines Softwareagenten setzt an dem Verhältnis zwischen genutzten und vergebenen Ressourcen an. Werden über einen gewissen Zeitraum (beispielsweise von $t_1 = 17$ bis $t_2 = 25$ in Abbildung 3.3) nur ein relativ geringer Anteil der zur Bereitstellung von Diensten an Multiagenten-Organisationen reservierten Ressourcen tatsächlich genutzt, kann es für den Softwareagenten unter Umständen vorteilhaft sein, mehr Verpflichtungen einzugehen als seine Ressourcen bereitstellen können, falls entsprechende Anfragen vorliegen. Dabei hat er das Risiko zu tragen, dass entgegen seiner Erwartung die Dienste abgerufen werden. Ist der Softwareagent zur Erbringung eines Dienstes verpflichtet und kann dieser Verpflichtung nicht nachkommen (z.B. wegen Überbuchung), müssen die vertraglich festgelegten Kosten (Vertragliche Nebenbedingungen siehe Abschnitt 2.4.3) hierfür dem Softwareagenten auferlegt werden.

Die Betrachtung von Überbuchungen benötigt eine Anpassung des in Abschnitt 3.2 vorgestellten Revenue-Management-basierten Modells der Beitrittsentscheidung. Während Austritte von Softwareagenten durch die a priori unbekannte Laufzeit von Mitgliedschaftsverträgen (siehe Abschnitt 3.2.6) in der Modellierung bereits abgedeckt sind und durch die unmittelbare Wirksamkeit neuer Beitrittsentscheidungen nicht in die Überbuchung einfließen, erfordert die Betrachtung von kurzfristig nicht genutzten Ressourcen (No-Shows) eine Erweiterung des Modells. Hierzu ist insbesondere die tatsächliche Dienstnutzung

durch die Multiagenten-Organisationen, in denen der Softwareagent Mitglied ist, von Interesse. Diese wird durch den $|S|$ -dimensionalen Vektor $\mathbf{u}_{t,b}$ für jede Dienstgüteklasse b und Zeitabschnitt t definiert als

$$\mathbf{u}_{t,b} = \begin{pmatrix} u_1^{t,b} \\ \vdots \\ u_{|S|}^{t,b} \end{pmatrix}.$$

Hieraus ergeben sich die durch $\mathbf{u}_{t,b}$ genutzten Ressourcen als

$$\sum_{b=1}^{|B|} \mathbf{M}_b^\top \mathbf{u}_{t,b} = \sum_{b=1}^{|B|} \begin{pmatrix} m_{1,1}^b & \cdots & m_{|S|,1}^b \\ \vdots & \ddots & \vdots \\ m_{1,|R|}^b & \cdots & m_{|S|,|R|}^b \end{pmatrix} \begin{pmatrix} u_1^{t,b} \\ \vdots \\ u_{|S|}^{t,b} \end{pmatrix}.$$

Das Überbuchen von Ressourcen im Kontext von IT-Systemen wurde insbesondere im Rahmen von Cloud Computing bereits in verschiedenen Arbeiten untersucht: z.B. Urgaonkar et al. (2002), Anandasivam und Prem (2009), Urgaonkar et al. (2009) sowie Tomas und Tordsson (2013). Diese Ansätze lassen sich ebenfalls auf die Beitrittsentscheidung von Softwareagenten übertragen. Dabei ist das Überbuchen von Ressourcen unabhängig von der in Abschnitt 3.4 vorgestellten Kapazitätssteuerung anwendbar und somit zwar ein Teil der Beitrittsentscheidung eines Softwareagenten, jedoch nicht Kern des in dieser Arbeit entwickelten Verfahrens. Das in diesem Abschnitt vorgestellte erweiterte Modell ermöglicht es jedoch, beispielsweise auf Basis einer empirisch beobachteten Wahrscheinlichkeitsverteilung der tatsächlichen Nutzung, eine Beitrittsentscheidung für oder wider einem Beitritt zu beeinflussen. Die Umsetzung hängt dabei wesentlich von den domänenspezifischen Gegebenheiten und vertraglichen Gestaltungen ab.

4. Evaluation

Dieses Kapitel überprüft das in Abschnitt 3 entwickelte Verfahren auf dessen Wirksamkeit unter unterschiedlichen Bedingungen (vgl. Abschnitt 1.3). Das „*Artefakt wird gegen die identifizierte Forschungslücke*“ (Riege et al., 2009, S. 75) der Beitrittsentscheidungen von Softwareagenten aus dem vorangegangenen Kapitel 2 simulativ evaluiert. Es findet somit kein Einsatz unter Realweltbedingungen statt. Hingegen wird das entwickelte Verfahren gegen die in Kapitel 2 herausgearbeiteten Anforderungen evaluiert. Hierzu zählt insbesondere die Zielorientierung von Softwareagenten (vgl. Abschnitt 2.1.1.4). Zunächst wird die verwendete Evaluationsmethode aufgezeigt (Abschnitt 4.1). Abschnitt 4.2 beschreibt den Aufbau der Simulation. Abschnitt 4.3 präsentiert die Simulationsergebnisse, die abschließend diskutiert werden (Abschnitt 4.4).

4.1 Evaluationsmethode

Die Evaluation des entwickelten Verfahrens kann in verschiedenen Dimensionen erfolgen, welche nicht-abschließend beispielsweise durch Hevner et al. (2004) definiert werden: *IT artifacts can be evaluated in terms of functionality, completeness, consistency, accuracy, performance, reliability, usability, fit with the organization, and other relevant quality attributes* (Hevner et al., 2004, S. 85). Die Evaluation des in dieser Arbeit entwickelten Verfahrens wird durch zwei wesentliche Faktoren beeinflusst: (i) Softwareagenten arbeiten zielgerichtet, verfolgen somit eine Art Nutzenfunktionen sowie (ii) das entwickelte Verfahren ist domänenunabhängig einsetzbar. Hieraus lassen sich zwei Ziele für die Evaluation herausbilden:

1. Ziel der Evaluation ist es, die **Wirksamkeit** des Verfahrens unter gegebenen Bedingungen zu demonstrieren. Hierzu zählt insbesondere der Vergleich mit anderen Arten der Beitrittsentscheidung und den jeweils erreichten Zielwerten. Als Zielgröße wird dabei die erwirtschaftete Kompensation gesetzt.
2. Aufgrund der domänenunabhängigen Entwicklung des Verfahrens ist es möglich, dass sich die oben angeführte Wirksamkeit beim Einsatz in unterschiedlichen Domänen durch entsprechende Umgebungsbedingungen unterscheidet. Ziel der Evaluation ist

es daher, festzustellen, unter welchen **Bedingungen** der Einsatz des Verfahrens für den Softwareagenten vorteilhaft ist und wann es gegenüber anderen Verfahren keine Vorteile oder gar Nachteile mit sich bringt.

Als Evaluationsmethode wird die Simulation gewählt, welche im Rahmen der gestaltungsorientierten Wirtschaftsinformatik den experimentellen Evaluationsmethoden zugeordnet wird (Hevner et al., 2004, S. 86). *Simulation* wird hier als Ergebnis des Modellierungsprozesses eines vorgeschlagenen oder realen Systems verstanden: *Simulation is the process of modeling a proposed or real dynamic system and observing its behavior over time* (Pollacia, 1989, S. 8). Dabei stellt das Simulationsmodell durch seine Verkürzungseigenschaft stets nur eine unvollständige Abbildung der Realität dar, die im Allgemeinen durch künstliche Daten angereichert wird (Banks, 1998, S. 3 f.). Der Aufbau der Simulation orientiert sich an der VDI-Richtlinie 3633 für „*Simulation von Logistik-, Materialfluß- und Produktionssystemen*“ (VDI, 1997).

Durch den domänenunabhängigen Verfahrensentwurf steht ein breites Anwendungsfeld zur Verfügung, die Evaluation des entwickelten Verfahrens kann jedoch nur unter vorgegebenen Bedingungen erfolgen. Daher werden verschiedene Szenarien simuliert, die sich in wesentlichen Parametern unterscheiden. Dabei wird das oben angeführte zweite Ziel verfolgt, das Anwendungsfeld des entwickelten Verfahrens einzugrenzen. Mit Hilfe dieses Evaluationsvorgehens werden Softwareagenten in die Lage versetzt, selbstständig entscheiden zu können, wann der Einsatz des Verfahrens gegenüber anderen Verfahren einen Vorteil verspricht und somit bei der Beitrittsentscheidung Anwendung finden sollte. Die Anzahl der zu untersuchenden Einflussfaktoren ist jedoch zu groß, um Abhängigkeiten zwischen allen auftretenden Parameterkombinationen in vertretbarer Zeit untersuchen zu können. Um jedoch nicht nur einzelne Parameter isoliert zu betrachten, werden jeweils zwei Einflussfaktoren parallel verändert und deren Einfluss untersucht.

4.2 Simulationsaufbau

Zur Simulation der Beitrittsentscheidung von Softwareagenten zu Multiagenten-Organisationen wird das Multiagenten-Simulationsframework Repast Symphony in der Version 2.4 verwendet. Das Verhalten der Softwareagenten und Multiagenten-Organisationen ist in JAVA implementiert. Abbildung 4.1 zeigt den Ablauf der Beitrittsverhandlungen als UML Sequenzdiagramm: Multiagenten-Organisationen senden bei Bedarf Anfragen an Softwareagenten zur Bereitstellung von Diensten im Rahmen einer Mitgliedschaft. Die Softwareagenten evaluieren die Anfrage und geben gegebenenfalls ein Angebot für einen Beitritt ab. Erhält die anfragende Multiagenten-Organisation mehrere Angebote von verschiedenen Softwareagenten, so hat sie die Wahl zwischen den anbietenden potentiellen Mitgliedern. Bei Angeboten, die sich nur in der Kompensation unterscheiden, entscheidet sich die Multiagenten-Organisation für das günstigere Angebot. Bei exakt gleichen Angeboten wählt sie zufällig.

Softwareagenten Für die Simulation wird angenommen, dass zwei unterschiedliche Arten von Softwareagenten existieren: (i) Softwareagenten, die als Beitrittsentschei-

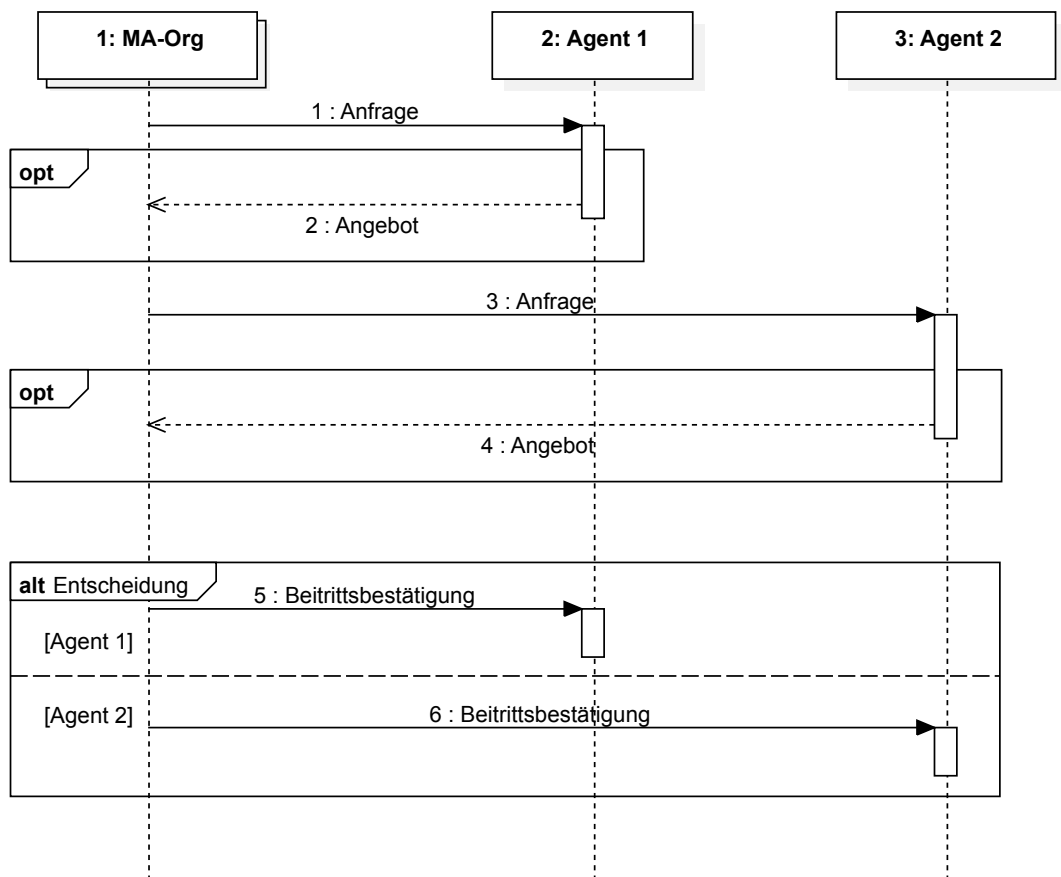


Abbildung 4.1: Sequenzdiagramm des Anfrage-/Angebotsablaufs im Simulationsmodell

dungsverfahren ausschließlich die Vorteilhaftigkeitsbewertung bezüglich der einzelnen Beitrittsentscheidung heranziehen und somit bei positiver Kompensation einer Multiagenten-Organisation beitreten, falls ausreichend Ressourcen zur Dienstbereitstellung zur Verfügung stehen (Typ *SimpleAgent*) sowie (ii) solche, die das in Kapitel 3 vorgestellte Verfahren anwenden und somit auch Anfragen unter Spekulation auf kommende höherwertigere Anfragen ablehnen (Typ *RMAgent*). Beide Typen handeln zielorientiert (vgl. Abschnitt 2.1.1.4) mit dem Ziel der Maximierung des erwirtschafteten Gewinns, wenden jedoch unterschiedliche Entscheidungsverfahren an. Dabei wird angenommen, dass Gewinn ausschließlich durch den Erhalt von Kompensationsleistungen – gemessen in Geldeinheiten (GE) – durch Multiagenten-Organisationen realisiert werden kann und die Verwendung der jeweiligen eigenen Ressourcen dem Softwareagenten keine Kosten verursacht. Softwareagenten vom Typ *RMAgent* verzichten dabei auf die Anpassung des Verfahrens durch den Parameter β_t^r : Dieser wird gesetzt als $\beta_t^r = 1 \forall r, t$. Betrachtet wird ausschließlich die Beitrittsentscheidung des Softwareagenten, Austrittsentscheidungen werden keine durch den Softwareagenten getroffen, sondern werden als extern – durch die Multiagenten-Organisation – bestimmt angenommen.

Die geforderte Kompensation $\mathbf{p}_b$ wird durch den Softwareagenten für jeden Dienst festgelegt (vgl. Tabelle 4.1, Nr. viii. bis x.) und bei der Verwendung von mehreren Dienstgüteklassen wird jeweils ein prozentualer Nachlass auf die nächst niederwertigere Dienstgütekategorie gewährt (vgl. Tabelle 4.1, Nr. xi.). Die zu Beginn jedes Simulationsdurchlaufs vorhandene Kapazität $\mathbf{x}_0$ der Ressourcen ist für alle Softwareagenten gleich (vgl. Tabelle 4.1, Nr. vii.). Dabei wird davon ausgegangen, dass jeder Softwareagent drei Ressourcentypen zu verwalten hat, die jeweils durch die zu Beginn festgelegte Kapazität beschränkt sind und nicht im Zeitverlauf der Simulation erweitert werden können: (i) Rechenkapazität (Ressource 1), (ii) Hauptspeicher (Ressource 2), (iii) Festplattenspeicher (Ressource 3). Die in Abschnitt 3.5 vorgestellte Möglichkeit des Überbuchens von Ressourcen wird durch die Softwareagenten nicht angewandt, da sie kein notwendiges Verfahren der Kapazitätssteuerung nach Abschnitt 3.4 darstellt und unabhängig davon auch bei anderen Entscheidungsverfahren Anwendung finden kann. Die Matrix $\mathbf{M}_1$ wird gefüllt durch die Werte $m_{s,x}^1 = 1 \forall s, x$. Es wird folglich angenommen, dass jeder Dienst jeweils eine Einheit der drei Ressourcen in der höchsten Dienstgütekategorie nutzt. Es wird angenommen, dass eine niederwertige Dienstgütekategorie weniger Ressourcen benötigt, als eine höherwertige: Für alle Ressourcen gilt, dass für die nächst niederwertigere Dienstgütekategorie 10 Prozent weniger Ressourcen aufzuwenden sind als für jeweils höherwertigere.

Die Simulation lässt die Untersuchung verschiedener Marktsituationen zu: (i) monopolistischer Markt, bei dem jeder Softwareagent Anfragen unabhängig von anderen Softwareagenten bearbeiten kann, (ii) duopolistischer Markt und (iii) polypolistischer Markt, bei denen jeweils zwei bzw. mehrere Softwareagenten bei einer Mitgliedschaftsanfrage in Konkurrenz zueinander stehen (vgl. Tabelle 4.1, Nr. iii.). Soweit nicht anders angegeben, wird in den Simulationen ein monopolistischer Markt angenommen.

Die Softwareagenten vom Typ *SimpleAgent* und *RMAgent* erhalten somit unabhängig voneinander die gleichen Anfragen von Multiagenten-Organisationen und können ohne Konkurrenzsituation darüber entscheiden. Hierdurch wird erreicht, dass nach einem Simulationsdurchlauf die erzielten Ergebnisse beider Verfahren direkt miteinander verglichen werden können.

Multiagenten-Organisationen Für die Simulation wird weiter angenommen, dass zwei unterschiedliche Typen von Multiagenten-Organisationen existieren, die sich ausschließlich in der angeforderten Dienstgütekategorie unterscheiden. Dabei fragen Multiagenten-Organisationen vom Typ *MA-Org-1* Dienste in der höherwertigen Dienstgütekategorie 1 und Multiagenten-Organisationen vom Typ *MA-Org-2* Dienste in der niederwertigeren Dienstgütekategorie 2 an. Bei einer Anfrage an einen Softwareagenten übermittelt jede Multiagenten-Organisation die benötigten Dienste und deren Menge. Es wird angenommen, dass pro Mitgliedschaftsanfrage nur ein Dienstyp angefragt wird und dessen Menge normalverteilt mit variablem Mittelwert (vgl. Tabelle 4.1, Nr. vi.) und einer Standardabweichung von 25% des Mittelwerts ist. Jede Multiagenten-Organisation setzt in jedem Zeitabschnitt mit einer gleichverteilten Wahrscheinlichkeit (vgl. Tabelle 4.1, Nr. iv.) eine Anfrage an alle Softwareagenten ab. Die Laufzeit der Mitgliedschaft wird durch die Multiagenten-Organisation zufällig bestimmt: Sie ist entsprechend der Nachfragemenge normalverteilt mit einem variablen Mittelwert (vgl. Tabelle 4.1, Nr. v.) und einer Standardabweichung von 25% des Mittelwerts.

Ein Simulationsdurchlauf entspricht 15.000 Zeiteinheiten, so dass bei der längsten untersuchten durchschnittlichen Mitgliedschaftsdauer (Parameter v.) ca. 5 Mitgliedschaften möglich sind. Jede Parameterkonstellation wird in 100 Simulationsdurchläufen untersucht. Aus den vorangegangenen Erläuterungen ergeben sich die in Tabelle 4.1 zusammengetragenen möglichen Einflussgrößen und Merkmalsausprägungen. Soweit in den nachfolgenden Szenarien nicht anders angegeben, werden die in der Tabelle 4.1 unterstrichenen Merkmalsausprägungen herangezogen.

In den nachfolgenden Abschnitten werden verschiedene Szenarien definiert, um die Einflussparameter auf die Beitrittsentscheidung von Softwareagenten zu untersuchen. Dabei werden – soweit möglich – zwei der in Tabelle 4.1 angegebenen Parameter variiert, um Korrelationen aufdecken zu können:

- 1. Verhältnis der nachfragenden Multiagenten-Organisationen** Hierbei wird der Einfluss der Anzahl der jeweiligen Typen von Multiagenten-Organisationen auf die erzielte Kompensation untersucht. Die Anzahl von Multiagenten-Organisationen vom Typ *MA-Org-1* und *MA-Org-2* wird dabei jeweils exponentiell erhöht (Parameter i.).
- 2. Nachfragehäufigkeit und Mitgliedschaftsdauer** In diesem Szenario wird die Nachfragewahrscheinlichkeit (Parameter iv.) und die durchschnittliche Mitgliedschaftsdauer (Parameter v.) in den angegebenen Merkmalsausprägungen jeweils exponentiell variiert.
- 3. Ressourcenverfügbarkeit und Nachfragemenge** Hier wird die vorhandene Kapazität x_0 der Ressourcen (Parameter vii.) sowie die durchschnittliche Dienstmenge

Tabelle 4.1: Simulationsparameter (Standardwerte unterstrichen)

Nr.	Einflussgröße	Merkmalausprägungen
i.	Anzahl Multiagenten-Organisationen Typ 1 / Typ 2	1, 2, 4, 8, 16, 32, <u>64</u> , 128, 256, 512, 1024
ii.	Anzahl der Softwareagenten Typ SimpleAgent	<u>1</u> , 2, 4, 8, 16, 32
iii.	Marktkonstellation	<u>Monopol</u> , Duopol, Polypol
iv.	Nachfragewahrscheinlichkeit Multiagenten-Organisation	0,0025, 0,005, <u>0,01</u> , 0,02, 0,04, 0,08, 0,16, 0,32
v.	Durchschnittliche Mitgliedschaftsdauer	25, 50, 100, 200, 400, <u>800</u> , 1600, 3200
vi.	Durchschnittliche Dienstmenge pro Anfrage	1, 2, <u>4</u> , 8, 16, 32, 64, 128
vii.	Ressourcenkapazität pro Ressource	2,5, 5, <u>10</u> , 20, 40, 80, 160, 320, 640
viii.	Kompensation pro Dienstseinheit für Dienst 1	2,5, 5, <u>10</u> , 20, 40, 80, 160
ix.	Kompensation pro Dienstseinheit für Dienst 2	2,5, 5, 10, <u>20</u> , 40, 80, 160
x.	Kompensation pro Dienstseinheit für Dienst 3	2,5, 5, 10, 20, <u>40</u> , 80, 160
xi.	Rabatt für günstigere Dienstgüteklassen	2%, 4%, 8%, 16%, <u>32%</u> , 64%
xii.	Anzahl angebotener Dienste	1, 2, <u>3</u>
xiii.	Anzahl angebotener Dienstgüteklassen	1, <u>2</u>

pro Anfrage (Parameter vi.) in den angegebenen Merkmalsausprägungen jeweils exponentiell variiert. Die vorhandene Kapazität der Ressourcen bezieht sich dabei stets auf alle drei Ressourcentypen.

- 4. Kompensation** Die zu erhaltende Kompensation ist bestimmt durch die Kompensation für jeden Dienst in der höchsten Dienstgüteklasse (Parameter viii. bis x.) sowie den Rabatt für günstigere Dienstgüteklassen (Parameter xi.). In diesem Szenario werden nur zwei Dienste angeboten und es wird sowohl der Rabatt auf günstigere Dienstgüteklassen als auch das Verhältnis zwischen den Kompensationen für Dienst 1 (variabel) und Dienst 2 (fest, 10 GE) variiert.
- 5. Dienstgüteklassen und Dienste** Dieses Szenario variiert die Anzahl angebotener Dienste (Parameter xii.) sowie Dienstgüteklassen (Parameter xiii.).
- 6. Dienstanbieterstruktur** Neben dem angenommenen monopolistischen Markt sind auch Situationen für Softwareagenten möglich, bei denen sie während der Beitrittsentscheidung in Konkurrenz mit anderen Softwareagenten stehen (Parameter iii.). Daher wird hier die Anzahl der im Markt befindlichen Softwareagenten vom Typ *SimpleAgent* variiert (Parameter ii.).

4.3 Simulationsergebnisse

Die Ergebnisse der Simulationsdurchläufe der in Abschnitt 4.2 aufgeführten Szenarien werden in den nachfolgenden Abschnitten präsentiert. Diese umfassen das Verhältnis zwischen den verschiedenen Typen nachfragender Multiagenten-Organisationen (Abschnitt 4.3.1), die Nachfragehäufigkeit und Vertragslaufzeit (Abschnitt 4.3.2), die Ressourcenverfügbarkeit und Nachfragemenge (Abschnitt 4.3.3), die Kompensationen (Abschnitt 4.3.4), die angebotenen Dienstgüteklassen und Dienste (Abschnitt 4.3.5) sowie die Dienstanbieterstruktur (Abschnitt 4.3.6). Die Auswertung der Daten wurde mit der Software SPSS Statistics des Herstellers IBM in der Version 24 durchgeführt. Die aufgeführten Korrelationen sind – soweit nicht anders angegeben – auf einem Niveau von 0,01 signifikant.

4.3.1 Verhältnis der nachfragenden Multiagenten-Organisationen

Zur Untersuchung des Einflusses des Verhältnisses der nachfragenden Multiagenten-Organisationen auf die Steigerung der erhaltenen Gesamtkompensation wurde die Anzahl der Multiagenten-Organisationen vom Typ *MA-Org-1* und *MA-Org-2* jeweils exponentiell in den Parameterausprägungen 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024 erhöht. Bei jeweils 11 Parameterausprägungen und jeweils 100 Durchläufen ergeben sich insgesamt 12.100 Simulationsdurchläufe. Abbildung 4.2 zeigt die Steigerung der durchschnittlich erzielten Gesamtkompensation des Softwareagenten vom Typ *RMAgent* gegenüber derer vom Typ *SimpleAgent* über die jeweils 100 Durchläufe für jede der angegebenen Parameterkombinationen. Im Bereich bis 8 Multiagenten-Organisationen jeden Typs liegt die durchschnittliche Steigerung bei -0,05%. Ab einer Anzahl von 32 und mehr Multiagenten-Organisationen liegt die Steigerung der Kompensation stets im positiven Bereich und steigt im Allgemeinen mit der Anzahl der nachfragenden Multiagenten-Organisationen an. Eine Ausnahme

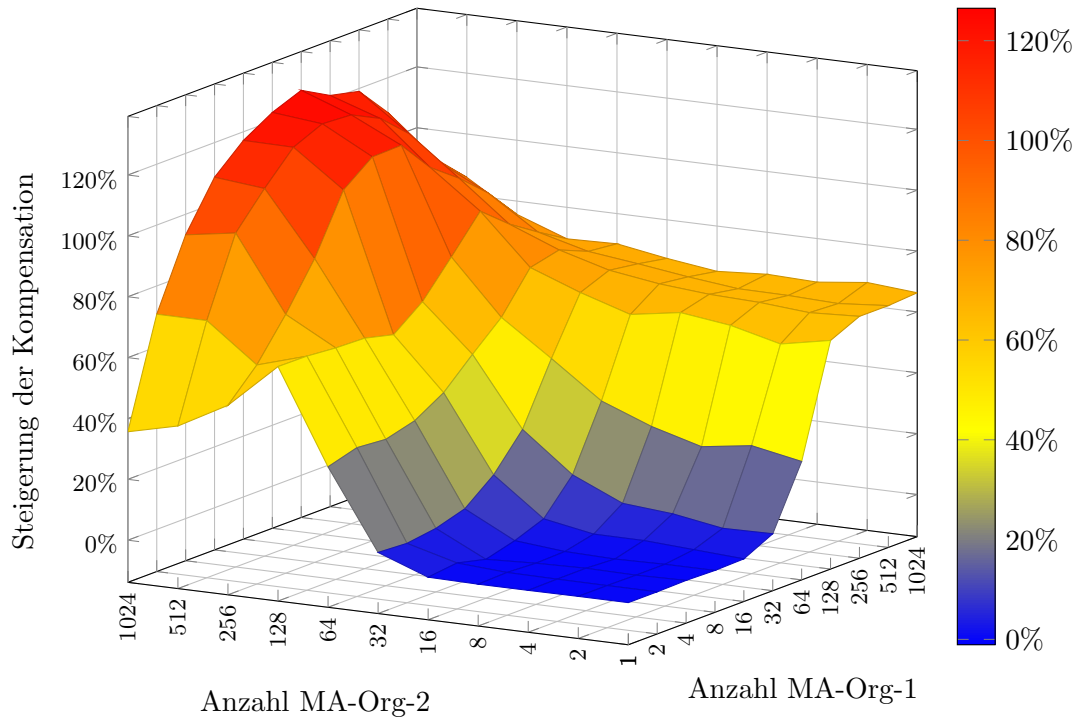


Abbildung 4.2: Steigerung der erzielten Kompensationen unter Veränderung der Anzahl nachfragender Multiagenten-Organisationen vom Typ 1 und 2

bildet der Bereich mit sehr wenigen Multiagenten-Organisationen vom Typ *MA-Org-1* und sehr vielen vom Typ *MA-Org-2*. Hier korreliert die erzielte Steigerung negativ mit der Anzahl vom Typ *MA-Org-2*. Abbildung 4.3 zeigt die erzielte Kompensation bei nur einer Multiagenten-Organisation vom Typ 1 in Abhängigkeit von der Anzahl vom Typ 2. Die durch den Softwareagenten vom Typ *SimpleAgent* erzielte Kompensation bei mehr als 128 Multiagenten-Organisationen vom Typ 2 steigt mit weiter steigender Anzahl an, während sie beim Typ *RMAgent* sinkt.

Die Abbildungen 4.4 und 4.5 zeigen die Ergebnisse als Boxplots.²³ Dabei wird deutlich, dass in den für die Parameterkonstellationen von jeweils 1 bis 8 Multiagenten-Organisationen vom Typ *MA-Org 1* und *MA-Org 2* nur Ausreißer vom Median von 0% abweichen (vgl. Abbildung 4.4, (a) - (d)). In diesen Fällen konnte das Verfahren meist keine Steigerung der Kompensation erzielen, verringerte sie jedoch auch nicht. In den übrigen Fällen der Abbildungen 4.4 und 4.5 ist die Streubreite der erzielten Steigerung hingegen sehr ähnlich und die Werte unterscheiden sich im Wesentlichen in der Höhe der erzielten Steigerung.

Zur Überprüfung des Einflusses der Anzahl der Multiagenten-Organisationen auf die erzielte Gesamtkompensation wurde eine Korrelationsanalyse durchgeführt. Tabelle 4.2 stellt das Ergebnis dieser Analyse dar. Die erzielte Kompensation ist somit im untersuchten Bereich wesentlich stärker von der Anzahl an Multiagenten-Organisationen vom Typ *MA-Org-2* abhängig als von der Anzahl an Multiagenten-Organisationen vom Typ *MA-Org-1*.

²³ Zur Darstellung von Boxplots wird in dieser Arbeit die Whisker-Darstellung verwendet mit den dargestellten Werten Minimum, 25%-Perzentil, Median, 75%-Perzentil sowie Maximum.

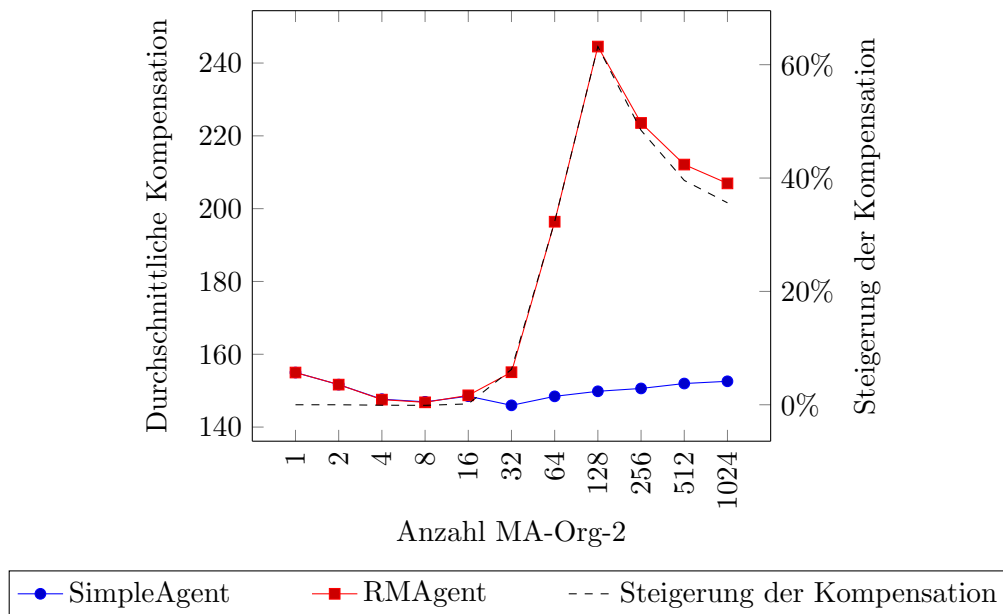


Abbildung 4.3: Erzielte durchschnittliche Kompensation abhängig von der Anzahl von Multiagenten-Organisationen vom Typ 2 bei einer vom Typ 1

Tabelle 4.2: Korrelationsanalyse zum Verhältnis der nachfragenden Multiagenten-Organisationen

Einflussgröße	Kompensationssteigerung
Anzahl MA-Org-1	0,311
Anzahl MA-Org-2	0,612
Beobachtungen	12.100
R^2	0,472
Korrigiertes R^2	0,472

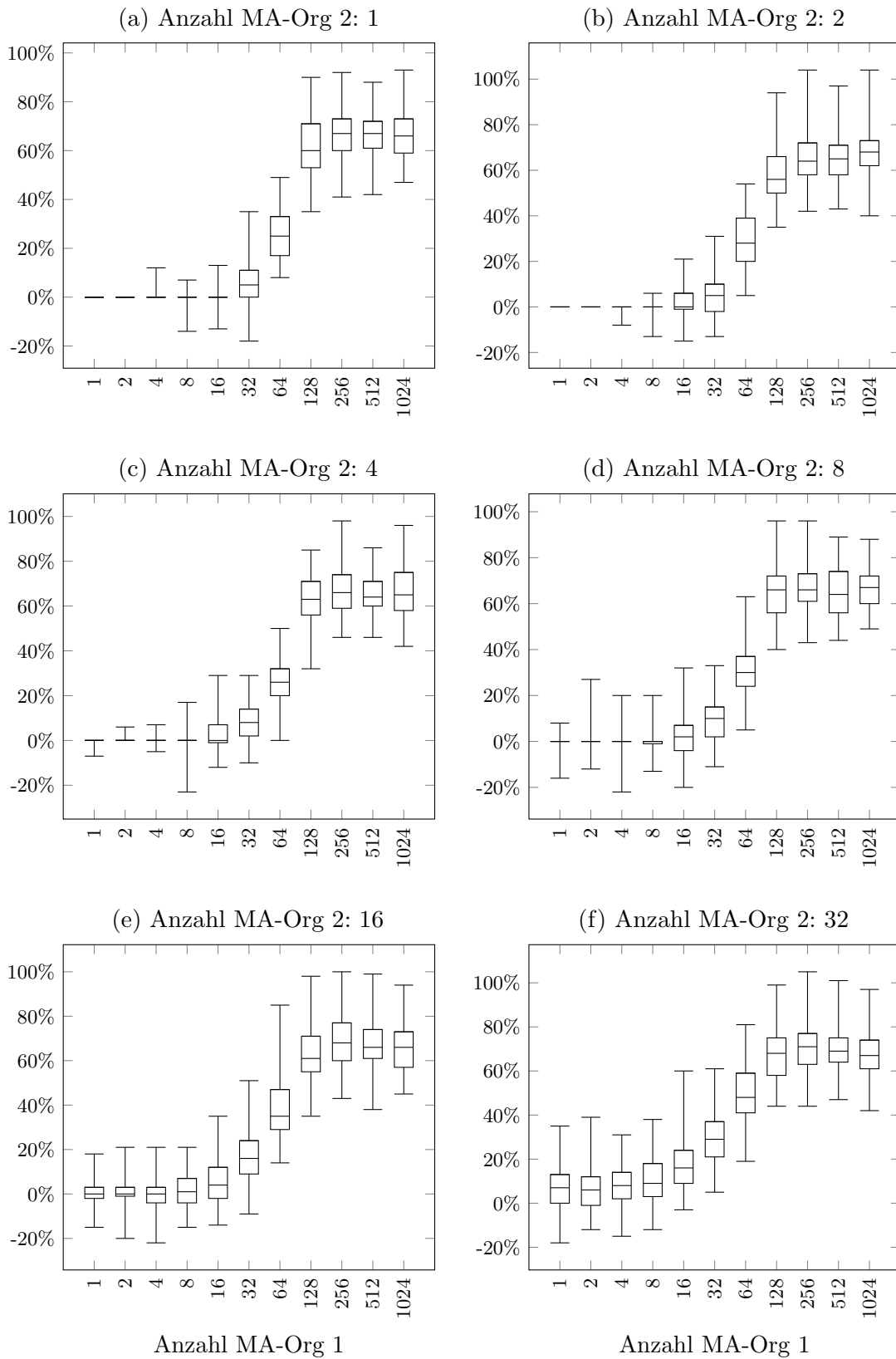


Abbildung 4.4: Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der Anzahl nachfragender Multiagenten-Organisationen vom Typ 1 und 2

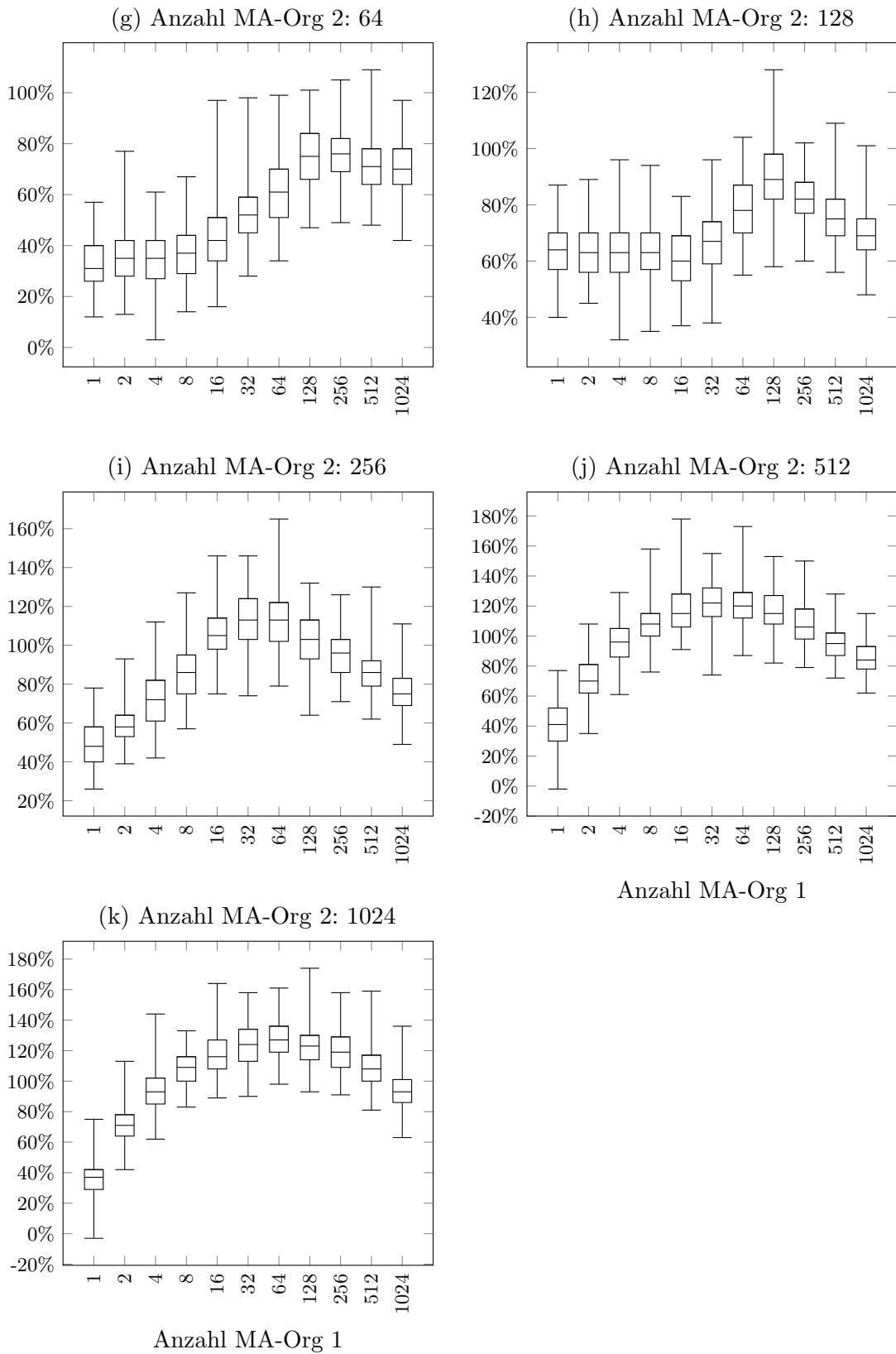


Abbildung 4.5: Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der Anzahl nachfragender Multiagenten-Organisationen vom Typ 1 und 2 (Fortsetzung von Abbildung 4.4)

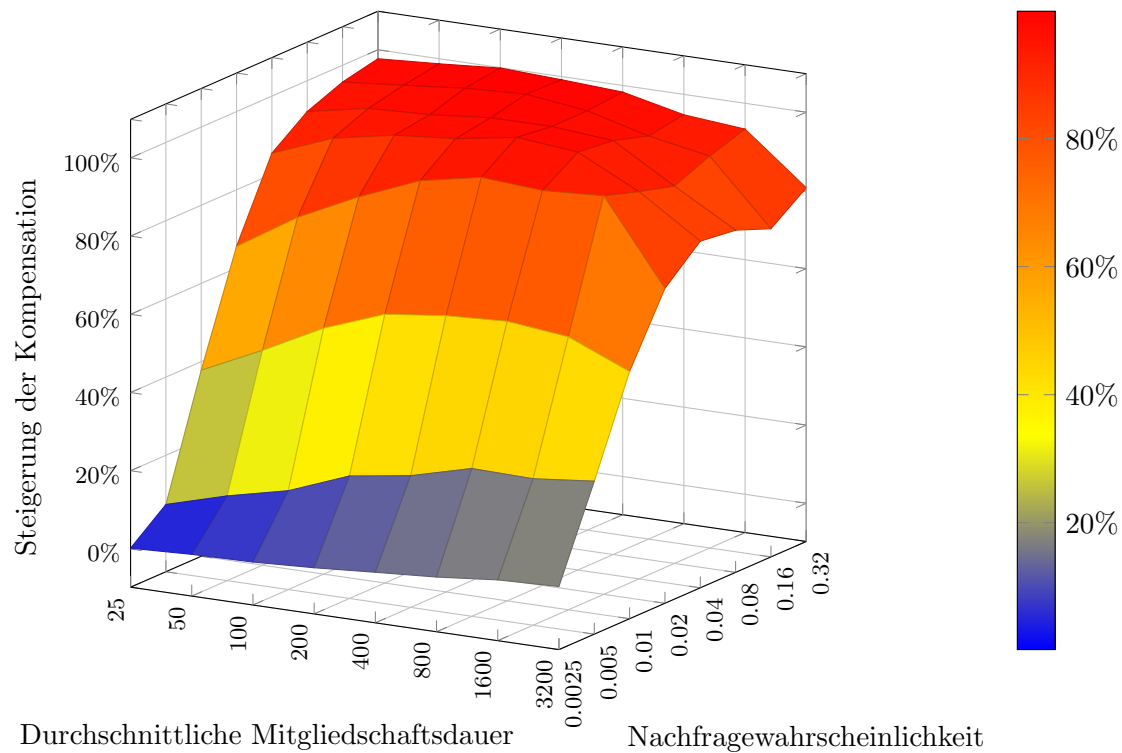


Abbildung 4.6: Steigerung der erzielten Kompensationen unter Veränderung der durchschnittlichen Mitgliedschaftsdauer und der Nachfragewahrscheinlichkeit je Zeiteinheit

4.3.2 Nachfragehäufigkeit und Mitgliedschaftsdauer

Abbildung 4.6 zeigt die Steigerung der erzielten Kompensationen unter Veränderung der durchschnittlichen Mitgliedschaftsdauer (Parameter v .) und der Nachfragewahrscheinlichkeit je Zeiteinheit (Parameter iv .). Dabei liegt die durchschnittlich erzielte Steigerung des entwickelten Verfahrens stets im positiven Bereich, ist jedoch abhängig von den veränderten beiden Parametern.

Abbildung 4.7 zeigt die Ergebnisse als Boxplots. Dabei wird deutlich, dass die Streubreite der erzielten Steigerung der Kompensation von der durchschnittlichen Mitgliedschaftsdauer abhängig ist. Je länger die durchschnittliche Mitgliedschaftsdauer ist, desto höher ist die Streubreite der erzielten Steigerung der Kompensation.

Tabelle 4.3 zeigt die Korrelationsanalyse zur Nachfragewahrscheinlichkeit und Mitgliedschaftsdauer. Augenscheinlich ist die Korrelation zwischen durchschnittlicher Mitgliedschaftsdauer und Steigerung der Kompensation jedoch abhängig von der Nachfragewahrscheinlichkeit (vgl. Abbildung 4.6). Die Korrelation zwischen Vertragslaufzeit und Steigerung der Kompensation bei sehr seltenen Nachfragen (Nachfragewahrscheinlichkeit je Zeiteinheit 0.0025) beträgt 0,273. Somit ist die Korrelation deutlich über der in Tabelle 4.3 angegebenen -0,028 (Signifikanzniveau 0,05). Dabei ist in Abbildung 4.6 ersichtlich, dass jede Verlängerung der Vertragslaufzeit bei niedriger Nachfragewahrscheinlichkeit eine Steigerung der durchschnittlich erzielten Kompensation des *RMAgent* gegenüber dem *SimpleAgent* mit

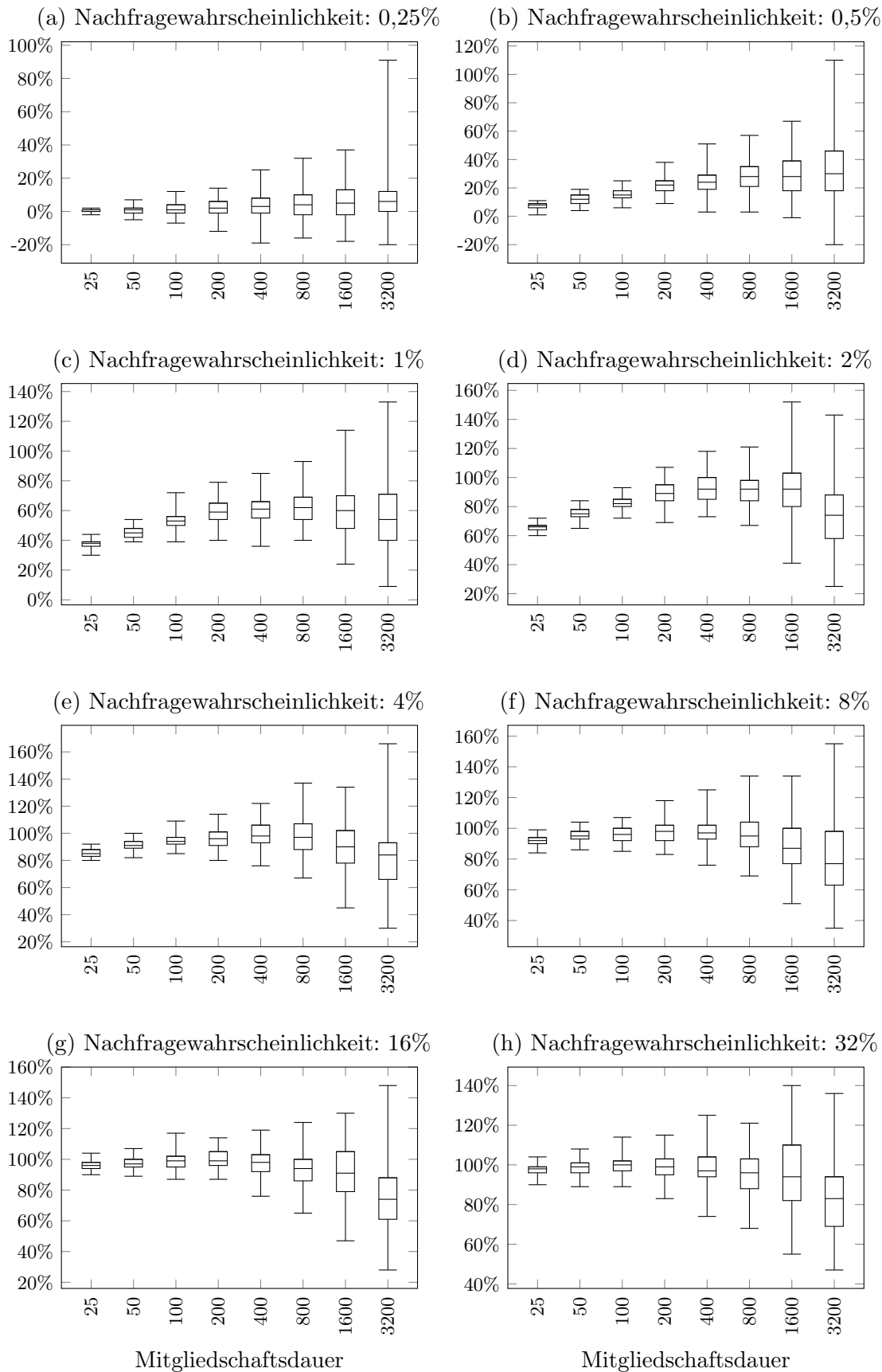


Abbildung 4.7: Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der durchschnittlichen Mitgliedschaftsdauer und der Nachfragewahrscheinlichkeit je Zeiteinheit

sich bringt. Im Gegensatz hierzu fällt bei einer hohen Nachfragewahrscheinlichkeit von mehr als 0,02 und hohen Vertragslaufzeiten der Vorteil des vorgestellten Verfahrens wieder ab.

Tabelle 4.3: Korrelationsanalyse zur Nachfragewahrscheinlichkeit und Mitgliedschaftsdauer

Einflussgröße	Kompensationssteigerung
Nachfragewahrscheinlichkeit	0,536
Mitgliedschaftsdauer	-0.028
Beobachtungen	6.400
R^2	0,289
Korrigiertes R^2	0,288

4.3.3 Ressourcenverfügbarkeit und Nachfragemenge

Abbildung 4.8 zeigt die Ergebnisse zur Untersuchung der Steigerung der erzielten Kompensationen unter Veränderung der durchschnittlichen Dienstmenge pro Anfrage (Parameter vi.) und der Ressourcenkapazität (Parameter vii.). Dabei wird deutlich, dass das vorgestellte Verfahren nicht in allen Parameterkonstellationen einen Vorteil bietet, sondern es auch zu einer Verringerung der erzielten Kompensation gegenüber dem *SimpleAgent* kommen kann. Insbesondere bei sehr hohen Anfragemengen, aber nur sehr geringer zur Verfügung stehender Ressourcenkapazität, wie auch bei sehr hoher Ressourcenkapazität und sehr kleinen Anfragemengen, hat das entwickelte Verfahren negative Werte erzielt.

Abbildung 4.9 zeigt die Ergebnisse als Boxplots. Dabei ist die Tendenz zu beobachten, dass die Streuung der erzielten Steigerung der Kompensation mit steigender Ressourcenkapazität sinkt. Ausnahmen hiervon sind die Werte für eine Ressourcenkapazität von 2,5 bis 5 bzw. 10 und hohen Dienstmengen pro Anfrage (vgl. Abbildung 4.9, (g)-(h)): Der Median sowie die deckungsgleichen 25%- und 75%-Perzentile liegen bei 0% und es existieren nur einzelne Ausreißer.

Tabelle 4.4 zeigt die Korrelationsanalyse des Einflusses der beiden Parameter vi. und vii. (vgl. Tabelle 4.1) auf die Steigerung der erzielten Gesamtkompensation des *RMAgent*. Hierbei sind zwar nur niedrige Korrelationen zu erkennen, jedoch legt Abbildung 4.8 augenscheinlich einen Bezug des Verhältnisses der beiden Parameter zur Steigerung der Kompensation nahe. Eine Korrelationsanalyse zwischen dem Quotienten $\frac{\text{Nachfragemenge}}{\text{Ressourcenkapazität}}$ und der Steigerung der Kompensation ergibt einen Korrelationskoeffizient von -0.434.

4.3.4 Kompensationen

Abbildung 4.10 zeigt die Ergebnisse der Untersuchung zur Steigerung der erzielten Kompensationen unter Veränderung der Kompensation für Dienst 1 (Parameter viii.) sowie des Rabatts für günstigere Dienstgüteklassen (Parameter xi.). Die Abbildung zeigt auf, dass in fast allen Parameterkonstellationen positive Steigerungswerte erzielt werden konnten, jedoch in den drei Konstellationen mit einer Kompensation von 10 GE und einem

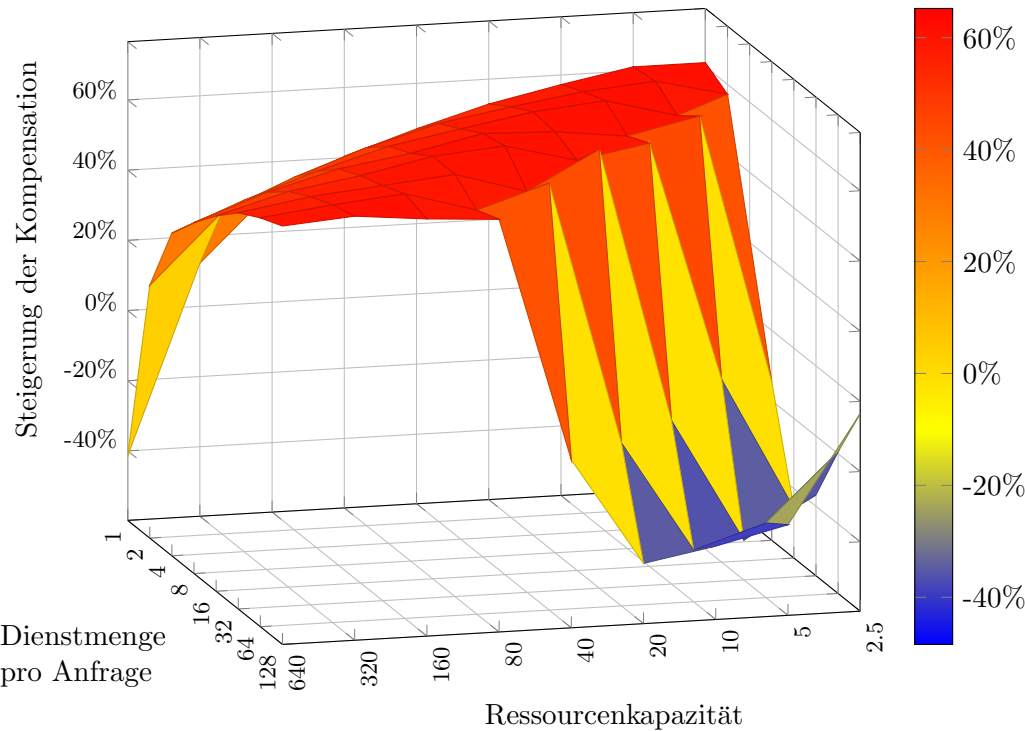


Abbildung 4.8: Steigerung der erzielten Kompensationen unter Veränderung der durchschnittlichen Dienstmenge pro Anfrage und der Ressourcenkapazität

Tabelle 4.4: Korrelationsanalyse zur Ressourcenverfügbarkeit und Nachfragemenge

Einflussgröße	Kompensations- steigerung
Ressourcenkapazität	0,069
Nachfragemenge	-0,267
Beobachtungen	7.200
R^2	0,076
Korrigiertes R^2	0,076

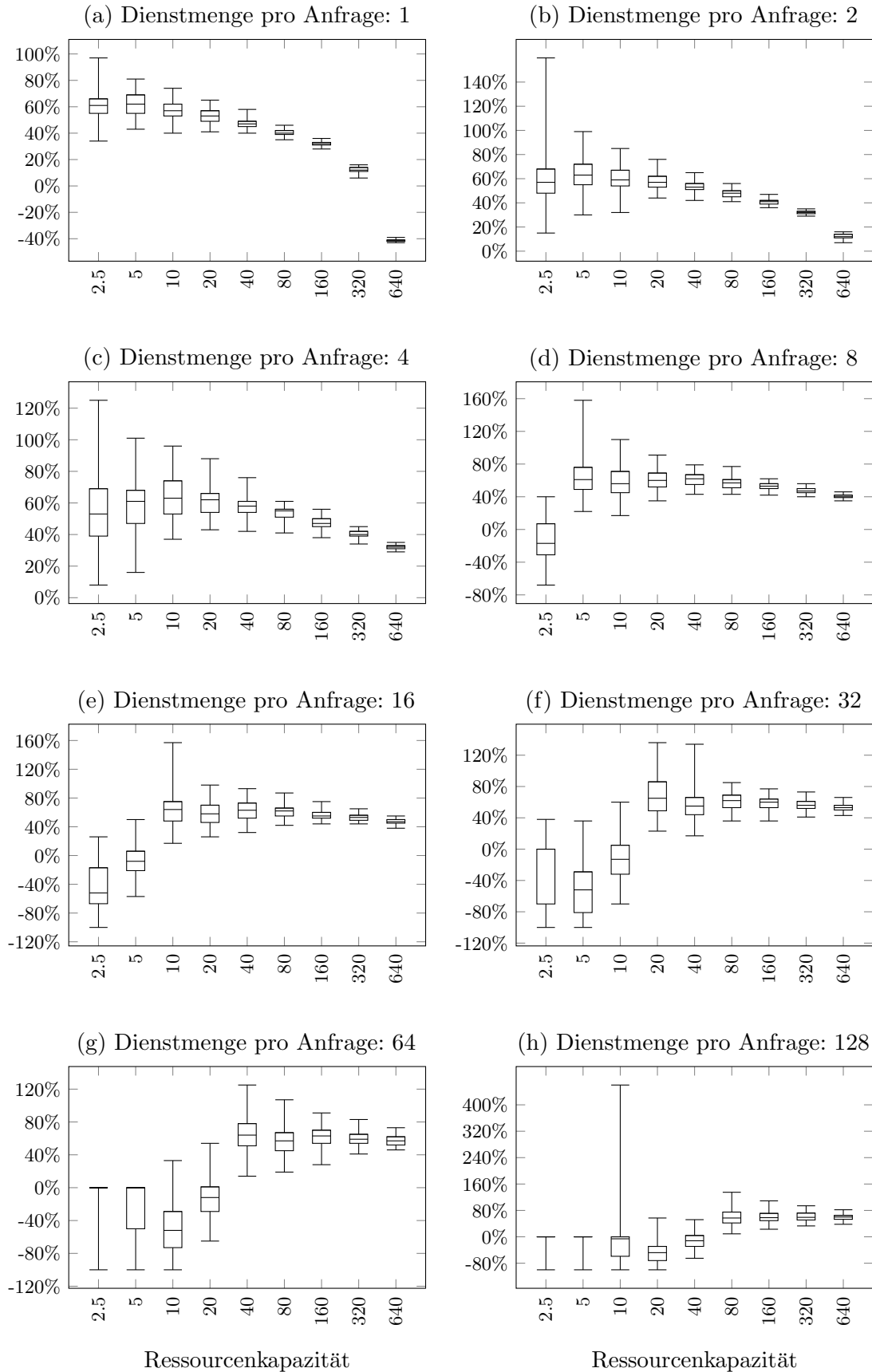


Abbildung 4.9: Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der durchschnittlichen Dienstmenge pro Anfrage und der Ressourcenkapazität

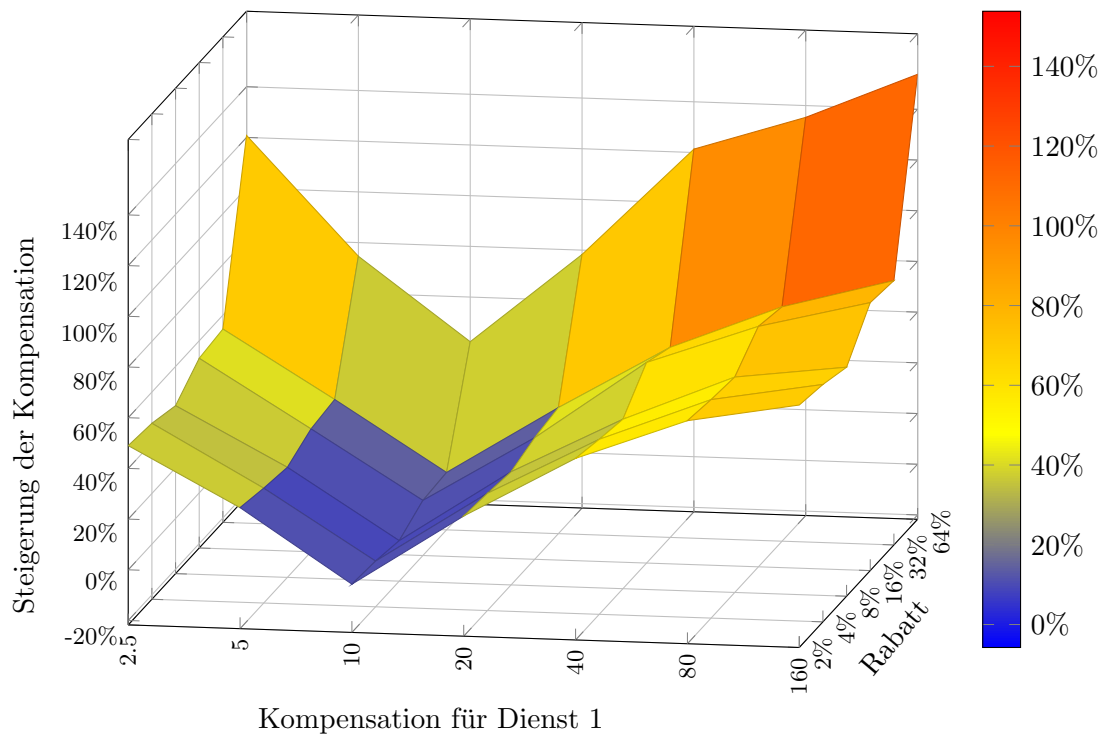


Abbildung 4.10: Steigerung der erzielten Kompensationen unter Veränderung der Kompensation für Dienst 1 sowie des Rabatts für günstigere Dienstgüteklassen

Rabatt je Dienstgüteklasse von 2%, 4% bzw. 8% negative Werte von -2,69%, -3,71% bzw. -5,77% erzielt wurden. Für höhere Rabatte konnte das Verfahren hingegen auch bei einer Kompensation von 10 GE eine nicht-negative Steigerung der Gesamtkompensation erzielen.

Abbildung 4.11 zeigt die Ergebnisse als Boxplots. Dabei wird deutlich, dass die Streuung der erzielten Steigerung der Kompensation abhängig ist von der Kompensation für Dienst 1. Insbesondere sind alle Werte – einschließlich der Ausreißer – im positiven Bereich für Kompensationen ungleich 10 GE.

Tabelle 4.5 zeigt die Korrelationsanalyse zur Kompensationsgestaltung. Beide Einflussparameter weisen dabei eine im Vergleich zu anderen Parametern hohe Korrelation zur erzielten Kompensationssteigerung auf. Aus Abbildung 4.10 wird jedoch augenscheinlich klar, dass die Korrelation zwischen der Kompensation für Dienst 1 und der Kompensationssteigerung bereichsabhängig von der Kompensation ist. Eine abschnittsweise Korrelationsanalyse ergibt eine Korrelation von -0,694 für den Bereich 2,5 bis 10 sowie von 0,620 für den Bereich 10 bis 160.

4.3.5 Dienstgüteklassen und Dienste

Abbildung 4.12 zeigt die Steigerung der erzielten Kompensationen unter Veränderung der Anzahl der angebotenen Dienste (Parameter xii.) und Dienstgüteklassen (Parameter xiii.) als Boxplots. Während im Falle von 2 oder 3 angebotenen Diensten, das Verfahren stets deutliche Steigerungen der Kompensation erzielen konnte, ist bei nur einem angebotenen Dienst die Vorteilhaftigkeit nicht offensichtlich: Bei einer Dienstgüteklasse entsprach das

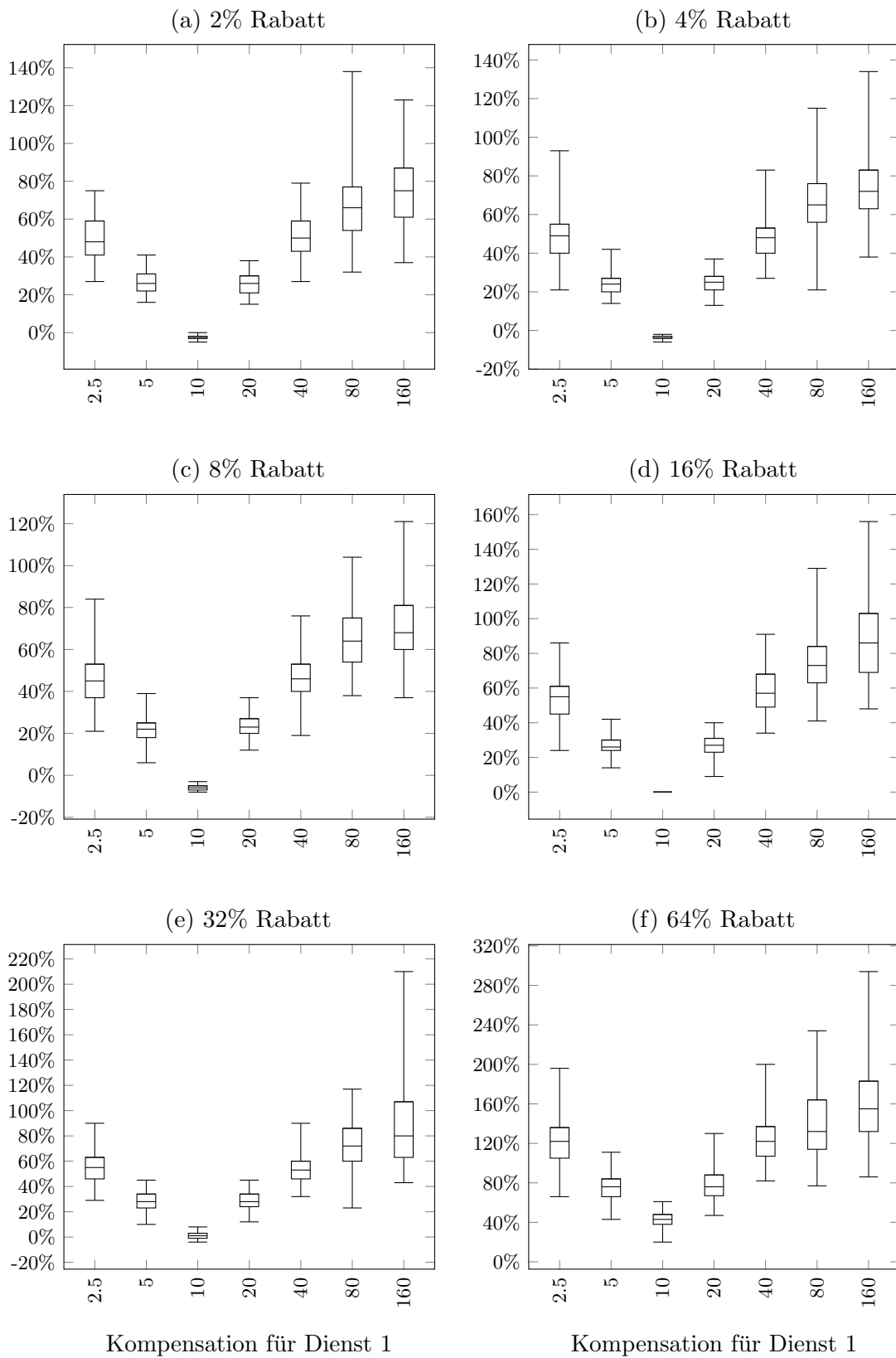
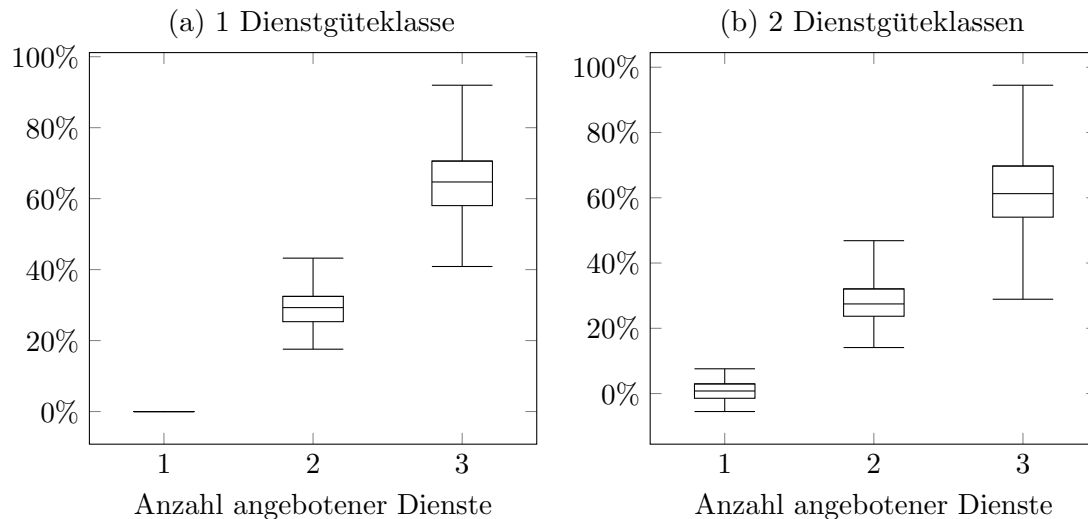


Abbildung 4.11: Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der Kompensation für Dienst 1 sowie des Rabatts für günstigere Dienstgüteklassen

Tabelle 4.5: Korrelationsanalyse zur Kompensationsgestaltung

Einflussgröße	Kompensations- steigerung
Kompensation Dienst 1	0,529
Rabatt je Dienstgüteklasse	0,550
Beobachtungen	4.200
R^2	0,582
Korrigiertes R^2	0,582

**Abbildung 4.12:** Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der Anzahl der angebotenen Dienste und Dienstgüteklassen

Verhalten des *RMAgent* exakt dem des *SimpleAgent*, bei zwei Dienstgüteklassen konnte hingegen im Median ein Vorteil erzielt werden.

Tabelle 4.6 zeigt die Korrelationsanalyse zu Diensten und Dienstgüteklassen. Die Anzahl angebotener Dienste korreliert dabei stark mit der Steigerung der Kompensation. Die Anzahl an Dienstgüteklassen hingegen korreliert nicht signifikant mit dieser Steigerung.

Tabelle 4.6: Korrelationsanalyse zu Diensten und Dienstgüteklassen

Einflussgröße	Kompensations- steigerung
Anzahl Dienste	0,956
Anzahl Dienstgüteklassen	-0,023
Beobachtungen	600
R^2	0,915
Korrigiertes R^2	0,914

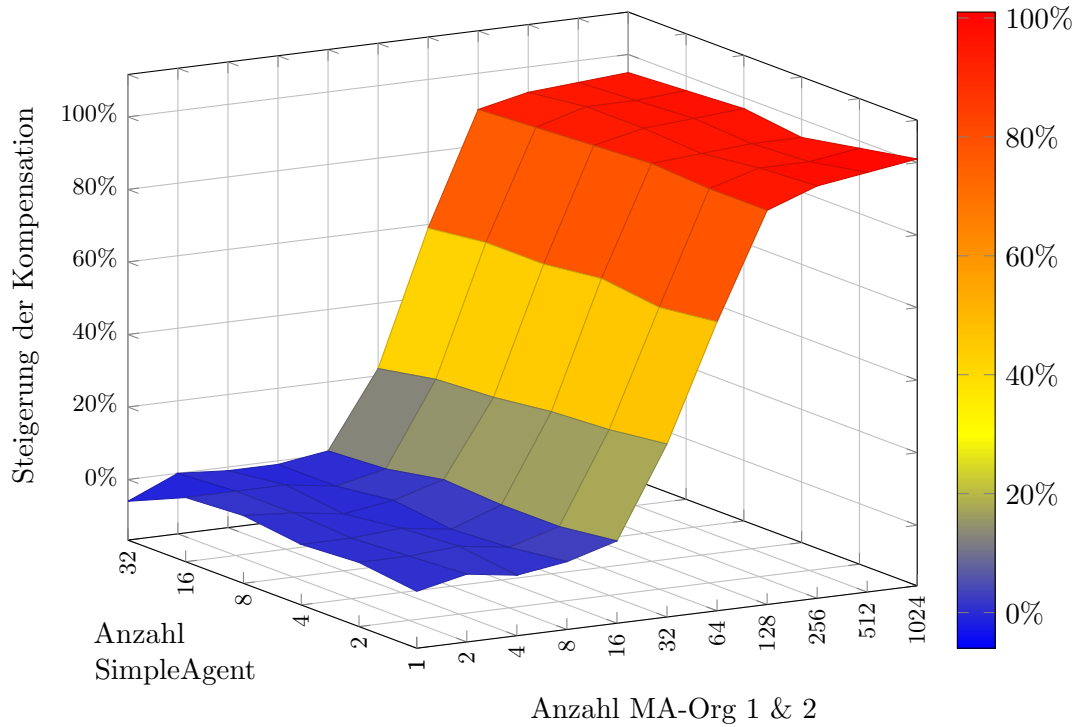


Abbildung 4.13: Steigerung der erzielten Kompensationen unter Veränderung der Anzahl nachfragender Multiagenten-Organisationen und anbietender Softwareagenten vom Typ *SimpleAgent*

4.3.6 Dienstanbieterstruktur

Abbildung 4.13 zeigt die Ergebnisse der Untersuchung zur Steigerung der erzielten Kompensationen des *RMAgent* auf einem duo- bzw. polypolistischen Markt (Parameter iii.) unter Veränderung der Anzahl an Softwareagenten vom Typ *SimpleAgent* (Parameter ii.) sowie der Anzahl an Multiagenten-Organisationen vom Typ 1 und 2 (Parameter i.). Dabei wird die Steigerung in dieser Untersuchung als Steigerung der erzielten Kompensation des Softwareagenten vom Typ *RMAgent* gegenüber dem Durchschnitt der erzielten Kompensation der Softwareagenten vom Typ *SimpleAgent* betrachtet. Während es durch die Abbildung der Mittelwerte in Abbildung 4.13 so scheint, als wäre die erzielte Steigerung annähernd unabhängig von der Anzahl an Softwareagenten vom Typ *SimpleAgent*, wird in der Darstellung der Boxplots in Abbildung 4.14 deutlich, dass die Streuung der erzielten Steigerung sich stark unterscheidet. Augenscheinlich ist bei einer kleinen Anzahl an Softwareagenten vom Typ *SimpleAgent* die Streuung der Steigerung unabhängig von der Anzahl der nachfragenden Multiagenten-Organisationen. Bei 16 bzw. 32 Softwareagenten vom Typ *SimpleAgent* hingegen ist die Streuung der Steigerung bei wenigen Multiagenten-Organisationen wesentlich größer als bei entsprechend vielen.

Tabelle 4.7 zeigt die Korrelationsanalyse zur Dienstanbieterstruktur und unterstützt die oben vorgenommene Einschätzung bezüglich Abbildung 4.14: Während die Steigerung der erzielten Kompensation leicht negativ (Signifikanzniveau 0,05) mit der Anzahl an anderen Softwareagenten korreliert, korreliert die Steigerung deutlich stärker positiv mit der Anzahl an Multiagenten-Organisationen.

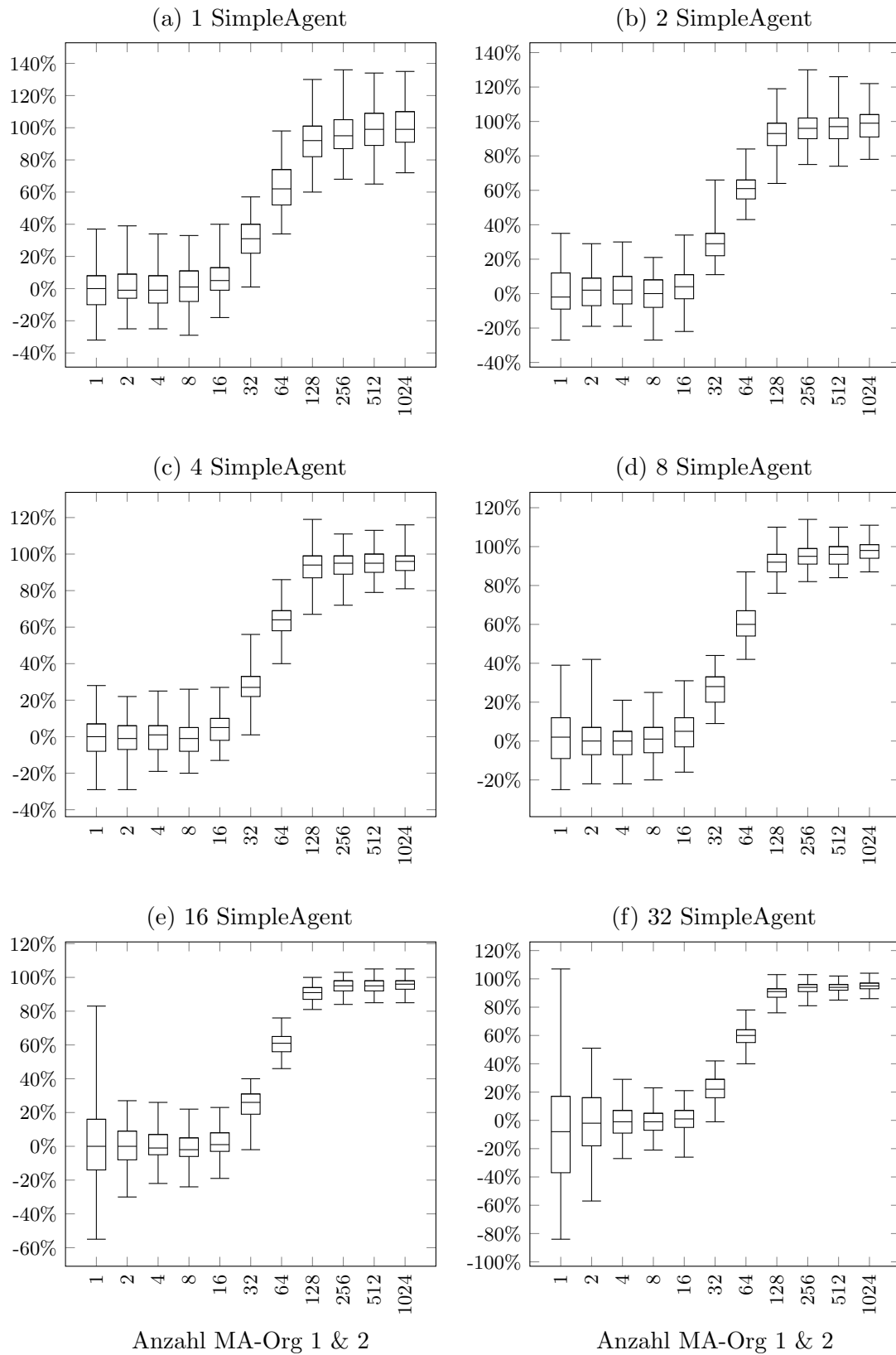


Abbildung 4.14: Boxplot-Darstellung der Steigerung der erzielten Kompensationen unter Veränderung der Anzahl der Softwareagenten vom Typ *SimpleAgent* sowie der Anzahl an Multiagenten-Organisationen vom Typ 1 und 2

Tabelle 4.7: Korrelationsanalyse zur Dienstanbieterstruktur

Einflussgröße	Kompensations- steigerung
Anzahl SimpleAgent	-0,026
Anzahl MA-Org 1 & 2	0,683
Beobachtungen	6.600
R^2	0,467
Korrigiertes R^2	0,466

4.4 Diskussion

Die simulative Untersuchung des entwickelten Verfahrens in Abschnitt 4.3 hat aufgezeigt, dass die Vorteilhaftigkeit des Verfahrens in Bezug auf die Steigerung der erzielten Kompensation insbesondere vom Umfang der durch die Multiagenten-Organisationen generierten Nachfrage nach Diensten und somit nach neuen Mitgliedern abhängig ist. Dies wurde besonders in den Untersuchungen zum *Verhältnis der nachfragenden Multiagenten-Organisationen* (Abschnitt 4.3.1), zur *Nachfragehäufigkeit und Mitgliedschaftsdauer* (Abschnitt 4.3.2) sowie zur *Dienstanbieterstruktur* (Abschnitt 4.3.6) deutlich. So haben beispielsweise die Ergebnisse zur Untersuchung der Dienstanbieterstruktur in Abschnitt 4.3.6 ergeben, dass eine deutliche Korrelation zwischen der Anzahl an nachfragenden Multiagenten-Organisationen und der erzielten Steigerung der Kompensation existiert. Dies ist insbesondere durch die damit verbundene Steigerung der Nachfrage nach Diensten und Mitgliedschaften zu erklären. Ebenso verhält es sich mit der leicht negativen Korrelation mit der Anzahl an konkurrierenden Softwareagenten: Die Nachfrage muss bei mehreren potentiell für eine Multiagenten-Organisation in Frage kommenden Softwareagenten zwischen diesen aufgeteilt werden und verringert sich aus Sicht jedes einzelnen Softwareagenten dementsprechend.

Während zwar die erzielte Kompensation positiv mit der Anzahl an Multiagenten-Organisationen vom Typ 2 korreliert (vgl. Tabelle 4.2), sinkt diese für sehr wenige Multiagenten-Organisationen vom Typ 1 ab einem bestimmten Punkt (vgl. Abbildung 4.3). In diesem Fall ist das entwickelte Verfahren aufgrund der übermäßigen Anzahl an Anfragen mit niederwertiger Dienstgüteklasse nicht in der Lage, die höherwertigen Anfragen herauszufiltern. Genauere Prognosemodelle könnten hier Abhilfe schaffen. Unabhängig von der sinkenden Tendenz der Steigerung in diesem Bereich erzielt das Verfahren eine signifikante Steigerung der erzielten Kompensation gegenüber dem *SimpleAgent*.

Bei der Abhängigkeit zwischen Ressourcenkapazität und Nachfragemenge je Anfrage, kann das entwickelte Verfahren nur eine Steigerung der erzielten Gesamtkompensation realisieren, wenn keine extremen Verhältnisse zwischen den beiden Parametern vorherrschen. Bei einer sehr hohen Kapazität an Ressourcen und gleichzeitig nur kleinen Anfragemengen schützt das Verfahren durch einen zu hohen Bid-Price zu viele Ressourcen für höherwertige Anfragen, die allerdings nicht realisiert werden können. Bei sehr kleiner Kapazität und sehr hohen Anfragemengen ist die Wahrscheinlichkeit, eine Anfrage zu erhalten, die überhaupt bedient werden kann, sehr klein. Durch die hohe Anfragemenge wird ein hoher Bid-Price berechnet

(vgl. Abschnitt 3.3) und ebenfalls die ohnehin nur in geringem Maß vorhandenen Ressourcen in übermäßiger Weise geschützt, sodass Ressourcen ungenutzt verbleiben. Ein ausgewogenes Verhältnis zwischen der Anfragemenge sowie der vorhandenen Ressourcenkapazität ist somit Voraussetzung für die Anwendung des in Abschnitt 3 entwickelten Verfahrens.

Die durchschnittliche Mitgliedschaftsdauer korreliert leicht negativ mit der Steigerung der erzielten Kompensation (vgl. Abschnitt 4.3.2). Dies ist damit zu erklären, dass mit einer längeren Mitgliedschaftsdauer bei gleicher Ressourcenkapazität weniger Mitgliedschaften eingegangen werden können und somit in das Prognosemodell weniger Informationen über die erwartete Dauer der Mitgliedschaft einfließen. Eine möglichst genaue Prognose der zukünftig eingehenden Mitgliedschaften ist jedoch eine Grundvoraussetzung, um eine Vorteilhaftigkeit des Verfahrens zu erzielen. Dies ist ebenfalls die Erklärung für die erhöhte Streubreite der Steigerung der Kompensation bei einer längeren durchschnittlichen Mitgliedschaftsdauer: Werden durch die längere Mitgliedschaftsdauer weniger Mitgliedschaften eingegangen, erhöht sich automatisch die statistische Varianz der erzielten Ergebnisse.

Eine weitere essentielle Voraussetzung, um höhere Kompensationen gegenüber einfacheren Verfahren zu erzielen, ist die Nutzbarkeit von Differenzierung zwischen den angebotenen Diensten. Diese Differenzierung findet sich in der Kompensationsgestaltung wieder: Nur wenn sich die zu realisierenden Kompensationszahlungen oder Ressourcennutzungen zwischen den angebotenen Diensten unterscheiden, kann das Verfahren Vorteile realisieren. Diese Differenzierung kann jedoch alternativ durch die Einführung von Dienstgüteklassen und deren ausreichender Differenzierung erreicht werden (vgl. Abschnitt 4.3.1). Während zwar bei einer fast ausschließlich vorhandenen Nachfrage nach Diensten der niederwertigen Dienstgüteklasse 2 durch die unterschiedliche Bepreisung der verschiedenen Dienste Vorteile realisiert werden können, steigt dieser Vorteil bei einem umgekehrten Verhältnis der Multiagenten-Organisationen vom Typ 1 und 2 nochmals deutlich an.

Abbildung 4.12 macht deutlich, dass die erzielte Gesamtkompensation stark von der Anzahl der angebotenen Dienste abhängig ist. Dabei ist jedoch nicht nur die Anzahl der Dienste entscheidend, sondern insbesondere deren Differenzierung im Verhältnis der jeweils erzielten Kompensation und verursachten Ressourcennutzung. Nur wenn zwischen den angebotenen Diensten, welche für Mitgliedschaftsbeziehungen zu Multiagenten-Organisationen eingesetzt werden, Unterschiede bezüglich der erzielten Kompensation bestehen, kann das entwickelte Verfahren diese Unterschiede zum Vorteil des beitretenden Softwareagenten nutzen. Aus der Untersuchung zur Kompensationszusammensetzung (vgl. Abbildung 4.10) ergibt sich, dass die Steigerung der Kompensation stark mit der Kompensation für Dienst 1 und damit mit dem Verhältnis der Kompensationen für verschiedene Dienste korreliert (die Kompensation für Dienst 2 bleibt konstant). Unabhängig in welche Richtung sich die Kompensation für Dienst 1 von dem für Dienst 2 festgesetzten Wert von 10 GE unterscheidet, ist es dem entwickelten Verfahren möglich, deutliche Steigerung der Kompensation zu erzielen. Nur falls keine Steigerung durch Differenzierung mit Hilfe unterschiedlicher Dienste – Kompensation für Dienst 1 entspricht der für Dienst 2 – und kaum Differenzierung durch Rabatte für Dienstgüteklassen möglich sind, bleibt das Ergebnis des entwickelten Verfahrens knapp hinter dem Referenzwert des *SimpleAgent* zurück.

Die Untersuchung der Konstellation zur Anzahl von Dienstgüteklassen und Diensten bringt ein ähnliches Ergebnis: Je mehr angebotene Dienste – und das damit zusammenhängende höhere Verhältnis zwischen der Kompensation verschiedener Dienste – desto höher fällt die Steigerung der Kompensation aus (vgl. Abbildung 4.12). Den Einfluss der Anzahl der Dienstgüteklassen muss jedoch differenziert betrachtet werden: Während in diesem Fall von einem Rabatt von 32% ausgegangen wurde, offenbarte bereits die Untersuchung zur Zusammensetzung der Kompensation, dass erst bei der höchsten untersuchten Parameterbelegung von 64% eine signifikante Steigerung der Kompensation ausschließlich durch Rabattierung von Dienstgüteklassen erzielt wurde (vgl. Abbildung 4.10).

Die vorangegangene Simulation nimmt an, dass jeder Dienst jeweils eine Einheit jeder Ressource zur Bereitstellung benötigt. Eine Differenzierung der Dienste kann auch bereits automatisch erzielt werden, falls die Bereitstellung verschiedener Dienste unterschiedliche Mengen von Ressourcen benötigen, beispielsweise benötigt Dienst 1 hauptsächlich Rechenleistung, Dienst 2 Speicherplatz. Dabei gewinnt die Bewertung der einzelnen Ressourcen durch Bid-Prices (vgl. Abschnitt 3.4.2) an Bedeutung. Die Komplexität der Beitrittsentscheidung wird weiter erhöht, wenn neben dem unterschiedlichen Ressourcenverbrauch $\mathbf{M}_b$ sich auch die zur Verfügung stehenden Gesamtkapazitäten $\mathbf{x}_0$ unterscheiden. Die Berechnungskomplexität des vorgestellten Bid-Price-Verfahrens ändert sich jedoch nicht und hat somit das Potential weitere Vorteile zu erwirtschaften.

5. Zusammenfassung und Ausblick

Die Ergebnisse dieser Arbeit werden im folgenden Abschnitt 5.1 zusammengefasst. Abschnitt 5.2 schließt diese Arbeit mit einer kritischen Würdigung der erzielten Ergebnisse sowie dem sich hieraus ergebenden weiteren Forschungsbedarf ab.

5.1 Ergebnisse

Diese Arbeit präsentiert ein Verfahren zur Optimierung von Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen. Hierfür wurde zunächst in Kapitel 2 der Stand der Forschung im Forschungsbereich Multiagentensysteme sowie organisations-theoretische Ansätze mit Bezug zur Beitrittsentscheidung beschrieben und analysiert. Das in Kapitel 3 entwickelte Verfahren basiert auf Ansätzen des Revenue Management als Teilbereich des Operations Research. Die in der Literatur vorhandenen Verfahren des Revenue Management sind dabei nicht in der Lage, die grundlegend verschiedenen Gegebenheiten von Beitrittsentscheidungen von Softwareagenten abzubilden. Das in Abschnitt 3.2 entwickelte Revenue Management-basierte Modell der Beitrittsentscheidung versetzt Softwareagenten in die Lage, mit Mitgliedschaften in Multiagenten-Organisationen umzugehen, deren Dauer a priori unbekannt ist, und ihre Mitgliedschaft anhand verschiedener Dienstgüteklassen abzugrenzen. Die auf dieser Basis vorhandenen Möglichkeiten der Optimierung von Beitrittsentscheidungen nutzt das in Abschnitt 3.4 entwickelte Verfahren, das den Ansatz von Ressourcen-bezogenen Reservationspreisen (so genannte *Bid-Prices*) aus dem Revenue Management aufgreift und auf die Gegebenheit von Beitrittsentscheidungen von Softwareagenten zu Multiagenten-Organisationen anpasst.

Das in Kapitel 3 entwickelte Verfahren wird durch ein Simulationsexperiment evaluiert (vgl. Kapitel 4). Dabei stehen zwei Ziele im Vordergrund: Zum einen soll die Evaluation die **Wirksamkeit** des Verfahrens demonstrieren – insbesondere durch den Vergleich mit anderen Arten der Beitrittsentscheidung – und zum anderen soll sie feststellen, unter welchen **Bedingungen** der Einsatz des Verfahrens für den Softwareagenten vorteilhaft ist, da das Verfahren in unterschiedlichen Domänen eingesetzt werden kann. Zur Erreichung

dieser Ziele wird ein einfaches Entscheidungsverfahren als Referenz herangezogen und verschiedene Parameter der Simulation (vgl. Tabelle 4.1) jeweils paarweise variiert. Dabei werden insgesamt 329 verschiedene Parameterkonstellationen untersucht, die in jeweils 100 Simulationsdurchläufen über je 15.000 Zeitabschnitte die Wirksamkeit des Artefakts in Bezug auf die erzielte Kompensation testen.

In der Mehrzahl der untersuchten Parameterkonstellationen erzielt das entwickelte Verfahren eine Steigerung der erwirtschafteten Kompensation. Softwareagenten, die in einer Domäne Dienste anbieten, in der diese Parameterkonstellationen vorzufinden sind, werden durch die Anwendung des Verfahrens in die Lage versetzt, höhere Kompensationen durch Mitgliedschaften in Multiagenten-Organisationen zu erzielen, als dies mit einfachen Entscheidungsverfahren der Fall wäre. Selbst wenn einzelne Parameter a priori dem Softwareagenten nicht bekannt sind, kann mit Hilfe der präsentierten simulativen Evaluation bereits abgeschätzt werden, welches Risiko besteht und inwiefern eine Anwendung des Verfahrens durch den Softwareagenten trotz unbekannter Parameter erwogen werden kann. Das entwickelte Verfahren konnte jedoch nicht für alle Parameterkonstellationen einen Vorteil erwirtschaften, so dass für die Anwendung einige Voraussetzungen zu beachten sind. Diese Voraussetzungen ergeben sich aus den Simulationsergebnissen in Abschnitt 4.3 und setzen sich wie folgt zusammen:

Differenzierungsmöglichkeiten. Grundvoraussetzung für die Anwendung des Verfahrens ist die Möglichkeit der Differenzierung und den damit verbunden Unterschieden in der Höhe der erzielten Kompensation bei gleichem Ressourceneinsatz. Dies ist ebenfalls verbunden mit der in Abschnitt 3.1.2.1 vorgestellten Voraussetzung der *Heterogenität der Kunden* zur Anwendung von Revenue Management Ansätzen. Diese Differenzierung kann durch verschiedene Maßnahmen des Softwareagenten erreicht werden: (i) Für verschiedene Dienste mit gleichem Ressourcenbedarf werden unterschiedlich hohe Kompensationsforderungen gestellt, (ii) ein oder mehrere Dienste werden in unterschiedlichen Dienstgüteklassen angeboten, deren Kompensationen sich in ausreichendem Maße unterscheiden oder (iii) verschiedene Dienste nutzen eine unterschiedliche Menge an Ressourcen. Insbesondere bei unterschiedlichem Ressourcenbedarf der Dienste macht sich die Unterscheidung der jeweils vorhandenen Ressourcenkapazitäten bemerkbar und führt zu einer weiteren Differenzierung zwischen den verfügbaren Ressourcen.

Nachfrage. Das hier vorgestellte Bid-Price-Verfahren kann nur durch Ablehnen von bestimmten Anfragen Vorteile gegenüber anderen Ansätzen generieren. Voraussetzung ist somit eine entsprechende Nachfrage nach den angebotenen Diensten eines Softwareagenten und somit nach dessen Mitgliedschaft in Multiagenten-Organisationen. Steht diese Nachfrage nicht in ausreichendem Maße zur Verfügung, kann im Allgemeinen kein Vorteil gegenüber anderen Ansätzen erzielt werden. Abhängig von der individuellen Situation an angebotenen Diensten wirkt sich auch ein bestimmtes Verhältnis an nachfragenden Multiagenten-Organisationen positiv auf das Ergebnis des vorgestellten Verfahrens aus: Ist die überwiegende Zahl der Anfragen auf niederwertige Dienste oder Dienstgüteklassen ausgerichtet, jedoch auch eine ausreichend

hohe Zahl an höherwertigen Anfragen vorhanden, erzielt das Verfahren deutliche Steigerungen gegenüber dem Referenzverfahren.

Die voran genannten Voraussetzungen haben sich in der simulativen Evaluation als wesentlich für eine Vorteilhaftigkeit der Anwendung des entwickelten Verfahrens herausgestellt. Wenden Softwareagenten das entwickelte Verfahren für Beitrittsentscheidungen zu Multiagenten-Organisationen in Domänen an, die diese Voraussetzungen erfüllen, ist eine Steigerung der erwirtschafteten Kompensation gegenüber einfachen Verfahren wahrscheinlich. Falls beispielsweise die Nachfrage kleiner als erwartet ausfallen sollte, erzielt das Verfahren häufig die gleiche Kompensation wie das Referenzverfahren und erwirtschaftet nur in einzelnen Fällen deutlich weniger, so dass selbst bei a priori unbekannten oder unsichereren Parametern eine Anwendung des entwickelten Verfahrens möglich ist.

5.2 Kritische Würdigung und weiterer Forschungsbedarf

Die oben präsentierten Ergebnisse zeigen, dass das entwickelte Verfahren in der Mehrzahl der Parameterkonstellationen Vorteile gegenüber dem Referenzverfahren erwirtschaften konnte. Die potentielle Steigerung der erzielten Kompensation ist dabei in starkem Maße von dieser Parameterkonstellation abhängig. Das in Abschnitt 3.2 entwickelte Revenue Management-basierte Modell der Beitrittsentscheidung schafft dabei die Voraussetzung zur Anwendung des Verfahrens. Softwareagenten, die die oben genannte Voraussetzung der Differenzierungsmöglichkeit in ihrer Domäne nicht unmittelbar vorfinden, werden bereits durch dieses Modell in die Lage versetzt, ihre angebotenen Dienste beispielsweise durch verschiedene Dienstgüteklassen zu erweitern. Diese Erweiterung ist jedoch ebenfalls abhängig von der Domäne und so kann es sein, dass eine geringere Dienstgüte in manchen Domänen keine Akzeptanz auf Seiten der Multiagenten-Organisationen findet. Wie oben aufgezeigt, ist die Unterscheidung in Dienstgüteklassen jedoch nicht die einzige Möglichkeit der Differenzierung und somit keine direkte Voraussetzung für die Anwendung des entwickelten Verfahrens (beispielsweise durch unterschiedliche Kompensationsforderungen bei mehreren Diensten).

Bei der Entwicklung des Verfahrens wurde angenommen, dass separate Ressourcen für die Anwendung des Verfahrens zum Einsatz kommen (vgl. Abschnitt 3.1.2). Oftmals wird jedoch die gleiche dem Softwareagenten zur Verfügung stehende Infrastruktur – Prozessor, Hauptspeicher, Speicherplatz – für die Anwendung des Verfahrens als auch für die Bereitstellung von Diensten im Rahmen von Mitgliedschaften zur Anwendung kommen. Dies erhöht die Komplexität des Verfahrens, da die Berechnung des Bid-Prices und dessen Anwendung selbst die zur Dienstbereitstellung benötigten Ressourcen belegt und somit die Möglichkeit zur Erwirtschaftung von Kompensationen schmälert. Ebenfalls wird angenommen, dass Gewinn nur durch Kompensationen im Rahmen von Mitgliedschaften in Multiagenten-Organisationen erwirtschaftet werden kann (vgl. Abschnitt 4.2). Ein Einsatz der Ressourcen für Dienstleistungen, die nicht in einem organisatorischen Kontext stehen, ist jedoch ebenso möglich. Hierdurch entsteht für das entwickelte Verfahren weiteres Potential, da bei einer Ablehnung einer Mitgliedschaftsanfrage die vorhandenen Ressourcen für kurzfristige und weniger verbindliche Kooperationen genutzt und somit die Ablehnung durch diese kurzfristigen Dienstbereitstellungen an Dritte kompensiert werden kann.

Das in Abschnitt 3.2 vorgestellte Modell der Beitrittsentscheidung geht davon aus, dass Multiagenten-Organisationen mit Mitgliedschaftsanfragen auf Softwareagenten zugehen, die ihnen bereits bekannt sind. Das Modell sowie das entwickelte Verfahren ist jedoch auf alternative Verhandlungsabläufe anwendbar bzw. anpassbar: (i) Der Softwareagent bietet seine Dienste aus eigener Initiative einer ihm bekannten Multiagenten-Organisation an. In diesem Fall könnte das entwickelte Verfahren direkt angewandt werden, da die Entscheidung, welcher Dienst aus der zeitpunktbezogenen Sicht den höchsten Nutzen abwirft, aus dem Bid-Price-Vektor π_t abgelesen werden kann. Die Prognose der Nachfrage hingegen muss grundlegend angepasst werden, welche jedoch nicht zentraler Bestandteil des entwickelten Verfahrens ist. (ii) Multiagenten-Organisationen fragen Dienste beim Softwareagenten an und nennen bereits bei Anfrage die zu erhaltende Kompensation. Hier kann der Softwareagent ebenfalls das entwickelte Verfahren unverändert einsetzen, da der Bid-Price-Vektor π_t die Bewertung jeder eintreffenden Anfrage ermöglicht. (iii) Multiagenten-Organisationen fragen Dienste beim Softwareagenten an und dieser nutzt den Bid-Price-Vektor zur Bestimmung der geforderten Kompensationszahlung. Während das entwickelte Verfahren von vorberechneten Kompensationsforderungen je Dienst und Dienstgütekategorie ausgeht, ist es ebenfalls möglich, mit dem Bid-Price-Verfahren ebendiese zu bestimmen. Da es sich bei Bid-Prices um Reservationspreise handelt (Talluri und van Ryzin, 2004, S. 89), sind sie ein direkter Anhaltspunkt für die Bewertung einer Dienstbereitstellung und somit der Mitgliedschaft in einer Multiagenten-Organisation. Vorgegebene Kompensationsforderungen sind jedoch eine zentrale Annahme, deren Änderung tiefgreifende Anpassungen an dem entwickelten Verfahren benötigen würden. Neben diesen unabhängig nebeneinander stehenden Verhandlungsabläufen sind auch jegliche Mischungen daraus denkbar, was die Komplexität der Entscheidung selbst weiter erhöht.

Zahlreiche Einflussfaktoren sind von der Domäne abhängig, in der ein Softwareagent mit Multiagenten-Organisationen verhandelt. Eine Anpassung an die Domäne ist unter anderem durch den Korrekturfaktor β_t^r und insbesondere auch durch spezifische Prognoseverfahren möglich. Der Einsatz lernender Verfahren verspricht hier weitere Vorteile für das entwickelte Verfahren (Bao et al., 2013, 203 f.) und bedarf einer domänenspezifischen Entwicklung.

Literaturverzeichnis

- Acker, H. B. (1969). Stelle. Siehe Grochla (1969b), S. 1577–1582.
- Anandasivam, A. und M. Premm (2009). Bid price control and dynamic pricing in clouds. In: *17th European Conference on Information Systems (ECIS)*, Verona, S. 328–341.
- Andersson, M. R. und T. W. Sandholm (1998). Leveled Commitment Contracting among Myopic Individually Rational Agents. In: *Proceedings of the 3rd International Conference on Multi Agent Systems (ICMAS)*, Paris.
- Andersson, M. R. und T. W. Sandholm (2001). Leveled commitment contracts with myopic and strategic agents. *Journal of Economic Dynamics & Control* 25, S. 615–640.
- Anhalt, C. (2011). *Bezahlungssysteme für Mobile Software-Agenten: Eine an der doppelten Rechnungstheorie orientierte Gestaltungsempfehlung*. Dissertation, Universität Hohenheim.
- Arrow, K. J. (1970). *Essays in the theory of risk-bearing*. Amsterdam: North-Holland.
- Bala, R. und S. Carr (2010). Usage-based pricing of software services under competition. *Journal of Revenue and Pricing Management* 9(3), S. 204–216.
- Banks, J. (1998). Principles of Simulation Modeling. In: J. Banks (Hrsg.), *Handbook of Simulation – Principles, Methodology, Advances, Applications, and Practice*, Kapitel 1, S. 3–30. New York: John Wiley & Sons.
- Bao, T., J. Duffy und C. Hommes (2013). Learning, forecasting and optimizing: An experimental study. *European Economic Review* 61, S. 186–204.
- Barnard, C. I. (1938). *The Functions of the Executive*. Cambridge: Harvard University Press.
- Barney, J. (1991). Firm Resources and Sustained Competitive Advantage. *Journal of Management* 17(1), S. 99–120.
- Bea, F. X. (2004). Ziele und Zielkonflikte. Siehe Schreyögg (2004a), S. 1674–1680.
- Bea, F. X. und E. Göbel (2002). *Organisation: Theorie und Gestaltung* (2. Aufl.). Stuttgart: Lucius und Lucius.

- Beavers, G. und H. Hexmoor (2001). Teams of agents. *2001 IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace 1*, S. 574–582.
- Bellman, R. (1957). *Dynamic Programming*. Princeton, New Jersey: Princeton University Press.
- Bleicher, K. (1980a). Kompetenz. Siehe Grochla (1980), S. 1056–1064.
- Bleicher, K. (1980b). Verantwortung. Siehe Grochla (1980), S. 2283–2292.
- Bond, A. H. und L. Gasser (1988). *Readings in Distributed Artificial Intelligence*. San Mateo, CA: Morgan Kaufmann Publishers.
- Botimer, T. und P. Belobaba (1999). Airline pricing and fare product differentiation: A new theoretical framework. *Journal of Operational Research Society* 50, S. 1085–1097.
- Bratman, M. E., D. J. Israel und M. E. Pollack (1988). Plans and resource-bounded practical reasoning. *Computational Intelligence* 4(3), S. 349–355.
- Bresser, R. K. (2004). Ressourcenbasierter Ansatz. Siehe Schreyögg (2004a), S. 1269–1279.
- Bronner, R. (1992). Verantwortung. Siehe Frese (1992), S. 2503–2513.
- Brooks, C. und E. Durfee (2003). Congregation formation in multiagent systems. *Autonomous Agents and Multiagent Systems* 7(1-2), S. 145–170.
- Bruhn, M. (2014). Determinanten des Kaufverhaltens im Dienstleistungsbereich. In: T. Tomczak und W. Heidig (Hrsg.), *Revenue Management aus der Kundenperspektive - Grundlagen, Problemfelder und Lösungsstrategien*, Wiesbaden. Springer.
- Castelfranchi, C. (1995). Guarantees for Autonomy in Cognitive Agent Architecture. In: M. Wooldridge und N. Jennings (Hrsg.), *Intelligent Agents: Theories, Architectures, and Languages*, Volume 890 of *Lecture Notes in Artificial Intelligence*, S. 56–70. Springer.
- Chen, L. und S. M. Alexander (2010). Capacity-driven pricing mechanism in special service industries. *Journal of Revenue and Pricing Management* 9(3), S. 260–275.
- Chiang, W.-C., J. Chen und X. Xu (2007). An overview of research on revenue management: current issues and future research. *International Journal of Revenue Management* 1(1), S. 97–128.
- Coase, R. H. (1937). The Nature of the Firm. *Economica* 4(16), S. 386–405.
- Coase, R. H. (1991). The Nature of the Firm: Meaning. In: O. E. Williamson und S. G. Winter (Hrsg.), *The Nature of the Firm*, Kapitel 4, S. 48–60. New York: Oxford University Press.
- Cyert, R. M. und J. G. March (1963). *A Behavioral Theory of the Firm*. Blackwell.

- Dastani, M., V. Dignum und F. Dignum (2003). Role-assignment in open agent societies. In: *Proceedings of the second international joint conference on Autonomous agents and multiagent systems (AAMAS)*, New York. ACM Press.
- Demsetz, H. (1976). Toward a Theory of Property Rights. *The American Economic Review* 57(2), S. 347–359.
- Dignum, V. (2004). *A Model for Organizational Interaction: Based on Agents, Founded in Logic*. Dissertation, Utrecht University.
- Dignum, V. und J. Padget (2013). Multiagent Organizations. Siehe Weiss (2013), Kapitel Multiagent Organizations, S. 51–97.
- Diller, H. (2008). *Preispolitik*, Volume 4. Stuttgart: W. Kohlhammer.
- Dudenredaktion (Hrsg.) (2003). *Duden Informatik - Ein Fachlexikon für Studium und Praxis*. Dudenverlag.
- Dudenredaktion (Hrsg.) (2006). *Duden - Das Fremdwörterbuch* (9. Aufl.). Mannheim: Dudenverlag.
- Durfee, E. und J. Rosenschein (1994). Distributed problem solving and multi-agent systems: Comparisons and examples. Forschungsbericht, University of Michigan.
- Durfee, E. H., V. R. Lesser und D. D. Corkill (1987). Coherent Cooperation Among Communicating Problem Solvers. *IEEE Transactions on Computers C-36*, S. 1275–1291.
- Durfee, E. H., V. R. Lesser und D. D. Corkill (1989). Trends in cooperative distributed problem solving. *IEEE Transactions on Knowledge and Data Engineering* 1(1), S. 63–83.
- Eisenführ, F., M. Weber und T. Langer (2010). *Rationales Entscheiden* (5. Aufl.). Heidelberg: Springer.
- Erlei, M., M. Leschke und D. Sauerland (2007). *Neue Institutionenökonomik* (2. Aufl.). Stuttgart: Schäffer Poeschel.
- Erman, L. D., F. Hayes-Roth, V. R. Lesser und D. R. Reddy (1980). The Hearsay-II Speech-Understanding System: Integrating Knowledge to Resolve Uncertainty. *ACM Computing Surveys* 12(2), S. 213–253.
- Fama, E. F. und M. C. Jensen (1983). Separation of Ownership and Control. *Journal of Law and Economics* 26(2), S. 301–325.
- Ferber, J. (2001). *Multitagentensysteme*. München: Addison-Wesley.
- Ferber, J. und A. Droghoul (1992). Using Reactive Multi-Agent Systems in Simulation and Problem Solving. In: N. M. Avouris und L. Gasser (Hrsg.), *Distributed Artificial Intelligence: Theory and Praxis*. Kluwer Academic Publishers.

- Ferber, J. und O. Gutknecht (1998). A meta-model for the analysis and design of organizations in multiagent systems. In: *Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS-98)*.
- Ferber, J., O. Gutknecht und F. Michel (2004). From Agents to Organizations: An Organizational View of Multi-agent Systems. In: P. Giorgini, J. P. Müller, und J. Odell (Hrsg.), *Agent-Oriented Software Engineering IV, 4th International Workshop, AOSE 2003, Melbourne, Australia, July 15, 2003, Revised Papers*, Volume 2935 of *Lecture Notes in Computer Science*.
- Ferstl, O. K. und E. J. Sinz (1984). *Software-Konzepte der Wirtschaftsinformatik*. Berlin: DeGruyter.
- Ferstl, O. K. und E. J. Sinz (2008). *Grundlagen der Wirtschaftsinformatik* (6. Aufl.). München: Oldenbourg.
- FIPA (2002). FIPA Contract Net Interaction Protocol Specification. Foundation for Intelligent Physical Agents.
- Fox, M. S. (1981, Januar). An organizational view of distributed systems. *IEEE Transactions on Systems, Man, and Cybernetics* 11(1), S. 70–80.
- Franklin, S. und A. Graesser (1997). Is It an agent, or just a program?: A taxonomy for autonomous agents. In: J. Müller, M. Wooldridge, und N. Jennings (Hrsg.), *Intelligent Agents III Agent Theories, Architectures, and Languages*, Volume 1193 of *Lecture Notes in Computer Science*, S. 21–35. Springer Berlin Heidelberg.
- Frese, E. (Hrsg.) (1992). *Handwörterbuch der Organisation* (3. Aufl.). Stuttgart: C.E. Poeschel Verlag.
- Frese, E., M. Graumann und L. Theuvsen (2012). *Grundlagen der Organisation - Entscheidungsorientiertes Konzept der Organisationsgestaltung* (10. Aufl.). Wiesbaden: Gabler.
- Gabor, A. und C. Granger (1966). Price as an Indicator of Quality: Report on an Enquiry. *Economica* 33(129), S. 43–70.
- Gasser, L. (2001). Perspectives on organizations in multi-agent systems. *Multi-Agents Systems and Applications 2006*, S. 1–16.
- Genesereth, M. R. und S. P. Ketchpel (1994). Software agents. *Communications of the ACM* 37(7), S. 48–53.
- Georgeff, M. P. und F. F. Ingrand (1989). Decision-Making in an Embedded Reasoning System. In: *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, S. 972–978.
- Gregor, S. und A. R. Hevner (2013). Positioning and Presenting Design Science Research for Maximum Impact. *MIS Quarterly* 37(2), S. 337–355.

- Griffiths, N. und M. Luck (2003). Coalition formation through motivation and trust. In: *Proceedings of the 2nd International Conference on Autonomous Agents and Multiagent Systems*.
- Grochla, E. (1966). *Automation und Organisation*. Wiesbaden: Gabler.
- Grochla, E. (1969a). Automation. Siehe Grochla (1969b), S. 249–257.
- Grochla, E. (Hrsg.) (1969b). *Handwörterbuch der Organisation* (1. Aufl.). Stuttgart: C.E. Poeschel Verlag.
- Grochla, E. (1969c). Technik und Organisation. Siehe Grochla (1969b), S. 1636–1648.
- Grochla, E. (1972). *Unternehmensorganisation*. Reinbek: Rowohlt.
- Grochla, E. (Hrsg.) (1980). *Handwörterbuch der Organisation* (2. Aufl.). Stuttgart: C.E. Poeschel Verlag.
- Gutenberg, E. (1983). *Grundlagen der Betriebswirtschaftslehre*. Heidelberg: Springer.
- Hall, R. (1992). The Strategic Analysis of Intangible Issues. *Strategic Management Journal* 13(2), S. 135–144.
- Hauschildt, J. (1969). Verantwortung. Siehe Grochla (1969b), S. 1693–1702.
- Heidhardt, F. (1980). Soziale und sozio-technische Systeme. Siehe Grochla (1980), S. 2077–2087.
- Hevner, A. R., S. T. March, J. Park und S. Ram (2004). Design Science in Information Systems Research. *MIS Quarterly* 28(1), S. 75–105.
- Hewitt, C. (1986). Offices Are Open Systems. *ACM Transactions on Office Information Systems* 4(3), S. 271–287.
- Hill, W., R. Fehlbaum und P. Ulrich (1994). *Organisationslehre* (5. Aufl.). Berlin: Paul Haupt.
- Hoffmann, F. (1980). Begriff der Organisation. Siehe Grochla (1980), S. 2283–2292.
- Horling, B. und V. Lesser (2004). A Survey of multiagent Organizational Paradigms. *The Knowledge Engineering Review* 19:4, S. 281–316.
- Huhns, M. N. und L. M. Stephens (1999). Multiagent Systems and Societies of Agents. In: G. Weiss (Hrsg.), *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, S. 79–120. MIT Press.
- Huynh, T. D., N. R. Jennings und N. R. Shadbolt (2006). An integrated trust and reputation model for open multi-agent systems. *Autonomous Agents and Multi-Agent Systems* 13(2), S. 119–154.
- Irle, M. (1969). Soziale Systeme. Siehe Grochla (1969b), S. 1505–1509.

- Jennings, N. (2000). On Agent-Based Software Engineering. *Artificial Intelligence* 117, S. 277–296.
- Jennings, N. R. (1995). Controlling cooperative problem solving in industrial multi-agent systems using joint intentions. *Artificial Intelligence* 75, S. 195–240.
- Jensen, M. C. und W. H. Meckling (1976). Theory of the Firm: Managerial Behavior, Agency Costs and Ownership Structure. *Journal of Financial Economics* 3(4), S. 305–360.
- Karänke, P. (2014). *Multiagent resource allocation in service networks*. Dissertation, Universität Hohenheim.
- Karänke, P. und S. Kirn (2010). A Multi-tier negotiation protocol for logistics service chains. In: *18th European Conference on Information Systems*.
- Ketchpel, S. P. (1993). Coalition Formation Among Autonomous Agents. In: *Proceedings of the 5th European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, S. 73–88.
- Kieser, A. und H. Kubicek (1992). *Organisation* (3. Aufl.). Berlin: Walter de Gruyter.
- Kieser, A. und P. Walgenbach (2007). *Organisation* (5. Aufl.). Stuttgart: Schäffer-Poeschel Verlag.
- Kirn, S. (1996a). Gestaltung von Multiagenten-Systemen: Ein organisationszentrierter Ansatz. Habilitation. Westfälischen Wilhelms-Universität Münster.
- Kirn, S. (1996b). Integration von Organisation und Informationssystem: Benötigen wir eine Re-Vitalisierung des maschinellen Aufgabenträgers? Arbeitsbericht, Technische Universität Ilmenau.
- Kirn, S. (2002). Kooperierende intelligente Softwareagenten. *Wirtschaftsinformatik* 44(1), S. 53–63.
- Kirn, S. (2006). Flexibility of Multiagent Systems. Siehe Kirn et al. (2006), Kapitel 3, S. 53–69.
- Kirn, S. und L. Gasser (1998). Organizational Approaches to Coordination in Multi-Agent Systems. *it + ti, Themenheft „Intelligente Agenten“* 4, S. 23–29.
- Kirn, S., O. Herzog, P. Lockemann und O. Spaniol (Hrsg.) (2006). *Multiagent Engineering*. Heidelberg: Springer.
- Kirn, S. und C. Müller-Hengstenberg (2015). Technische und rechtliche Betrachtungen zur Autonomie kooperativ-intelligenter Softwareagenten. *KI - Künstliche Intelligenz* 29(1), S. 59–74.
- Kirsch, W. (1971). *Entscheidungsprozesse - Dritter Band*. Wiesbaden: Gabler.
- Klett, G. (1989). *Kooperierende Expertensysteme mit Kontraktnetzarchitektur und ihr Einsatz in technischen Anlagen*. Dissertation, Fernuniversität Hagen.

- Klusch, M. und A. Gerber (2002). Dynamic coalition formation among rational agents. *IEEE Intelligent Systems* 17(3), S. 42–47.
- Kosiol, E. (1962). *Organisation und Unternehmung*. Wiesbaden: Gabler.
- Kosiol, E. (1969a). Aufgabenanalyse. Siehe Grochla (1969b), S. 199–212.
- Kosiol, E. (1969b). Aufgabenträger. Siehe Grochla (1969b), S. 232–236.
- Laux, H. (1979). *Grundfragen der Organisation - Delegation, Anreiz und Kontrolle*. Heidelberg: Springer.
- Laux, H. (2005a). *Entscheidungstheorie* (6. Aufl.). Heidelberg: Springer.
- Laux, H. (2005b). *Grundlagen der Organisation* (6. Aufl.). Heidelberg: Springer.
- Lesser, V. R. und L. D. Erman (1988). Distributed Interpretation: A Model and Experiment. In: A. H. Bond und L. Gasser (Hrsg.), *Readings in Distributed Artificial Intelligence*, S. 120–139. Morgan Kaufmann Publishers.
- Littlewood, K. (1972). Forecasting and control of passenger bookings. *Proceedings of the Twelfth Annual AGIFORS Symposium*.
- Lockemann, P. C. (2006). Agents. In: S. Kirn, O. Herzog, P. Lockemann, und O. Spaniol (Hrsg.), *Multiagent Engineering*, Kapitel 1, S. 17–34. Springer.
- Luhmann, N. (1976). *Funktion und Folgen formaler Organisation* (3. Aufl.). Berlin: Duncker & Humblot.
- Luhmann, N. (1984). *Soziale Systeme - Grundriß einer allgemeinen Theorie*. Frankfurt am Main: Suhrkamp.
- Luhmann, N. (1987). *Soziologische Aufklärung 4 - Beiträge zur funktionalen Differenzierung der Gesellschaft*. Westdeutscher Verlag.
- Luhmann, N. (2000). *Organisation und Entscheidung*. Wiesbaden: Opladen.
- Malone, T. W. (1987, Oktober). Modeling Coordination in Organizations and Markets. *Management Science* 33(10), S. 1317–1332.
- Malone, T. W. (1990). Organizing information processing systems: parallels between human organizations and computer systems. In: S. P. Robertson, W. Zachary, und J. B. Black (Hrsg.), *Cognition, Computing, and Cooperation*, S. 56–83. New Jersey: Ablex Publishing.
- Malone, T. W. und K. Crowston (1994). The Interdisciplinary Study of Coordination. *ACM Computing Surveys* 26(1), S. 87–119.
- March, J. G. und H. A. Simon (1958). *Organizations* (2. Aufl.). Oxford, England: Wiley.
- Marschak, J. (1955). Elements for a Theory of Teams. In: G. Eberlein und W. Leinfellner (Hrsg.), *Economic Information, Decision, and Prediction*, Kapitel 21, S. 63–76. Dordrecht-Holland: D. Reidel Publishing Company.

- Marschak, J. und R. Radner (1972). *Economic Theory of Teams*. New Haven: Yale University Press.
- Martinez, E., I. Kwiatkowski und Clermont-Ferrand (2010). Towards a Model of Social Coherence in Multi-Agent Organizations. In: *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*.
- Meffert, H. (1995). Dienstleistungsmarketing. In: B. Tietz, R. Köhler, und J. Zentes (Hrsg.), *Handwörterbuch des Marketing*, Volume 2. Stuttgart: Schäffer-Poeschel Verlag.
- Mintzberg, H. (1979). *The Structuring of Organizations*. Prentice Hall.
- Müller, J. (1993). *Verteilte Künstliche Intelligenz - Methoden und Anwendungen*. Mannheim: BI Wissenschaftsverlag.
- Müller-Hengstenberg, C. D. und S. Kirn (2016). *Rechtliche Risiken autonomer und vernetzter Systeme*. De Gruyter Oldenbourg.
- Netessine, S. und R. Shumsky (2002). Introduction to the Theory and Practice of Yield Management. *INFORMS Transactions on Education* 3, S. 34–44.
- Nickles, M., M. Rovatsos und G. Weiss (Hrsg.) (2003). *Agents and Computational Autonomy*, Volume 2969 of *Lecture Notes in Computer Science*. Springer.
- Nissen, M. und K. Sengupta (2006). Incorporating software agents into supply chains: experimental investigation with a procurement task. *MIS Quarterly* 30(1), S. 145–166.
- Nordsieck, F. (1931). *Die schaubildliche Erfassung und Untersuchung der Betriebsorganisation*. Stuttgart: C.E. Poeschel Verlag.
- Nordsieck, F. und H. Nordsieck-Schröer (1969). Aufgabe. Siehe Grochla (1969b), S. 191–199.
- Österle, H., J. Becker, U. Frank, T. Hess, D. Karagiannis, H. Krcmar, P. Loos, P. Mertens, A. Oberweis und E. J. Sinz (2010). Memorandum zur gestaltungsorientierten Wirtschaftsinformatik. *Zeitschrift für betriebswirtschaftliche Forschung* 6(62), S. 664–672.
- Osterloh, M. (2004). Entscheidungsorientierte Organisationstheorie. Siehe Schreyögg (2004a), S. 222–229.
- Pak, K. (2005). *Revenue Management: New Features and Models*. Dissertation, Erasmus University Rotterdam.
- Pechtl, H. (2005). *Preispolitik*. Stuttgart: Lucius und Lucius.
- Petsch, M. (2006). *Offenheit von Multiagentensystemen - Eine integrative Betrachtung unter Berücksichtigung technischer, systemischer, sozialer und organisatorischer Aspekte*. Dissertation, Technische Universität Ilmenau.
- Pfeffer, J. und G. R. Salancik (1978). *The External Control of Organizations: A Resource Dependence Perspective*. New York: Harper & Row.

- Picot, A. (2005). *Organisation*. Stuttgart: Schäffer-Poeschel Verlag.
- Pollacia, L. F. (1989). A Survey of Discrete Event Simulation and State-of-the-Art Discrete Event Languages. *ACM SIGSIM Simulation Digest* 20(3), S. 8–25.
- Premm, M. (2015). Bid-Price Control for Energy-Aware Pricing of Cloud Services. In: *Proceedings of the 12th International Conference on Wirtschaftsinformatik*, S. 963–977.
- Premm, M. und S. Kirn (2017). Autonomous Agents in Multiagent Organizations. In: *Proceedings of the 9th International Conference on Agents and Artificial Intelligence (ICAART) – Volume 1*, S. 121–128. Scitepress.
- Premm, M., T. Widmer und P. Karänke (2013). Bid-Price Control for the Formation of Multiagent Organisations. In: M. Klusch, M. Thimm, und M. Paprzycki (Hrsg.), *Proceedings of the 11th German Conference on Multiagent System Technologies*, Volume 8076 of *Lecture Notes in Computer Science*, Heidelberg, S. 138–151. Springer.
- Reichwald, R. (2004). Organisationsgrenzen. Siehe Schreyögg (2004a), S. 998–1008.
- Richter, R. und E. G. Furubotn (2010). *Neue Institutionenökonomik - Eine Einführung und kritische Würdigung* (4. Aufl.). Tübingen: Mohr Siebeck.
- Riege, C., J. Saat und T. Bucher (2009). Systematisierung von Evaluationsmethoden in der gestaltungsorientierten Wirtschaftsinformatik. In: J. Becker, H. Krcmar, und B. Niehaves (Hrsg.), *Wissenschaftstheorie und gestaltungsorientierte Wirtschaftsinformatik*, S. 69–86. Heidelberg: Physica-Verlag.
- Russell, S. und P. Norvig (2010). *Artificial Intelligence: A Modern Approach* (3. Aufl.). Pearson.
- Sandholm, T. (1993). An Implementation of the Contract Net Protocol Based on Marginal Cost Calculations. In: *Eleventh National Conference on Artificial Intelligence (AAAI)*, Washington, S. 256–262.
- Sandholm, T. und V. Lesser (2002). Leveled-Commitment Contracting: A Backtracking Instrument for Multiagent Systems. *AI Magazine* 23(3), S. 89–100.
- Sandholm, T. W. und V. R. Lesser (1995). Issues in automated negotiation and electronic commerce: Extending the contract net framework. In: *Proceedings of the International Conference on Multi-Agent Systems*, S. 328–335. MIT Press.
- Schreyögg, G. (1996). *Organisation: Grundlagen moderner Organisationsgestaltung* (1. Aufl.). Gabler.
- Schreyögg, G. (Hrsg.) (2004a). *Handwörterbuch Unternehmensführung und Organisation* (4. Aufl.). Stuttgart: C.E. Poeschel Verlag.
- Schreyögg, G. (2004b). Organisation. Siehe Schreyögg (2004a), S. LV, 1767 S.
- Schreyögg, G. (2004c). Organisationstheorie. Siehe Schreyögg (2004a), S. 1069–1088.

- Schreyögg, G. (2008). *Organisation: Grundlagen moderner Organisationsgestaltung* (5. Aufl.). Wiesbaden: Gabler.
- Schwarz, H. (1980a). Aufgabenträger. Siehe Grochla (1980), S. 217–224.
- Schwarz, H. (1980b). Stelle. Siehe Grochla (1980), S. 2113–2118.
- Schwarz, H. (1983). *Betriebsorganisation als Führungsaufgabe* (9. Aufl.). Landsberg am Lech: Verlag Moderne Industrie.
- Scott, W. R. (1981). *Grundlagen der Organisationstheorie*. Frankfurt: Campus Verlag.
- Searle, J. (1969). *Speech Acts – An Essay in the Philosophy of Language*. Cambridge: Cambridge University Press.
- Shoham, Y. (1993). Agent-oriented programming. *Artificial Intelligence* 60, S. 51–92.
- Simon, H. A. (1996). *The Sciences of the Artificial* (3. Aufl.). Cambridge: MIT Press.
- Smith, R. G. (1980). The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver. *IEEE Transactions on Computers* C-29(12), S. 1104–1113.
- Smith, R. G. und R. Davis (1978). Distributed Problem Solving: The Contract Net Approach. Forschungsbericht, Stanford University.
- Smith, R. G. und R. Davis (1981). Frameworks for Cooperation in Distributed Problem Solving. *IEEE Transactions on Systems, Man, and Cybernetics* 11(1), S. 61–70.
- Sorge, C. (2005). *Softwareagenten – Vertragsschluss, Vertragsstrafe, Reugeld*. Schriften des Zentrums für angewandte Rechtswissenschaften, Universität Karlsruhe. Karlsruhe: Universitätsverlag Karlsruhe.
- Talluri, K. und G. van Ryzin (1998). An Analysis of Bid-Price Controls for Network Revenue Management. *Management Science* 44(11), S. 1577–1593.
- Talluri, K. T. und G. J. van Ryzin (2004). *The Theory and Practice of Revenue Management*. New York: Springer.
- Thom, N. (1992). Stelle, Stellenbildung und -besetzung. Siehe Frese (1992), S. 2321–2333.
- Timm, I. J. und F. Hillebrandt (2006). Reflexion als sozialer Mechanismus zum strategischen Management autonomer Softwaresysteme. In: M. Schmitt, M. Florian, und F. Hildebrandt (Hrsg.), *Reflexive soziale Mechanismen – Von soziologischen Erklärungen zu sozionischen Modellen*, S. 255–288. Wiesbaden: VS Verlag für Sozialwissenschaften.
- Tomas, L. und J. Tordsson (2013). Improving Cloud Infrastructure utilization through Overbooking. In: *Proceedings of the 2013 ACM Cloud and Autonomic Computing Conference*.
- Ulrich, H. (1969). Kompetenz. Siehe Grochla (1969b), S. 852–856.

- Urgaonkar, B., P. Shenoy und T. Roscoe (2002). Resource overbooking and application profiling in shared hosting platforms. *ACM SIGOPS Operating Systems Review* 36(SI), S. 239–254.
- Urgaonkar, B., P. Shenoy und T. Roscoe (2009). Resource overbooking and application profiling in shared hosting platforms. *ACM Transactions on Internet Technology* 9(1), S. 1–45.
- van Riemsdijk, M. B., K. Hindriks und C. Jonker (2009). Programming Organization-Aware Agents – A Research Agenda. In: H. Aldewereld, V. Dignum, und G. Picard (Hrsg.), *Engineering Societies in the Agents World X*, Volume 5881 of *Lecture Notes in Artificial Intelligence*, S. 98–112. Heidelberg: Springer.
- VDI (1997). *VDI 3633 Blatt 3: Simulation von Logistik-, Materialfluß- und Produktionssystem – Experimentplanung und -auswertung*. Düsseldorf: Verein Deutscher Ingenieure.
- von Martens, T. (2008). *Kundenwertorientiertes Revenue Management im Dienstleistungsbereich*. Dissertation, Technische Universität Dresden.
- Weatherford, L. R. und S. E. Bodily (1992). A Taxonomy and Research Overview of perishable-asset Revenue Management: Yield Management, Overbooking, and Pricing. *Operations Research* 40(4), S. 831–844.
- Weber, M. (1922). *Wirtschaft und Gesellschaft*. Tübingen: J.C.B. Mohr.
- Weigand, H. und V. Dignum (2003). I am Autonomous, You are Autonomous. In: M. Nickles, M. Rovatsos, und G. Weiss (Hrsg.), *Agents and Computational Autonomy*, Volume 2969 of *Lecture Notes in Computer Science*, S. 227–236.
- Weiss, G. (Hrsg.) (1999). *Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence*. MIT Press.
- Weiss, G. (Hrsg.) (2013). *Multiagent Systems* (2. Aufl.). MIT Press.
- Wernerfelt, B. (1984). A resource-based view of the firm. *Strategic Management Journal* 5, S. 171–180.
- Weyns, D., H. V. D. Parunak, F. Michel, T. Holvoet und J. Ferber (2005). Environments for Multiagent Systems State-of-the-Art and Research Challenges. In: *The First International Workshop on Environments for Multiagent Systems*, S. 1–47.
- Williamson, E. L. (1992). *Airline Network Seat Inventory Control: Methodologies and Revenue Impacts*. Dissertation, Massachusetts Institute of Technology.
- Williamson, O. E. (1985). *The Economic Institutions of Capitalism: Firms, Markets, Relational Contracting*. New York: The Free Press.
- Wooldridge, M. (1999). Intelligent Agents. In: G. Weiss (Hrsg.), *Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence* (2. Aufl.). Cambridge: MIT Press.

- Wooldridge, M. (2009). *An Introduction to MultiAgent Systems* (2. Aufl.). Chichester: John Wiley & Sons.
- Wooldridge, M. und N. R. Jennings (1995). Intelligent agents: theory and practice. *The Knowledge Engineering Review* 10, S. 115–152.
- Wooldridge, M., N. R. Jennings und D. Kinny (2000). The Gaia Methodology for Agent-Oriented Analysis and Design. *Autonomous Agents and Multi-Agent Systems* 3(3), S. 285–312.
- Zimmermann, H. (1980). OSI Reference Model-The ISO Model of Architecture for Open Systems Interconnection. *IEEE Transactions on Communications* 28(4), S. 425–432.