

Agentenbasierte marktliche Koordination
von Wertschöpfungsnetzwerken

Dissertation

zur Erlangung des Grades
„Doktor der Ingenieurwissenschaften (Dr.-Ing.)“
der Naturwissenschaftlich-Technischen Fakultät I
der Universität des Saarlandes

vorgelegt von

Dipl.-Inform. Christian Ruß

Saarbrücken, 2015

Tag des Kolloquiums

6. Juni 2016

Dekan der Naturwissenschaftlich-
Technischen Fakultät I

Univ.-Prof. Dr. Markus Bläser

Vorsitzender des Prüfungsausschusses

Univ.-Prof. Dr. Andreas Zeller

Erstgutachter / Doktorvater

Univ.-Prof. Dr. Jörg H. Siekmann

Zweitgutachter

Univ.-Prof. Dr. Torsten Eymann

(Universität Bayreuth)

Akademischer Beisitzer

Dr. Klaus Fischer (DFKI)

Meinen Eltern gewidmet.

Kurzzusammenfassung

Diese Arbeit stellt die Multiagenten-Supply-Chain-Simulationsumgebung MACSIMA vor, die das Design und die Simulation von großen Wertschöpfungsnetzwerken ermöglicht, die aus einer Vielzahl an autonomen Geschäftsagenten bestehen. MACSIMA stattet die Agenten, die miteinander über einen elektronischen Markt interagieren, mit einem adaptiven Verhandlungsmodul für bilaterale Verhandlungen aus, das mittels evolutionärer Algorithmen realisierte Lernfähigkeiten aufweist. Dadurch können Agenten ihre Verhandlungsstrategien an unterschiedliche Verhandlungspartner und sich dynamisch ändernde Marktbedingungen adaptieren. MACSIMA erlaubt es, die Lernfähigkeiten der Agenten sowie ihre Möglichkeiten zum Austausch und zur Nutzung von Informationen detailliert einzustellen und das sich daraus ergebende Systemverhalten zu simulieren und zu evaluieren. MACSIMA erweitert bekannte Ansätze um neue Konzepte, Methoden und Einstellmöglichkeiten. Simulationsergebnisse werden auf einer Makro-, Meso- und Mikroebene gemessen, mit Hilfe einer Evaluationsmethodik aufbereitet und schließlich im Hinblick auf die Systemperformanz und die Co-Evolution von Verhandlungsstrategien quantitativ und qualitativ analysiert. Die Evaluationsergebnisse zeigen, dass das Ergebnis der Selbstkoordination der Agenten stark von ihren Lern- und Informationsaustausch-Einstellungen abhängt und die Systemperformanz durch die Erweiterungen, die MACSIMA bereitstellt, im Vergleich zu bisherigen Ansätzen deutlich gesteigert werden kann.

Abstract

This work presents the multiagent supply chain simulation framework MACSIMA that allows the design of large-scale supply networks consisting of a multitude of autonomous business agents. MACSIMA equips all the agents interacting with each other via an electronic marketplace with an adaptive negotiation module for bilateral negotiations. This module provides the agents with learning capabilities realised by evolutionary algorithms. Thereby the agents can adapt their negotiation strategies to varying negotiation partners and dynamically changing market conditions. MACSIMA features the in-depth adjustment of the agents' capabilities for learning as well as the exchange and use of information. The resulting system behaviour can be simulated and evaluated. MACSIMA extends previous approaches by adding new concepts, methods and adjustment possibilities. Simulation outcomes are measured on a macro-, meso-, and micro-level, refined by an evaluation methodology, and finally analysed quantitatively and qualitatively with regard to the overall system performance and the co-evolution of negotiation strategies. The evaluation results show that the outcome of the agents' self-coordination depends strongly on the settings for their learning and information exchange capabilities. In comparison to previous approaches the overall system performance can be increased significantly by the extensions provided by MACSIMA.

Danksagung

An erster Stelle möchte ich mich bei Herrn Professor Dr. Jörg Siekmann für seine Unterstützung während der Fertigstellung dieser Forschungsarbeit bedanken. Ohne die Möglichkeiten, die er mir eröffnet hat und ohne seine Motivation und seinen Rat wäre diese Arbeit nicht möglich gewesen.

Mein besonderer Dank gilt ebenso Herrn Professor Dr. Torsten Eymann, auf dessen Forschungsergebnissen die in MACSIMA realisierten Ideen und Konzepte aufbauen. Da er sich in seinen Forschungsarbeiten intensiv mit der katallaktischen Selbstkoordination in einem agentenbasierten Wertschöpfungsnetzwerk beschäftigt hat, ist er für das bearbeitete Forschungsgebiet ein ausgewiesener Experte. Dankenswerterweise hat er sich spontan dazu bereit erklärt, die vorliegende Arbeit als Zweitgutachter zu beurteilen.

Darüber hinaus möchte ich mich bei der Multiagenten-System Gruppe am DFKI und insbesondere Herrn Dr. Klaus Fischer für die fruchtbare Zusammenarbeit über viele Jahre bedanken, in der auch die Grundlagen für diese Forschungsarbeit gelegt wurden.

Außerdem möchte ich mich bei meinen früheren Diplomanden Alexander Walz und Heiko Graeber dafür bedanken, dass sie mir auch nach dem Abschluss ihrer Diplomarbeiten freundschaftlich verbunden blieben und ich mit ihnen Ideen und Lösungsansätze diskutieren konnte.

Des Weiteren danke ich Herrn Professor Dr. Thomas Korne und Christian Korne für ihre Freundschaft und Motivation sowie das Korrekturlesen von Teilen dieser Arbeit.

Den größten Dank schulde ich aber meinen Eltern. Ohne ihr Verständnis und ihre großartige Unterstützung, die mich mein ganzes Leben begleitet, hätte ich mich dieser Promotion nicht widmen können.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation und Problemstellung	1
1.2	Aufbau der Arbeit	5
2	Verteilte KI & Multiagenten-Systeme	7
2.1	Einleitung und Übersicht	7
2.2	Verteilte Künstliche Intelligenz	7
2.2.1	Verteiltes Problemlösen	9
2.2.2	Multiagenten-Systeme	10
2.3	Software-Agenten	13
2.3.1	Grundlegende Definitionen	13
2.3.2	Agenten-Eigenschaften	14
2.3.3	Umwelt-Charakteristika	17
2.4	Agenten-Klassen und -Typen	17
2.4.1	Einfache Reflex-Agenten	17
2.4.2	Modellbasierte Reflex-Agenten	18
2.4.3	Zielbasierte Agenten	18
2.4.4	Nutzenbasierte Agenten	19
2.4.5	Lernende Agenten	20
2.4.6	Weitere Agenten-Klassen und -Typen	21
2.5	Das Koordinationsproblem in MAS	22
2.5.1	Kooperation	23
2.5.2	Kommunikation	23
2.5.3	Koordination	26
2.6	Holonische Multiagenten-Systeme	29
2.6.1	Holonische Systeme	30
2.6.2	Holonische Agenten	31
2.6.3	Holonische MAS-Strukturen	32
2.6.3.1	Holon als Gruppe autonomer kooperierender Agenten	32
2.6.3.2	Holon als vollständige Verschmelzung von Agenten	33
2.6.3.3	Holon als Kopf-gesteuerter Verbund von Agenten	34
2.7	Marktliche MAS-Koordination	35
3	Ökonomische Grundlagen	39
3.1	Einleitung und Übersicht	39
3.2	Grundlagen des Supply Chain Management	39
3.2.1	Supply Chains	40
3.2.2	Wertschöpfungsnetzwerke	42
3.2.2.1	Von Supply Chains zu Wertschöpfungsnetzwerken	42
3.2.2.2	Begriffsabgrenzung für diese Arbeit	46
3.2.3	Supply Chain Management (SCM)	48
3.2.4	E-SCM und Supply Webs	50
3.2.5	Supply Chain Modelle	53
3.3	Grundlagen marktlicher Koordination	56

3.3.1	Das Konzept Markt	56
3.3.2	Die Bildung von Katallaxien.....	58
3.3.3	Elektronische Märkte	60
3.3.3.1	Arten elektronischer Märkte.....	60
3.3.3.2	Phasen marktlicher Transaktion.....	61
3.3.4	Marktliche Koordinationsmechanismen	62
3.3.4.1	Bilaterale Verhandlungen.....	63
3.3.4.2	Auktionsmechanismen	64
4	Agenten in ökonomischen Wertschöpfungsstrukturen	67
4.1	Einleitung und Übersicht.....	67
4.2	Einsatzszenarien	68
4.2.1	Supply Chain Management	68
4.2.2	Elektronische Märkte	71
4.2.2.1	Innerbetriebliche elektronische Märkte.....	72
4.2.2.1.1	Produktion	72
4.2.2.1.2	Logistik.....	72
4.2.2.2	Überbetriebliche elektronische Märkte	73
4.2.2.2.1	Agentenbasierte Verhandlungen und Auktionen.....	73
4.2.2.2.2	Computergesteuerter Börsen- und Wertpapierhandel	75
4.2.2.2.3	Virtuelle Unternehmen und Unternehmensnetzwerke	75
4.3	Holonische Agenten in elektronischen Märkten	75
4.3.1	Holonische Geschäftsagenten	76
4.3.1.1	Koordinationsagent	76
4.3.1.2	Bündelagenten.....	77
4.3.1.3	Ausführungsagenten.....	78
4.3.1.4	Informations- und Kontrollfluss.....	78
4.3.1.5	Interaktionen auf elektronischen Märkten.....	79
4.3.2	Der holonische Koordinationsserver.....	80
4.3.2.1	Koordinations-Matchmaker-Agent	81
4.3.2.2	Koordinationsmechanismus-Agenten	82
4.3.2.3	Koordinationsausführungsagenten	82
4.3.2.4	Interaktionen und Prozesse.....	83
5	MACSIMA: Ein MAS für die marktliche Koordination von Wertschöpfungsnetzwerken	89
5.1	Einleitung und Übersicht.....	89
5.2	Die Architektur von MACSIMA.....	93
5.2.1	Aufbau von Simulationsszenarien.....	94
5.2.2	Generische Agenten-Typen.....	98
5.2.3	Realisierung des elektronischen Marktes	100
5.2.4	Kontrollzyklus eines MACSIMA-Agenten.....	102
5.3	Verhandlungen in MACSIMA	103
5.3.1	Verhandlungsprotokoll.....	103
5.3.2	Parameter der Verhandlungsstrategie.....	106
5.3.3	Ablauf einer bilateralen Verhandlung	110
5.3.3.1	Die Datenstruktur PriceDistribution.....	110
5.3.3.2	Bestimmung des Verhandlungsintervalls.....	112
5.3.3.3	Start einer Verhandlung	113

5.3.3.4	Reaktion mit Gegenangeboten (PROPOSE-Fälle).....	114
5.3.3.5	Reaktion mit Verhandlungsabbrüchen (REJECT-Fälle).....	115
5.3.3.6	Erfolgreicher Verhandlungsabschluss (ACCEPT-Fälle).....	116
5.3.3.7	Preisautomat	117
5.3.3.7.1	Kennzahlen des Preisautomaten	119
5.3.3.7.2	Senkung des Startpreises ULP_{VK} des Verkäufers	120
5.3.3.7.3	Senkung des Mindestpreises LLP_{VK} des Verkäufers.....	123
5.3.3.7.4	Erhöhung des Startpreises LLP_K des Käufers.....	124
5.3.3.7.5	Erhöhung des Maximalpreises ULP_K des Käufers	126
5.4	Lernen in MACSIMA.....	128
5.4.1	Adaptives Verhandlungsmodul der Agenten.....	129
5.4.1.1	1-Pool-Modus	130
5.4.1.2	2-Pool-Modus mit Pool-Separierung.....	132
5.4.1.3	Informationsaustausch-Einstellungen.....	136
5.4.1.4	Informationsnutzungs-Modus.....	137
5.4.2	Fitness-Bewertungsmethoden.....	138
5.4.2.1	Die PmA-Methode.....	140
5.4.2.2	Die AvP-Methode.....	140
5.4.2.3	Die BaP-Methode	141
5.4.2.4	Die MdP-Methode	141
5.4.2.5	Die RMF-Methode	141
5.4.3	Genetische Operatoren	142
5.4.3.1	Selektion: Standardmethoden	142
5.4.3.2	Deterministic Minority Selection	145
5.4.3.3	Rekombination	149
5.4.3.4	Mutation	150
5.4.4	Ersetzungsschemata.....	152
5.4.4.1	Elitismus	153
5.4.4.2	Schwacher Elitismus	153
5.4.5	Spezialverfahren	153
5.4.5.1	Highest-Fitness-First-Verfahren.....	154
5.4.5.2	Ticket-Limitierung.....	155
5.4.6	Das STDEA-Adaptionsverfahren	157
6	Simulationsinfrastruktur	159
6.1	Einleitung und Übersicht	159
6.2	MACSIMA-Simulationsumgebung.....	160
6.2.1	Grundlegende Einstellungen.....	161
6.2.1.1	Einstellungen für Güter und Agenten.....	161
6.2.2	User Interface	163
6.2.2.1	Strategie-Einstellungen.....	164
6.2.2.2	Einstellungen für das evolutionäre Lernen.....	166
6.2.2.3	Einstellungen für den Simulationsstart.....	168
6.3	Notation für die Simulationsszenarien	169
6.3.1	Nicht-optionale Notationselemente	172
6.3.2	Optionale Notationselemente	175
6.4	Protokollierung von Simulationsläufen.....	176
6.4.1	Generierung der Protokoll-Dateien	176
6.4.2	Format und Verarbeitbarkeit der Protokoll-Dateien.....	177

6.4.3	Inhalte der Protokoll-Dateien	177
7	Evaluationsmethodik	179
7.1	Einleitung und Übersicht.....	179
7.2	Evaluationsebenen.....	181
7.2.1	Untersuchungsperspektiven aus Sicht der Sozionik.....	181
7.2.2	Makro-, Meso- und Mikro-Ebene	183
7.3	Quantitative Evaluation.....	184
7.3.1	Generierung von Kennzahlen.....	184
7.3.2	Ergebnistabellen	185
7.3.2.1	Makro-Ergebnistabellen	186
7.3.2.1.1	Makro-Umsatz-Ergebnistabelle.....	186
7.3.2.1.2	Makro-Preis-Ergebnistabelle	188
7.3.2.2	Meso-Ergebnistabelle.....	190
7.3.2.3	Mikro-Ergebnistabelle.....	192
7.3.3	Verwendete Maße und Kennzahlen der Statistik	194
7.3.3.1	Herfindahl-Index	194
7.3.3.2	Weitere Lage- und Streuungsmaße	195
7.3.3.2.1	Lagemaße.....	195
7.3.3.2.2	Streuungsmaße.....	195
7.4	Qualitative Evaluation.....	196
7.4.1	Visualisierungsoptionen für Simulationsergebnisse	196
7.4.2	Visualisierung von Kennzahlen	198
8	Evaluation.....	203
8.1	Einleitung und Übersicht.....	203
8.1.1	Veröffentlichungen.....	204
8.2	Systemverhalten von AVALANCHE und MACSIMA	205
8.3	Variation der Gen-Startwerte	209
8.3.1	Isolierte Variation des Gens Acquisitiveness (AQ)	210
8.3.2	Isolierte Variation des Gens Satisfaction (SF).....	211
8.3.3	Variation von Startwert-Kombinationen.....	211
8.4	Parametrisierung der Lern-Komponente.....	212
8.4.1	Effekte der Relation von Pool und Population.....	213
8.4.2	Effekte der Fitness-Methoden	216
8.4.3	Effekte der Selektions-Methoden.....	217
8.4.4	Effekte der Rekombinations-Einstellungen	219
8.4.5	Effekte der Pool-Separierung.....	224
8.4.6	Effekte der Ersetzungsschema-Einstellungen	228
8.4.7	Effekte der Mutations-Methoden	230
8.5	Informationsaustausch und –nutzung.....	236
8.5.1	Informationsaustausch und Mutation	237
8.5.2	Informationsaustausch bei heterogenem Lernen.....	240
8.6	Selbstkoordination mit MACSIMA-Erweiterungen	243
8.7	Zusammenfassung der Evaluationsergebnisse	247
9	Zusammenfassung & Ausblick	251
9.1	Zusammenfassung.....	251

9.2	Erzielte Erkenntnisse	253
9.3	Ausblick.....	255
10	Literaturverzeichnis	257
11	Anhang	279
11.1	Testreihen	279
11.1.1	AVALANCHE-Experimente	279
11.1.2	Testreihe zu Startwert-Effekten.....	281
11.1.3	Testreihe zur Relation von Pool- und Populations-Größe.....	282
11.1.4	Testreihe zu den Effekten der Fitness-Methoden.....	284
11.1.5	Testreihe zu den Selektions-Methoden	285
11.1.5.1	Statische Initialisierung der Gen-Startwerte.....	285
11.1.5.2	Randomisierte Initialisierung der Gen-Startwerte.....	285
11.1.6	Testreihe zu Rekombinations-Einstellungen	286
11.1.7	Testreihe zur Pool-Separierung	287
11.1.8	Testreihe zu Ersetzungsschema-Einstellungen.....	288
11.1.9	Testreihe zu Mutations-Methoden.....	288
11.1.10	Testreihe zu Informationsaustausch und Mutation.....	289
11.1.11	Testreihe zum Informationsaustausch bei heterogenem Lernen	289
11.1.12	Testreihe zur Selbstkoordination mit MACSIMA-Erweiterungen.....	290
11.2	Evaluations-Diagramme	293
11.3	Makro-Ergebnistabellen	296
11.3.1	Makro-Tabellen	296
11.3.2	Medio-Tabelle	297
11.3.3	Mikro-Tabellen.....	299

Abbildungsverzeichnis

Abbildung 1: Interaktion eines Agenten mit seiner Umwelt.....	8
Abbildung 2: Schnittmengen des Forschungsgebiets der Multiagenten-Systeme mit anderen Forschungsdisziplinen	12
Abbildung 3: Allgemeines Modell eines modell- und nutzenbasierten Agenten.....	19
Abbildung 4: Allgemeines Modell eines lernenden Agenten.....	20
Abbildung 5: Struktur einer KQML-Nachricht	25
Abbildung 6: Überblick über die Koordinations-Modi nach Moulin.....	29
Abbildung 7: Der Holon als Gruppe autonomer kooperierender Agenten.....	33
Abbildung 8: Das Verschmelzen mehrerer Agenten zu einem Holon	33
Abbildung 9: Der Holon als Kopf-gesteuerter Verbund	34
Abbildung 10: Wirkungszusammenhang zwischen dem Systemergebnis eines MAS und dem Design der Agenten und des verwendeten Koordinationsmechanismus.....	37
Abbildung 11: Schematische Darstellung der Verzahnung von Unternehmen in einer Supply Chain	41
Abbildung 12: Strukturelle Sicht auf eine Supply Chain	42
Abbildung 13: Beispiel für ein Wertschöpfungsnetzwerk	43
Abbildung 14: Von der Wertschöpfungskette zum –netz	43
Abbildung 15: Charakteristika adaptiver und ubiquitärer Supply Networks im Vergleich mit traditionellen Supply Chains	44
Abbildung 16: Die adaptive Supply Chain.....	45
Abbildung 17: Liefernetzwerk.....	50
Abbildung 18: Zusammenwirken von E-Business und SCM im E-SCM.....	51
Abbildung 19: Wandel im Supply Chain Management Ansatz	52
Abbildung 20: Supply Chain Modelle nach [Mentzer et al. 2001]	53
Abbildung 21: Struktur eines Wertschöpfungsnetzwerks nach [Lambert et al. 1998].....	54
Abbildung 22: Allgemeines Supply-Chain-Modell nach [Bowersox et al. 2002]	55
Abbildung 23: Übergreifende Koordinationsfunktion der Märkte.....	57
Abbildung 24: Phasen einer marktlichen Transaktion	62
Abbildung 25: Bilaterale Preisverhandlungen.....	64
Abbildung 26: Der holonische Koordinationsagent	77
Abbildung 27: H-MAS basierte Koordinations-Infrastruktur für elektronische Märkte.....	80
Abbildung 28: Der holonische Koordinationsserver	81
Abbildung 29: Interaktionsdiagramm zum Ablauf einer Auktion.....	84
Abbildung 30: EPK zum Start einer Auktion durch den CMM	85
Abbildung 31: EPK zur Abgabe eines ersten Gebots durch einen Geschäftsagenten.....	86
Abbildung 32: Abschluss einer Auktion durch den CX-Agenten	87
Abbildung 33: MACSIMA-Simulationsszenario mit generischen Bausteinen.....	94
Abbildung 34: Basis-Topologie von Simulationsszenarien für Wertschöpfungsnetzwerke in der Anwendungsdomäne Computer	95
Abbildung 35: Einstellung von Agentengruppen mit der jeweiligen Anzahl von Agenten und deren Informationsaustausch- sowie Lern-Parametrisierung	95
Abbildung 36: Matchmaking und Ontologie-Abbildung über den ISA-Agenten.....	100
Abbildung 37: Kontrollzyklus eines MACSIMA-Produzenten-Agenten	102
Abbildung 38: Bilaterales Preisverhandlungsprotokoll.....	105
Abbildung 39: Verhandlungsintervall von Käufer und Verkäufer.....	113
Abbildung 40: Die Phasen des Preisautomaten für den Startpreis des Verkäufers.....	121
Abbildung 41: Die Phasen des Preisautomaten für den Mindestpreis des Verkäufers	123
Abbildung 42: Die Phasen des Preisautomaten für den Startpreis des Käufers	125

Abbildung 43: Die Phasen des Preisautomaten für den Maximalpreis des Käufers	126
Abbildung 44: Pool und Population eines Agenten	130
Abbildung 45: Im Verhandlungsmodul ablaufende Prozesse im 1-Pool-Modus.....	131
Abbildung 46: UML-Aktivitätsdiagramm für den 2-Pool-Modus.....	133
Abbildung 47: Austausch des Genotyps im 2-Pool-Modus	135
Abbildung 48: Adaptionszyklus eines MACSIMA-Agenten.....	139
Abbildung 49: Beispiel für ein Roulette-Rad beim RWS-Verfahren	143
Abbildung 50: Schematischer Ablauf der Deterministic Minority Selection für DMS....	148
Abbildung 51: Kreuzung zweier Genotypen beziehungsweise Individuen	150
Abbildung 52: Beispiel für die Mutation eines Gens.....	152
Abbildung 53: UML-Aktivitätsdiagramm für das Ticket-Limitierungs-Verfahren.....	156
Abbildung 54: Einstellungen für Güter und deren Preisgrenzen in MACSIMA	161
Abbildung 55: Einstellungen für Supplier-Agenten in MACSIMA	162
Abbildung 56: Einstellungen für Deposit-Agenten in MACSIMA	163
Abbildung 57: Einstellungen für Branch-Agenten in MACSIMA	163
Abbildung 58: Einstellungen für die Erzeugung von initialen Verhandlungsstrategien in MACSIMA.....	164
Abbildung 59: Kombination von Initialisierungsfunktionen für die initiale Befüllung des Strategie-Pools von MACSIMA-Agenten	165
Abbildung 60: Detail-Einstellungen für das Evolutionäre Lernen in MACSIMA im Tab <i>EvolutionaryAlgorithms</i>	166
Abbildung 61: Informationsaustausch-, Pool- und Population-Einstellungen für das Evolutionäre Lernen in MACSIMA im Tab <i>Learning</i>	167
Abbildung 62: Einstellungen für den Simulationsstart in MACSIMA	168
Abbildung 63: Visualisierung der Entwicklung der Güterpreise während eines Simulationslaufs mit dem Skript <i>preisPlotSales.sh</i>	198
Abbildung 64: Preisentwicklung für eine andere Szenario-Parametrisierung – erzeugt mit dem Skript <i>preisPlotSales.sh</i>	198
Abbildung 65: Entwicklung der internen Maximalpreis-Schranken (MaxPrice) von Chip- Produzenten – erzeugt mit dem Skript <i>genSales.sh</i>	199
Abbildung 66: Entwicklung der Verhandlungsdauer bzw. der Rundenzahl, die eine Verhandlungsstrategie bis zu einem erfolgreichen Abschluss benötigt.....	199
Abbildung 67: Entwicklung des Gens <i>Acquisitiveness</i> von Käufern und Verkäufern des Gutes Chip.....	200
Abbildung 68: Entwicklung des Gens <i>Delta_Jump</i> von Verkäufern bzw. Produzenten des Gutes Chip sowie von Käufern (Prozessor-Produzenten).....	201
Abbildung 69: Entwicklung aller Gene im Einkaufspool der Chip-Produzenten.....	201
Abbildung 70: Entwicklung aller Gene im Verkaufspool der Chip-Produzenten	202
Abbildung 71: Verlauf des AVALANCHE-Experiments <i>Eymann_1</i> mit festgelegten Startpreisen und homogener Strategieverteilung (ohne Lernen).....	205
Abbildung 72: Verlauf des MACSIMA-Experiments mit den Einstellungen von <i>Eymann_1</i> (ohne die Verwendung von MACSIMA-Erweiterungen).....	206
Abbildung 73: Preisentwicklung in AVALANCHE bei einem AQ-Wert von 0,9	206
Abbildung 74: Verlauf des MACSIMA-Experiments mit den Einstellungen von <i>Eymann_2</i> (ohne MACSIMA-Erweiterungen)	207
Abbildung 75: Verlauf des MACSIMA-Experiments mit AQ = 0,95 und den sonstigen Einstellungen von <i>Eymann_2</i> (ohne MACSIMA-Erweiterungen)	207
Abbildung 76: AVALANCHE-Preisentwicklung in einem Szenario mit hartnäckigen Zimmermann-Agenten (mit einem höheren AQ-Wert als die übrigen Agenten)	208
Abbildung 77: MACSIMA-Preisentwicklung mit hartnäckigen Prozessor-Produzenten.	208
Abbildung 78: Effekte bei Variation des Startwerts des Gens <i>Acquisitiveness</i>	210

Abbildung 79: Effekte bei Variation des Startwerts des Gens Satisfaction	211
Abbildung 80: Verteilung von Gesamtumsatz, Verkaufsabschlüssen und Verhandlungsabbrüchen bei den durchgeführten Gen-Startwert-Tests.....	212
Abbildung 81: P:Pn-Relationstests für $P = 3$	213
Abbildung 82: P:Pn-Relationstests für $P = 5$	214
Abbildung 83: P:Pn-Relationstests für $P = 7$	215
Abbildung 84: P:Pn-Relationstests für $P = 10$	215
Abbildung 85: Fitness-Methoden-Ergebnisse für Pool- und Populations-Größe $P5Pn30$	216
Abbildung 86: Fitness-Methoden-Ergebnisse für Pool- und Populations-Größe $P5Pn35$	216
Abbildung 87: Selektions-Methoden-Testergebnisse 1	217
Abbildung 88: Selektions-Methoden-Testergebnisse 2	218
Abbildung 89: Selektions-Methoden-Testergebnisse 3	219
Abbildung 90: Effekte der Rekombination auf Gesamtumsatz und -gewinn sowie die Anzahl an Verkaufsabschlüssen	220
Abbildung 91: Effekte der Rekombination auf die Verhandlungsergebnis-Kennzahlen ..	220
Abbildung 92: Effekte der besten und schlechtesten Rekombination auf Gesamtumsatz und -gewinn u. das Verhältnis der Verhandlungsergebnis-Kennzahlen.....	221
Abbildung 93: Effekte der besten und schlechtesten Rekombination auf den durchschnittlichen Umsatz und Gewinn pro Transaktion	221
Abbildung 94: Effekte der besten Einstellungen für $nPC(1-1-X)$ und $RaC(X-X-0,8)$ in Kombination mit fünf Selektions-Methoden.....	222
Abbildung 95: Effekte der Rekombination auf die Stabilität des Gesamtgewinns	222
Abbildung 96: Effekte der Rekombination auf die Stabilität des Gesamtumsatzes.....	223
Abbildung 97: Effekte von Rekombinations-Einstellungen auf die Stabilität von Umsatz und Gewinn pro Transaktion	223
Abbildung 98: Effekte der Pool-Separierung auf Gesamtumsatz, Gesamtgewinn und Verhandlungsergebnis-Kennzahlen.....	224
Abbildung 99: Effekte der Pool-Separierung auf Umsatz und Gewinn pro Abschluss	225
Abbildung 100: Effekte der Pool-Separierung auf die Stabilität der Anzahl an Verkaufsabschlüssen und Verhandlungsabbrüchen	225
Abbildung 101: Effekte der Pool-Separierung auf den durchschnittlichen Gesamtumsatz und -gewinn	226
Abbildung 102: Effekte der Pool-Separierung auf die Kennzahlen für Verhandlungsergebnisse	226
Abbildung 103: Effekte der Pool-Separierung auf die Stabilität des Gesamtumsatzes	227
Abbildung 104: Effekte der Pool-Separierung auf die Stabilität der Kennzahlen für Verhandlungsergebnisse	227
Abbildung 105: Effekte der Pool-Separierung auf die Stabilität des Umsatzes und Gewinns pro Verkaufsabschluss.....	227
Abbildung 106: Gesamtumsatz und Gesamtgewinn in Abhängigkeit von den Ersetzungsschema-Einstellungen	228
Abbildung 107: Verkaufsabschlüsse und Verhandlungsabbrüche in Abhängigkeit von den Ersetzungsschema-Einstellungen	229
Abbildung 108: Prozentualer Anteil von Verhandlungsabbrüchen und Verkaufsabschlüssen an den geführten Verhandlungen.....	229
Abbildung 109: Durchschnittlicher Gesamtumsatz und -gewinn in Abhängigkeit von den Einstellungen für die Mutations-Methode GMU	231
Abbildung 110: Entwicklung von Gesamtumsatz, -gewinn und Verhandlungsergebnissen bei Variation der Gauß-Weite für GMU.....	231
Abbildung 111: Vergleich der Entwicklung des Gesamtumsatzes für GMU mit den Mutationsraten 0,5 und 1,0 bei Variation der Mutations-Weite.....	232

Abbildung 112: Entwicklung der Abschlüsse und Nachrichtenaustausche pro Verhandlung für die Mutationsraten 0,5 und 1,0 bei Variation der Mutations-Weite	232
Abbildung 113: Gesamtumsatz, Gesamtgewinn und Verhandlungsergebnisse in Abhängigkeit von den Einstellungen für die Mutations-Methode SMU	233
Abbildung 114: Unterschiede bei der Entwicklung von Umsatz, Gewinn und Verhandlungsergebnissen für SMU bei Variation der Mutationsrate	233
Abbildung 115: Gesamtumsatz, Gesamtgewinn und Verhandlungsergebnissen für die Methoden GMU und SMU bei Variation von Mutations-Rate u. –Weite	234
Abbildung 116: Entwicklung von Gesamtumsatz, Gesamtgewinn und Verhandlungsergebnissen für die Methoden GMU und SMU bei Variation der Mutations-Rate und –Weite	234
Abbildung 117: Vergleich der Entwicklung verschiedener Variationskoeffizienten für die Methoden GMU und SMU bei Variation von Mutations-Rate u. –weite	235
Abbildung 118: Variationskoeffizienten für die Mutations-Methoden GMU und SMU bei Variation von Mutations-Rate u. –Weite	236
Abbildung 119: Informationsaustausch-Effekte 1	237
Abbildung 120: Informationsaustausch-Effekte 2	238
Abbildung 121: Informationsaustausch und General Mutation (GMU)	239
Abbildung 122: Informationsaustausch und Single Mutation (SMU)	239
Abbildung 123: Gesamtübersicht kombinierter Effekte von variierenden Informationsaustausch- und Mutations-Einstellungen im Detail.....	240
Abbildung 124: Ermittlung der über alle Informationsaustausch-Einstellungen hinweg besten Mutations-Einstellung	240
Abbildung 125: Gesamtumsatz der Agenten auf Stufe 3 in Abhängigkeit von den Informationsaustausch-Einstellungen	241
Abbildung 126: Gesamtgewinn der Agenten auf Stufe 3 in Abhängigkeit von den Informationsaustausch-Einstellungen	241
Abbildung 127: Umsätze der Agenten auf Stufe 3 in Abhängigkeit von den Informationsaustausch- und Fitness-Einstellungen.....	242
Abbildung 128: Gewinne der Agenten auf Stufe 3 in Abhängigkeit von den Informationsaustausch- und Fitness-Einstellungen.....	243
Abbildung 129: Vergleich der Selbstkoordinationsleistung von an [Eymann 2000] orientierten Szenarien mit Szenarien mit MACSIMA-Erweiterungen, Teil 1.....	245
Abbildung 130: Vergleich der Selbstkoordinationsleistung von an [Eymann 2000] orientierten Szenarien mit Szenarien mit MACSIMA-Erweiterungen, Teil 2.....	246
Abbildung 131: Verkaufsabschlüsse und Verhandlungsabbrüche in an [Eymann 2000] orientierten Szenarien und Szenarien mit MACSIMA-Erweiterungen.....	246
Abbildung 132: Gesamtumsatz, Gesamtgewinn und Verhandlungsergebnisse für GMU und SMU bei Variation von Mutations-Rate u. –Weite.....	293
Abbildung 133: Anzahl an Verhandlungsabschlüssen und pro Verhandlung ausgetauschten Nachrichten für GMU(0,5-X) bei Variation der Mutations-Weite .	293
Abbildung 134: Variationskoeffizienten für die Mutations-Methoden GMU und SMU bei Variation von Mutations-Rate und –Weite	294
Abbildung 135: Variationskoeffizienten für die Mutations-Methoden GMU und SMU bei ausgewählten Parameter-Einstellungen.....	294
Abbildung 136: Gesamtübersicht kombinierter Effekte von variierenden Informationsaustausch- und Mutations-Einstellungen im Detail.....	295

Tabellenverzeichnis

Tabelle 1: Überblick über verschiedenen Supply-Chain-Modelle	55
Tabelle 2: Wesentliche Parameter für die Definition von Simulationsszenarien.....	97
Tabelle 3: Die Variablen der Datenstruktur PriceDistribution.....	111
Tabelle 4: Einstellungen für die Instanziierung des Adaptionprozesses eines Agenten..	154
Tabelle 5: Teil 1 der Notation für die MACSIMA-Simulationsszenarien (zu Basis- Einstellungen bzw. grundlegenden Namenskomponenten).....	170
Tabelle 6: Teil 2 der Notation für die MACSIMA-Simulationsszenarien (zu den Detail- Einstellungen für die evolutionären Lern-Algorithmen).....	171
Tabelle 7: Teil 3 der Notation für die MACSIMA-Simulationsszenarien (zu weiteren optionalen Einstellungen bzw. Namenskomponenten)	172
Tabelle 8: In den Ergebnistabellen verwendete Abkürzungen.....	186
Tabelle 9: Teil 1 der Struktur der Makro-Umsatz-Ergebnistabelle.....	187
Tabelle 10: Teil 2 der Struktur der Makro-Umsatz-Ergebnistabelle.....	188
Tabelle 11: Struktur der Makro-Preis-Ergebnistabelle.....	189
Tabelle 12: Wesentliche Spalten der Meso-Ergebnistabelle	191
Tabelle 13: Struktur der Mikro-Ergebnistabelle.....	193
Tabelle 14: Übersicht der mit AVALANCHE durchgeführten Experimente	280
Tabelle 15: Gen-Wert-Einstellungen für die AVALANCHE-Experimente	281
Tabelle 16: An [Eymann 2000] orientierte Szenario-Definitionen zur Untersuchung von Gen-Startwert-Effekten	282
Tabelle 17: Ausschnitt der Szenario-Definitionen mit einer breiten Verteilung von Gen- Startwerten.....	282
Tabelle 18: Untersuchte Kombinationen von Pool- und Populations-Größe.....	283
Tabelle 19: Basis-Parametrisierungen zur Testreihe „Relation von Pool- und Populations- Größe“	283
Tabelle 20: Szenario-Grundmuster der Testreihe zu den Effekten der Fitness-Methoden	284
Tabelle 21: Untersuchte Mutations-Parameter-Einstellungen.....	288
Tabelle 22: Innerhalb der Testreihe „Performanzvergleich“ untersuchte Szenarien	290
Tabelle 23: Makro-Preis-Tabelle (links) und Makro-Umsatz-Tabelle (rechts).....	296
Tabelle 24: Medio-Tabelle – Überblick	297
Tabelle 25: Medio-Tabelle – Details.....	298
Tabelle 26: Mikro-Tabelle – Simulationslauf 1	299
Tabelle 27: Mikro-Tabelle – Simulationslauf 2	299
Tabelle 28: Mikro-Tabelle – Simulationslauf 3	300
Tabelle 29: Mikro-Tabelle – Simulationslauf 4	300

Abkürzungsverzeichnis

AARIA	Autonomous Agents for Rock Island Arsenal
ABS	Agentenbasierte Simulation
ACL	Agent Communication Language
AI	Artificial Intelligence
aLLP	AbsoluteLowerPrice (im Deutschen: absolute untere Preisgrenze)
ANS	Agent Name Server
ANTS	Agent Network for Task Scheduling
AQ	Acquisitiveness (im Deutschen: Gewinnstreben)
AT	Algorithmic Trading (im Deutschen: Algorithmen-basierter Handel)
aULP	AbsoluteUpperPrice (im Deutschen: absolute obere Preisgrenze)
AVG	Average (im Deutschen: Durchschnitt)
AvP	Average Price Method (im Deutschen: Durchschnittspreis-Methode)
BaP	Basic Price Method (im Deutschen: Basispreis-Methode)
BDI	Belief, Desire and Intentions (im Deutschen: Glaube, Wünsche, Intention)
BiC	Binary Competition
bspw.	beispielsweise
bzw.	beziehungsweise
B2B	Business to Business
B2C	Business to Consumer
CA	Cash (im Deutschen: Geldbestand)
CMech	Coordination Mechanismus Agent (im Deutschen: Koordinationsmechanismusagent)
CMM	Coordination Matchmaker Agent (im Deutschen: Koordinations-Matchmaker-Agent)
CNP	Contract Net Protocol (im Deutschen: Kontrakt-Netz-Protokoll)
COOL	COOrdination Language
CPFR	Collaborative Planning, Forecasting and Replenishment
CPU	Central Processing Unit
CRM	Customer Relationship Management (im Deutschen: Kundenbeziehungsmanagement)
csv	Character-separated values
CX	Coordination Execution Agent (im Deutschen: Koordinationsausführungsagent)
DAI	Distributed Artificial Intelligence (im Deutschen: VKI, s. u.)
DASCh	Dynamical Analysis of Supply Chains
DC	Delta_Change
DFG	Deutsche Forschungsgemeinschaft
DJ	Delta_Jump
DESK	Density-based Emergent Scheduling Kernel
DFKI	Deutsches Forschungszentrum für Künstliche Intelligenz
d. h.	das heißt
DMS	Deterministic Minority Selection
DPS	Distributed Problem Solving
DtA	Deterministic Average (im Deutschen: Deterministische Durchschnittsauswahl)
DtS	Deterministic Selection (im Deutschen: Deterministische Auswahl)
DwS	Delete worst Selection (im Deutschen: Selektion mit Löschen des schlechtesten Individuums)

E	Elitism (im Deutschen: Elitismus)
E	Everybody
EC	Electronic Commerce
ECR	Efficient Consumer Response
engl.	englisch
E-SCM	E-Supply-Chain-Management
EA	Evolutionärer Algorithmus
EPK	Ereignis-gesteuerte Prozesskette
etc.	et cetera
evtl.	eventuell
Ext	Externes Lernen
f.	folgende (Seite)
ff	folgende (Seiten)
FIPA	Foundation for Intelligent Physical Agents
FPSB	First-Price Sealed Bid (im Deutschen: verdeckte Ersterpreis-Auktion)
G	Gruppe
GE	Geldeinheiten
ggf.	gegebenenfalls
GMU	Generelle Mutation
GUI	Graphical User Interface (im Deutschen: Benutzeroberfläche)
Hrsg.	Herausgeber
H-MAS	Holonisches Multiagentensystem
HFF	Highest-Fitness-First-Verfahren
HFT	Hochfrequenzhandel
IC	Initial Cash (im Deutschen: Anfangsausstattung mit Geld)
i. d. R.	in der Regel
Int	Internes Lernen
ISA	Interoperabilitäts-Service-Agent
IT	Informationstechnologie
JIT	Just In Time (im Deutschen: gerade rechtzeitig)
KI	Künstliche Intelligenz
KPI	Key Performance Indicator (im Deutschen: Leistungskennzahl)
KQML	Knowledge Query and Manipulation Language
LLP	LowerLimitPrice (im Deutschen: Preisuntergrenze)
MASCOT	Multi-Agent Supply Chain Coordination Tool
MACSIMA	Multi-Agent Supply Chain Simulation Framework
MAGMA	Minnesota Agent Marketplace Architecture
MAGNET	Multi-Agent Negotiation Testbed
MAS	Multiagenten-System
MdP	Medium Price Method
MIS	Managementinformationssystem
Mix	Gemischtes Lernen
N	NoExchange
NoR	No Recombination (im Deutschen: ohne Rekombination)
nPC	n-Point Crossover (im Deutschen: n-Punkt-Crossover)
OEM	Original Equipment Manufacturer (im Deutschen: Erstausrüster)
oLLP	Original Lower Limit Price (im Deutschen: originale untere Preisgrenze)
oULP	Original Upper Limit Price (im Deutschen: originale obere Preisgrenze)
P	Pool-Größe
PC	Personal Computer
PD	Price Distribution (Datenstruktur)

PmA	Price Minus Average Method
Pn	Populations-Größe
PPS	Produktionsplanung und -steuerung
PW	ProfitWish (im Deutschen: Gewinnwunsch)
RaC	Random Crossover (im Deutschen: zufällige Kreuzung)
RMF	Relative Medium Price Fraction Method
RP	Reputation
RWS	Roulette Wheel Selection (im Deutschen: Auswahl mittels Roulettekessel)
s.	siehe
s. a.	siehe auch
S	Stufe
sce	Endung für die Szenario-Dateien
SCGA	Generic Supply Chain Agent (im Deutschen: Generischer Supply-Chain-Agent)
SCM	Supply Chain Management
SF	Satisfaction (im Deutschen: Zufriedenheit)
SMU	Single Mutation (im Deutschen: Einzelmutation)
sog.	sogenannt
STDEA	Dezentraler evolutionärer Algorithmus von [Smith & Taylor 1998] (Smith and Taylor's distributed evolutionary algorithm)
T	Ticket-Limitierungs-Verfahren
s. u.	siehe unten
u. a.	unter anderem
ULP	UpperLimitPrice (im Deutschen: Preisobergrenze)
UML	Unified Modeling Language
usw.	und so weiter
u. U.	unter Umständen
vgl.	vergleiche
VKI	Verteilte Künstliche Intelligenz
VPL	Verteiltes Problemlösen
VPS	Verteiltes Problemlöse-System
VU	Virtuelles Unternehmen
wAVG	weighted_Average (im Deutschen: gewichteter Durchschnittspreis)
WE	Weak Elitism (im Deutschen: Schwacher Elitismus)
WM	WeightMemory
XETRA	Exchange Electronic Trading
z. B.	zum Beispiel

1 Einleitung

1.1 Motivation und Problemstellung

Eine *Supply Chain (Versorgungskette)* ist eine Kette von Geschäftsentitäten, die gemeinsam bestimmte Produkte beschaffen, weiterverarbeiten und distribuieren.

Traditionelle Supply Chains sind von ihrer Natur her relativ linear, sequentiell und statisch aufgebaut. Für häufig wiederkehrende Warenflüsse innerhalb von Supply Chains werden zwischen den teilnehmenden Unternehmen zumeist langfristige Rahmenverträge geschlossen, weil die Kosten zur flexiblen Auswahl und Einbindung neuer Teilnehmer, beispielsweise für die Deckung eines kurzfristig auftretenden unerwarteten Bedarfs, die möglichen Vorteile und Einsparungen in den meisten Fällen übertreffen.

Entsprechende Versorgungsketten sind daher sehr inflexibel und ermöglichen keine schnellen Reaktionen auf sich kurzfristig ergebende Marktchancen. Insbesondere sind in traditionellen Supply Chains außerhalb der rahmenvertraglich zentral festgelegten Grenzen dezentrale autonome Entscheidungen der teilnehmenden Unternehmen nur in begrenztem Maße möglich. Da heutige Märkte und Wertschöpfungsstrukturen nicht nur global, sondern auch vernetzt und hochdynamisch sind, sind Versorgungsketten aber dazu gezwungen, immer flexibler und schneller auf sich ändernde Kundenanforderungen und Nachfrageschwankungen zu reagieren und ihre Adaptivität insgesamt wesentlich zu steigern.

Eine Möglichkeit, dies zu erreichen, besteht darin, sich auf der Grundlage von flexiblen kooperativen Partnerschaften zu organisieren. Insbesondere besteht durch die Entwicklungen im E-Commerce die Möglichkeit, die statischen Organisationsformen traditioneller Supply Chains in zunehmenden Maße durch sich über elektronische Märkte dynamisch bildende Strukturen zu ersetzen, die sich besser an unvorhergesehene Nachfrageschwankungen, Produktvariationen und kurze Produktlebenszyklen adaptieren können.

Aus diesen Gründen wird davon ausgegangen, dass sogenannte *Supply Webs* [Laseter 1998] – das heißt B2B-basierte dynamische Netzwerke von Supply-Chain-Einheiten – oder *adaptive Supply Chains* [Christopher 2011] die heutigen statischen Supply Chains in einem zunehmenden Ausmaß ersetzen werden. Dabei handelt es sich um flexible Organisationsformen, die auf Elementen der Selbstorganisation fußen.

Idealerweise würden in solchen, sich selbst koordinierenden Organisationsformen Aufgaben mit hohen Transaktionskosten weitgehend automatisiert von Informationssystemen übernommen werden. Sie sollten Güter und deren zugehörige aktuelle Marktpreise kennen und in der Lage sein, die Güter selbständig zu kaufen und zu verkaufen sowie logistische Prozesse autonom zu steuern.

Eine solche Automatisierung führt zur Senkung von Transaktionskosten, ermöglicht das flexible Handeln ohne langfristige Rahmenverträge und trägt zur Agilität und Adaptivität einer Versorgungs- beziehungsweise Wertschöpfungskette bei.

Daher wird von vielen Forschern vorgeschlagen, die Supply-Chain-Einheiten beziehungsweise Geschäftsentitäten (zum Beispiel Lieferanten, Produzenten, Einzel- und Großhändler

sowie Logistik-Dienstleister) als intelligente Agenten zu modellieren, um diesen Anforderungen gewachsen zu sein. Agenten bieten den Vorteil, auf Änderungen in ihrem Umfeld schnell und flexibel reagieren zu können, da sie in der Lage sind, Aufgaben im Auftrag ihrer Nutzer autonom durchzuführen.

Aus der erweiterten Autonomie der Supply-Chain-Einheiten und der Flexibilisierung und Dynamisierung ihrer Interaktionen resultiert aber auch ein zu lösendes Koordinationsproblem. Denn ohne eine effiziente und effektive Koordination der Akteure können die höhere Dynamik und die weitgehend verteilten Prozesse, die einem Supply Web innewohnen, auf Kostensteigerungen und eine Zunahme von Verzögerungen anstatt auf eine höhere Leistungsfähigkeit und Flexibilität hinauslaufen.

Eine Quelle möglicher Konflikte besteht insbesondere darin, dass die autonomen und eigennützigen Akteure gleichzeitig in mehreren Wertschöpfungsstrukturen aktiv sein können, so dass nicht angenommen werden kann, dass sie ein übergeordnetes gemeinsames Ziel verfolgen beziehungsweise zu dessen Erfüllung in gleichem Maße beitragen. Daher müssen geeignete Organisationsstrukturen und Koordinationsmechanismen gefunden werden, die dabei helfen, die lokalen Aktivitäten der Supply-Chain-Agenten geeignet zu koordinieren.

Entsprechende Koordinationsmechanismen können zentraler Art sein und zum Beispiel durch einen Auktionator instanziiert werden, der Angebot und die Nachfrage zentral bündelt. Dieser kann die Koordination eigennütziger, nutzenorientierter Akteure unter gewissen Rahmenbedingungen – sogar theoretisch abgesichert – effizient und nutzenmaximierend organisieren.

Im Anwendungsszenario eines agilen adaptiven Wertschöpfungsnetzwerks könnte die Rolle der zentralen Auktionator- oder Koordinator-Instanz dabei von agentenbasierten Koordinationsservern (siehe Kapitel 4.3) übernommen werden, die Teil einer agentenbasierten Koordinationsinfrastruktur sind, die den Supply-Chain-Entitäten – ob menschlich oder von einer KI gesteuert – eine Reihe von Koordinationsdienstleistungen (zum Beispiel Matchmaking Services) und Koordinationsmechanismen zur Verfügung stellt.

Noch flexibler und effizienter können die Elemente einer solchen Koordinationsinfrastruktur gestaltet werden, wenn man sie nicht nur agentenbasiert realisiert, sondern ihren Entwurf auch an den Design-Prinzipien für *holonische Multiagentensystemen (H-MAS)* ausrichtet (siehe Abschnitt 2.6). Entsprechende Modellierungsansätze wurden von der Multiagenten-Forschungsgruppe des DFKI unter der Leitung von Prof. Dr. Jörg Siekmann und Dr. Klaus Fischer entwickelt (siehe u. a. [Gerber et al. 1999a; Gerber et al. 1999b; Fischer et al. 2003]) und in vielen Praxisprojekten erfolgreich eingesetzt (siehe Abschnitt 2.6).

Auf dieser Basis wurden im Rahmen dieser Arbeit holonische Geschäftsagenten und Koordinationsserver-Agenten für den Einsatz in elektronischen Märkten und Wertschöpfungsnetzwerken entwickelt. Die Struktur der Agenten und ihr Zusammenspiel werden in Abschnitt 4.3 erläutert. Dabei handelt es sich um die Erweiterung und Aktualisierung von Konzepten, die auf eigenen Veröffentlichungen beruhen (siehe [Ruß & Gerber 2000; Ruß & Gerber 2001b; Ruß & Gerber 2001c; Ruß 2001; Ruß et al. 2003; Ruß et al. 2001]). Beiträge zur Entwicklung der vorgestellten holonischen Agenten wurden im Rahmen der betreuten Diplomarbeiten [Deßloch 2002] und [Puin 2004] geleistet.

Eine weitere Möglichkeit zur Koordination der Supply-Chain-Agenten besteht darin, einen dezentralen marktlichen Ansatz zu verfolgen, den Moulin (siehe [Moulin 1995]) *Modus direkter Verhandlungen* nennt. Bei diesem erreichen die Teilnehmer beziehungsweise Agenten eine Lösung des Koordinationsproblems ohne die Unterstützung durch eine zentrale Koordinator-Instanz, indem sie direkt miteinander interagieren. Mittels bilateraler Verhandlungen stimmen sie sich über den Tausch von Ressourcen und Gütern ab. Finden Transaktionen

statt, so variieren die Preise von Transaktion zu Transaktion, da sie sich bei dieser Art der Koordination gemäß des Verlaufs der jeweiligen Verhandlung individuell bilden. Dabei orientieren sich die Akteure an einem auf ihren eigenen Erfahrungen beruhenden „gefühlten“ Marktpreis.

Eine zentrale Frage ist, zu welchen Ergebnissen ein solcher Modus direkter Verhandlungen als dezentraler Koordinationsmechanismus führt, wenn die autonomen, eigennützig agierenden Teilnehmer an einem Wertschöpfungsnetzwerk sich über ihn selbständig und frei koordinieren.

Dies ist nur schwer zu beantworten. Insbesondere liegt auf der Hand, dass der Ausgang eines solchen Experiments nicht nur von den Teilnehmern abhängt, sondern auch von ihren Fähigkeiten, sich beim Verwenden des Koordinationsmechanismus strategisch geschickt zu verhalten. Zudem haben auch die Menge und die Qualität an Informationen über aktuelle Marktpreise, die die Akteure im Modus direkter Verhandlungen nutzen können, einen nicht unbedeutenden Einfluss auf ihr Abschneiden.

Um die aufgeworfene Fragestellung unter variierenden Einflüssen wissenschaftlich zu untersuchen, bieten sich daher Simulationen an. Denn bei ihnen kann man sicher sein, alle Einflussparameter korrekt zu erfassen. Zudem lassen sich mit ihrer Hilfe Experimente bezüglich variierender Einflüsse und Fähigkeiten der Teilnehmer ohne finanzielles Risiko durchführen, bevor man entsprechende Konzepte auf elektronischen Märkten in der Realität ausprobiert. Im Idealfall lassen sich aus der Evaluation der Simulationsergebnisse Design-Empfehlungen für die Konzeption und Implementierung realer Systeme ableiten, so dass Fehler vermieden werden.

Wesentliche Beiträge zur Untersuchung von Fragestellungen der genannten Art konnte Professor Dr. Torsten Eymann leisten, der sich eingehend mit der katallaktischen Selbstkoordination in agentenbasierten Wertschöpfungsnetzwerk-Szenarien beschäftigt hat.

In seinen zentralen Arbeiten (siehe [Eymann 2000; Eymann 2003; Eymann 2013]) nutzen die Agenten zur marktlichen Selbstkoordination ein bilaterales Verhandlungsprotokoll, in dessen Verlauf sie auf der Grundlage von verschiedenen Strategieparametern Entscheidungen treffen. Eine rudimentäre Anpassung ihrer Strategieparameter an sich ändernde Marktbedingungen ist den Geschäftsagenten anhand eines einfachen dezentralisierten Adaptionsverfahrens möglich (siehe [Smith & Taylor 1998]).

Professor Dr. Eymann hat anhand des implementierten Multiagentensystems *AVALANCHE (Agent-based Value Chain Experiment)* in [Eymann 2000] gezeigt, dass Supply-Chain-Agenten beziehungsweise Geschäftsagenten auf dieser Basis dazu in der Lage sind, sich über die Teilnahme an einem elektronischen Markt sinnvoll selbst zu koordinieren. Entlang einer Wertschöpfungskette bilden sich auf der Grundlage der in *AVALANCHE* integrierten Konzepte für die von den Geschäftsagenten angebotenen und nachgefragten Güter Preise und ein Güterstrom. In [Eymann 2003; Eymann 2013] wird darüber hinaus ausführlich erläutert, wie ein agentenbasierter elektronischer Marktplatz und daran teilnehmende Software-Agenten in Java realisiert werden können. Die grundlegenden und wichtigen Ansätze und Erkenntnisse, die Professor Dr. Eymann vorgestellt hat, bieten jedoch Raum für darauf aufbauende, ergänzende Forschungsfragen.

Insbesondere ist dabei von Interesse, wie sich die von Eymann im *AVALANCHE*-System realisierten Ansätze erweitern lassen, um die Agenten noch realitätsnäher miteinander interagieren zu lassen und im Idealfall eine Selbstkoordinationsleistung der Geschäftsagenten zu erzielen, die einen Einsatz der entwickelten Konzepte in realen Supply-Chain-beziehungsweise Wertschöpfungsnetzwerk-Szenarien sinnvoll erscheinen lässt.

Die Effekte unterschiedlicher Erweiterungen auf die Selbstkoordinationsfähigkeiten von agentenbasierten Wertschöpfungsnetzwerken sollten dabei simulativ untersuchbar, quantitativ und statistisch analysierbar sowie miteinander gut vergleichbar sein.

Zur Beantwortung entsprechender Fragestellungen wurde das *MACSIMA*-System (*Multi Agent Supply Chain Simulation Framework*) entwickelt. *MACSIMA* erlaubt die Simulation agentenbasierter Wertschöpfungsnetzwerke mit nutzenbasierten Geschäftsagenten, die die Fähigkeit besitzen, aus gemachten Erfahrungen mit Hilfe komplexer *evolutionärer Algorithmen (EA)* zu lernen und Verhandlungserfahrungen sowie Kenntnisse über Marktpreise innerhalb von Agentengruppen auszutauschen. *MACSIMA* beinhaltet die in [Eymann 2000; Eymann 2003; Eymann 2013] beschriebenen Konzepte und die Funktionalitäten des *AVALANCHE*-Systems, bietet aber darüber hinaus auch eine Reihe von Erweiterungen.

Wesentliche Erweiterungen, die im Vergleich zu *AVALANCHE* in das *MACSIMA*-System integriert und untersucht werden, bestehen in einer signifikanten Verbesserung der Lernfähigkeiten der Geschäftsagenten mittels evolutionärer Algorithmen, die sehr flexibel und fein einstellbar sind. Zudem werden die sozialen Fähigkeiten der *MACSIMA*-Agenten so erweitert, dass sie Informationen über auf dem Markt getätigte Transaktionen und die dabei eingesetzten erfolgreichen Verhandlungsstrategien mit anderen Agenten innerhalb von definierbaren Gruppen gezielt austauschen können. Dabei können insbesondere auch der Umfang und die Häufigkeit des Informationsaustauschs in *MACSIMA* eingestellt werden.

Die ausgetauschten Informationen können dann von den *MACSIMA*-Agenten zur Adaption ihrer Verhandlungsstrategien genutzt werden. Im Unterschied zum *AVALANCHE*-System können *MACSIMA*-Agenten durch das realisierte Konzept des *2-Pool-separierten Lernens* die von ihnen für den Kauf und Verkauf von Gütern genutzten Verhandlungsstrategien getrennt voneinander adaptieren, so dass ihre Lernprozesse zielgerichteter ablaufen können. Die genannten Erweiterungen werden in Kapitel 5 ausführlich beschrieben.

Im *AVALANCHE*-System waren den Geschäftsagenten im Vergleich dazu nur eingeschränkte Fähigkeiten zur Adaption und zum Informationsaustausch auf Basis des *STDEA*-Adaptionsverfahrens (siehe Abschnitt 5.4.6) gegeben.

Daneben beinhaltet Kapitel 5 auch weitere neuartige Konzepte. Dazu gehört zum Beispiel ein Preisautomat, der korrigierend eingreift, wenn es wahrscheinlich ist, dass sich die „gefühlten“ Marktpreise eines Agenten zu weit von den tatsächlichen Marktpreisen entfernt haben (siehe Abschnitt 5.3.3.7). Zudem wurden im Rahmen der Arbeit mit den Varianten der *Deterministic Minority Selection* auch besondere Selektions-Methoden entwickelt, die von den Agenten beim evolutionären Lernen verwendet werden können (siehe Abschnitt 5.4.3.2). Daneben wurden weitere Verfahren als optionale Einstellmöglichkeiten realisiert, mit denen das evolutionäre Lernen optimiert werden kann (zum Beispiel durch den Einsatz des *Highest-Fitness-First-Verfahrens*, das in Abschnitt 5.4.5.1 beschrieben wird oder des in Abschnitt 5.4.5.2 erläuterten *Ticket-Limitierungs-Verfahrens*).

Da die Simulationsläufe in *AVALANCHE* und *MACSIMA* nichtdeterministisch verlaufen, ergab sich für die vorliegende Arbeit der Anspruch, die mit *MACSIMA* erzielten Ergebnisse quantitativ und statistisch abzusichern. Aus diesem Anspruch leiten sich die in Kapitel 6 vorgestellte Simulationsinfrastruktur sowie die in Kapitel 7 erläuterte Evaluationsmethodik ab, mit deren Hilfe die in Kapitel 8 durchgeführten Evaluationen nicht nur stichprobenhaft, sondern auf der Basis umfangreicher Testreihen automatisiert durchgeführt wurden.

Um eine detaillierte quantitative und statistische Evaluation der Simulationsergebnisse durchführen zu können, wurden im Vergleich zu *AVALANCHE* auch eine Vielzahl an ergänzenden Ergebnis- und Statistik-Kennzahlen eingeführt (siehe Abschnitt 7.3), die für Simulationsszenarien auch über mehrere Simulationsläufe hinweg aggregiert und berechnet

werden. Darüber hinaus wurden auch mehrere Evaluationsebenen, nämlich eine Makro-, eine Meso- und eine Mikro-Ebene eingeführt, um Simulationsergebnisse aus der Perspektive des Gesamtsystems, aber auch aus der Sicht von Wertschöpfungsstufen und Agentengruppen sowie von einzelnen Agenten bewerten zu können. Auf dieser Basis können die unterschiedlichen Auswirkungen von heterogen definierten Simulationsszenarien auf mehreren Evaluationsebenen differenziert betrachtet werden.

Darüber hinaus verfolgt MACSIMA einen rundenbasierten Simulationsansatz anstatt eines zeitbasierten Ansatzes, wie er in AVALANCHE realisiert wurde. Dies hat den Vorteil, dass Ressourcenengpässe, die zur Laufzeit einer Simulation auf einem dafür genutzten Server auftreten, in der Regel keine störenden Einflüsse auf die Simulationsergebnisse entwickeln können.

Die wesentliche mit MACSIMA untersuchte Forschungsfrage ist dabei, wie sich die in MACSIMA realisierten Erweiterungen der grundlegenden in AVALANCHE realisierten Ansätze auf die Selbstkoordinationsleistung von agentenbasierten Wertschöpfungsnetzwerken auswirken. Hierbei wird quantitativ genau analysiert, welche Vorteile und Leistungssteigerungen sich aus ihrer Verwendung ergeben. Aus den hieraus gewonnenen Erkenntnissen lassen sich dann Empfehlungen für die Gestaltung von Geschäftsagenten für den operativen Einsatz in realen Wertschöpfungsnetzwerk-Szenarien ableiten.

Beiträge zur Entwicklung von MACSIMA wurden im Rahmen mehrerer betreuter Diplom-, Bachelor- und Masterarbeiten geleistet (siehe [Zenner 2004; Graeber 2006; Walz 2007; Steudter 2010; Steudter 2013]). Mit dem System erzielte Forschungsergebnisse wurden auf mehreren Konferenzen und Workshops präsentiert und in den zugehörigen Tagungsbänden sowie in einem Buch aus der Reihe *Lecture Notes in Business Information Processing (LNBIP 98)* des Springer-Verlags veröffentlicht (siehe [Ruß & Walz 2009a; Ruß & Walz 2009b; Ruß & Walz 2009c; Ruß & Walz 2009d]). Das in Kapitel 5 beschriebene MACSIMA-System, die zu ihm gehörige Simulationsinfrastruktur (siehe Kapitel 6) und Evaluationsmethodik (siehe Kapitel 7) sowie die damit in Kapitel 8 erzielten Evaluationsergebnisse stellen den Kern dieser Arbeit dar.

1.2 Aufbau der Arbeit

Da die bearbeiteten Fragestellungen einen interdisziplinären Charakter aufweisen, gibt die Arbeit zunächst in den Kapitel 2 und 3 eine Einführung in die relevanten Grundlagen.

Während in Kapitel 2 die Grundlagen der Verteilten Künstlichen Intelligenz und der Multiagentensysteme erläutert werden, behandelt Kapitel 3 die ökonomischen Grundlagen, zu denen unter anderem das Supply Chain Management sowie die marktliche Koordination gehören.

Kapitel 4 gibt einen Überblick über den Einsatz von Agenten in ökonomischen Wertschöpfungsstrukturen und stellt als ersten Beitrag der Arbeit das Konzept holonisch strukturierter Geschäftsagenten und Koordinationsserver-Agenten für den Einsatz in elektronischen Märkten und marktlich koordinierten Wertschöpfungsnetzwerken vor.

Anschließend wird in Kapitel 5 das MACSIMA-System detailliert beschrieben, mit dem katalaktische Selbstkoordinationsprozesse in agentenbasierten Wertschöpfungsnetzwerken simuliert werden können.

Zur Durchführung von Simulationen und zur sowohl qualitativen als auch quantitativen Analyse der generierten Simulationsergebnisse wurden eine Simulationsinfrastruktur und eine Evaluationsmethodik entwickelt, denen sich die Kapitel 6 und 7 widmen.

Abschließend wird in Kapitel 8 untersucht, welche Auswirkungen sich auf die Koordinationsleistung ergeben, wenn die von den Agenten im dezentralen Selbstkoordinationsprozess einsetzbaren Lern- und Informationsaustauschfähigkeiten variiert und miteinander sowie mit weiteren entwickelten Konzepten und Verfahren kombiniert werden.

2 Verteilte KI & Multiagenten-Systeme

2.1 Einleitung und Übersicht

Dieses Kapitel beginnt mit einem kurzen Überblick über das Forschungsgebiet der *Verteilten Künstlichen Intelligenz (VKI)* bzw. *Distributed Artificial Intelligence (DAI)* in Abschnitt 2.2. Dabei werden die Teilgebiete der VKI, nämlich das *Verteilte Problemlösen (VPL)* sowie das Forschungsgebiet der *Multiagenten-Systeme (MAS)* erläutert und voneinander abgegrenzt.

Anschließend wird in Abschnitt 2.3 der Begriff des *Software-Agenten* grundlegend definiert. Es handelt sich dabei um Software-Objekte mit speziellen Eigenschaften, die in der Folge ausführlich erläutert werden.

In Abhängigkeit von ihren Fähigkeiten und den sie konstituierenden Komponenten lassen sich Agenten in eine Reihe unterschiedlicher Klassen und Typen einordnen, von denen einige für diese Arbeit besonders wichtige in Abschnitt 2.4 vorgestellt werden.

Einzelne Agenten können miteinander interagieren und dadurch Agenten-Gesellschaften beziehungsweise Multiagenten-Systeme bilden, woraus in der Regel Kooperationsmöglichkeiten und Koordinationsprobleme entstehen. Daher werden in Abschnitt 2.5 die Begriffe Kooperation, Kommunikation und Koordination mit ihrer Bedeutung für den Kontext der Multiagenten-Systeme erläutert. In diesem Zusammenhang wird die Relevanz der Koordinationstheorie und der Mechanismus-Design-Theorie für die Lösung von Koordinationsproblemen in Multiagenten-Systemen verdeutlicht.

Die Anwendung holonischer Konzepte auf das Design von Multiagenten-Systemen kann dabei helfen, Agenten effizient zu koordinieren und ermöglichen die Schaffung effizient organisierter, skalierbarer Multiagenten-Systeme [Jennings 1999]. Daher werden in Abschnitt 2.6 die grundlegenden Konzepte *Holonischer Multiagentensysteme (H-MAS)* erläutert. Diese sind gerade für betriebswirtschaftliche Anwendungsszenarien von besonderer Relevanz.

Abschließend wird in Abschnitt 2.7 die Bedeutung von marktlichen Koordinationsmechanismen zur Lösung von Koordinationsproblemen zwischen autonomen nutzenbasierten Geschäftsagenten besprochen, wie sie auf elektronischen Märkten, in Wertschöpfungsnetzwerken und in sonstigen betriebswirtschaftlichen MAS-Anwendungsdomänen anzutreffen sind.

2.2 Verteilte Künstliche Intelligenz

Bei den Forschungsgebieten *Intelligente Agenten* und *Multiagenten-Systeme* handelt es sich um noch recht junge Forschungsgebiete, die Teilgebiete der *Künstlichen Intelligenz (KI)* sind. Dieses Teilgebiet der Informatik, dessen Ursprünge bis in die Mitte der Fünfzigerjahre des letzten Jahrhunderts zurückreichen, hat zum Ziel, menschliche Intelligenzleistungen soweit zu automatisieren und in Computersystemen nachzubilden, dass diese Computersysteme zu menschenähnlichen intelligenten Problemlösungen fähig sind bzw. menschliche Problemlösungsfähigkeiten sogar noch übertreffen.

Dies ist in Spezialbereichen bereits heute schon der Fall, wie die Leistungen des Schachcomputers Deep Blue [Hsu 2002] oder des WATSON-Systems¹ verdeutlichen, die beide von IBM entwickelt wurden.

Bis in die frühen Achtzigerjahre hinein hat die traditionelle KI-Forschung in erster Linie untersucht, wie Ausschnitte von intelligentem menschlichen Verhalten simuliert werden können, indem einzelne KI-Methoden wie heuristische Suche, Methoden der Wissensrepräsentation, Planung, Lernen, Perzeption sowie die Erzeugung und das Verständnis gesprochener Sprache genutzt werden.

Innerhalb der letzten drei Jahrzehnte hat sich diese traditionelle methodenzentrierte Interpretation von KI zu einem Verständnis von KI als „... the science of agent design ...“ erweitert (s. [Caglayan & Harrison 1998; Ferber 2001]). Dabei versteht man unter einem Agenten

“... anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors.” [Russell & Norvig 1995]

Das heißt, dass gegenwärtige KI-Forschung untersucht, wie ein einzelner Agent gebaut werden kann, der in seiner Umwelt - in Abhängigkeit von seiner Perzeption dieser Umwelt und seinen Interaktionsmöglichkeiten mit ihr – mit Hilfe einer geeigneten Kombination der einzelnen KI-Methoden ein „erfolgreiches“ Verhalten entwickelt.

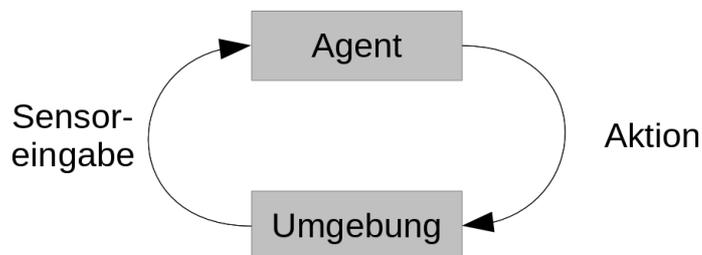


Abbildung 1: Interaktion eines Agenten mit seiner Umwelt²

Darüber hinaus hat sich die gegenwärtige MAS-Forschung auch mit den sozialen Aspekten des Agentendesigns zu beschäftigen, nämlich dem Design und dem Verhalten von Gruppen von Agenten, welches sich aus deren Interaktionen ergibt.

Während die Entwicklung und Erforschung intelligenter Agenten bis in die Siebzigerjahre des letzten Jahrhunderts zurückreicht, entstand durch die Untersuchung dieser sozialen Aspekte seit den Achtzigerjahren zunehmend das Forschungsgebiet der Verteilten Künstlichen Intelligenz als Teilgebiet der Künstlichen Intelligenz.

Dieses wird von [Bond & Gasser 1988] als „... the subfield of AI concerned with concurrency in AI computations, at many levels ...“ charakterisiert.

Dessen Entwicklung wurde insbesondere durch die Entwicklung leistungsfähiger Computersysteme und den Siegeszug der Computer-Netzwerke sowie des E-Commerce vorangetrieben. Die VKI-Forschung wird in einer vernetzten Welt zunehmend bedeutsamer:

The role of DAI systems with cooperative and competitive agents is likely to increase ... allowing autonomous, even competitive, enterprises to cooperate through the on-line, dynamic establishment of contracts ...“ [Sandholm 1993]

¹ Eine Beschreibung des WATSON-Systems bietet [Ferruci et al. 2012].

² Quelle: In Anlehnung an [Russell & Norvig 2003, S. 33]

Der Forschungsbereich der Verteilten Künstlichen Intelligenz lässt sich laut [Bond & Gasser 1988] in zwei grundlegende Hauptbereiche unterteilen:

- **Verteiltes Problemlösen (VPL):**

Forschung in diesem VKI-Bereich, auch *Distributed Problem Solving (DPS)* genannt, untersucht, „... *how the work of solving a particular problem can be divided among a number of modules, or 'nodes' that cooperate at the level of dividing and sharing knowledge about the problem and about the developing of a solution.*“ [Bond & Gasser 1988]

- **Multiagenten-Systeme (MAS):**

Forschung in diesem VKI-Bereich beschäftigt sich mit “... *coordinating intelligent behaviour among a collection of autonomous (possibly heterogeneous) intelligent agents:*” [Kraus et al. 1995]

Im Folgenden werden diese beiden Hauptbereiche der VKI-Forschung weiter erläutert.

2.2.1 Verteiltes Problemlösen

Verteilte Problemlöse-Systeme (VPS) sind auf die Lösung eines speziellen Problems ausgerichtet [von Martial 1992]. Die von VPS zu lösenden Probleme zeichnen sich dadurch aus, dass sie sich auf natürliche Weise, z. B. auf der Basis funktionaler oder regionaler Abhängigkeiten, in mehrere unabhängig voneinander lösbare Teilprobleme zerlegen lassen, deren Bearbeitung anschließend gemäß eines fest vorgegebenen Zuordnungsschemas auf eine Menge kooperativer *Problemlöse-Module* verteilt wird.

Zu betonen ist an dieser Stelle, dass es sich bei diesen Problemlöse-Modulen nicht unbedingt um Agenten handelt, da ihnen in der Regel wichtige Eigenschaften fehlen, die einen Agenten von anderen Software-Systemen differenzieren (siehe hierzu die Ausführungen zu Agenten-Eigenschaften in Abschnitt 2.3.2) oder die entsprechenden Eigenschaften bei ihnen nur schwach ausgeprägt sind. Zumeist mangelt es ihnen bereits an der für einen Agenten wesentlichen Eigenschaft der *Autonomie*.

VPS verfolgen bei der Problemlösung also einen zentral gesteuerten Top-Down-Ansatz, bei der mehrere gleichartige oder sich unterscheidende Problemlöse-Module eines VPS kollektiv ein Problem lösen und der die folgenden Schritte umfasst:

1. Ein Gesamtproblem wird analysiert (*Analyse und ggf. Zielspezifikation*) und
2. hierarchisch in Teilprobleme dekomponiert, die wiederum möglicherweise in weitere Teilprobleme zerlegt werden (*Dekomposition bzw. Zerlegung*), bis diese
3. jeweils spezialisierten Problemlöse-Modulen zugewiesen werden können, die auf die Lösung dieser Teilprobleme hin optimiert sind (*Allokation bzw. Zuweisung*) und
4. eine *Lösung der Teilprobleme* durchführen (*Erledigung*). Anschließend erfolgt in einem letzten Schritt
5. die Zusammenfügung der Lösungen der Teilprobleme zu einer Gesamtlösung (*Ergebnissynthese*).

Gemäß [Durfee & Rosenschein 1994] kann bei VPS prinzipiell angenommen werden, dass, wenn man die verteilten Problemlöse-Module als Agenten bezeichnen „will“, sich diese *kooperativ* verhalten. Dies bedeutet, dass sie *benevolent* handeln und ein gemeinsames Ziel verfolgen.

Dies rührt daher, dass in der Regel ein zentraler Designer für die Konzeption des gesamten VPS und seiner Komponenten verantwortlich ist, der deren Problemlöse-, Entscheidungs- und Interaktionsverhalten im Voraus fest definiert.

2.2.2 Multiagenten-Systeme

Im Gegensatz zum Verteilten Problemlösen beschäftigt sich das Forschungsgebiet der *Multiagenten-Systeme* mit Systemen aus autonomen, heterogenen, selbständig entscheidenden und häufig physisch (z. B. über ein Netzwerk) verteilten Software-Entitäten, welche die wesentlichen, einen Agenten ausmachenden Eigenschaften aufweisen³ (s. Abschnitt 2.3.2).

Die MAS-Forschung geht daher weder von der Annahme aus, dass Agenten in jedem Fall kooperativ handeln und die ihnen zur Bearbeitung angebotenen Aufgaben stets annehmen, noch von der Annahme, dass sie von einem Designer auf das Erreichen eines gemeinsamen Ziels hin ausgerichtet und konzipiert worden sind.

Insbesondere ist in ihnen weder die Problemdekomposition noch die Zuteilung zu lösender Teilprobleme bzw. von Aufgaben zentral organisiert oder a priori anhand eines Schemas vorgegeben. Stattdessen erfolgen in MAS die Entscheidungen und die Handlungen der Agenten dezentral und verteilt, so dass sich die Mitglieder eines MAS auf Basis ihrer Interaktionen selbst organisieren und ihre Ziele, Pläne und Aktionen miteinander abstimmen müssen, um koordiniert zu handeln.

“MAS studies how individual, self-interested decision makers might discover (...) stable, predictable, and desirable ways of interacting among themselves.” [Durfee & Rosenschein 1994]

Da Agenten in Multiagenten-Systemen in der Regel über eine eigenständige Repräsentation ihrer Ziele verfügen, ist zu erwarten, dass sie bei ihren Interaktionen und der Auswahl und Durchführung ihrer Aufgaben grundsätzlich *rational* und damit auch *eigennützig*⁴ handeln, da sie primär danach streben, ihre eigenen Ziele zu erreichen und dadurch ihren damit verbundenen Nutzen zu maximieren.

“MAS research borrowed from the social science literature the underlying assumption that an agent should be rational: That, whatever it is doing, it should endeavor to maximize its own benefit/payoff.” [Durfee & Rosenschein 1994]

Erwünschte übergeordnete gemeinsame Ziele des Systems sind für die Agenten nur von sekundärem Interesse, so dass deren Erreichen und die Auflösung von Zielkonflikten zwischen den Agenten durch die Wahl geeigneter *Koordinationsmechanismen* (siehe die Abschnitte 2.5.3 und 3.3.4) ermöglicht bzw. erreicht werden muss, mittels derer die Agenten miteinander interagieren. Im Idealfall werden die Agenten durch einen Koordinationsmechanismus dazu gebracht, auf ein gemeinsames übergeordnetes bzw. globales Ziel hinzuarbeiten.

“Multiagent Systems are concerned with coordinating intelligent behaviour among a collection of autonomous intelligent agents, how they coordinate their knowledge, goals, skills, and plans jointly to take action or solve problems.” [Bond & Gasser 1998, S. 3-35]

Die *Koordination* und *Kooperation* zwischen den Agenten spielt daher in der MAS-Forschung eine zentrale Rolle (s. dazu Abschnitt 2.5). Insbesondere die Auswahl der Koordinationsmechanismen ist für die Emergenz eines wünschenswerten Systemverhaltens entscheidend. Diese müssen für die Agenten einen Anreiz erzeugen oder sie im Extremfall dazu

³ In der Biologie finden sich auch Beispiele für natürliche Multiagenten-Systeme (z. B. Insektenstaaten)

⁴ Die englischsprachige Literatur verwendet hierfür den Begriff „*self-interested*“.

zwingen, sich an der Problemlösung des Gesamtsystems auf eine kooperative Art und Weise zu beteiligen.

Durch die auf diese Weise geförderte Kooperation der Agenten wird dann ein globales Problem gelöst oder ein wünschenswerter Systemzustand durch eine Annäherung an eine global optimale Lösung erreicht.

Während dies im Vergleich zum Verteilten Problemlösen zu einem erhöhten Kommunikations- und Koordinationsaufwand führt, ergibt sich daraus andererseits für MAS auch eine erhöhte Flexibilität beim Problemlösen, z. B. bei der Verteilung oder Umverteilung von Aufgaben.

Denn im Unterschied zum verteilten Problemlösen existiert in MAS keine im Voraus fest vorgegebene Zuordnung von zu lösenden Teilproblemen zu einzelnen Agenten, so dass Teilprobleme sowie durchzuführende Aufgaben (sogenannte *Tasks*) zur Laufzeit zwischen den Agenten ausgetauscht werden können.

Im Unterschied zum klassischen VPL ist es sogar für viele MAS charakteristisch, dass es sich um offene Systeme handelt, deren Agenten-Gesellschaft sich zur Laufzeit ändern kann, da Agenten ein- oder aus ihm austreten dürfen bzw. möglicherweise ausfallen (Ausfallsicherheit, Robustheit). Dies ist beispielsweise in agentenbasierten elektronischen Märkten der Fall, ohne dass dies zu Komplikationen führt.

Die o. g. Eigenschaften ermöglichen es MAS, die zur Verfügung stehenden Ressourcen effizient und effektiv einzusetzen und neuartige Problemlösungswege zu finden, an die ein Systemdesigner a priori evtl. gar nicht gedacht hätte oder auch Aufgaben zu lösen, die zum Zeitpunkt des Systementwurfs noch gar nicht bekannt waren [Stickel et al. 1998].

Daher können MAS auch besser als VPS mit unvollständigen Informationen über den Zustand ihrer Umwelt und dem Status anderer Agenten sowie Störeinflüssen umgehen und besitzen im Gegensatz zu verteilten Problemlöse-Systemen eine wesentlich höhere *Flexibilität* und *Robustheit* (s. dazu auch die Ausführungen zu einer partiell beobachtbaren und dynamischen Umwelt in Abschnitt 2.3.3).

In ihnen entwickelt sich daher häufig ein emergentes Verhalten, das zu einer Leistungsfähigkeit des MAS führt, welches die Summe der individuell möglichen Leistungen seiner Mitglieder deutlich übersteigt. Denn durch die Selbstorganisationsprozesse in einem MAS lassen sich der Einsatz von Spezialfähigkeiten einzelner Agenten sowie sonstiger Ressourcen im Idealfall so steuern, dass damit quantitative Einsparungen, Nutzenzuwächse oder qualitative Verbesserungen einhergehen.

Typische Einsatzszenarien von MAS zeichnen sich u. a. dadurch aus, dass

- es keine zentrale Steuerungs-, Koordinations- oder Kontrollinstanz gibt,
- jeder Agent nur ein partielles und häufig unsicheres Wissen über den Umweltzustand der Anwendungsdomäne und den Zustand der übrigen Agenten besitzt,
- ein einzelner Agent nicht in der Lage ist, die gegebenen Probleme zu lösen, da sein Wissen sowie seine Fähigkeiten und die ihm zur Verfügung stehenden Ressourcen dafür zu eingeschränkt sind,
- die Agenten primär individuelle, zueinander in Konflikt stehende Ziele verfolgen und rational handeln, d. h. sich u. a. nur dann kooperativ verhalten, wenn dies (zumindest mittelfristig) zu ihrem Vorteil ist und
- die Berechnungen und Handlungen der Agenten autonom, parallel und asynchron erfolgen.

Ganz grundlegend lassen sich die Einsatzszenarien von MAS danach differenzieren, ob in ihnen ausschließlich Software-Agenten miteinander interagieren oder wie im Fall von Artificial Societies [Malsch 1998] auch menschliche Akteure Bestandteile des MAS sind, was beispielsweise in Auktionen auf elektronischen Märkten der Fall sein kann, wenn an ihnen sowohl menschliche Bieter als auch Biet-Agenten teilnehmen.

Je nach Aufgabenstellung und Systemdesign sind die Grenzen zwischen VPS und MAS nicht in jedem Fall ganz klar definierbar. Manche Forscher betrachten MAS sogar als ein Teilgebiet oder Spezialfall des Verteilten Problemlösens [Corsten & Gössinger 1997].

Insbesondere aufgrund der Bedeutung, welche die Interaktionsmöglichkeiten für das Verhalten von MAS haben, sind zur Entwicklung und Erforschung von Multiagenten-Systemen auch Erkenntnisse aus vielen anderen Forschungsdisziplinen hilfreich oder erforderlich, welche die Arbeitsteilung und die Interaktionen zwischen Menschen im Beruf und allgemein in der Gesellschaft zum Gegenstand haben.

Wesentliche Schnittmengen und Austausch- bzw. Transferbeziehungen bestehen dabei natürlich mit der KI und VKI, aber beispielsweise auch mit der *Organisationstheorie*, der *Spiel- und Entscheidungstheorie* oder der *Sozionik* (s. Abschnitt 7.2.1), um nur die wichtigsten zu nennen (siehe die folgende Abbildung).

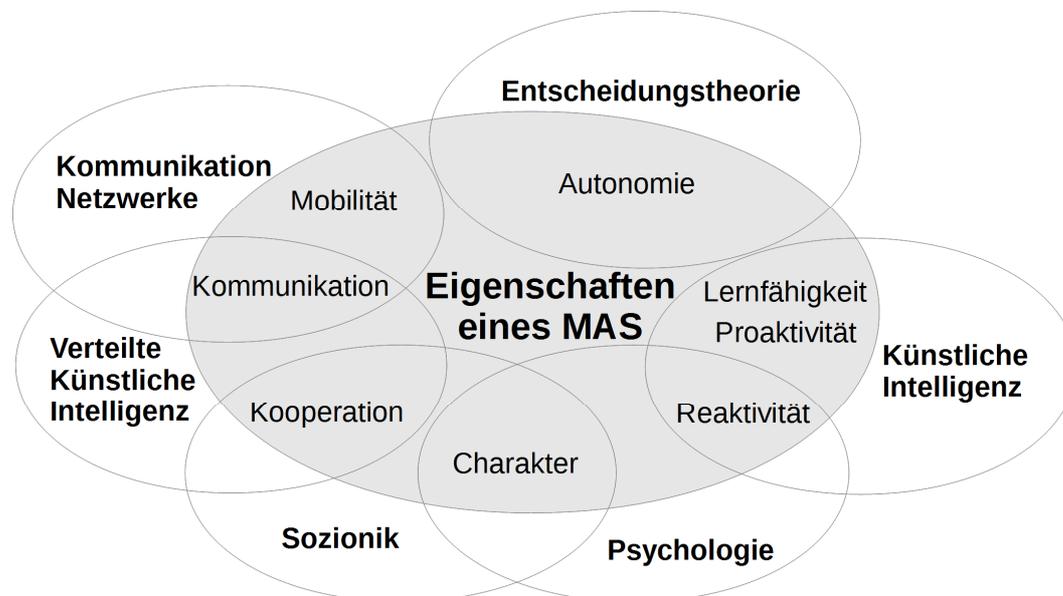


Abbildung 2: Schnittmengen des Forschungsgebiets der Multiagenten-Systeme mit anderen Forschungsdisziplinen⁵

Hierdurch weist das Forschungsgebiet der Multiagenten-Systeme einen besonders interdisziplinären Charakter auf, was die Realisierung von MAS zu einer komplexen Aufgabenstellung macht. Da MAS aus Software-Agenten aufgebaut sind, wird im Folgenden ausführlich beschrieben, was unter diesem Begriff genau zu verstehen ist.

⁵ Quelle: [Gerber 2005, S. 7]

2.3 Software-Agenten

2.3.1 Grundlegende Definitionen

Seinen Ursprung findet der Begriff des *Agenten* im Begriff "agens" (im Lateinischen „das Handelnde“). Laut dem Oxford Dictionary⁶ und der Merriam Webster Enzyklopädie⁷ versteht man unter einem Agenten ganz allgemein „... *a person who acts on behalf of another*“. „Handeln mit dem Ziel, einen Auftrag zu erfüllen“ ist also die wesentliche Aufgabe eines Agenten. Im Kontext dieser Arbeit geht es nicht um natürliche Personen, die eine Agentenfunktion ausüben, sondern um Software-Programme, die dies tun bzw. als Bestandteil eines verteilten Systems selbständig handeln und mit anderen Agenten des Systems kommunizieren und daher *Software-Agenten*⁸ genannt werden. Ebenso wie für einen menschlichen Agenten stellt sich auch bei der Betrachtung eines Software-Agenten die Frage, welche Art von Handlungen der Agent ausführt und wie er die Entscheidungen darüber, welche Handlungen er ausführt, trifft.

Gemäß Michael Wooldridge [Wooldridge 2002], aber auch gemäß [Franklin & Graesser 1996; Priestersbach 1999, S. 23] existiert für den Begriff des *Agenten* bzw. *Software-Agenten* keine allgemein anerkannte Definition. Gemeinsam mit seinem Kollegen Nicholas Jennings definiert er einen Agenten als ein in seiner Umgebung zielorientiert und v. a. selbständig, d. h. autonom agierendes bzw. handelndes Computersystem:

„An agent is a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives.“
[Jennings & Wooldridge 1998, S. 115-152]

Die am Anfang des Kapitels genannte Definition von [Russell & Norvig 1995] war eine sehr umfassende, da sie unter einem *Agenten* alles versteht, was seine Umgebung durch *Sensoren* wahrnehmen und durch sogenannte *Effektoren* verändern kann (s. a. [Russell & Norvig 2003]). Dies beinhaltet auch biologische Agenten wie z. B. Menschen, Tiere und Pflanzen. Um die Agenten-Definition auf die für das Forschungsgebiet der Multiagenten-Systeme wesentlichen Entitäten einzugrenzen, haben [Fischer et al. 1996a] die folgende Definition vorgeschlagen:

„Agenten sind autonome oder teilautonome Hardware- oder Softwaresysteme, die Aufgaben in einer komplexen, sich dynamisch verändernden Umwelt, durchführen.“
[Fischer et al. 1996a]

Nach Stuart Russell und Peter Norvig nimmt ein Agent seine Umwelt durch Sensoren wahr und reagiert durch seine Handlungen auf diese Umwelt. Die grundlegende Funktionalität eines Agenten besteht also in der Fähigkeit, mit seiner Umwelt interagieren zu können.

Um dies tun zu können, muss ein Agent auch über eine gewisse Logik verfügen, die darüber entscheidet, welche Handlung er als nächstes ausführt. [Schwaiger 2006] unterscheidet daher drei wesentliche Architekturkomponenten, die ein Software-Agent besitzen muss, nämlich eine *Sensorkomponente*, eine *Entscheidungskomponente* (evtl. inklusive einer Wissensbasis) sowie eine *Aktionskomponente*.

Die *Sensorkomponente* besteht aus einer Menge von Sensoren, mit denen ein Agent Ausschnitte seiner Umwelt wahrnimmt und, sofern er über eine Art Gedächtnis verfügt, Wissen über diese Umwelt sammeln kann.

⁶ <http://www.oxforddictionaries.com/definition/english/agent>

⁷ <http://www.merriam-webster.com/dictionary/agent>

⁸ Vgl. <http://wirtschaftslexikon.gabler.de/Definition/agent.html>

Die *Entscheidungskomponente* kann je nach der Agenten-Klasse, welcher der Agent angehört (s. Abschnitt 2.4) auf unterschiedlich hohen Komplexitätsstufen realisiert sein. Während sie im einfachsten Fall, dem sogenannten *einfachen Reflex-Agenten* (vgl. Abschnitt 2.4.1) lediglich auf der Basis simpler Wenn-Dann-Regeln Entscheidungen trifft, was in relativ einfachen Umgebungen ausreichend sein kann (s. u.), kann sie auch komplexe Modelle zur Repräsentation des Umweltzustands, des internen Zustands sowie der Ziele eines Agenten umfassen und darüber hinaus über ausgefeilte Schlussfolgerungs-, Planungs-, Entscheidungs- und Lern-Funktionalitäten verfügen.

Die von seiner *Entscheidungskomponente* ausgewählten Aktionen setzt der Agent dann mit Hilfe seiner *Aktionskomponente* in seiner Umgebung bzw. Umwelt um. Diese Komponente hat Zugriff auf *Hardware-* und/oder *Software-Effektoren*. Im Falle eines physischen Agenten wäre beispielsweise ein Roboter-Arm ein Hardware-Effektor. Ein Software-Effektor wäre z. B. ein Funktionsaufruf, der in einem E-Commerce-Szenario ein Kaufangebot macht oder eine Finanztransaktion tätigt.

2.3.2 Agenten-Eigenschaften

Nach den vorgenannten, grundlegenden, aber auch recht weit gefassten Definitionen des Begriffs *Agent* könnten auch Systeme, die gemeinhin nicht unbedingt als besonders intelligent gelten, wie z. B. Anlagen zur Heizungssteuerung, als Agenten bezeichnet werden.

Daher macht eine darüber hinaus gehende Differenzierung von Agenten anhand von eher allgemeinen (und damit schwächeren) und spezifischeren (stärkeren, teilweise KI-spezifischen) Charakteristika Sinn (für Details bzgl. Agenten-Klassen und -Typen s. Kapitel 2.4).

[Wooldridge & Jennings 1995] unterscheiden anhand der Eigenschaften, die ein Agent besitzt, zwischen einer schwachen und einer starken Definition des Agentenbegriffs. Gemäß der schwachen Definition ist eine Hard- oder Software-Entität ein Agent, wenn sie die folgenden vier Eigenschaften bzw. Fähigkeiten aufweist:

- *Autonomie*: Der Agent führt seine Berechnungen und Aktivitäten sowie seine Entscheidungsfindung selbständig und unabhängig aus.
- *Soziale Fähigkeiten* bzw. *Interaktionsfähigkeit*: Agenten sind zu *Interaktionen* und zur *Kommunikation* mit anderen Entitäten wie anderen Agenten in ihrer Umwelt (oder Menschen) fähig, um ihre Ziele zu erreichen.
- *Reaktivität*: Agenten perzipieren ihre Umwelt und reagieren auf Veränderungen ihrer sich (ggf. dynamisch) ändernden Umwelt, indem sie agieren (für Details bzgl. der Umwelt-Charakteristika von Agenten s. Abschnitt 2.3.3).
- *Proaktivität*: Agenten sind nicht nur reaktiv, sondern können auch spontan von selbst aktiv werden und die Initiative ergreifen, um ihre Ziele zu verfolgen.

[Wooldridge 2002] erachtet das Vorhandensein der genannten vier Eigenschaften gar als notwendige Anforderungen, um eine Entität als Agenten bezeichnen zu können.

Wie es die in den vorhergehenden Abschnitten vorgestellten Definitionen deutlich gemacht haben, stellt die Autonomie-Eigenschaft eines Agenten das ihn von anderen Entitäten (wie Computersystemen und Software-Objekten) am wesentlichsten differenzierende Charakteristikum dar.

Die Fähigkeit der *Autonomie* (aus dem Altgriechischen (αὐτός) „autos“, „selbst“, und (νόμος) „nomos“, „Gesetz“) bedeutet zunächst, dass Agenten eine *Operationsautonomie* aufweisen, d.h. (zumindest über einen längeren Zeitraum) unabhängig von der Kontrolle oder den Eingriffen eines Benutzers oder anderer Agenten bestimmte Aufgaben selbständig

durchführen können. Synonyme für Operationsautonomie sind *Verhaltens-* und *Handlungsautonomie*.

Hierfür benötigen die Agenten den Zugriff auf ihren internen Zustand und die Fähigkeit zur Steuerung ihrer Aktionen über Effektoren. Wenn Agenten darüber hinaus auch auf Basis ihrer eigenen internen Zustände, Ziele und Erfahrungen die Entscheidung darüber treffen können, welche Aktionen sie als nächstes ausführen, d. h. zwischen Verhaltensalternativen auswählen können, so verfügen sie zusätzlich auch über eine sogenannte *Entscheidungsautonomie* [Steels 1994; Steels & Brooks 1995]. Jedoch wird diese Entscheidungsautonomie häufig durch die menschlichen Nutzer oder Designer der Agenten auf ein bestimmtes gewünschtes Maß begrenzt. Je stärker ein Agent dazu fähig ist, selbständig zu lernen und neue Ziele und Entscheidungsregeln zu generieren, die nur noch indirekt einer äußeren Kontrolle unterliegen, umso stärker ist sein Grad an Entscheidungsautonomie. Das Vorliegen einer Operationsautonomie ist die Vorbedingung für eine Entscheidungsautonomie; umgekehrt gilt dieses jedoch nicht.

Reaktivität bedeutet, dass ein Agent auf der Basis von Verhaltensregeln bzw. -mustern definierte Aktionen ausführt, sobald bestimmte Sensorinformationen vorliegen. Im einfachsten Fall handelt es sich dabei um einfache, unveränderliche Wenn-Dann-Regeln, auf deren Basis ein Agent bestimmte Aktionen reflexartig ausführt, ohne dass weitere Prozesse des Schlussfolgerns oder Planens stattfinden. Solche Agenten nennt man *einfache Reflex-Agenten* (s. Abschnitt 2.4.1).

Die Fähigkeit zum *sozialen Verhalten* ist wesentlich, um die Zusammenarbeit mehrerer Agenten und damit auch die gemeinsame Lösung von Problemstellungen zu ermöglichen. Durch die Kooperation mehrerer Agenten können bestimmte Problemlösungen überhaupt erst umgesetzt und/oder bzgl. ihrer Ressourceneffizienz und ihres qualitativen Zielerreichungsgrades optimiert werden.

Hierbei sind natürlich die Möglichkeiten zur *Kommunikation* (zwischen den Agenten und/oder zwischen Agenten und ihren menschlichen Nutzern) von wesentlicher Bedeutung, wobei in der Regel bestimmte standardisierte Kommunikationssprachen und -protokolle (z.B. FIPA, KQML (vergleiche [Finin et al. 1994; Finin et al. 1995])) zum Einsatz kommen.

Die starke Definition des Agenten-Begriffs erfordert nach [Wooldridge & Jennings 1995] das Vorhandensein weiterer mentaler Eigenschaften wie

- *Wissen*,
- *Glauben* und
- *Intention*

Agenten, die entsprechende Eigenschaften aufweisen besitzen die Eigenschaft der

- *Deliberativität* (aus dem Lateinischen „*deliberare*“, „erwägen, überlegen; sich entscheiden, beschließen“)

Im Unterschied zu rein reaktiven Agenten besitzen *deliberative Agenten* eine Repräsentation ihres internen Zustands sowie des Zustands ihrer Umwelt und können auf dieser Grundlage symbolische Schlussfolgerungs- und Planungsprozesse durchführen [Newell & Simon 1976].

[Weiß und Jakob 2005] betrachten in diesem Zusammenhang mentale Zustände und gliedern diese in drei Kategorien:

- Informationsbezogene Zustände (Wissen, Vermutungen)

- Konative Zustände (Intentionen, Pläne, Verpflichtungen)
- Affektive Zustände (Ziele, Präferenzen, Wünsche)

Zu der Klasse *deliberativer Agenten* gehören beispielsweise *BDI-Agenten*, auf die in Abschnitt 2.4.6 kurz eingegangen wird. Neben dieser Klasse werden auch reaktive und hybride Agenten unterschieden, auf die ebenfalls in Abschnitt 2.4.6 eingegangen wird.

Eine noch KI-spezifischere Auffassung des Agentenbegriffs fordert zusätzliche mentale Eigenschaften bzw. Charakteristika wie *Intelligenz* sowie *Lern- und Adaptionsfähigkeit*, damit sich ein Agent auf der Basis gemachter Erfahrungen an neue Situationen anpassen kann.

Zudem werden von Agenten häufig die folgenden Charaktereigenschaften gefordert:

- *Kooperationsbereitschaft (Gutwilligkeit)*
- *Nutzenorientiertheit*
- *Rationalität*: Damit ist Fähigkeit gemeint, sich so zu verhalten, dass eigene Ziele erreicht werden und dabei der eigene Nutzen maximiert wird.

Die Eigenschaft der *Rationalität* bedeutet, dass ein Agent sinnvolle Entscheidungen trifft, die der Erreichung seiner Ziele und damit der Optimierung seines eigenen Nutzens dienen. Idealerweise besitzt ein rationaler Agent ein umfangreiches Wissen über seine Umgebung und seine Handlungsmöglichkeiten und ist zudem dazu in der Lage, zu erkennen, welche seiner Ziele aus einem gegebenen Zustand heraus überhaupt erreichbar sind. Voraussetzung hierfür ist, dass der Agent über Entscheidungs- und Planungsfunktionalitäten verfügt sowie über ein Performanz-Maß, mit dem er (beispielsweise durch eine Nutzenfunktion) aktuelle Zustände und die zu erwartenden Effekte seiner Aktionen bewerten kann. Rationalität bedeutet dabei nicht, dass ein Agent allwissend ist, da sowohl die Effekte seiner Handlungen als auch seine Umwelt oftmals *nicht-deterministisch* (vgl. Abschnitt 2.3.3) sind.

Weitere Eigenschaften, die oft genannt werden, sind:

- *Effizienz*: Agenten versuchen, ihre Aufgaben im Rahmen der ihnen nur beschränkt zur Verfügung stehenden Zeit und sonstiger Ressourcen (Speicher, Energie, monetäre Ressourcen) möglichst effizient zu erreichen.
- *Mobilität*: Mobile Agenten besitzen die Fähigkeit, ihre physikalische Position in der realen Welt zu ändern. So können sie als Roboter ihren Standort verändern oder sich als Software-Agenten in einem Rechnernetz zwischen verschiedenen Servern bewegen, z. B. weil ein leistungsfähigerer Server ein performanteres Arbeiten ermöglicht oder weil bestimmte Arbeiten (beispielsweise aufgrund der zeitweisen Kappung der Außenverbindung zu einem lokalen Rechnernetz, in dem eine Informationssuche [Mattern 1998] oder eine Systemdiagnose vorgenommen werden soll) ansonsten nicht möglich sind.

Wünschenswert ist darüber hinaus, dass ein Agent über eine gewisse *Lernfähigkeit* verfügt, d. h. sein Verhalten adaptieren und verbessern kann, indem er auf der Grundlage seiner gemachten Erfahrungen die Inhalte seiner Entscheidungskomponente bzw. seine Entscheidungslogik adaptiert (siehe auch Abschnitt 2.4.5).

Die oben genannten Eigenschaften eines Agenten können je nach Klasse oder Typ des Agenten (siehe hierzu Kapitel 2.4) unterschiedlich stark ausgeprägt sein.

2.3.3 Umwelt-Charakteristika

Die Umwelt eines Agenten kann anhand ihrer Charakteristika folgendermaßen klassifiziert werden:

- *vollständig beobachtbar* oder *partiell beobachtbar*: In der Regel ist die Umwelt eines Agenten nur partiell beobachtbar, da in den seltensten Fällen vollständige, akkurate und aktuelle Informationen über den Umweltzustand vorliegen. In den Veröffentlichungen von Wooldridge [Wooldridge 2002; Wooldridge 2009] und Weiss [Weiss 2000] werden diese Eigenschaften auch „*accessible*“ und „*inaccessible*“ genannt.
- *deterministisch* oder *stochastisch* bzw. *nicht-deterministisch*: Eine Umwelt ist deterministisch, wenn jede Aktion eines Agenten garantiert immer zu der gleichen Änderung des Umweltzustands führt. In einer stochastischen Umwelt ist hingegen der aus einer Aktion resultierende Umweltzustand nicht garantiert prognostizierbar, so dass stets eine Unsicherheit bezüglich der aus einer Aktion resultierenden Effekte besteht.
- *statisch* oder *dynamisch*: In einer statischen Umwelt erfolgen Änderungen des Umweltzustands exklusiv durch die Aktionen des Agenten. In einer dynamischen Umwelt erfolgen hingegen Zustandsänderungen auch unabhängig vom Agenten durch weitere Agenten sowie sonstige Einflüsse.

In Abhängigkeit von den Eigenschaften und den damit einhergehenden Herausforderungen seiner Umwelt steigen auch die Anforderungen an die Fähigkeiten eines Agenten, über die er verfügen muss, um in und mit seiner Umwelt erfolgreich zu (inter)agieren. Je nach Ausprägung ihrer Fähigkeiten bzw. nach Grad der Komplexität ihrer Entscheidungskomponente lassen sich Agenten in unterschiedliche Klassen einordnen. Im folgenden Abschnitt werden daher einige wesentliche Agenten-Klassen und -Typen vorgestellt.

2.4 Agenten-Klassen und -Typen

Während Corsten und Gössinger *primitive, technische, kognitive* und *soziale* Agenten [Corsten & Gössinger 1997] unterscheiden, teilen Stuart Russell und Peter Norvig Agenten nach der Qualität ihrer Fähigkeiten zur Auswahl ihrer Handlungen in vier Klassen ein [Russell & Norvig 2003, S. 46 ff.]:

1. Einfache Reflex-Agenten (*simple reflex agents*),
2. Modellbasierte Reflex-Agenten (*model-based reflex agents*),
3. Zielbasierte Agenten (*goal-based agents*) und
4. Nutzenbasierte Agenten (*utility-based agents*).

Im Anschluss daran definieren [Russell & Norvig 2003] zusätzlich als fünfte Agenten-Klasse die Klasse der lernenden Agenten. Die Eigenschaften und Unterschiede dieser fünf Agenten-Klassen werden im Folgenden kurz beschrieben.

2.4.1 Einfache Reflex-Agenten

Einfache Reflex-Agenten sind die am einfachsten agierenden Agenten. Ein einfacher Reflex-Agent besitzt keine Perzeptionshistorie bzw. Gedächtnis, sondern agiert ausschließlich auf der Basis seiner aktuellen Perzeption der Umwelt und einer Menge von exakt definierten *Verhaltensmustern* bzw. *-regeln*.

Er transformiert seine Sensorinformationen in Handlungen, indem er seine sensorischen Wahrnehmungen fortwährend mit den in seinen Verhaltensregeln beschriebenen, zustandsbezogenen Bedingungen vergleicht. Sobald der Bedingungsteil einer seiner Regeln erfüllt ist bzw. ein bestimmter Zustand eintritt, führt er die im Aktionsteil der Regel festgelegten Aktionen reflexartig aus, ohne weitere Prozesse zwischenschalten (z. B.: „Wenn die Ampel rot ist, dann halte das Auto an.“).

Insbesondere besitzt diese Agenten-Klasse keine Vorstellung davon, wie sich seine Umwelt durch eine von ihm ausgeführte Aktion verändert und er hat auch keine Kenntnis über Aspekte des Umweltzustands, die er beispielsweise aufgrund der Reichweite oder Lage seiner Sensoren nicht unmittelbar wahrnehmen kann.

Agenten dieser Klasse können daher in einer nur partiell beobachtbaren oder gar dynamischen Umwelt lediglich in überaus eingeschränktem Maße erfolgreich sein. Da sie weder über ein Gedächtnis verfügen, noch mehrere Schritte im Voraus planen können, laufen sie Gefahr, erfolglose Handlungen immer wieder zu wiederholen und so in eine Endlosschleife zu geraten (wie beispielsweise eine Stubenfliege, die unablässig versucht, durch eine Fensterscheibe zu fliegen).

Ihre einzige Möglichkeit, diese Gefahr zumindest abzumildern, besteht darin, die Ausführung ihrer Regeln beziehungsweise die Auswahl der durchzuführenden Aktionen geeignet zu randomisieren. Aber auch in diesem Fall bleiben ihre Fähigkeiten und Einsatzmöglichkeiten im Vergleich zu den übrigen Agenten-Klassen sehr eingeschränkt.

2.4.2 Modellbasierte Reflex-Agenten

Modellbasierte Reflex-Agenten verfügen darüber hinaus über Wissen über ihre Umwelt, welches in einem sogenannten *internen Modell* repräsentiert wird. Das Modell speichert den aktuellen Zustand des Agenten sowie seiner Umwelt und beinhaltet auch eine Perzeptionshistorie mit dem vom Agenten gesammeltem Wissen über seine Umwelt. In der Regel beinhaltet es auch Wissen über einige Teile davon, die er aktuell nicht wahrnehmen kann. Darüber hinaus beschreibt es, wie sich die Aktionen des modellbasierten Agenten auf seine Umwelt auswirken und diese verändern – der Agent besitzt also eine gewisse Kenntnis darüber, „wie die Welt funktioniert“.

Daher ist ein modellbasierter Reflex-Agent viel besser dazu geeignet, in einer partiell beobachtbaren Umwelt erfolgreich zu agieren. Die Auswahl seiner Aktionen erfolgt zwar grundsätzlich auf die gleiche Art wie beim einfachen Reflex-Agenten, jedoch kann auf Basis des internen Modells die situationsbezogene Auswahl geeigneter Aktionen deutlich differenzierter erfolgen.

Denn er hat durch sein internes Modell einerseits eine Kenntnis über den aktuellen Umweltzustand, der über die aktuellen Sensorinformationen hinausgeht und ihn in die Lage versetzt, zwei unterschiedliche Umweltzustände auch dann voneinander abzugrenzen, wenn diese die gleichen sensorischen Reize liefern. Und andererseits gibt ihm sein Modell auch die Fähigkeit, die Auswirkungen seiner Handlungen genauer abzuschätzen.

2.4.3 Zielbasierte Agenten

Zielbasierte Agenten erweitern modellbasierte Reflex-Agenten um die Beschreibung konkreter, vom Agenten gewünschter Zielzustände, an denen der Agent die Auswahl der durchzuführenden Aktionen ausrichten kann. Dies ermöglicht es dem Agenten, zielgerichtet aus einer Menge von möglichen Handlungen diejenige(n) auszuwählen, die ihn am ehesten oder effizientesten seinen Zielen näher bringt. Hierzu kann er Methoden wie z. B. *Suchverfahren*,

Planungsverfahren und dazugehörigen *Heuristiken* aus den entsprechenden Teilgebieten der KI einsetzen, welche ihm dabei helfen, geeignete Aktionssequenzen zu bestimmen.

Der Einsatz entsprechender Methoden ist zwar mit gewissen Aufwänden verbunden (Berechnungskomplexität und -zeit), jedoch erhöht der Einsatz solcher Methoden die Flexibilität, Effektivität und Effizienz entsprechender Agenten außerordentlich, da sie auf dieser Basis nicht nur Wissen über wünschenswerte Zielzustände besitzen, sondern mit Hilfe der Methoden auch mehrere zielerfüllende Aktionssequenzen berechnen und zwischen ihnen differenzieren und auswählen können. Jedoch können sie nur zwischen Zielzuständen und Nicht-Zielzuständen unterscheiden.

2.4.4 Nutzenbasierte Agenten

Nutzenbasierte Agenten (siehe Abbildung 3) sind aufbauend darauf und darüber hinaus in der Lage, verschiedene Zielsituationen anhand einer *Nutzenfunktion* (*utility function*) zu bewerten. Ihre Nutzenfunktion bildet verschiedene interne sowie externe Zustände (Umweltzustände) auf eine Skala ab (in der Regel auf reelle Zahlen). Hierdurch beschreibt sie die Präferenzen der Agenten bzgl. verschiedener Zustände und gibt ein Maß dafür an, wie wünschenswert ein bestimmter Zustand – im Vergleich mit anderen – ist. Auf diese Weise kann ein nutzenbasierter Agent mit Hilfe seiner Nutzenfunktion verschiedene Zustände, Pläne und Ziele evaluieren und zwischen diesen differenzieren.

Ein nutzenorientierter Agent kann insbesondere versuchen, seine Nutzenfunktion über seinen Zielen zu maximieren und so zwischen mehreren wünschenswerten Zielen auftretende Zielkonflikte lösen. Ein *rational handelnder*, nutzenbasierter Agent wird immer diejenige Aktion oder Aktionssequenz auswählen, die zu Zuständen führt, die seinen erwarteten Nutzen maximieren.

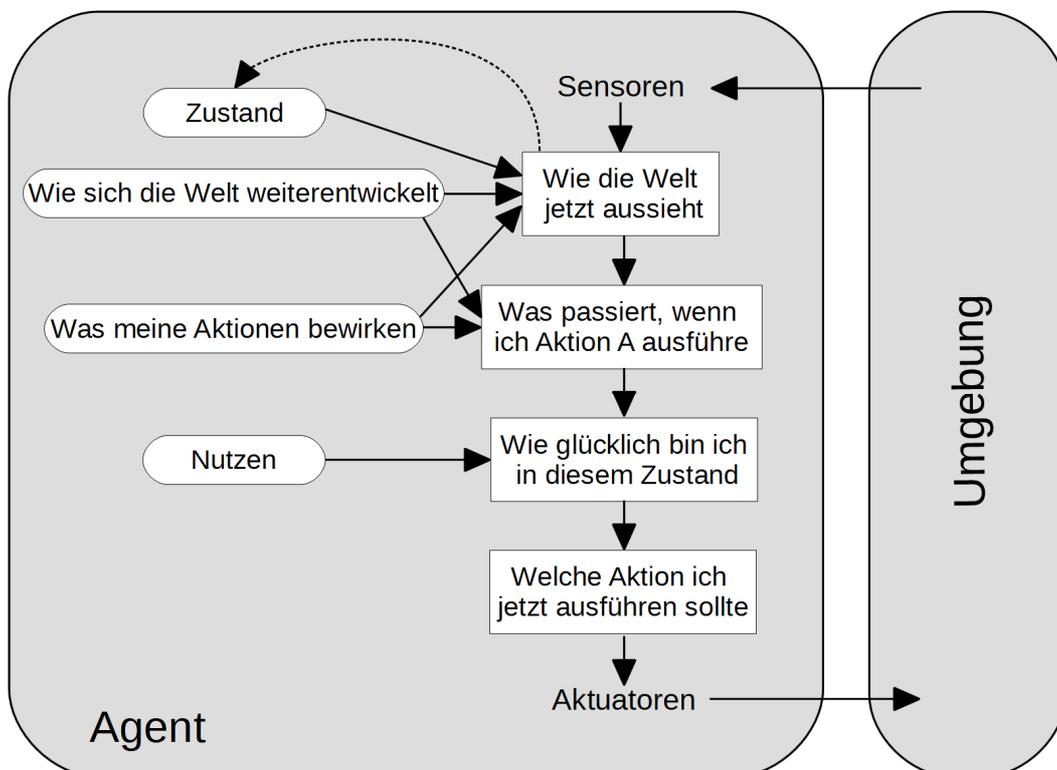


Abbildung 3: Allgemeines Modell eines modell- und nutzenbasierten Agenten⁹

⁹ Quellen: [Russell & Norvig 2004, S. 79]

2.4.5 Lernende Agenten

Alan Turing schlug bereits 1950 vor, intelligente Maschinen nicht von Hand zu programmieren, sondern lernende Maschinen zu bauen und diesen dann etwas beizubringen [Turing 1950, S. 433-460].

Die Lernfähigkeit hat für einen Agenten den Vorteil, dass sie ihm erlaubt, auch in einer anfangs für ihn unbekanntem Umwelt zu operieren und in dieser mit der Zeit kompetenter und geschickter zu werden, als ihm dies auf der Grundlage seines Anfangswissens alleine möglich gewesen wäre.

Lernende Agenten bestehen nach [Russell & Norvig 2003] neben ihren Sensoren und Effektoren aus vier Hauptkomponenten (s. a. die folgenden Abbildung), nämlich einer

- *ausführenden Komponente (Leistungselement)*, einer
- *Lern-Komponente (Lernelement)*, einer
- *Kritik-Komponente* sowie einer
- *Problemgenerator-Komponente*.

Bei der ausführenden Komponente handelt es sich im Wesentlichen um die gleiche Logik, die in den vorhergehenden Abschnitten den gesamten Agenten ausgemacht hat, d. h. Perzeptionen verarbeitet und Entscheidungen bzgl. auszuführender Aktionen trifft.

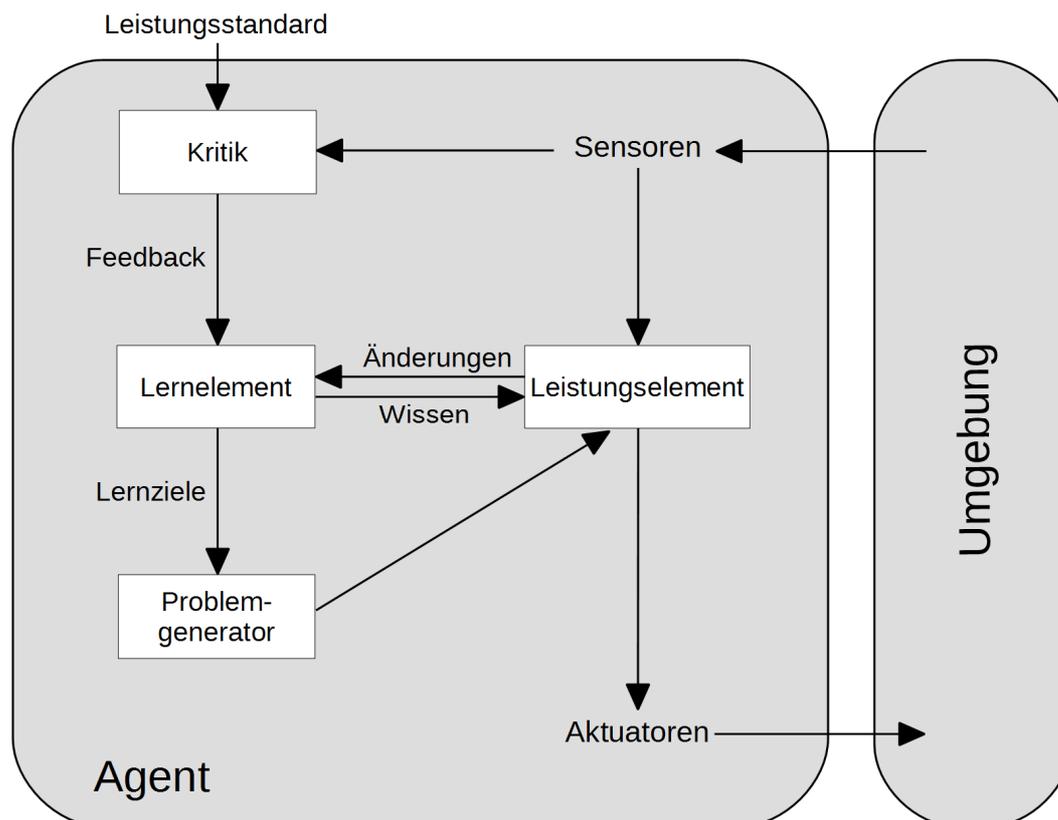


Abbildung 4: Allgemeines Modell eines lernenden Agenten¹⁰

Das zweite Kernstück eines lernenden Agenten ist seine Lern-Komponente selbst, die dafür verantwortlich ist, dass der Agent seine Fähigkeiten mit zunehmender Zeit verbessert bzw. sich besser an die Umwelt adaptiert. Hierzu nutzt die lernende Komponente Feedback über

¹⁰ Quellen: [Russell & Norvig 2004, S. 80]

die Performanz des Agenten, welches sie von der Kritik-Komponente erhält und ihr dabei hilft, die Konsequenzen ausgeführter Aktionen zu bewerten.

Auf der Grundlage dessen bestimmt die lernende Komponente, wie die ausführende Komponente, zum Beispiel was die Verhaltensregeln mit ihren Bedingungen und auszuführenden Aktionen angeht, zu erweitern und/oder zu modifizieren ist, damit sich die Leistungsfähigkeit des Agenten verbessert.

Die letzte Komponente eines lernenden Agenten stellt die Problemgenerator-Komponente dar, die als eine Art Impulsgeber die Durchführung von Handlungen vorschlägt, die den Erfahrungsschatz eines Agenten geeignet erweitern.

Das Zusammenspiel dieser Komponenten ist in Abbildung 4 veranschaulicht, die [Russell & Norvig 2004, S. 80] entnommen ist.

2.4.6 Weitere Agenten-Klassen und -Typen

Neben den oben beschriebenen und allgemein anerkannten grundlegenden Agenten-Klassen lassen sich in der Literatur weitere Klassifizierungen und Typisierungen finden.

So wird beispielsweise auch zwischen *reaktiven*, *deliberativen* und *hybriden Agenten* unterschieden [Müller 1996a; Müller 1996b; Brenner et al. 1998].

Reaktive Agenten besitzen analog zu den einfachen Reflex-Agenten weder ein internes symbolisches Modell ihrer Umwelt noch die Fähigkeit, komplexe Schlussfolgerungsprozesse durchzuführen. Ebenso können sie keine Pläne erstellen, sondern führen in Abhängigkeit von ihren Perzeptionen Aktionen aus einer Menge von vordefinierten Verhaltensmustern aus, mit denen sie ausgestattet sind.

Deliberative (überlegende) Agenten verfügen hingegen wie modellbasierte Agenten über ein internes symbolisches Modell ihrer Umwelt und die Fähigkeit, logische Schlüsse zu ziehen und erwartbare, durch Umwelteinflüsse oder andere Agenten induzierte, Zustandsänderungen ihrer Umgebung in ihre Entscheidungen einzubeziehen. Durch letztere stellen sie konzeptionell eine Erweiterung des oben beschriebenen modellbasierten einfachen Reflex-Agenten dar.

Hybride Agenten enthalten sowohl reaktive als auch deliberative Anteile bzw. Komponenten. Mit hybriden Architekturen wie INTERRAP [Müller & Pischel 1993a; Müller & Pischel 1993b; Müller 1996] werden die Vorteile beider Ansätze miteinander verschmolzen.

[Müller 1996] ergänzt diese Komponenten noch um Ziele und Pläne. Eine dynamische Modifikation und Erweiterung der Fähigkeiten der beschriebenen Agenten war jedoch im Gegensatz zu den oben beschriebenen lernenden Agenten nicht möglich. Da sich die von ihm beschriebenen Agenten in einer dynamischen Umwelt bewegt haben, waren sie eher auf eine schnelle Reaktionsfähigkeit als auf die Erstellung optimaler Pläne ausgelegt.

Bei den *BDI-Agenten* [Rao & Georgeff 1991; Rao & Georgeff 1995] handelt es sich um eine besondere Form deliberativer Agenten, deren interne Zustände sich aus technischen Äquivalenten von *Beliefs* (*Wissen über ihre Umwelt*), *Desires* (*erstrebenswerte Zustände* bzw. *Ziele*) und *Intentions* (*aktuell verfolgte Absichten*) zusammensetzen (s. a. [Bratman 1987; Bratman et al. 1987]).

Die *Beliefs* eines BDI-Agenten bestehen aus Wissen und Annahmen über den aktuellen Umweltzustand, den internen Zustand des Agenten sowie Hintergrundwissen, das der Agent für seine Schlussfolgerungsprozesse benötigt (beispielsweise bezüglich der Auswirkungen der von ihm durchführbaren Aktionen). Die Speicherung dieses Weltwissens kann in einer beliebigen Wissensrepräsentationsstruktur erfolgen.

Seine *Desires* beschreiben die vom Agenten präferierten Umweltzustände bzw. eine Menge optionaler Ziele in einer Datenstruktur, wobei verschiedene seiner optionalen Ziele auch zueinander in Konflikt stehen oder aktuell unerfüllbar sein können. Aus dieser Datenstruktur wählt der Agent dann ein – z. B. aufgrund seiner aktuellen Ressourcen-Planung erfüllbares – konkretes Ziel aus, welches er in der Folge zu erreichen versucht.

Um ein ausgewähltes Ziel zu erreichen, wählt der BDI-Agent aus einer Plan-Datenbank einen hierarchisch strukturierten Plan aus, der in der Regel aus Teilzielen und diesen zugeordneten, ebenfalls hierarchisch aufgebauten Teilplänen besteht und dazu geeignet ist, dieses Ziel zu erreichen. Die Menge der aktuell verfolgten Pläne werden als *Intentions* bezeichnet. Neben den eigentlichen Aktionsplänen verwaltet ein BDI-Agent auch deren aktuellen Bearbeitungszustand.

[Brenner et al. 1998] sehen eine weitere wichtige Unterscheidung in *stationären* und *mobilen Agenten*. Die Mobilität beeinflusst nicht nur die interne Architektur eines Agenten sondern auch die des gesamten Agentensystems. Bei mobilen Agentensystemen sind die wesentlichen Aspekte die Migrationsfähigkeit der Agenten sowie die Agenten- und Sicherheitsschichten auf den Servern. Im Gegensatz dazu sind stationäre Agenten nicht in der Lage, sich auf einen anderen Rechner zu begeben und dadurch Netzwerkkommunikation einzusparen. Dafür entfallen die Implementierungseinschränkungen und die bei mobilen Agenten gegebenen Sicherheitsprobleme. Für Transaktionen in E-Commerce-Szenarien, mit denen sich diese Arbeit beschäftigt, sind daher stationäre Agenten vorzuziehen.

Darüber hinaus können Agenten auch aufgrund ihres Zwecks bzw. ihrer Tätigkeit typisiert werden. So suchen *Informationsagenten* selbstständig nach vorgegebenen Informationen (z. B. Internet, Web-Services oder Datenbanken) und tauschen sich mit anderen Informationsagenten aus. *Assistenz-Agenten* versuchen, Anwender bei ihrer Arbeit zu unterstützen, indem sie selbstständig erkennen, was diese planen und daraufhin geeignete Vorschläge anbieten oder im Hintergrund entsprechende Vorbereitungen durchführen. Bei Online-Auktionen werden immer häufiger so genannte *Bieter-Agenten* verwendet, die nach vorgegebenen Regeln im Auftrag von menschlichen Benutzern bei Auktionen mitbieten.

Im Wesentlichen kombinieren die in diesem Abschnitt aufgezeigten alternativen Agenten-Klassifizierungen und -Typen die Eigenschaften der oben beschriebenen und in [Russell & Norvig 2003] definierten fünf Agentenklassen in einer jeweils auf ihren Einsatzzweck und ihre Einsatzumgebung abgestimmten Form. Insbesondere die Ausführungen zu deliberativen Agenten und BDI-Agenten ergänzen die vorgestellte Sichtweise um Aspekte, die die Deliberativität der Agenten betreffen.

Bei den im Rahmen dieser Arbeit realisierten Agenten handelt es sich um nutzenbasierte lernende Agenten, die in einer nur partiell beobachtbaren Umgebung agieren und danach streben, Aktionen (Verhandlungsstrategien und -schritte) auszuwählen, die ihren an Umsatz und Gewinn orientierten Nutzen im Durchschnitt maximieren (Details hierzu finden sich insbesondere in Kapitel 5).

2.5 Das Koordinationsproblem in MAS

In den MAS-Anwendungsdomänen treten typischerweise Probleme auf, die nur von mehreren Agenten gemeinsam gelöst werden können. Hieraus entsteht in MAS die Notwendigkeit zur *Kooperation*, d. h. zur freiwilligen *Zusammenarbeit* mit anderen selbständigen Agenten, wie es in arbeitsteilig organisierten Gesellschaften die Regel ist. Gemäß [Bond & Gasser 1988] ist die *Kooperation* zwischen Agenten daher ein wichtiger Bestandteil von Multiagenten-Systemen.

Die Zusammenarbeit der Agenten bedingt, dass die kooperierenden Agenten Aktionssequenzen ausführen müssen, die zur Problemlösung bzw. einer gemeinsamen Problemlösung bzw. Zielerreichung beitragen.

Die entsprechenden Aktionen müssen, zum Beispiel bezüglich ihrer zeitlichen Reihenfolge, aufeinander abgestimmt werden, so dass die zwischen den Aktionen bestehenden Zusammenhänge bzw. Abhängigkeiten Berücksichtigung finden und nicht verletzt werden.

Unter Koordination versteht man Mechanismen und Prozesse, die zur Abstimmung zwischen den Agenten und zur Erzeugung einer Ordnung über ihren Aktionen dienen.

Während in hierarchisch organisierten Strukturen eine Koordination der oftmals abhängigen Akteure häufig durch Anweisungen und ein Regelwerk an Vorschriften erfolgt, bedarf es bei vernetzten, dezentralen Strukturen weiterer Koordinationsmechanismen.

2.5.1 Kooperation

Zum Begriff der Kooperation¹¹ existiert eine Vielzahl an Definitionen, die den Begriff aus Sicht verschiedener Wissenschaftsdisziplinen wie z. B. der Spieltheorie, der Betriebswirtschaftslehre, der Psychologie, der Organisationstheorie usw. zu erklären versuchen.

Laut [Wikipedia 2015] versteht man darunter ganz allgemein „... das zweckgerichtete Zusammenwirken von Handlungen zweier oder mehrerer Lebewesen, Personen oder Systeme, in Arbeitsteilung, um ein gemeinsames Ziel zu erreichen“. Adam Smith [Smith 1776], Begründer der Nationalökonomie, betrachtet Kooperation in Verbindung mit Arbeitsteilung als wichtige Produktivkraft.

In der betriebswirtschaftlichen Forschung werden zumeist zwischenbetriebliche Kooperationen betrachtet, denen [Rupprecht-Däullary 1994] die folgenden Charakteristika zuordnet:

- *"Freiwilligkeit der Kooperationsentscheidung,*
- *weitgehende Unabhängigkeit und Selbständigkeit der Kooperationspartner auch während der Kooperation und*
- *die im Vergleich zum Alleingang bessere Zielerreichung als Zweck bzw. Ziel der Kooperation".*

Diese Charakteristika gelten auch für die Kooperation von autonomen, selbstständig entscheidenden nutzenbasierten und bzgl. ihrer Eigenschaften und Ziele unterschiedlicher Software-Agenten – gerade, wenn sie, wie es diese Arbeit zum Gegenstand hat, in ökonomisch-betriebswirtschaftlichen Anwendungsszenarien beziehungsweise -domänen miteinander interagieren.

Durch die Kooperation der Agenten können die individuellen Fähigkeiten der Agenten und sonstige Ressourcen effizienter eingesetzt und verteilt werden, so dass gemeinsame Ziele effizienter und häufig auch in größerer Zahl erreicht werden können. Kooperation hat für ein MAS daher sowohl quantitative als auch qualitative Vorteile.

2.5.2 Kommunikation

Die Fähigkeit zur Kommunikation ist für einen Agenten ganz grundlegend, da sie seine sozialen Fähigkeiten maßgeblich bestimmt, die nach [Wooldridge 2002] bei jedem Agenten vorhanden sein müssen.

¹¹ Lateinischer Ursprung: cooperatio, Zusammenwirkung)

Da die Kommunikation zwischen Agenten (z. B. bei einem Nachrichtenaustausch) auch einen Teil ihrer Ressourcen verbraucht, ist ein möglicher Kommunikationsnutzen immer auch gegen die entstehenden Kosten abzuwägen.

Die Kommunikation zwischen den Agenten ist auf der Basis eines Kommunikationsmechanismus realisiert. Dieser beschreibt

„... a communication protocol, in effect defining who can communicate with whom and what message types are allowed.“ [Wellman & Wurman 1997, S. 5]

Nach [Ollmert & Schinzer 2001] lassen sich Kommunikationsmechanismen in *nachrichtenorientierte* und *Blackboard-orientierte* Kommunikationsmechanismen differenzieren.

Während im nachrichtenorientierten Fall Nachrichten zwischen Agenten Peer-to-Peer bzw. direkt ausgetauscht werden, erfolgt der Nachrichtenaustausch im Blackboard-orientierten Fall bereits gruppenorientiert. Beim *direkten Nachrichtenaustausch* nutzen die Agenten in der Regel die bekannten Computernetzwerke mit ihren Standards (Netzwerkadressen etc.).

Im klassischen Blackboard-Fall haben alle Agenten eines MAS (oder VPS) einen Lese- und Schreib-Zugriff auf einen gemeinsamen Speicherbereich (das sogenannte *Blackboard*, z. B. realisiert durch eine Datenbank), über das sie – für alle lesbar – Nachrichten bzw. Informationen austauschen können. Es gibt aber auch Realisierungen, in denen Agenten nur Zugriffsrechte auf für sie relevante Teilbereiche des Blackboards erhalten. Auf diese Weise kann eine gruppenorientierte Kommunikation umgesetzt werden.

Der klassische Blackboard-Ansatz wird in Domänen mit heterogenen eigennützigen Agenten (z. B. in der E-Commerce-Domäne oder allgemein in marktlich koordinierten Systemen) kaum verwendet, da er die Gefahr birgt, dass vertrauliche Informationen publik werden, aus denen andere Agenten Vorteile ziehen können.

Neben dem Blackboard-Ansatz lässt sich ein gruppenorientiertes Senden und Empfangen von Nachrichten auch mittels *Broadcasting* und *Multicasting* realisieren. Beim *Broadcasting*-Ansatz schickt ein Agent alle seine Nachrichten direkt an alle Agenten seiner Gruppe, so dass alle Gruppenmitglieder über den gleichen Informationsstand verfügen [Foner 1999].

Um die Interpretierbarkeit der gesendeten Nachrichten zu vereinfachen und den Nachrichtenaustausch und die damit verbundenen Aktivitäten der Agenten zu synchronisieren, werden verschiedenen, für den jeweiligen Anwendungszweck geeignete Kommunikationsprotokolle verwendet. In marktlich koordinierten Systemen kommen häufig das Kontraktnetz-Protokoll (englisch *Contract Net Protocol* bzw. *CNP*) [Smith 1980], Erweiterungen davon [Sandholm & Lesser 1995b; Sandholm 1993] oder auch klassische Auktionsprotokolle und bilaterale Verhandlungsprotokolle (siehe Abschnitt 3.3.4.1 sowie Abschnitt 5.3.1) zum Einsatz. Sie definieren, welche Art von Nachrichten zu welchem Zeitpunkt zwischen welchen Agenten bzw. Agenten-Rollen ausgetauscht werden dürfen. Die genannten marktlichen Kommunikationsprotokolle ermöglichen eine Verteilung von Ressourcen und Aufgaben unter den Agenten eines MAS.

Im Falle des Contract-Net-Protokolls schreibt ein sogenannter Manager-Agent Problemlösungsaufgaben aus, um die eine Gruppe von Bieter-Agenten konkurrieren kann. Es besteht aus den Phasen

- *Recognition Phase*: Der Manager-Agent stellt fest, dass er ein Problem alleine nicht oder nur zu hohen Kosten lösen kann. Daher unterteilt er das Problem in Teilprobleme bzw. Teilaufgaben.
- *Announcement Phase*: Der Manager-Agent spezifiziert jede der Teilaufgaben genau (Aufgabenbeschreibung, zur Lösung zur Verfügung stehende Zeit, Anforderungen

an die Lösungsqualität usw.). Eine spezifizierte Aufgabe bietet er dann einer Gruppe von Bieter-Agenten an, die zur Lösung der Aufgabe prinzipiell geeignet sind.

- *Bidding Phase*: Die Bieter-Agenten entscheiden auf Basis ihrer individuellen Ressourcen sowie Kosten- und Nutzenberechnungen, ob sie um die Lösung der Aufgabe konkurrieren wollen und geben ggf. wie bei einer Auktion Gebote dafür ab.
- *Awarding & Expediting Phase*: Der Manager-Agent vergibt die Aufgabe an den Bieter-Agenten, der das beste Gebot gemacht hat. Der erfolgreiche Bieter integriert die Erfüllung dieser Aufgabe daraufhin in seine individuellen Plan-Strukturen und informiert den Manager-Agenten, wenn er dies erfolgreich getan hat.

Die Kommunikation zwischen Agenten basiert in der Regel auf der Sprechakttheorie [Weiss 1999] und wird durch eine Abfolge von Sprechakten realisiert. Beispielsweise orientiert sich die Agenten-Kommunikationssprache *KQML* (*Knowledge Query Manipulation Language*), die als Protokoll für den Austausch von Wissen entwickelt wurde, an der Sprechakttheorie. *KQML* (vgl. [Finin et al. 1994; Finin et al. 1995]) verwendet 36 verschiedene Nachrichtentypen (wie beispielsweise Anfragen, Zusagen oder Absagen), sogenannte *Performatives*.

Neben einem Performative, die dem Empfänger der Nachricht hilft, diese semantisch einzuordnen, enthält eine *KQML*-Nachricht auch Informationen zum Inhalt (z. B. auf welche Aufgabe sich eine Anfrage bezieht) sowie zum Sender und Empfänger der Nachricht sowie bzgl. der Sprache, in der der Inhalt definiert ist und anhand derer der Empfänger den Inhalt daher auch interpretieren muss.

Zuletzt erfolgt die Angabe eines Wissensgebiets (Ontologie), anhand dessen die Nachricht zu interpretieren ist, da bestimmte Begriffe in unterschiedlichen Wissensgebieten eine andere Bedeutung haben können.

```
(<Performative>
:content <speech act>
:sender <name>
:receiver <name>
:language <text>
:ontology <text>
)
```

Abbildung 5: Struktur einer *KQML*-Nachricht

Kommunikation ist für die Realisierung von Kooperation und Koordination in einem MAS wesentlich. Grundsätzlich kann Koordination zwar auch ohne Kommunikation erfolgen (siehe [Bond & Gasser 1998]), aber je eingeschränkter ein Agent seine Umwelt beobachten kann, umso schwieriger und ineffizienter wird dies. Diese Form von Koordination ist daher nur dann sinnvoll, wenn keine Möglichkeit zu direkter Kommunikation zwischen den Agenten gegeben ist [Nwana et al. 1996].

Gerade in marktlich organisierten MAS ist eine Koordination der Agenten ohne Kommunikation und dazugehörige marktliche Protokolle kaum möglich.

2.5.3 Koordination

Der Begriff *Koordination* bezeichnet die Abstimmung der von mehreren Agenten durchzuführenden Aktionen unter Berücksichtigung der zwischen den Aktionen bestehenden Abhängigkeiten (sogenannte Interdependenzen) sowie räumlicher, zeitlicher, ressourcenbezogener und sonstiger Restriktionen.

In seinen Arbeiten zur Koordinationstheorie [Malone 1987a; Malone 1987b; Malone et al. 1988; Malone & Crowston 1992] hat Malone eine Anzahl von Prinzipien definiert, mit denen *Koordinationsprobleme* beschrieben und gelöst werden können. In [Malone & Crowston 1994] werden Koordinationsprozesse untersucht und als deren wesentliche zu koordinierende Komponenten Akteure beziehungsweise Agenten sowie aus ihnen gebildete Gruppen, deren Ziele, Aktivitäten und Ressourcen identifiziert. Koordination wird von Malone und Crowston als Prozess, der das Management von Abhängigkeiten leistet, verstanden. Abhängigkeiten zwischen Agenten bestehen zum Beispiel, wenn sie die gleichen Ziele verfolgen, so dass sie sich in der Folge möglicherweise behindern. Zwei Aktivitäten sind voneinander abhängig, wenn sie auf die gleiche Ressource zugreifen, welche dann möglicherweise nicht in ausreichender Menge zur Verfügung steht, um beide Aktivitäten auszuführen, oder wenn Aktivitäten nur in einer bestimmten temporalen Abfolge ausgeführt werden können. In der Anwendungsdomäne eines Wertschöpfungsnetzwerks besteht eine solche Abhängigkeit von Aktivitäten beispielsweise darin, dass ein produzierender Agent die Produktion seines Output-Gutes erst beginnen kann, nachdem er die dafür notwendigen Ressourcen beziehungsweise Input-Güter bei einem anderen Agenten eingekauft hat und ihm diese auch geliefert wurden.

Zur Lösung von Koordinationsproblemen werden geeignete *Koordinationsmechanismen* eingesetzt. So können marktliche Koordinationsmechanismen wie beispielsweise Auktionen eine sinnvolle, nutzenmaximierende Zuteilung von Ressourcen auf Wirtschaftssubjekte leisten, wenn die Nachfrage nach ihnen das Angebot übersteigt. [Fleisch 2013] beschreibt das Ziel der Koordinationstheorie folgendermaßen:

„Das Ziel der Koordinationstheorie ist die Identifikation und Beschreibung von allgemeingültigen Koordinationsproblemen und die Identifikation, Beschreibung bzw. Ableitung von Koordinationsmechanismen zur Lösung dieser Probleme.“ [Fleisch 2013]

Im Gegensatz zu VPL-Systemen kann in MAS (siehe Abschnitt 2.2.2) von den interagierenden Agenten nicht erwartet werden, dass diese zentral gestaltet werden bzw. sich kooperativ und benevolent verhalten, um die Systemperformanz zu verbessern.

Stattdessen sind Agenten in MAS in ihren Entscheidungen absolut autonom und es kann angenommen werden, dass diese grundsätzlich egoistisch handeln. Zudem gibt es keine zentrale Steuerungs- oder Kontrollinstanz. Daher erfolgt die Verteilung eines gemeinsam erarbeiteten Gesamtnutzens in MAS auch nicht unbedingt gleichmäßig oder fair.

In multiagent systems, the task of coordination can be quite difficult, for there may be situations (in ... open systems) where there is no possibility for global control, globally consistent knowledge, globally shared goals or global success criteria, or even a global representation of a system.” [Bond & Gasser 1988]

Die Charakteristika von Multiagenten-Systemen bedingen einen systeminhärenten Mangel an global übergreifenden gemeinsamen Zielen und Erfolgskriterien der Agenten. Dies führt oft zu Zielkonflikten zwischen den Agenten in einem MAS. Aufgrund dieser Tatsache besteht das grundlegende Problem in der MAS-Forschung darin, die Agenten so zu koordinie-

ren, dass sie kohärent handeln. Dies erfolgt in der Regel auf der Grundlage von Verhandlungs- bzw. Koordinationsmechanismen, gerade wenn es sich um marktlich organisierte MAS handelt beziehungsweise Allokationsprobleme zu lösen sind.

Für die Auswahl oder den Entwurf von geeigneten Koordinationsmechanismen sind die Erkenntnisse oben beschriebenen Koordinationstheorie sowie der *Mechanismus-Design-Theorie* (englisch *Coordination Mechanism Design*) hilfreich [Hehenkamp 2007; Russ 1997], welche ein Teilgebiet der *Spiel- und Entscheidungstheorie* darstellt [Varian 1999]. Letztere hat das Ziel, Koordinationsmechanismen so zu gestalten, dass für die daran teilnehmenden Akteure Anreize geschaffen werden, die sie dazu bewegen, freiwillig ein Verhalten zu entwickeln, das zu einem gewünschten gemeinschaftlichen Gesamtverhalten führt.

Die Mechanismus-Design-Theorie kann genutzt werden, um Protokolle für rationale, kompetitive Agenten zu generieren, denen ein stabiles Lösungskonzept zugrunde liegt und die dadurch theoretisch abgesichert zu wünschenswerten Ergebnissen führen. Entsprechende Anwendungen finden sich beispielsweise in den Arbeiten von Zlotkin und Rosenschein (siehe u. a. [Zlotkin & Rosenschein 1994]). Details zu den Grundlagen dieser Theorie finden sich beispielsweise in [Hehenkamp 2007].

Auswahlkriterien für Koordinationsmechanismen sind u. a.

- *Effizienz-Merkmale des Mechanismus* (Berechnungs-Effizienz, Kommunikationseffizienz, Geschwindigkeit bzw. Unmittelbarkeit, Einfachheit des Mechanismus usw.)
- *Effizienz-Merkmale der erzielten Lösungen* (Allokationseffizienz, Pareto-Effizienz, Gesamtnutzen-Maximierung usw.)
- *Stabilitäts-Eigenschaften* (Existenz dominanter Strategien und/oder spieltheoretischer Gleichgewichte wie z. B. das *Nash-Gleichgewicht*)
- *Teilnahme-Anreize der Akteure* (Teilnahme-Kosten, Nutzenverteilungs-Eigenschaften der erzielten Lösungen etc.)
- *Verhaltens-Anreize* (z. B. Motivation zur wahrheitsgemäßen Kommunikation bzw. Offenbarung von Präferenzen, *Anreiz-Kompatibilität*)
- *Sicherheits-Eigenschaften* (Transparenz, Manipulierbarkeit, Vertraulichkeit bzgl. offener Präferenzen, Verhinderung spekulativen Verhaltens etc.)
- *Symmetrie-Eigenschaften* (Gleichbehandlung aller Teilnehmer, Fairness)
- *Skalierbarkeit*

Prinzipiell kann die Koordination von Agenten *zentral* oder *dezentral* erfolgen. Im zentralen Fall sammelt eine zentrale Instanz alle Informationen, die für die Generierung einer Lösung relevant sind, berechnet diese und teilt sie den teilnehmenden Agenten mit, die diese dann umzusetzen haben. Dies hat zur Folge, dass die Teilnehmer bei einer zentralen Koordination ihre Autonomie bis zu einem gewissen Grad aufgeben, je nachdem wie hoch der Zwang zur Offenbarung von Informationen sowie zur Umsetzung der fremdbestimmten Lösung ist.

Im dezentralen Fall bleibt die Autonomie der Teilnehmer vollständig erhalten, da die Koordination dann über Verhandlungen erfolgt, welche die Agenten jederzeit aufnehmen oder abbrechen können. Darüber hinaus ist es ihnen überlassen, welches Maß an Information sie während der Verhandlungen offenbaren, wie viele Verhandlungen sie beginnen, welche Verhandlungspartner sie auswählen und ob sie ggf. mehrere Verhandlungen parallel führen.

Dies macht dezentrale Koordinationsansätze wesentlich flexibler und ausfallsicherer als zentrale. Zudem besteht auch nicht die Gefahr, dass die zentrale Instanz für das MAS zu

einer Art Flaschenhals oder Sicherheitslücke wird. Jedoch geht dies häufig zu Lasten der Koordinationseffizienz, da in der Regel die Kommunikationskosten steigen, der Weg der Lösungsfindung langwieriger ist und häufig auch keine global optimalen Lösungen gefunden werden, da die Agenten nur aus ihrer individuellen Sicht heraus agieren.

[Moulin 1995] unterscheidet bei Koordinationsprozessen darüber hinausgehend drei Modi (vgl. hierzu auch die Zusammenfassung von [Eymann 2000, S. 33 ff.]):

- *Gerechtigkeits-Modus (Justice Mode)*: Hier existiert eine zentrale Koordinator-Instanz (sogenannter *Arbitrator*), welcher eine möglichst optimale Allokationslösung für das Koordinationsproblem berechnet und gemäß einer Formel eine gerechte bzw. faire Verteilung des Koordinationsgewinns durchführt.

Damit für die Akteure bzw. Agenten die Teilnahme an einer solchen über eine zentrale Instanz realisierten Koordination individuell rational ist, dürfen die Kosten dieser Instanz, die letztlich von den Akteuren zu tragen sind, nicht größer sein als der von den Akteuren dadurch zu erwartende Nutzenzuwachs. Ansonsten entsteht bei den Akteuren der Anreiz, andere Koordinationsmechanismen bzw. -modi (s. u.) auszuprobieren (s. dazu auch [Rasmusson & Janson 1999, S. 143-150]).

Weitere Probleme einer Arbitrator-gesteuerten Koordination bestehen darin, dass die Agenten dem Arbitrator gegenüber ihre Präferenzen bzw. ihre Nutzenfunktionen offenbaren und der Arbitrator daher sehr vertrauenswürdig sein muss, damit er seine hervorgehobene Stellung nicht für Manipulationen ausnutzt oder entsprechende Informationen weitergibt.

- *Modus direkter Verhandlungen (Direct Agreement Mode)*: Hier erreichen die Teilnehmer eine Lösung des Koordinationsproblems ohne die Unterstützung durch eine zentrale Koordinator-Instanz, indem sie direkt miteinander interagieren bzw. verhandeln (z. B. stimmen sie sich mittels bilateraler Verhandlungen über den Tausch von Ressourcen und Gütern ab). Durch geschicktes strategisches Verhalten ist es für die Teilnehmer möglich, sich Vorteile zu verschaffen, die in der Regel zu einer weniger fairen Lösung führen als im Gerechtigkeits-Modus. Finden Transaktionen statt, so variieren die Preise von Transaktion zu Transaktion, da sie sich gemäß des Verlaufs der jeweiligen Verhandlung individuell bilden.
- *Modus des dezentralen Verhaltens (decentralized behavior mode)*: Hier kommunizieren die Agenten nicht miteinander, sondern über einen Intermediär, in der Regel einen sogenannten Walras-Auktionator [Walras 1874], als einzigen Kommunikationspartner.

Dieser sendet für ein zu tauschendes Gut Preisvorschläge an die Akteure und diese teilen ihm daraufhin auf Basis ihrer individuellen Präferenzen mit, wie viele Einheiten des Gutes sie bereit wären, zu diesem Preis anzubieten oder nachzufragen. Dies tut der Intermediär solange, bis ein Gleichgewichtspreis gefunden ist, bei dem die Angebotsmenge der Nachfragemenge entspricht. Alle Nachfrager erhalten anschließend die von ihnen gewünschte Menge des Gutes.

Im Unterschied zum Modus direkter Verhandlungen zahlen dabei alle den gleichen Preis (Gleichgewichtspreis).

Und im Unterschied zum Gerechtigkeitsmodus berechnet der Intermediär keine Allokationslösung, sondern diese ergibt sich durch das dezentrale Entscheidungsverhalten der Akteure von selbst.

Abbildung 6 gibt einen Überblick über die von Moulin definierten Koordinations-Modi. Für die Lösung von Ressourcenallokations- und Aufgabenverteilungs-Problemen in MAS hat sich der Einsatz von marktlichen Koordinationsmechanismen (siehe Kapitel 2.7) bewährt.

Die im Rahmen dieser Arbeit relevanten marktlichen Koordinationsmechanismen werden in Kapitel 3.3 betrachtet. Im Fokus der Arbeit steht dabei die Koordination der Agenten mittels des Modus direkter (bilateraler) Verhandlungen. Das zur Koordination eingesetzte Verhandlungsprotokoll wird in Kapitel 5.3 beschrieben. Als zentrale Instanz ist lediglich ein Verzeichnisdienst implementiert, der den Agenten Auskunft über in Frage kommende Verhandlungspartner innerhalb des MAS gibt.

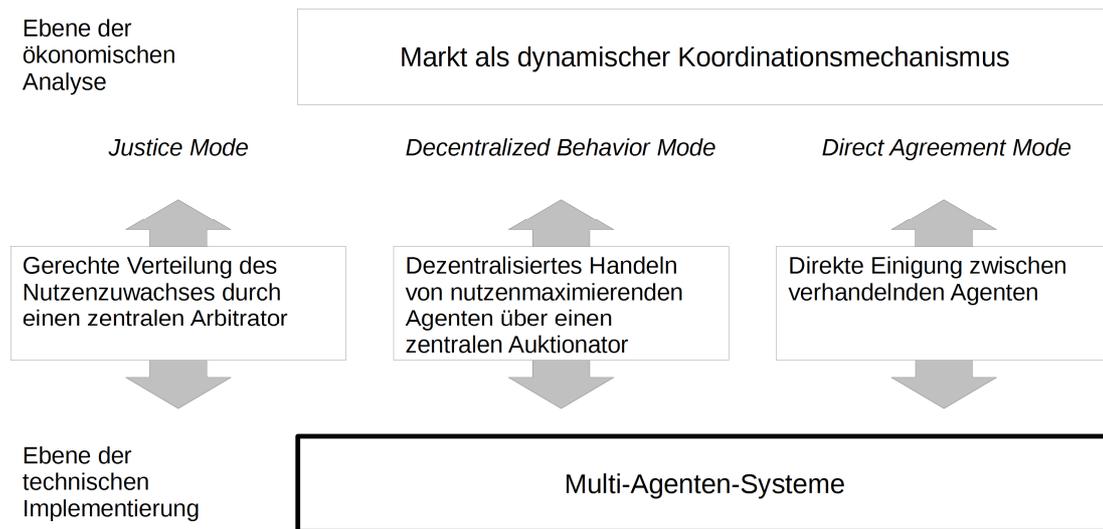


Abbildung 6: Überblick über die Koordinations-Modi nach Moulin¹²

2.6 Holonische Multiagenten-Systeme

Im Folgenden wird der Begriff *Holonische Multiagenten-Systeme (H-MAS)* erläutert. Die Nutzung holonischer Konzepte ermöglicht die Schaffung effizient organisierter MAS, die der wachsenden Komplexität skalierbarer Systeme Rechnung tragen (siehe [Jennings 1999]). Effiziente MAS-Strukturen entstehen in H-MAS dadurch, dass die Agenten den Grad ihrer Kooperation dynamisch anpassen können, indem sich beispielsweise autonome Agenten temporär zu einer übergeordneten Einheit formieren, um eine gemeinsame Aufgabe effizient zu lösen. Dabei kann eine Vielzahl an Organisationsformen emergieren, die menschlichen Organisations- oder Unternehmensformen ähneln (vgl. [Schillo 2004]).

In der MAS-Forschungsgruppe am DFKI in Saarbrücken wurden unter der Leitung von Prof. Dr. Jörg Siekmann und Dr. Klaus Fischer wesentliche konzeptionelle Beiträge zur Modellierung von *holonischen Agenten* und H-MAS (siehe u. a. [Gerber et al. 1999a; Gerber et al. 1999b; Fischer et al. 2003]) sowie von *Holonische Manufacturing Systems (HMS)* [Fischer 1998; Fischer 1999] geleistet.

Darüber hinaus wurde in der MAS-Forschungsgruppe eine Reihe von H-MAS für verschiedene Anwendungsszenarien entwickelt. Dazu gehören unter anderem Systeme zur Dispositions- und Ressourcenplanung in Speditionen (TELETRUCK) [Bürckert et al. 1998], zur forstwirtschaftlichen Logistik (CASA) und zur Einsatzplanung von Maschinen in der Landwirtschaft (AGRICOLA) [Gerber 2005].

¹² Quelle: [Eymann 2000, S. 33]

Weiterhin wurden holonische Geschäftsagenten, ein holonischer agentenbasierter Koordinationsserver und eine holonische Koordinationsinfrastruktur entwickelt, die sich für den Einsatz auf elektronischen Märkten oder in marktlich koordinierten Supply Webs (siehe Abschnitt 3.2.2) eignen (siehe [Ruß & Gerber 2000; Ruß & Gerber 2001a; Ruß & Gerber 2001b; Ruß & Gerber 2001c; Ruß et al. 2001; Ruß et al. 2003]). Wesentliche Aspekte ihrer Modellierung werden in Abschnitt 4.3 präsentiert.

Der dort vorgestellte holonische agentenbasierte Koordinationsserver wurde in der von mir betreuten Diplomarbeit [DeBloch 2002] implementiert. In der Diplomarbeit [Puin 2004] wurden generische Supply-Chain-Entitäten (d. h. Ressourcen-Produzenten, Produzenten, Händler und Konsumenten) agentifiziert, d. h. konzeptionell entwickelt und implementiert.

Darüber hinaus wurde in [Puin 2004] die Koordinationseffizienz für verschiedene Szenarien (bei Variation der vom Koordinationsserver angebotenen Auktionsmechanismen, der Angebots- und Nachfragesituation sowie den von den Supply-Chain-Agenten genutzten Strategien) simulativ untersucht und anhand von Kennzahlen auch quantitativ analysiert. Detailliertere Informationen hierzu finden sich in den genannten Diplomarbeiten.

Die von der MAS-Gruppe am DFKI erzielten Forschungsergebnisse haben gezeigt, dass H-MAS auf vielfältige Weise zur inner- und überbetrieblichen Koordination in Unternehmensnetzwerken beitragen können.

2.6.1 Holonische Systeme

Die Grundlagen holonischer Systeme basieren auf den Arbeiten des ungarischen Philosophen Arthur Koestler, der beispielsweise in seinem Buch „The Ghost in the Machine“ [Koestler 1967] untersucht, wie biologische Systeme organisiert sind. Seiner Ansicht nach bestehen solche Systeme, zum Beispiel ein biologischer Organismus, aus einer Vielzahl an einzelnen Komponenten mit spezifischen Aufgaben, die miteinander interagieren und sich zu immer größeren Organisationseinheiten mit dabei zunehmendem Aufgabenspektrum zusammenfügen.

Die Bezeichnung *Holon* setzt sich aus den griechischen Wörtern „holos“ (d. h. „Ganzes“) und der Nachsilbe „on“ (d. h. Teilchen) zusammen. [Koestler 1967] versteht unter einem Holon den Bestandteil eines größeren Systems beziehungsweise Konstrukts, welches wiederum aus mehreren anderen Elementen aufgebaut sein kann, die selbst wieder zusammengesetzte oder *atomare Holone* (s. u.) bilden.

Die organisatorische Struktur eines Holons und der es konstituierenden Elemente bzw. Sub-Holone sowie die Interaktionen zwischen diesen Elementen ermöglichen dabei im Idealfall eine flexible, effiziente und robuste Durchführung der von dem Holon zu leistenden Aufgaben. Eine besondere Eigenschaft holonischer Systeme besteht in einer gewissen, der Einsatzdomäne angepassten, Entscheidungsautonomie ihrer Sub-Systeme. Holonische Systeme weisen gegenüber auf andere Weise strukturierten und organisierten Systemen folgende Vorzüge auf:

- Im Unterschied zu nicht-holonischen Systemen (insbesondere hierarchisch organisierten) verfügen holonische Systeme über das Potenzial, hierarchische und heterarchische Eigenschaften in Abhängigkeit von den Erfordernissen der Einsatzdomäne geeignet zu kombinieren. Darüber hinaus ist ein einzelnes Holon nicht zwingend an eine bestimmte Hierarchie gebunden, sondern es kann gleichzeitig in mehreren Hierarchien aktiv sein und in diesen Aufgaben übernehmen. Durch diese Eigenschaft zeichnen sich holonische Systeme durch eine besondere Flexibilität und Robustheit aus (siehe u. a. [Schillo 2004]).

- Streng hierarchisch organisierte Systeme weisen häufig eine recht starre Kopplung ihrer Komponenten auf, wodurch die Performanz des Gesamtsystems sehr stark vom Funktionieren einzelner Teile beziehungsweise Module abhängt. Im Extremfall bedeutet damit der Ausfall einer kleinen Einheit das Versagen des Gesamtsystems im Hinblick auf die Erfüllung seines Einsatzzwecks. Die Leistung des Gesamtsystems würde hierdurch zumindest beeinträchtigt. Da Holone und ihre Sub-Holone im Gegensatz dazu einen deutlich höheren Grad an Autonomie genießen, besteht für ein holonisches System dagegen die Möglichkeit, sich an die Änderung oder den Ausfall von untergeordneten Elementen sowie auch an Schwankungen in der Menge der von ihm durchzuführenden Aufgaben zu adaptieren.
- Weiterhin weisen die Komponenten hierarchisch organisierter Systeme zumeist auch eine festgesetzte Anzahl an Komponenten auf. Da sich die Teile eines Holons durch Interaktionen koordinieren, können zur Systemlaufzeit weitere Teile hinzukommen oder Teile ausgetauscht werden. Dies führt zu einer hochgradigen Skalierbarkeit sowie auch zu einer besonderen Robustheit und Flexibilität solcher Systeme.

Aufgrund der geschilderten Eigenschaften holonischer Systeme werden die Ideen von Koestler bei der Entwicklung von MAS eingesetzt, für deren Anwendungsdomänen die dargelegten Eigenschaften (wie z. B. Adaptivität, Flexibilität, Stabilität und Effizienz) von besonderer Wichtigkeit sind (Beispiele finden sich oben sowie in Abschnitt 4.3).

2.6.2 Holonische Agenten

[Gerber et al. 1999a; Gerber et al. 1999b] geben eine exakte Definition für den Begriff eines holonischen Agenten. In holonischen Multiagenten-Systemen können einzelne Agenten die Entscheidung treffen, ihre Autonomie temporär einzuschränken oder aufzugeben und ein sogenanntes übergeordnetes Holon zu bilden, das eine kohärente fraktale Struktur [Koestler 1967] darstellt und ein gemeinsames Ziel verfolgt.

Das neu gebildete übergeordnete Holon agiert dann nach außen wie ein einziger autonomer Agent und wird nach außen durch einen ausgewählten Agenten repräsentiert, den Kopf des Holons (englisch „Head of Holon“). Dieser war entweder bereits Bestandteil des Holons oder er wurde für die Übernahme dieser Funktion neu in das Holon integriert. Letzteres kann durch die Neuerzeugung oder aber die Integration bereits bestehender Agenten von außerhalb erfolgen. Die weiteren Mitglieder des Holons bilden in ihrer Gesamtheit seinen Körper (englisch „Body of Holon“).

Ein Holon bildet sich in der Regel aus einer Menge von Sub-Holons (sogenannte *Holonisierung*). Ein Holon, welches in keine weiteren untergeordneten Agenten oder Sub-Holone zerlegbar ist, wird als atomares Holon bezeichnet. Agenten und Sub-Holone können parallel innerhalb von mehreren verschiedenen Holons agieren.

Die vorgestellte Definition holonischer Agenten führt im Vergleich zu den vorgenannten klassischen Agenten-Definitionen bzgl. der folgenden Merkmale zu Modifikationen und Erweiterungen:

- **Autonomie:** Ein Holon bildet sich in der Regel aus einer Menge von Agenten und agiert als selbständige Einheit. Insbesondere tritt ein Holon in Interaktionen mit seiner Umwelt als ein einziger autonomer Agent auf. Tritt ein Agent einem Holon bei, so passt er seine Aktivitäten dessen übergeordneten Zielen an bzw. beugt sich diesen. Zudem muss er in diesem Fall je nach Einsatzdomäne und Holon bereit sein, bestimmte Beschränkungen seiner Autonomie in Kauf zu nehmen (beispielsweise in Bezug auf seine Aktions-, Interaktions- und Kommunikationsfähigkeiten).

- **Zielgerichtetes Verhalten:** Agenten, die Teil eines Holons sind, können neben den übergeordneten Zielen des Holons auch individuelle Ziele verfolgen. Jedoch ist erforderlich, dass sie mit dem übergeordneten Holon zumindest ein Ziel gemeinsam haben und zu dessen Erfüllung beitragen. Die vom übergeordneten Holon verfolgten Ziele ergeben sich aus den Zielen seiner ihm untergeordneten, es konstituierenden Agenten und Sub-Holons. Dabei muss der Holon nicht notwendigerweise konkrete Ziele seiner agentenbasierten Sub-Strukturen unverändert übernehmen. Jedoch dürfen seine Ziele und die seiner Mitglieder nicht zueinander im Widerspruch stehen. Solange dies für einen Agenten als Mitglied eines Holons gewährleistet bleibt, kann er gleichzeitig Mitglied in mehreren Holonen sein.
- **Bessere Möglichkeiten durch Gruppenbildung:** Holonische Agenten besitzen die Fähigkeit zur Bildung von Gruppen und übergeordneten Holonen, in denen sie ihre Fähigkeiten bündeln.
- **Glauben:** Das Wissen bzw. die Vorstellung eines Agenten bzgl. seiner Umwelt kann sowohl explizit in einem spezifischen Teil des Holons repräsentiert werden als auch verteilt vorliegen. In letzterem Fall stellen die Teile des Holons ihre lokalen Wissensausschnitte dem Holon zur Verfügung, sobald dieses benötigt wird.
- **Begrenzte Rationalität:** Das übergeordnete Super-Holon darf das Ressourcenmanagement seiner Teil- bzw. Sub-Holone kontrollieren und hierzu Anweisungen oder zumindest Richtlinien ausgeben.
- **Kommunikationsfähigkeit:** Die Fähigkeit, mit anderen Agenten direkt zu kommunizieren, kann eingeschränkt sein. In betriebswirtschaftlichen Anwendungsdomänen kann es beispielsweise sinnvoll sein, bestimmte Teile eines Holons (z. B. die „Produktionsabteilung“ als Sub-Holon des Holons „Unternehmen“) nur eingeschränkt mit anderen Holonen Informationen austauschen zu lassen, die nicht Mitglied im gleichen Holon sind. Bestimmte Informationsaustausche sind nur dem Super-Holon und nicht seiner Sub-Holone gestattet.

2.6.3 Holonische MAS-Strukturen

Anhand des Autonomiegrades, welchen Agenten als Mitglieder eines Holons beibehalten, kann man zwischen verschiedenen Formen der Holonisierung differenzieren. Im Extremfall entsteht das Holon entweder durch einen Gruppenbildungsprozess oder im Gegensatz dazu durch vollständige Verschmelzung.

2.6.3.1 Holon als Gruppe autonomer kooperierender Agenten

Im ersten Fall (siehe Abbildung 7) basiert der Holon rein auf Kooperation und entspricht damit einer *Gruppe von autonomen kooperierenden Agenten* bzw. in weitgehendem Maße einem Multiagenten-System nach klassischer Definition, wobei die Mitglieder auf beliebige Arten holonisch aufgebaut sein können, aber innerhalb des Gruppen-Holons nichts von ihrer Autonomie aufgeben.

Im Fall des Holons als Gruppe autonomer Agenten dient die Verfolgung des holonischen Design-Ansatzes hauptsächlich dem Designer eines MAS hauptsächlich als Design-Hilfe, um das MAS strukturierter entwerfen und entwickeln zu können.

Durch den beibehaltenen hohen Autonomiegrad können die einzelnen Agenten gleichzeitig in mehreren Holonen Mitglied sein. Der hohe Grad an Autonomie und die Realisierung durch Kooperation bringt aber im praktischen Einsatz oft einen erhöhten Kommunikations- und Koordinationsaufwand mit sich.

Entsprechende Aufwände lassen sich gegebenenfalls durch gemeinsame Richtlinien beschränken, die regeln, welche Agenten zu welchem Zweck miteinander kommunizieren oder interagieren dürfen.

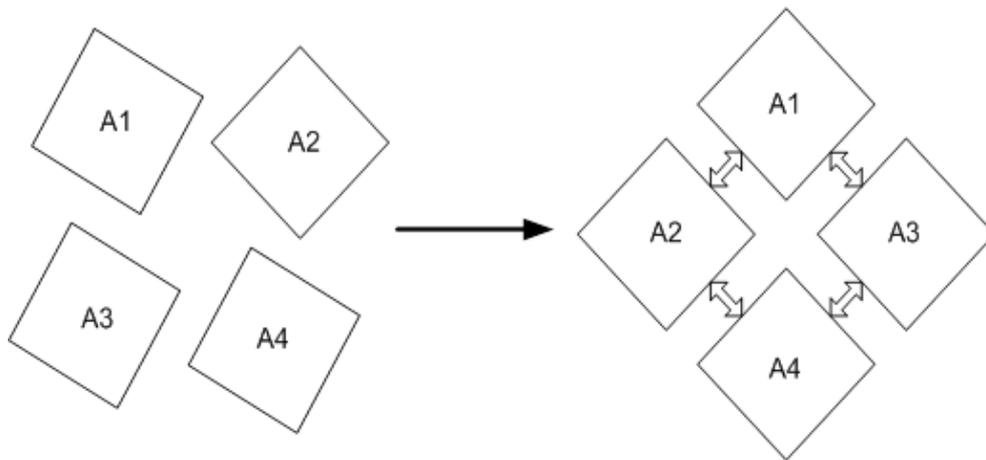


Abbildung 7: Der Holon als Gruppe autonomer kooperierender Agenten¹³

2.6.3.2 Holon als vollständige Verschmelzung von Agenten

Der zweite Extremfall besteht in der *vollständigen Verschmelzung* der das Holon bildenden Agenten (siehe Abbildung 8). Hierbei handelt es sich um eine Ausprägung der Holonisierung, bei der die Agenten auf ihre Autonomie vollständig verzichten.

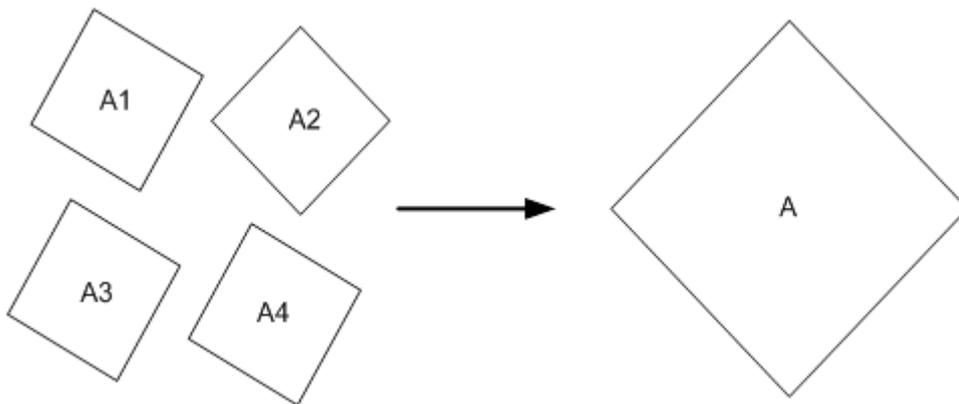


Abbildung 8: Das Verschmelzen mehrerer Agenten zu einem Holon¹⁴

Hierbei ist sogar vorstellbar, dass sich die Software-Agenten, wenn sie einem solchen Holon beitreten, selbst terminieren, um beispielsweise ihre bisherige Berechnungsautonomie aufzugeben und mit ihren Fähigkeiten als neue Instanz innerhalb einer vom Holon kontrollierten Umgebung „reinkarniert“ werden.

Bei der Verschmelzung erfolgt die Aggregation des vorhandenen lokalen Wissens der verschmelzenden Agenten. Hierbei müssen Konflikte bzgl. sich möglicherweise widersprechender Einträge in den lokalen Wissensbasen auf geeignete Weise (z. B. durch eine Art Durchschnittsbildung) gelöst werden. Durch das aggregierte Wissen verfügt der Holon über

¹³ aus [Gerber et al. 1999b]

¹⁴ aus [Gerber et al. 1999b]

ein vollständigeres und damit in der Regel umfassender nutzbares Weltwissen als die Agenten, aus denen er gebildet wurde.

Ein weiterer Vorteil der vollständigen Verschmelzung besteht in der Minimierung der Kommunikations- und Koordinationskosten. Ein Nachteil besteht darin, dass die Fähigkeit zum verteilten Problemlösen stark eingeschränkt wird.

Die beiden genannten Extremfälle sind jedoch selten von praktischer Relevanz. Meistens wird in Abhängigkeit von den Anforderungen der Anwendungsdomäne je nach Aufgabenstellung ein H-MAS realisiert, welches verschiedene Ausprägungen der Extremfälle auf geeignete Weise mischt. Insbesondere werden die Agenten in der Praxis ihre Autonomie nur zum Teil aufgeben wollen, was eine dritte, im Folgenden vorgestellte Ausprägungsform eines Holons erlaubt.

2.6.3.3 Holon als Kopf-gesteuerter Verbund von Agenten

Die dritte Form der Holonisierung besteht in der Realisierung als *Kopf-gesteuerter Verbund* (siehe Abbildung 9). Der Autonomiegrad der das Holon bildenden Agenten liegt bei dieser Form zwischen den genannten extremen Ausprägungsformen des Holons als *Gruppe autonomer kooperierender Agenten* bzw. als deren *vollständige Verschmelzung*. Die Agenten geben dabei ihre Autonomie lediglich teilweise auf.

Um die Koordinations- und Kommunikationseffizienz zu steigern, fungiert bei dieser Realisierungsform häufig einer der Agenten des Holons als dessen „Kopf“, der die Abläufe und Interaktionen innerhalb des Holons in sinnvollem Umfang kontrolliert und ggf. auch mit Weisungsbefugnissen gegenüber den Sub-Holons steuert. Oft besitzt der Kopf-Agent die Befugnis, seinen Sub-Holonen Ressourcen zuzuteilen.

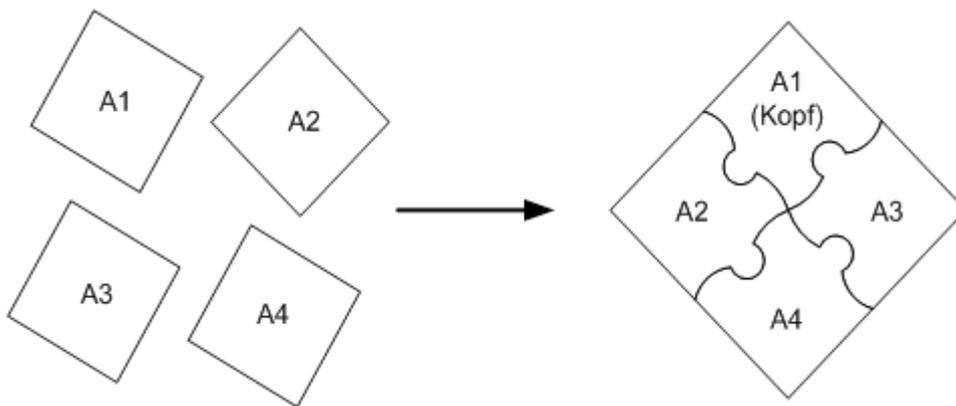


Abbildung 9: Der Holon als Kopf-gesteuerter Verbund¹⁵

Zusätzlich repräsentiert er das Holon in der Regel bei nach außen gerichteten Interaktionen als Einheit (siehe Abbildung 9) und kann auch (z. B. bei Verhandlungen mit externen Entitäten) autonom für das von ihm gesteuerte Holon Entscheidungen treffen. Weiterhin ist der Kopf oftmals dazu bevollmächtigt, für das Holon Pläne zu entwerfen und den Sub-Holonen Teilpläne und/oder Aufgaben zuzuweisen. Darüber hinaus verfügt er häufig über die Autorisation, weitere Agenten in das Holon zu integrieren oder bereits integrierte Agenten wieder auszugliedern [Fischer et al. 2003].

¹⁵ aus [Gerber et al. 1999b]

Der sich ergebende Holon kann in seiner Ausprägung sowohl einer moderierten Gruppe als auch einer streng hierarchisch organisierten hierarchischen Struktur entsprechen. Weitere theoretische Grundlagen zur Holonisierung sind in [Fischer et al. 2003] dargelegt.

Die beschriebenen konzeptionellen Grundlagen holonischer Multiagentensysteme dienen dem Verständnis der Strukturierung der in Kapitel 4.3 vorgestellten holonischen Geschäftsagenten und Koordinationsserver-Agenten.

2.7 Marktliche MAS-Koordination

Märkte sind in der Lage, effiziente Allokationen, d. h. den Austausch und die Zuteilung von Ressourcen, Gütern und Aufträgen auch bei fehlender oder nur in sehr geringem Maße vorliegender zentraler Steuerung zu erreichen.

Diese Eigenschaft von Märkten gilt insbesondere auch in nur partiell beobachtbaren, dynamischen und nicht-deterministischen Umgebungen (vgl. Abschnitt 2.3.3), wie z. B. in typischen E-Commerce-Szenarien oder auch dem in dieser Arbeit vorgestellten Multiagentensystem MACSIMA (s. Kapitel 5).

Entsprechende Umgebungen sind dadurch charakterisiert, dass nur ein sehr unvollständiges und unsicheres Wissen über die Fähigkeiten, Ziele, Präferenzen und Zustände der anderen Marktteilnehmer bzw. Agenten verfügbar ist.

[Moulin 1995] charakterisiert Märkte folgendermaßen:

„Markets describe the exchange and production of commodities in the private property regime. The ownership structure assigns each unit of good and each technology to exactly one participant. Under specialization and division of labor, the initial allocation of resources is not expected to be Pareto-optimal [...]. Markets are meant to exploit these cooperative opportunities by eliciting Pareto improving trades“ [Moulin 1995, S. 45]

Um eine effiziente Allokation zu erreichen, muss jeder am Markt teilnehmende Agent nur an sich selbst denken. Und die daraus resultierende Allokation wird sehr oft optimal oder nah an einer optimalen Lösung sein, d. h. von einem allwissenden Arbitrator bzw. zentral entscheidenden und steuernden Ressourcen-Allokator in vielen Fällen kaum zu verbessern sein. Vor diesem Hintergrund werden zur Koordination in MAS häufig marktliche Koordinationsmechanismen eingesetzt. Diese müssen aber durch geeignete marktliche Verhandlungsprotokolle realisiert werden.

„Markets need policies and protocols through which agents communicate their intentions for doing particular tasks, and reach agreement using negotiation.“ [Bond & Gasser 1988]

Das prominenteste Protokoll für die marktliche Organisation von Multiagentensystemen ist das *Contract Net Protocol (CNP)* von Smith [Smith 1980; Smith & Davis 1981; Sandholm 1993, S. 295-308], welches die Allokation von Aufgaben und Ressourcen durch deren Ausschreibung und kompetitives Bieten in einem Markt realisiert (das CNP wurde im Detail in Abschnitt 2.5.2 beschrieben).

Auf der Grundlage der Forschungsarbeiten von Smith sind von verschiedenen Autoren weitere Erweiterungen des CNP vorgeschlagen worden [Smith & Davis 1988; Sandholm 1993; Sandholm & Lesser 1995b; Malone 1987b].

Im ursprünglichen CNP waren die Agenten kooperativ und sie haben Aufgaben füreinander kostenlos ausgeführt. Aber im Bereich der betriebswirtschaftlichen MAS-Anwendungen [Fischer et al. 1996c; Gomber et al. 1998] wie in der Transport-Domäne [Fischer et al.

1996b], der unternehmensübergreifenden Fertigung [Sandholm & Lesser 1995a; Sandholm & Lesser 1995c] und im Electronic Commerce (EC) [Sandholm & Lesser 1995b] müssen sich Agenten gegenseitig für die Übernahme und die Erledigung von Aufgaben bezahlen. In diesen Umgebungen akzeptieren Agenten nur individuell rationale Verträge, die auf ihren lokalen Kostenberechnungen und Nutzenbewertungen basieren.

In diesen MAS-Domänen, die virtuelle elektronische Märkte beinhalten, repräsentieren Agenten im Allgemeinen Entitäten der realen Welt, von denen a-priori anzunehmen ist, dass diese eigennützig handeln. Das heißt, sie wurden von den verschiedenen Organisationen, in deren Auftrag sie handeln, analog zu deren Ziel- und Präferenzsystemen heterogen gestaltet, um ausschließlich die Interessen ihrer „Besitzer“ zu verfolgen. Daher fördern sie unter allen Umständen Problemlösungen, die nur im Hinblick auf ihr eigenes individuelles Präferenzsystem nutzenmaximierend sind.

Dabei sind die potenziellen Auswirkungen ihrer Strategien auf den von anderen Agenten realisierten Nutzen nicht Gegenstand der Betrachtung bzw. irrelevant, was die Zahl der erreichbaren gemeinsamen Problemlösungen im System einschränkt und damit die von den beteiligten Agenten realisierbare Gesamtnutzensumme.

Die Markt-Interaktionsregeln, die in den Interaktionsprotokollen spezifiziert sind, bestimmen sowohl die Allokationsentscheidungen (die Zuteilung der Aufgaben an die Agenten, die für deren Durchführung am besten geeignet sind) innerhalb des MAS als auch die Preisung der Aufgaben.

Insofern haben diese Regeln einen entscheidenden Einfluss auf die Performanz eines marktlichen MAS. Wenn daher die Interaktionsregeln eines marktlich organisierten MAS mit eigennützigen Agenten Spielraum für antisoziale Verhaltensweisen bieten (wie z. B. Lügen), wird dieses Verhalten unter den eigennützigen Agenten emergieren, wann immer dies für diese individuell vorteilhaft ist.

Ungünstig gewählte Interaktionsregeln führen daher zu suboptimalen Allokationsentscheidungen und damit zu einem Verlust an globaler System-Performanz. Dadurch besteht die Gefahr, dass bei den am MAS teilnehmenden Agenten die Motivation sinkt, an weiteren Interaktionen innerhalb des MAS teilzunehmen. Im schlimmsten Fall kann dies zu einem Boykott des Systems führen. MAS, die ein offenes System darstellen, sind in diesem Fall in ihrem Bestehen gefährdet, da Agenten das System verlassen können, wenn sie unzufrieden sind beziehungsweise ihnen ihre Teilnahme nur einen geringen Nutzen verspricht.

Daher müssen Konventionen für Agenten-Interaktionen gefunden werden, welche ein Verhalten der individuellen Agenten garantieren, die sowohl zu einem kohärenten, sozial gewünschten Verhalten des gesamten MAS führen, als auch zur Akzeptanz des Systems und insbesondere seiner Koordinationsmechanismen durch die Agenten.

Die Gestaltung der Regeln des Marktes hat entscheidende Auswirkungen auf die Performanz eines marktlichen MAS. Trotz dieser Tatsache fehlt es der VKI-Forschung weiterhin an dem Verstehen von „... *market rules that produce coherent system performance.*“ [Bond & Gasser 1988].

Die Idee, wie dies erreicht werden kann, basiert auf effizienten Koordinationsmechanismen, die auf der Basis der Erkenntnisse der Mechanismus-Design-Theorie (vgl. Abschnitt 2.5.3) je nach Aufgabenstellung bzw. dem zu lösenden Koordinationsproblem im MAS ausgewählt bzw. entworfen werden.

Es kann aufgrund ihrer konstituierenden Charakteristika angenommen werden, dass sich Agenten in betriebswirtschaftlichen Anwendungsszenarien, gerade wenn diese marktlich

koordiniert werden, ähnlich wie Menschen verhalten. Zudem ist es gemäß der Idee der *Artificial Societies* [Malsch 1997; Malsch 1998; Malsch 2002] gut vorstellbar, dass zukünftig in solchen Szenarien Software-Agenten mit menschlichen Agenten interagieren, konkurrieren und kooperieren.

„Die zentrale Idee ist, dass eine effiziente, gemäß gewünschter Charakteristika optimale Gesamtperformanz eines MAS durch ein angepasstes Design seines Koordinationsmechanismus realisiert werden kann.“ [Malsch 1999]

Wenn ein MAS ein offenes System darstellt, in welches ggf. (z. B. auf einem virtuellen Markt) externe Agenten jederzeit eintreten bzw. dieses verlassen können, so muss der Koordinationsmechanismus, über den die Agenten interagieren, so gestaltet sein, dass er übertrieben eigennütziges Verhalten verhindert. Stattdessen soll er die Agenten zu einem Verhalten motivieren bzw. zwingen, welches in sozial wünschenswerten Systemergebnissen (z. B. gesamtutzenmaximierende und individuell rationale Ressourcenallokationen) resultiert.

Dies kann dadurch realisiert werden, dass der Koordinationsmechanismus Interaktionsregeln enthält, die dazu führen, dass die global gewünschten Interaktionsstrategien (z. B. bei einer Vickrey-Auktion benötigte Informationen wahrheitsgemäß zu offenbaren) auch diejenigen Interaktionsstrategien sind, die für die individuellen Agenten die für sie nutzenmaximierenden Strategien sind.

Da eigennützige Agenten eben diese Strategien wählen werden, führt ein entsprechend gestalteter Interaktionsmechanismus dann zur Emergenz eines vom Designer eines solchen Systems gewünschten Ergebnisses des Gesamtsystems. Abbildung 10 verdeutlicht diesen Zusammenhang.

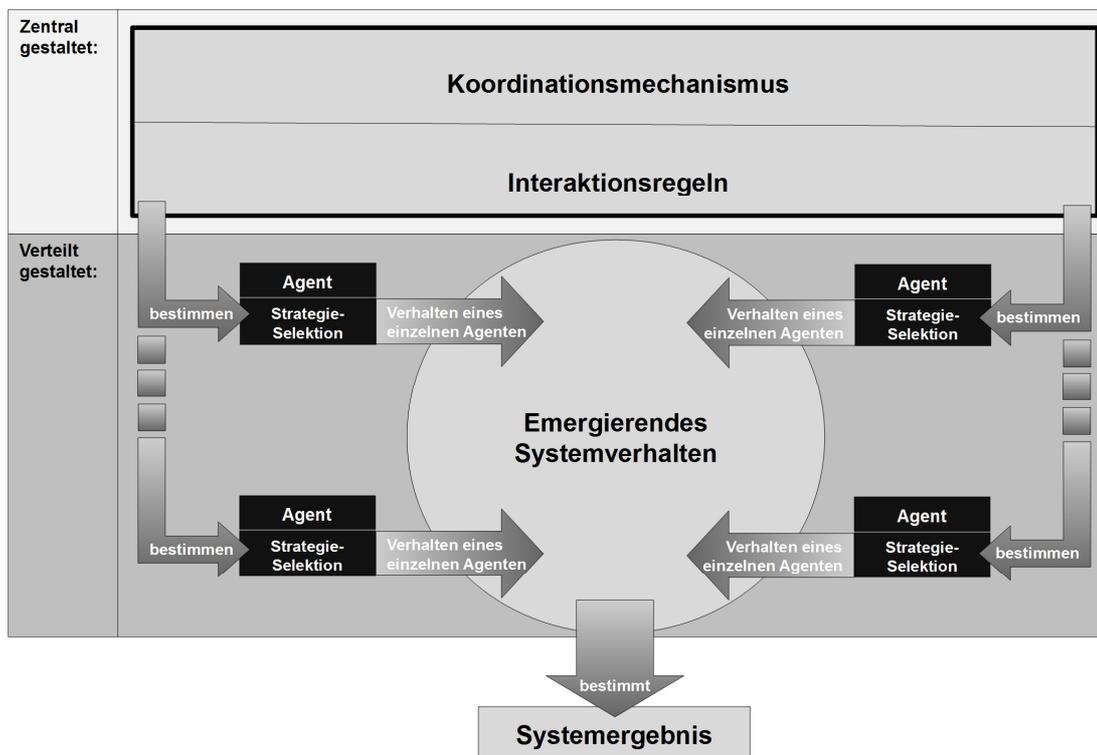


Abbildung 10: Wirkungszusammenhang zwischen dem Systemergebnis eines MAS und dem Design der Agenten und des verwendeten Koordinationsmechanismus¹⁶

¹⁶ Quelle: In Anlehnung an [Ruß 1997]

In dieser Arbeit erfolgt die Koordination der Agenten des in MACSIMA realisierten Wertschöpfungsnetzwerks bzw. der Austausch von Ressourcen und Gütern unter ihnen dezentral nach dem Modus direkter Verhandlungen (vgl. Kapitel 2.5.3 bzw. [Moulin 1995]).

Die Agenten, deren unterschiedliche Typen im Detail in Abschnitt 5.2.2 vorgestellt werden, sind *autonom*, *ziel-* und *nutzenbasiert* und besitzen *soziale Fähigkeiten*, d. h. sie sind *kommunikativ* und besitzen ein Verhandlungsmodul mit weitreichenden Verhandlungsfunktionalitäten.

Auf Basis dieser Eigenschaften und Funktionalitäten verhandeln sie miteinander auf direktem Wege über den Austausch von Ressourcen und Gütern unter Verwendung eines *bilateralen Verhandlungsprotokolls*, welches in Kapitel 5.3 erläutert wird. Weitere marktliche Protokolle beziehungsweise Auktionsmechanismen, nämlich die *Vickrey-Auktion*, die *verdeckte Erstpreis-Auktion*, die *Englische Auktion* sowie die *Holländische Auktion* (siehe Abschnitt 3.3.4.2) sind ebenfalls in MACSIMA implementiert, spielen jedoch im Rahmen dieser Arbeit keine zentrale Rolle.

Darüber hinaus sind die in MACSIMA handelnden Agenten *lernfähig*. Sie besitzen eine Lern-Komponente, die ihre Verhandlungsstrategien unter Verwendung evolutionärer Lernverfahren auf Basis der in den Verhandlungen gemachten Erfahrungen an die aktuellen Marktgegebenheiten, also ihren Umweltzustand, adaptiert.

Die MACSIMA-Agenten suchen *proaktiv* nach geeigneten Verhandlungspartnern und entscheiden *autonom* darüber, wann sie mit welchen Agenten verhandeln, welche Verhandlungsschritte sie machen und ob sie eine Verhandlung abbrechen oder erfolgreich abschließen.

Die Umwelt der Agenten, das heißt das realisierte Wertschöpfungsnetzwerk beziehungsweise der Markt, ist *dynamisch* sowie *nicht-deterministisch* und von den einzelnen Agenten nur *partiell beobachtbar*.

Die in dieser Arbeit vorgestellten Agenten und ihre Architektur sind daher dazu geeignet, um als sogenannte Geschäftsagenten auf elektronischen virtuellen Märkten eingesetzt zu werden und im Auftrag realer Wirtschaftssubjekte wie z. B. Unternehmen und ihrer Mitarbeiter potenzielle Geschäftspartner zu identifizieren und mit diesen Informationen auszutauschen, zu verhandeln und Verträge abzuschließen.

Zu den ersten Geschäftsagenten, die in diesem Sinne entworfen wurden, gehören zum Beispiel der Contact Finder [Burkey & Anagnost 1996b, Burkey & Anagnost 1995] sowie der BargainFinder [Burkey & Anagnost 1996a]¹⁷. Beide wurden im CStar Projekt [CSTaR 1997] von Andersen Consulting entwickelt.

Für das Teilproblem der Verhandlungsführung wurde von [Szapiro et al. 1991] ein Verhandlungsunterstützungssystem auf der Basis evolutionärer Algorithmen vorgeschlagen. Davon inspiriert bedient sich auch die Lern-Komponente des adaptiven Verhandlungsmoduls der MACSIMA-Agenten evolutionärer Lernverfahren, deren Komponenten zusammen mit den für sie gegebenen Einstellmöglichkeiten in Abschnitt 5.4 erläutert werden.

¹⁷ Der BargainFinder wurde entwickelt, um das World Wide Web nach interessanten Produkten zu durchsuchen, über deren Preise zu verhandeln und diese ggf. zu kaufen.

3 Ökonomische Grundlagen

3.1 Einleitung und Übersicht

Dieses Kapitel beginnt mit einer kurzen Einführung in die Grundlagen des Supply Chain Managements. Da dies ein sehr umfangreiches Gebiet ist, werden nur die für diese Arbeit relevanten begrifflichen Grundlagen beschrieben und wesentliche Konzepte kurz erklärt.

Zunächst wird in Abschnitt 3.2.1 erklärt, was unter einer *Supply Chain* zu verstehen ist. Dabei wird darauf hingewiesen, dass synonym hierzu auch viele andere Begriffe wie z. B. *Wertschöpfungskette*, *Wertschöpfungsnetzwerk* oder auch *Supply Network* oder *Supply Web* verwendet werden.

In Abschnitt 3.2.2.1 wird dargelegt, dass Supply Chains heutzutage eher netzwerkartige als lineare kettenartige Strukturen aufweisen. In der Folge wird in Abschnitt 3.2.2.2 erläutert, was im Rahmen dieser Arbeit unter dem Begriff Wertschöpfungsnetzwerk verstanden wird. Eine Abgrenzung des Begriffs ist erforderlich, da er in der Literatur nicht einheitlich definiert ist.

Im Anschluss daran wird in Abschnitt 3.2.3 erklärt, was unter dem Begriff Supply Chain Management zu verstehen ist. Auch hierfür liegen in der Literatur zahlreiche unterschiedliche Definitionen vor.

Darauffolgend gibt Kapitel 3.2.4 einen kurzen Überblick über das E-Supply Chain Management, welches zur Entstehung von sogenannten Supply Webs führt. In Abschnitt 3.2.5 werden verschiedene Supply-Chain-Modelle vorgestellt und besprochen. Von besonderem Interesse ist dabei ihre Eignung im Hinblick auf die Definition von Simulationsszenarien zur Durchführung empirischer Analysen. Vor diesem Hintergrund wird die grundlegende Struktur für Basis-Simulationsszenarien von MACSIMA abgeleitet (vgl. die Abschnitte 5.2.1 und 6.2.1).

Anschließend werden in Abschnitt 3.3 die Grundlagen marktlicher Koordination erläutert. Dazu wird ganz grundlegend besprochen, was unter einem Markt und dem Design von Koordinationsmechanismen zu verstehen ist.

Darauf aufbauend erfolgt eine Einführung in elektronische Märkte und die Vorstellung von bilateralen Verhandlungen und verschiedenen in MACSIMA implementierten Auktionsmechanismen als Beispiele für marktliche Koordinationsmechanismen.

3.2 Grundlagen des Supply Chain Management

Unternehmen stehen miteinander in einem immer schärfer werdenden globalen Wettbewerb. Die Kundenbedürfnisse werden zunehmend inhomogener bzw. individueller und verändern sich rasch, so dass auch die Produktlebenszyklen immer kürzer werden und in vielen Fällen ein Wechsel vom Verkäufer- bzw. Push-Markt zu einem Käufer- bzw. Pull-Markt erfolgt. Schnelligkeit, Kosteneffizienz, Flexibilität und Kundenorientierung sowie ein hohes Maß an Lieferfähigkeit und -treue sind daher entscheidende Erfolgsfaktoren.

Unternehmen müssen sich an diese Rahmenbedingungen anpassen, wenn sie ihre Profitabilität und Wettbewerbsfähigkeit erhalten wollen (vgl. [Beckmann 2003]). Entsprechende Anpassungsprozesse führen dazu, dass Unternehmen sich verstärkt auf ihre Kernkompetenzen konzentrieren und diese in arbeitsteilige Wertschöpfungsketten einbringen.

3.2.1 Supply Chains

Wertschöpfungsketten werden in der Literatur *Supply Chain* genannt [Ganeshan & Harrison 1995; Swam et al. 1998; Lee & Billington 1995]. Wörtlich kann man diesen Begriff auch als Liefer- oder Versorgungskette übersetzen.

Für den Begriff der Supply Chain existiert keine allgemein anerkannte Definition. [Beckmann 2003] definiert den Begriff folgendermaßen:

„Zu einer Supply Chain gehören alle Unternehmen, die an der Entwicklung, Erstellung und Lieferung eines Erzeugnisses (Produkt oder Dienstleistung) beteiligt sind. ... Die Supply Chain repräsentiert den Fluss von Leistungsobjekten durch ein Netzwerk von Wertschöpfungspartnern, das sich vom Rohstofflieferanten bis zum Endverbraucher erstreckt.“

[Tsyganova 2000] gibt einen Überblick über einige wesentliche der in der Literatur gebräuchlichen Definitionen. Sie schreibt:

„Der Begriff Supply Chain spiegelt die Verzahnung von Unternehmen bei der gemeinsamen Leistungserstellung wieder [...]. Der Supply Chain entsprechen in der deutschen Literatur die Begriffe Logistikkette, Lieferkette, Versorgungskette und unternehmensübergreifende Wertschöpfungskette.“ (vgl. [Tsyganova 2000, S. 3]).

[Stadtler 2010] versteht unter einer Supply Chain ein Produktions- und Logistiknetzwerk und betont die Vielzahl an Interdependenzen, zu denen es kommt, wenn die Mitglieder des Netzwerks Kundenaufträge gemeinsam an der Erfüllung von Kundenaufträgen arbeiten. [Hohrath 2013, S.15] bemerkt, dass sich die Definitionen des Begriffs Supply Chain in der Literatur, was ihre genauen Formulierungen angeht, durchaus unterscheiden. Er stellt aber zusammenfassend fest, dass die meisten Autoren eine Supply Chain als ein unternehmensübergreifendes Netzwerk beschreiben, dessen Elemente neben Herstellern auch ihre Lieferanten und Kunden umfassen. Einige der Definitionen beinhalten darüber hinaus auch logistische Elemente. Den Definitionen ist gemeinsam, dass die Elemente einer Supply Chain durch den Fluss bzw. den Austausch von Objekten (in der Regel Materialien bzw. Güter, Informationen und/oder Geld) miteinander verbunden werden. Dabei scheint in den gängigen Definitionen der Materialfluss strukturbestimmend für eine Supply Chain zu sein.

Eine Supply Chain besteht also aus Entitäten, welche Rohstoffe liefern bzw. beschaffen, diese (ggf. über mehrere Wertschöpfungsstufen) in Halbfertig- und Fertigprodukte transformieren und die Distribution an die Endkunden bzw. Konsumenten übernehmen (s. Abbildung 11 und Abbildung 12). Gemäß [Vahrenkamp 2008] gehören dazu

- Zulieferer (z. B. Rohstoff- oder Komponentenlieferanten),
- Produzenten bzw. Hersteller,
- Großhändler,
- Speditionen,
- Lagerhäuser und Warenverteilzentren,
- Logistik-Dienstleister und

- Filialen des Einzelhandels.

Darüber wird je nach Wertschöpfungskette auch der Endkunde bzw. Konsument als Entität betrachtet. Die Struktur der Supply Chain ist dabei von der interorganisatorischen Arbeitsteilung zwischen den teilnehmenden Unternehmen abhängig.

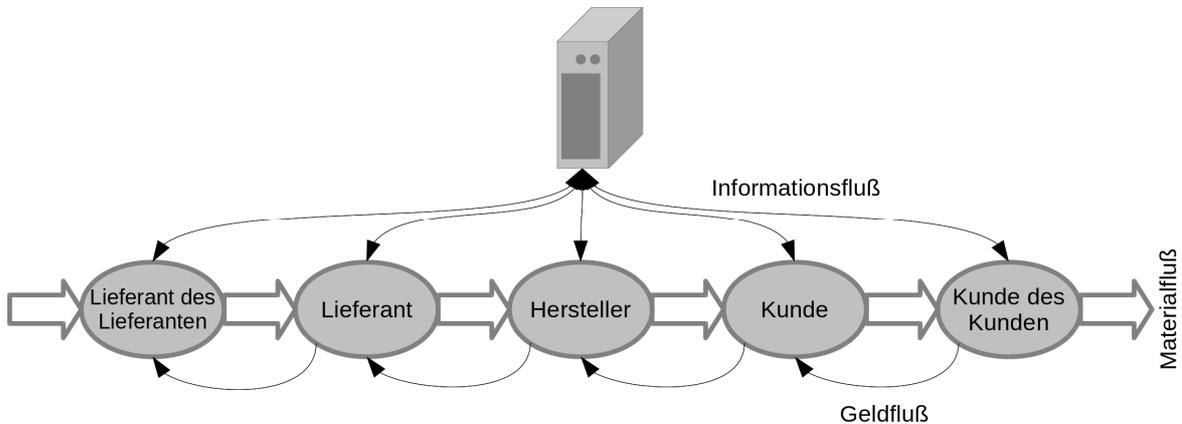


Abbildung 11: Schematische Darstellung der Verzahnung von Unternehmen in einer Supply Chain¹⁸

In einer Wertschöpfungskette erfolgt ein Güterfluss in Richtung der Konsumenten und ein entgegengesetzter Informations- und Finanzfluss von den Konsumenten weg in Richtung der Produzenten und Zulieferer. Durch Garantiefälle und Recycling kann jedoch auch ein Güterfluss vom Konsumenten zurück zum Produzenten erfolgen. Abbildung 12 zeigt eine schematische Darstellung einer Supply Chain (die im Vorgriff auf Abschnitt 3.2.2 eine netzwerkartige Struktur aufweist). Anzumerken ist, dass eine Supply Chain weniger durch die Lieferanten (englisch *Supplier*), sondern durch die Kundennachfrage (englisch *Demand*) gesteuert wird. Daher wäre die Bezeichnung „*Demand Chain*“ eigentlich angebrachter.

Nach [Beckmann 2003] ist eine Supply Chain durch folgende Eigenschaften gekennzeichnet:

- Der Gegenstand einer Supply Chain sind Entwicklungs-, Beschaffungs-, Produktions- und Distributionsprozesse.
- Ihr Kernziel ist die Schaffung von Kundennutzen, welcher in einem ausbalancierten Verhältnis zu Kosten und Gewinn steht. Die Einzelziele der beteiligten Partner werden durch die Leistungsfähigkeit der gesamten Kette erreicht.
- Sie überschreitet organisatorische Grenzen und umfasst alle Beteiligten und logistischen Prozesse, von der Beschaffung von Rohstoffen beim Zulieferer bis hin zu den Serviceleistungen beim Endkunden als ein ganzheitliches System.
- Ihre Koordination erfolgt über ein durchgängiges Informationssystem, das von allen Mitgliedern der Supply Chain genutzt wird.

Beispiele für Supply Chains finden sich in allen Industrien, zum Beispiel in der Automobilindustrie, der Textilindustrie oder der Elektronik- und Computerindustrie. In letzterer konnte beispielsweise Dell Computer durch eine konsequente Optimierung der Lieferkette signifikante Wettbewerbsvorteile erzielen (vgl. [Lawrenz et al. 2013a; Lawrenz et al. 2013b]).

¹⁸ Quelle: In Anlehnung an [Tsyganova 2000, S. 3, Darstellung 1]

[Vahrenkamp 2008] weist darauf hin, dass erst seit der Arbeit von Houlihan (siehe [Houlihan 1985]) Wertschöpfungsketten als Ganzes unter dem Begriff Supply Chain thematisiert werden. Vorher wurden in den klassischen Ansätzen der Betriebswirtschaftslehre die Entitäten bzw. Teilnehmer isoliert voneinander betrachtet. Dabei wurde angenommen, dass die Teilnehmer an einer Wertschöpfungskette autonome und eigennützige Entscheidungen auf der Basis von einzelwirtschaftlichen Entscheidungskalkülen treffen. Statt auf die Koordination des gesamten Netzwerks Rücksicht zu nehmen, konkurrieren sie miteinander.

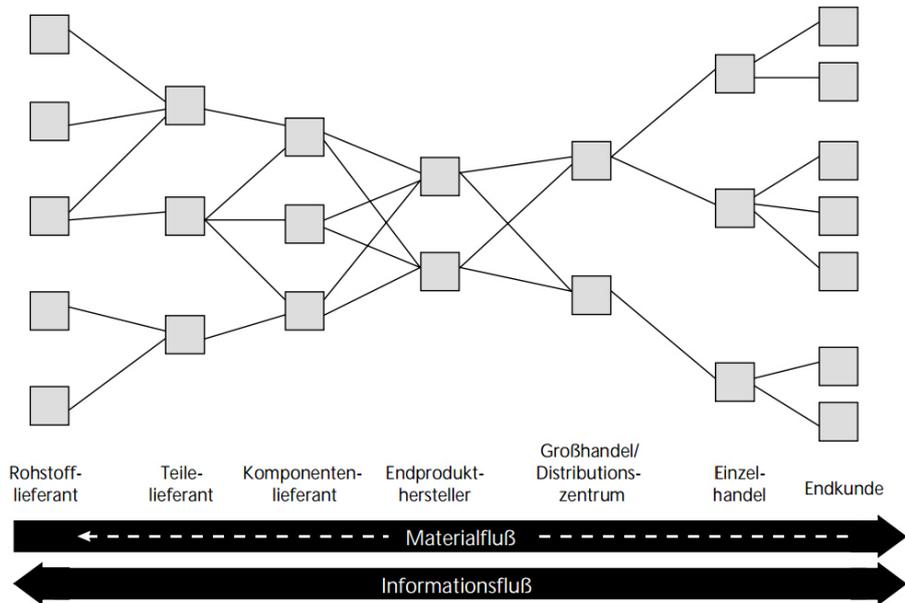


Abbildung 12: Strukturelle Sicht auf eine Supply Chain¹⁹

Aus einem solchen Handeln ergibt sich natürlich ein Koordinations- und Optimierungsproblem, dem im Rahmen dieser Arbeit mit einem agentenbasierten Koordinationsansatz begegnet wird. Durch den Einsatz von Agententechnologie können die Transaktionskosten innerhalb von Wertschöpfungsnetzwerken gesenkt und deren Flexibilität und Adaptivität gesteigert werden.

3.2.2 Wertschöpfungsnetzwerke

3.2.2.1 Von Supply Chains zu Wertschöpfungsnetzwerken

Der Begriff der Supply Chain bzw. Wertschöpfungskette impliziert eine eher lineare, starre Struktur zwischen den Entitäten auf den verschiedenen Wertschöpfungsstufen der Supply Chain dominieren (in der Regel bilaterale Beziehungen, siehe Abbildung 11). Weiterhin assoziiert man mit dem Begriff Supply Chain, zum Beispiel wenn man die Automobilindustrie betrachtet, eine Struktur, welche

- auf jeder Stufe nur einen oder wenige Kooperationspartner aufweist, die ihre Kernkompetenzen einbringen und kurzfristig kaum austauschbar sind, so dass die Supply Chain Struktur nur
- schwer veränderbar, kettenartig und eher auf Langfristigkeit ausgelegt ist.

¹⁹ Quellen: [Brodersen 2012] und Marktstudie SCM, Fraunhofer Anwendungszentrum für Logistikorientierte Betriebswirtschaft (ALB), 2000

[Hohrath 2013] stellt jedoch fest, dass eine lineare Struktur nicht mehr der gängigen Unternehmenspraxis entspricht. Viele weitere Autoren konstatieren, dass Supply Chains heutzutage eher netzwerkartige als kettenartige Strukturen aufweisen und die Vernetzung weiter zunimmt. Solche netzwerkartigen Wertschöpfungsstrukturen bezeichnen sie als *Wertschöpfungsnetzwerke*, *Supply Nets*, *Supply Networks* oder *Supply Webs* (vgl. z. B. [Laseter 1998; Grolik et al. 2001; Beckmann 2003]). Abbildung 13 beinhaltet eine einfache schematische Darstellung eines solchen Wertschöpfungsnetzwerks.

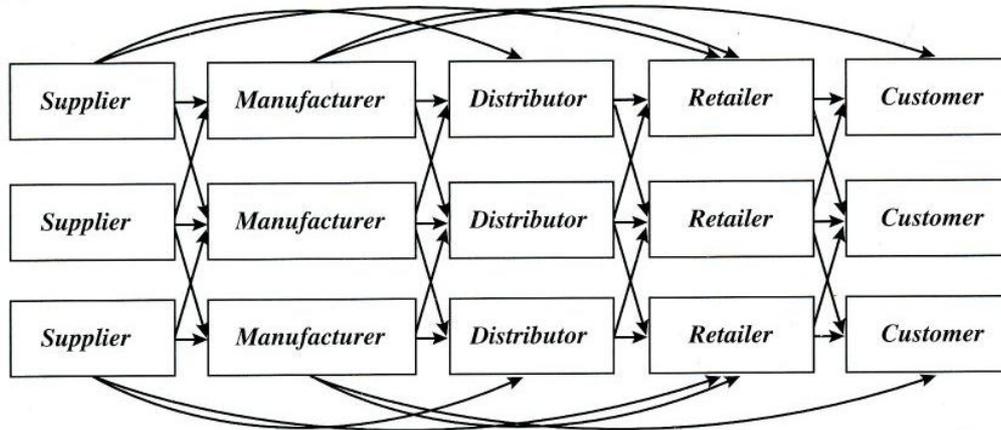


Abbildung 13: Beispiel für ein Wertschöpfungsnetzwerk²⁰

[Bullinger et al. 2009] fassen ihre aktuellen Erkenntnisse zu dieser Entwicklung folgendermaßen zusammen:

„Prozessketten, wie sie derzeit existieren, werden sich zu wandelnden Prozessnetzen entwickeln, d. h. von bilateralen Beziehungen zu zeitlich befristeten multilateralen Beziehungen. Von ihrer Ausprägung her sind diese sowohl „klein“ (im Sinne von flexibel) als auch „groß“ (im Sinne von mächtig). [...] es gibt einen Trend, den wir die „Dynamisierung der Beziehungen“ nennen. Kooperationsbeziehungen werden künftig deutlich weniger fest geknüpft sein als heutige Hersteller-Lieferanten-Beziehungen.“

[Beckmann 2003] stellt ebenfalls fest, dass solche Supply Nets keinesfalls starr und endgültig sind, sondern sich sowohl die beteiligten Wertschöpfungspartner als auch die Beziehungen innerhalb des Netzes ändern können.

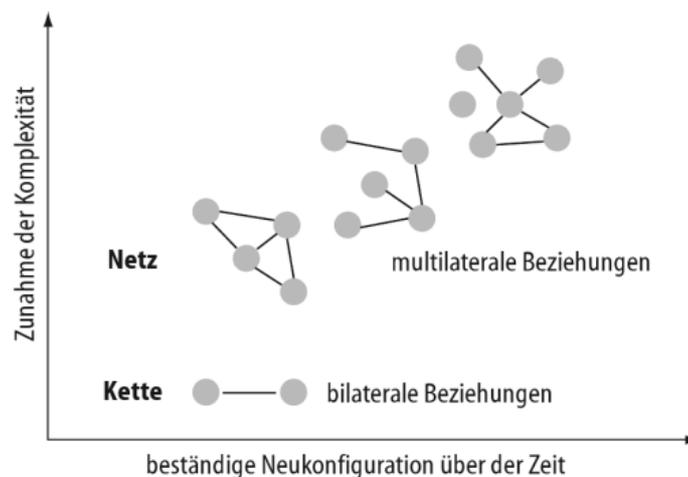


Abbildung 14: Von der Wertschöpfungskette zum -netz²¹

²⁰ Quelle: [Walz 2007, S. 4]

²¹ Quelle: [Bullinger et al. 2009]

Dies ist unter anderem den enormen Fortschritten im Bereich der Informationssysteme, insbesondere im Bereich des *E-Supply-Chain-Management (E-SCM)*, zu verdanken (z. B. bei eProcurement-Lösungen [Lawrenz et al. 2013a; Lawrenz et al. 2013b]), die das Management komplex vernetzter multilateraler Strukturen vereinfachen bzw. erst möglich machen. Was unter E-SCM im Detail zu verstehen ist, wird in Abschnitt 3.2.4 erläutert.

[Teuteberg 2005] betont ebenso die Entwicklung und zunehmende Ausdehnung von unternehmensübergreifenden Wertschöpfungsnetzwerken wie Lieferanten-Abnehmer-Netzwerken auf der Basis von Internet-Technologien und vor dem Hintergrund von zunehmendem Outsourcing. Er betont dabei, dass die Beteiligten eines solchen Netzwerks mit einer zunehmenden *Komplexität* und *Dynamik* konfrontiert werden, da *Supply Networks* aufgrund von „...immer kürzerer Produktlebenszyklen oder nur kurzfristig eingegangener Unternehmenskooperationen ... oftmals nur von temporärer Dauer ...“ sind, so dass sie häufiger rekonfiguriert werden müssen.

Darüber hinaus sind nach [Teuteberg 2005] Vertrauen und/oder zumindest Kooperationsbereitschaft in entsprechenden Netzwerken sehr wichtig, da die Akteure in einem Supply Network auch gleichzeitig an weiteren zueinander in Konkurrenz stehenden Supply Networks beteiligt sein können. Insgesamt kommt [Teuteberg 2005] zu dem Schluss, dass sich traditionelle Supply Chains zu *adaptiven Supply Networks* (vgl. dazu die Ausführungen von SAP in [SAP 2002] sowie [Ivanov & Sokolov 2009]) entwickeln werden, denn: „*Adaptive Supply Networks zeichnen sich [...] durch Flexibilität und Lernfähigkeit aus ...*“, während seiner Auffassung nach „... *traditionelle Supply Chains [...] durch eine zentrale Steuerung und Inflexibilität gekennzeichnet ...*“ sind.

Charakteristika	“Traditionelle” Supply Chains (Phase bis 2000)	Adaptive Supply Networks (Phase ab 2000)	Ubiquitäre Supply Networks (Phase ab 2006)
Informationsaustausch	Sequentiell + langsam	Parallel + dynamisch	Ubiquitär (“allgegenwärtig”)
Planungshorizont	Tage/Wochen	Stunden/Tage	Minuten/Stunden
Planung	Batch-Betrieb/ Zentraler Plan	Dynamisch	Dynamisch/Portfolio an Alternativplänen
Antwortzeiten	Tage/Stunden	Stunden	Minuten
Berechnungen	Sammlung und Berechnung historischer Daten	Interpretation in Realzeit	Interpretation gefilterter und kontextsensitiver Daten in Realzeit
Steuerung	Zentral	Verteilt	Verteilt
Event Management	Zentral/Manuell	Dynamisch/Verteilt	Dynamisch/Verteilt/ Fokus auf kritische Ausnahmen
Integration	Insellösungen	Unternehmensübergreifend	Unternehmensübergreifend
Standardisierung	Proprietär	Offen	Offen
Zugriff	Applikationen/ Web-Browser	Web-Browser/ Applikationen	Multimodal (Web, Mobil, Voice, externe Applikationen)
Rückkopplung und Lernen	Keine Rückkopplung/kein Lernen	Automatische Rückkopplung und Lernen	Automatische Rückkopplung und kontinuierliches Lernen
Auto-ID auf der Basis von RFID	Nicht eingesetzt	In Teilbereichen	Unternehmensübergreifend
Basis-Technologien	Planungssysteme in Teilbereichen	Agenten	Agenten, Web Services, RFID, VoiceXML, Semantic Web, P2P-Netzwerke, Mobile Computing

Abbildung 15: Charakteristika adaptiver und ubiquitärer Supply Networks im Vergleich mit traditionellen Supply Chains²²

²² Quelle: [Teuteberg 2005]

Außerdem führt [Teuteberg 2005] den Begriff der ubiquitären Supply Networks ein, welche über die Eigenschaften adaptiver Supply Networks hinausgehend Entscheidungsträgern „... zu jeder Zeit und an jedem Ort einen multimodalen Zugriff (Voice, Web, mobil) auf entscheidungsrelevante Daten ... in Realzeit ...“ ermöglichen. Eine vergleichende Übersicht der wesentlichen Charakteristika von traditionellen Supply Chains und adaptiven sowie ubiquitären Supply Networks findet sich in Abbildung 15.

Trotz der anhand der oben zitierten Erkenntnisse deutlich werdenden Unterschiede zwischen traditionellen Supply Chains und Wertschöpfungsnetzwerken ist zu beobachten, dass die Begriffe Supply Chain und Wertschöpfungskette in der Literatur häufig synonym zu den Begriffen Wertschöpfungsnetzwerk, Supply Network und Supply Web (siehe Abschnitt 3.2.4) verwendet und nicht klar voneinander abgegrenzt werden.

Es ist zu vermuten, dass dies daran liegt, dass die Grenze zwischen Wertschöpfungskette bzw. Supply Chain und Wertschöpfungsnetzwerk bzw. Supply Web (siehe unten sowie Abschnitt 3.2.4) je nach der Perspektive, die man einnimmt, relativ fließend ist und Supply Chains als eine spezielle Art der Netzwerkorganisation betrachtet werden können. [Ickerott 2007] schreibt dazu zum Beispiel:

„Aus der Perspektive der Organisationstheorie betrachtet, sind Supply Chains eine spezielle Form der Netzwerkorganisation. Sie stellen eine Zwischenform aus der reinen Marktorganisation und der Hierarchie dar. Im Ideal konzentrieren sich die Unternehmen jeweils auf ihre Kernkompetenzen [...] und passen sich durch Veränderung der Supply Chain Struktur flexibel an geänderte Marktanforderungen an.“

Christopher hat schon früh den Netzwerkcharakter von Supply Chains betont [Christopher 1992, S.12]. In seinen neueren Veröffentlichungen [Christopher 2011] bemerkt er, dass die Taktrate, in der sich Supply Chains den Kundenbedürfnissen anpassen müssen stark zugenommen hat, da sich immer mehr Verkäufermärkte zu Käufermärkten gewandelt haben (siehe auch Abschnitt 3.3.1) und zudem bisherige Massenprodukte in zunehmendem Maße kundenindividuell gefertigt werden. Daraus leitet er für die Supply Chains der Zukunft die Notwendigkeit ab, deutlich adaptiver und flexibler zu werden.

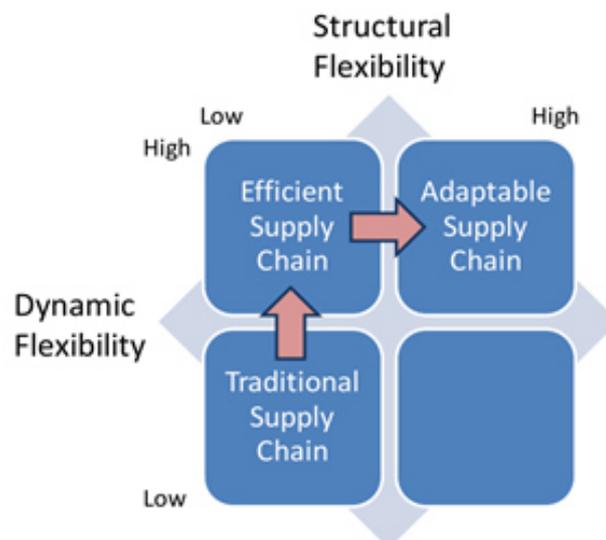


Abbildung 16: Die adaptive Supply Chain²³

²³ Quelle: Vortrag *Supply Chain 2.0: Managing Supply Chain Complexity in an Age of Uncertainty* von Professor Martin Christopher (Cranfield University) auf der Elemica Supply Chain Conference inSIGHT 2013 am 30. Mai 2013 in Philadelphia (USA).

Bzgl. der Flexibilität unterscheidet er zwischen einer dynamischen Flexibilität, die es der Supply Chain ermöglicht, mit Nachfrageschwankungen und einer zunehmenden Variantenvielfalt umzugehen und einer strukturellen Flexibilität, die es erlaubt, die Struktur des Gesamtsystems fundamental zu ändern, wenn sich eines der Partnerunternehmen stark verändert oder ihm ausscheidet.

Die Supply Chain der Zukunft ist für ihn die anpassungsfähige, sogenannte *Adaptable Supply Chain*, welche ein hohes Maß an dynamischer und struktureller Flexibilität aufweist (siehe Abbildung 16). Darüber hinaus wird sie seiner Auffassung nach zunehmend zu einem informationsbasierten virtuellen Netzwerk (siehe [Christopher 2011, S. 264]).

3.2.2.2 Begriffsabgrenzung für diese Arbeit

Vor dem oben dargelegten Hintergrund wird in dieser Arbeit unter einem Wertschöpfungsnetzwerk eine komplexe Form einer Supply Chain verstanden, welche eines oder mehrere der folgenden Charakteristika aufweist. Je mehr dieser Charakteristika auf eine Supply Chain zutreffen und je stärker diese ausgeprägt sind, umso eher wird diese Supply Chain im Rahmen dieser Arbeit nicht als Wertschöpfungskette, sondern als –netz bezeichnet. Die differenzierenden Charakteristika sind in Anlehnung an [Laseter 1998; Grolik et al. 2001; SAP 2002; Deßloch 2002; Teuteberg 2005; Ivanov 2009; Bullinger et al. 2009] die folgenden:

- *Netzwerkorganisation*: Laut [Gabler 2015a] stellt die Netzwerkorganisation „... die intermediäre Form der Organisation ökonomischer Aktivitäten zwischen Markt und Hierarchie dar.“

Im Falle einer intraorganisationalen Netzwerkorganisation gehören die Einheiten der Netzwerkorganisation alle zu dem gleichen, rechtlich selbstständigen Unternehmen. Im Gegensatz dazu schließen sich bei einer interorganisationalen Netzwerkorganisation mehrere selbstständige bzw. autonome Unternehmen zusammen, um ihre Fähigkeiten und/oder Kernkompetenzen zu bündeln bzw. zu ergänzen und dadurch gemeinsam Wettbewerbsvorteile zu erzielen.

In beiden Fällen werden zur Koordination der Einheiten der Netzwerkkoordination verstärkt Marktmechanismen verwendet. Insbesondere die interorganisationale Netzwerkorganisation stellt ein polyzentrisches System dar, da aufgrund der Autonomie ihrer Mitglieder komplexe Beziehungsgeflechte entstehen können, die mehr oder weniger kooperativ und stabil sein können.

- *Offenheit*: Die Menge der Teilnehmer an dem Wertschöpfungsnetzwerk kann variieren, da das Netzwerk prinzipiell für neue Teilnehmer offen ist, sofern sie mit ihren Gütern und Dienstleistungen beziehungsweise ihren Kernkompetenzen innerhalb des Wertschöpfungsnetzwerks Kooperationspartner finden. Teilnehmer können sowohl in das Netzwerk eintreten als es auch verlassen.
- *Adaptivität*: Um sich an die externen Anforderungen der Umwelt (z. B. dynamisch variierende Einflüsse des Wettbewerbs oder der Kundenbedürfnisse) adaptieren zu können, können die Teilnehmer innerhalb des Netzwerk zum einen bestehende Beziehungen zu ihren Netzwerkpartnern verändern oder abbrechen und zum anderen neue Beziehungen eingehen. Ein Wertschöpfungsnetzwerk ist darüber hinaus auch in der Lage, sich aufgrund interner dynamischer Veränderungen (z. B. einem Strategiewechsel einzelner oder mehrerer Teilnehmer) zu adaptieren.

Hervorzuheben ist, dass bei einer klassischen Supply Chain Design-Entscheidungen, d. h. Entscheidungen bzgl. der Auswahl der Teilnehmer sowie ihrer Beziehungen

zueinander, zu den langfristig wirksamen strategischen Entscheidungen zählen, wohingegen diese Entscheidungen in Wertschöpfungsnetzwerken einen mehr operativen Charakter haben.

- *Dynamisierung & Flexibilität:* Durch den globalen Wettbewerb, den Trend zur Fokussierung auf Kernkompetenzen sowie die zunehmende Verwendung von Informationssystemen mit standardisierten Datenaustausch-Formaten werden die Teilnehmer an Wertschöpfungsstrukturen zunehmend austauschbarer und ihre Beziehungen untereinander dynamischer.

Die (Kooperations-)Beziehungen innerhalb des Netzwerks werden daher weniger fest geknüpft bzw. sind leichter und deutlich kurzfristiger änderbar als in einer kettenartigen „traditionellen“ Supply Chain.

Partnerschaften zwischen autonomen Entitäten im Netzwerk können flexibler eingegangen oder zurückgezogen werden und sind oft auch kurzfristiger Natur. Dadurch sind Wertschöpfungspfade innerhalb der Netzwerk-Struktur nicht stabil, sondern ändern sich dynamisch. Die Beziehungen zwischen den Akteuren innerhalb von Wertschöpfungsnetzwerken wandeln sich in zunehmendem Maße von stabilen bilateralen hin zu sich dynamisch adaptierenden multilateralen Beziehungen (s. auch Abbildung 14).

Zwischen den Stufen des Wertschöpfungsnetzwerks, z. B. Herstellern und Lieferanten, sind nicht nur 1:1-, sondern auch 1:m- und n:m-Beziehungen erlaubt. Ggf. sind auch stufenübergreifende Beziehungen zwischen den Teilnehmern des Netzwerks erlaubt. Insgesamt weist ein Wertschöpfungsnetzwerk komplexe Beziehungsstrukturen auf.

- *Wettbewerb:* Die Teilnehmer auf einer Stufe stehen ggf. in Bezug auf den Absatz ihrer Güter und Dienstleistungen an nachgelagerte Stufen im Wettbewerb zueinander. Unter Umständen sind sich auch gleichzeitig Mitglieder in mehreren zueinander in Konkurrenz stehenden Wertschöpfungsnetzwerken.
- *Autonomie:* Die Teilnehmer sind nicht nur rechtlich selbständige Unternehmen, sondern sie verfügen bzgl. der Beziehungen zu anderen Netzwerk-Teilnehmern über eine hohe Entscheidungsautonomie, z. B. was die Bepreisung ihrer Güter und Dienstleistungen angeht oder die Frage, mit welchen Teilnehmern sie überhaupt kooperieren wollen.
- *Lernfähigkeit:* Idealerweise ist ein solches Netzwerk lernfähig, insbesondere im Hinblick auf seine Adaptionsprozesse.
- *Erhöhter Koordinationsbedarf:* Die vorgenannten Charakteristika verursachen aufgrund der entstehenden Dynamiken und Wechselwirkungen Koordinationsprobleme, denen mit geeigneten Informationssystemen begegnet werden muss.
- *Einsatz von Formen marktlicher Koordination* (s. auch Abschnitt 3.3) und eines Preismechanismus. [Ickerott 2007] bemerkt dazu, dass eine Situation des gegenseitigen Vorteils (sogenannte „Win-Win-Situation“) gegeben sein muss, damit zwei Unternehmen kooperieren. Besteht bzgl. des erzielbaren Vorteils zwischen den Partnern ein Ungleichgewicht, so kann dieses durch den Einsatz eines Preismechanismus kompensiert werden.

Zum letztgenannten Punkt ist die prominente *Move-to-the-Market-Hypothese* [Malone 1987b] zu erwähnen, die seit den 80er Jahren diskutiert wird. Diese besagt, dass der Einsatz

von Informations- und Kommunikationstechnologie durch seine – insbesondere für Markttransaktionen – transaktionskostenreduzierende Wirkung dazu führt, dass die wirtschaftliche Leistungserstellung in zunehmendem Maße vermarktet wird. Durch die IT-getriebene Senkung der Kosten für Markttransaktionen steigt die Effektivität und Effizienz von marktlicher Koordination. Dadurch wird es für Unternehmen interessanter, Aktivitäten auszulagern bzw. benötigte Dienste und Güter über einen Markt zu beschaffen. In der Folge führt dies zu einer zunehmenden Bedeutung von marktlicher Koordination und elektronischen Märkten (vgl. [Laudon et al. 2010]).

Dies lässt zusammengefasst vermuten, dass die Relevanz marktlicher Koordination von Wertschöpfungsnetzwerken und in Verbindung damit natürlich auch die Bedeutung elektronischer Märkte weiter zunehmen wird.

Da *Supply Webs* [Laseter 1998; Grolik et al. 2001; Deßloch 2002; Teuteberg 2005] mit Technologien und Methoden des E-Supply Chain Management (E-SCM) realisiert werden, wird im folgenden Abschnitt zunächst erklärt, was unter Supply Chain Management zu verstehen ist, bevor in Abschnitt 3.2.4 der Begriff des E-SCM genauer beleuchtet wird.

3.2.3 Supply Chain Management (SCM)

[Beckmann 2003] stellt fest, dass es im globalen Wettbewerb „... zur Erfüllung der veränderten Marktanforderungen zunehmend nicht mehr ausreicht, die unternehmensinternen Abläufe zu optimieren, sondern eine Verlagerung des Blickwinkels auf die unternehmensübergreifende Zusammenarbeit notwendig wird.“

Vor diesem Hintergrund versteht Beckmann unter *Supply Chain Management (SCM)* die „... Gestaltung, Lenkung und Entwicklung der Supply Chain über alle Wertschöpfungsstufen von der Rohstoffgewinnung bis hin zu Serviceleistungen beim Endverbraucher.“ [Beckmann 2003]

Gestaltung bedeutet dabei für ihn, die Supply Chain so zu organisieren und auszurichten, dass sie zu einer zweckgerichteten handlungsfähigen Einheit wird, die einerseits in der Lage ist, ihre aktuellen Aufgaben zu erfüllen, aber andererseits auch lenkungs- und entwicklungsfähig bleibt. Mit Lenkung meint er, die Abläufe innerhalb der Supply Chain so zu kontrollieren und zu steuern, dass die Wertschöpfungskette einen gewünschten Zustand erreicht und beibehält. Darüber hinaus versteht Beckmann unter Entwicklung, dass innerhalb der Supply Chain ein fortlaufender Lernprozess stattfinden soll, der auf ihre kontinuierliche Fortentwicklung und Verbesserung abzielt.

Nach [Beckmann 2003] verfolgt SCM das Ziel, Kooperationspartner auf Basis einer Win-Win Beziehung in ein gemeinsames Wertschöpfungs-system zu integrieren, um dann „... durch Abstimmung, Nutzung und Verbesserung der gemeinsamen Fähigkeiten die Wettbewerbsposition der gesamten Supply Chain zu steigern ...“. Ziel ist ein effizientes, Endkunden ausgerichtetes Handeln entlang der gesamten Wertschöpfungskette, welches erst auf Basis einer effektiven Integration und Koordination aller Partner (Zulieferer, Produzenten, Händler und Logistik-Dienstleistern) möglich wird.

Entgegen früherer Ansätze, die auf die Optimierung der unternehmensinternen Prozesse der Mitglieder bzw. einzelner Fragmente einer Supply Chain fokussierten, wird nach [Beckmann 2003] im SCM die gesamte Supply Chain als ein geschlossenes System betrachtet, welches aus einer ganzheitlichen Sicht heraus geeignet zu modellieren und zu optimieren ist. Neben intraorganisatorischen Prozessen betrachtet das SCM dabei insbesondere auch interorganisatorische Prozesse und versucht, alle Aktivitäten entlang der Wertschöpfungskette so nahtlos wie möglich zu integrieren und zu koordinieren.

Auch [Vahrenkamp 2008, S. 13] versteht unter dem Supply Chain Management eine Koordinationsaufgabe, die mittels einer ganzheitlichen Sicht auf die Logistikkette auf eine Abstimmung der Güterströme abzielt, die über das hinausgeht, was auf der Basis der lokalen Sichten der einzelnen Teilnehmer erreichbar ist.

[Vahrenkamp 1996] konkretisiert dies weiter und versteht unter SCM „... eine Optimierung des Gesamtsystems, das die folgenden Zielsetzungen verfolgt:

- *Orientierung am Nutzen des Endkunden.*
- *Steigerung der Kundenzufriedenheit durch bedarfsgerechte Anlieferung.*
- *Senkung der Bestände in der Logistikkette und eine damit verbundene Senkung der Kosten für das Vorhalten von Beständen.*
- *Verstetigung des Güterstroms und der damit möglichen Vereinfachung der Steuerung.*
- *Höhere Effizienz der Produktionssteuerung und der Kapazitätsplanung.*
- *Raschere Anpassung an Änderungen des Marktes.*
- *Verkürzung der Auftragsdurchlaufzeiten im Zeitwettbewerb.*
- *Vermeidung von „Out-Of-Stock“ Situationen.“*

Auch [Ickerott 2007] betont die Koordinationsfunktion des SCM und definiert den Begriff folgendermaßen:

„Supply Chain Management (SCM) verfolgt das Ziel, unternehmensinterne und unternehmensübergreifende logistische Wertschöpfungsketten zu optimieren. Das Hauptaugenmerk liegt hierbei auf der Koordination der Prozesse zwischen den einzelnen Akteuren einer Wertschöpfungskette und der Integration der Informationssysteme. In Netzwerken aus gleichberechtigten Kooperationspartnern wird in der Regel eine dezentrale, auf gegenseitigen Interessenausgleich zielende Koordination angestrebt. Für die Ausgestaltung derartiger Netze existieren bisher jedoch kaum geeignete Entscheidungs- bzw. Designhilfen. Ein Instrument, das die komplexe Aufgabe der Gestaltung von Supply Chain Prozessen unterstützen kann, ist die Simulation.“

Vor diesem Hintergrund kann MACSIMA als Designhilfe für die Gestaltung von Wertschöpfungsnetzwerken verstanden werden.

Was die oben oft genannte Koordination angeht, welche das SCM leisten soll, so geht es laut [Beckmann 2003] bei dieser „... um eine wechselseitige Abstimmung einzelner Prozesse in einem arbeitsteiligen System im Hinblick auf ein übergeordnetes Gesamtziel.“. Bei bestehenden SCM-Systemen kann dabei im Hinblick auf die Koordination zwischen Netzwerkpartnern zwischen einer heterarchischen und einer hierarchischen Koordination differenziert werden. [Beckmann 2003] schreibt hierzu:

- *„Koordination nach dem hierarchischen Prinzip ist dadurch gekennzeichnet, dass eine Abstimmung der Pläne der beteiligten Unternehmen so erfolgt, dass eine übergeordnete Planungsebene Rahmenpläne entwirft, die als Vorgabe für die untergeordneten Planungsebenen dienen.*
- *Koordination nach dem heterarchischem Prinzip erfolgt durch Abstimmung zwischen prinzipiell gleichberechtigten und relativ unabhängigen Entscheidungsträgern so, dass sie ihre Entscheidungen durch gegenseitige Übereinkunft fallen. Das Weisungsprinzip der Hierarchie wird durch das Verhandlungsprinzip der Heterarchie ersetzt. Diese Verfahrensweise wird durch die SCM-Systemfunktionalitäten im Bereich Collaborative Planning unterstützt.“*

[Beckmann 2003] führt weiter aus, dass heterarchische Koordinationsansätze in zunehmendem Maße in Supply Chains mit hohen dynamischen Anforderungen zum Einsatz kommen. In der Praxis dominiert jedoch noch die hierarchische Koordination.

[Vahrenkamp 2008] bemerkt, dass eine wesentliche Aufgabe des SCM darin besteht, die Dynamiken zu kontrollieren, die bei Produkten mit kurzen Lebenszyklen und starken Nachfrageschwankungen (wie z. B. bei elektronischen Komponenten und modischen Textilartikeln) entstehen können und beispielsweise zum *Peitschenschlageffekt* oder *Bullwhip-Effekt* [Herlyn 2014; Lee 1997] führen. [Vahrenkamp 2008] geht auch auf die Erweiterung des linearen Modells der Lieferkette zu einem Liefer- bzw. Wertschöpfungsnetzwerk ein (siehe Abbildung 17).

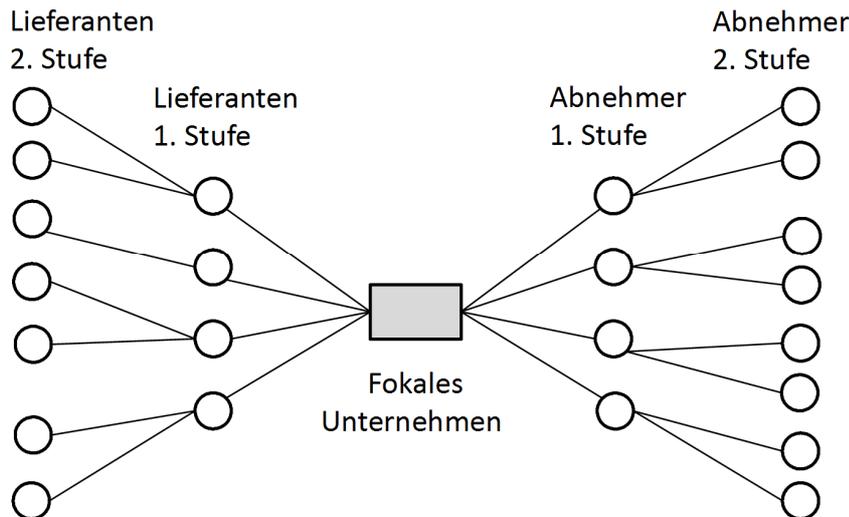


Abbildung 17: Liefernetzwerk²⁴

Das im Zentrum stehende Unternehmen stellt ein fokales Unternehmen dar. Ein solches nimmt (zum Beispiel durch seine Marktmacht) eine strategische Führungsrolle innerhalb des Netzwerks ein, durch die es die Netzwerkpartner selektieren sowie das Netzwerk kontrollieren und seine Aktivitäten hierarchisch koordinieren kann. Dies ist beispielsweise in der Regel in den Supply Chains der Automobilindustrie der Fall.

Die übrigen Elemente werden aufgefächert in die Lieferanten und Abnehmer der 1. und 2. Stufe (englisch *tier*). Sofern sich die Lieferanten logistisch und datentechnisch eng mit dem fokalen Unternehmen verbinden müssen, kann dies für sie hohe SCM-Investitionskosten führen, da sie sich an die von dem fokalen Unternehmen vorgegebenen Schnittstellen anpassen müssen.

3.2.4 E-SCM und Supply Webs

Nach [Bullinger et al. 2009] subsumiert der Begriff *E-Supply Chain Management (E-SCM)* „... *SCM-Technologie, die zur Integration von Kunden und Zulieferern Konzepte und Technologie des E-Business nutzt.*“. Sie schreiben weiter: „*Spezifiziert werden kann dieser Begriff als der Datenaustausch bzw. die Kopplung der IT-Systeme der Supply Chain Partner über das Internet, um Informationen schnell, asynchron, flexibel und angepasst auszutauschen und die Geschäftsprozesse der Wertschöpfungskette unternehmensübergreifend zu planen und zu steuern.*“

²⁴ Quelle: In Anlehnung an [Vahrenkamp 2008, S. 14]

[Bullinger et al. 2009] führen weiter aus, dass „... durch Vernetzung und Dezentralisierung [...] neue Formen des Beschaffungs- und Auftragsabwicklungsvorgangs kreiert ...“ werden „... und dadurch Transparenz (insbesondere Preistransparenz für die Käufer) [...] geschaffen ...“ wird.

Gemäß ihrer Ausführungen gehören auch die Begrifflichkeiten *E-Commerce*, *E-Procurement* und *E-Fulfillment* zum *E-SCM*. Unter *E-Commerce* verstehen [Bullinger et al. 2009] den Kauf- und Verkaufsprozesse, die über das Internet erfolgen und ausschließlich zwischen einem Unternehmen und dessen Endkunden, das heißt im Rahmen einer *B2C-Beziehung (Business to Consumer)*, stattfinden. Im Unterschied dazu geht es beim *E-Procurement* um Interaktionen, die zwischen einem Unternehmen und einem Lieferanten bei einem Beschaffungsprozess, das heißt im Rahmen einer *B2B-Beziehung (Business to Business)*, ablaufen.

Das *E-Fulfillment* schließt sich an eine bei einem *E-Commerce*- oder *E-Procurement*-Prozess getätigte Bestellung an und beinhaltet die durchgehende elektronische Abwicklung eines erteilten Auftrags, die bei der Bestellung im Internet beginnt, neben der Bezahlung auch den Transport sowie die Lieferung der Ware umfasst und zudem auch *After-Sales-* und *Entsorgungs-Dienstleistungen* beinhalten kann. Alle damit zusammenhängenden *intra-* und *interorganisatorischen* Aktivitäten werden dabei elektronisch miteinander verknüpft.

Das Zusammenspiel von *SCM* und *E-Business* ist in Abbildung 18 dargestellt. Beginnend mit einer Kundenanfrage über einen *Web-Store* wird über ein *ATP/CTP-Modul* die Lieferbarkeit einer Ware ermittelt. Kundeninformationen werden dabei (z. B. für *Rabatt-Entscheidungen*) über ein *Customer Relationship Management (CRM)* System eingeholt. Informationen über die Interaktionen des Kunden im *Web-Store* gehen dabei auch in die *Bedarfsplanung* und damit ggf. auch in *Abverkaufsprognosen* ein. Wird die Ware bestellt, so prüft die *Bestandsplanung*, ob diese auch tatsächlich verfügbar oder lieferbar ist. In Abhängigkeit davon wird entweder die Lieferung gestartet oder der Auftrag an die *Produktionsplanung* weitergegeben. Im letzteren Fall wird ein *E-Procurement*-Prozess gestartet, der sich um die Beschaffung der zur Produktion benötigten *Input-Güter* kümmert. Dieser Prozess beinhaltet entweder eine direkte bilaterale Interaktion mit einem Lieferanten oder er nutzt einen *Marktplatz*. Am Ende leistet das *E-Fulfillment* die Abwicklung des Auftrags.

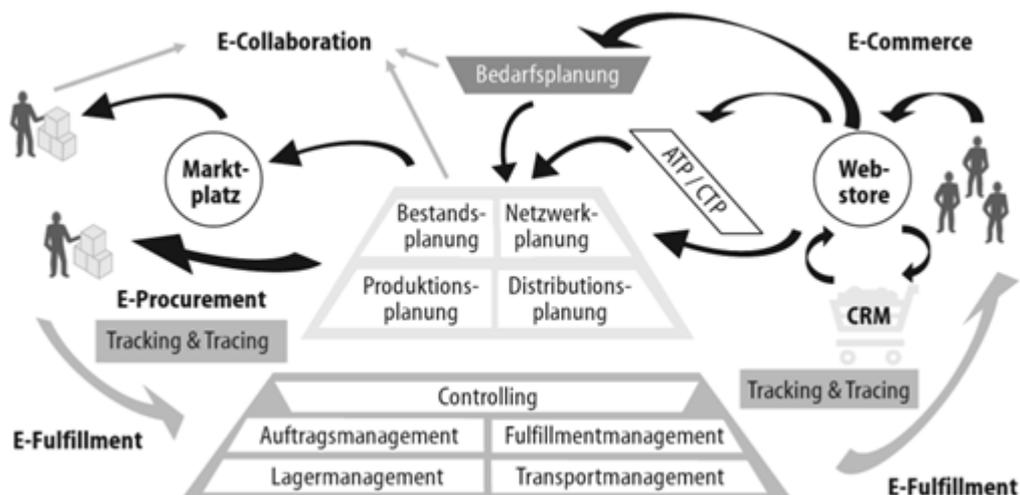


Abbildung 18: Zusammenwirken von E-Business und SCM im E-SCM²⁵

[Wannenwetsch & Nicolai 2013] gibt einen umfassenden Überblick über *E-SCM*-Anwendungen entlang der Wertschöpfungskette.

²⁵ Quelle: [Bullinger et al. 2009]

Supply Chains wandeln sich durch Nutzung der Möglichkeiten von *E-Business* und *E-SCM* zunehmend in sogenannten Supply Webs, die sich in vier Kernbereichen von traditionellen Supply Chains unterscheiden:

- *Partnerschaftliche Produktentwicklung* zur Reduktion der Produktkomplexität sowie zur Verbesserung der Effizienz und zur Reduktion von Kosten.
- *Anbindung an elektronische Marktplätze (E-Marketplaces)* für den Ein- und Verkauf von Rohstoffen, Materialien, Produkten und Dienstleistungen sowie zur Senkung von Transaktions-, Beschaffungs- und Distributionskosten.
- Nutzung eines *Collaborative Planning* Prozesses, bei dem die an einem Wertschöpfungsnetzwerk teilnehmenden Unternehmen Informationen über Aufträge, Bestände und Kapazitäten elektronisch austauschen, um ihre Einkaufs-, Logistik-, Produktions- und Distributions-Prozesse zu koordinieren und ihre Planungen aufeinander abzustimmen. Das Collaborative Planning kann dabei Bestandteil des noch umfassenderen *Collaborative Planning, Forecasting and Replenishment*²⁶ (CPFR) Konzepts (vgl. [Seifert 2006]) sein, bei dem es sich um eine Weiterentwicklung des *Efficient Consumer Response (ECR)* Konzeptes (siehe [Seifert 2006]) handelt. CPFR soll dabei die Effizienz der Geschäftsprozesse einer Wertschöpfungskette weiter steigern.
- Einsatz von *E-Fulfillment* (siehe oben) zur Verbesserung der Durchlaufzeiten zum Kunden.

Abbildung 19 verdeutlicht den zunehmenden Übergang von traditionellen Wertschöpfungsketten hin zu Wertschöpfungsnetzwerken bzw. Supply Webs. Die Wohlfahrt der Akteure in solchen Supply Webs hängt neben ihrer eigenen Leistungsfähigkeit auch sehr stark von der Leistung der anderen Akteure bzw. Entitäten ab, sowie von deren Kooperationsbereitschaft.

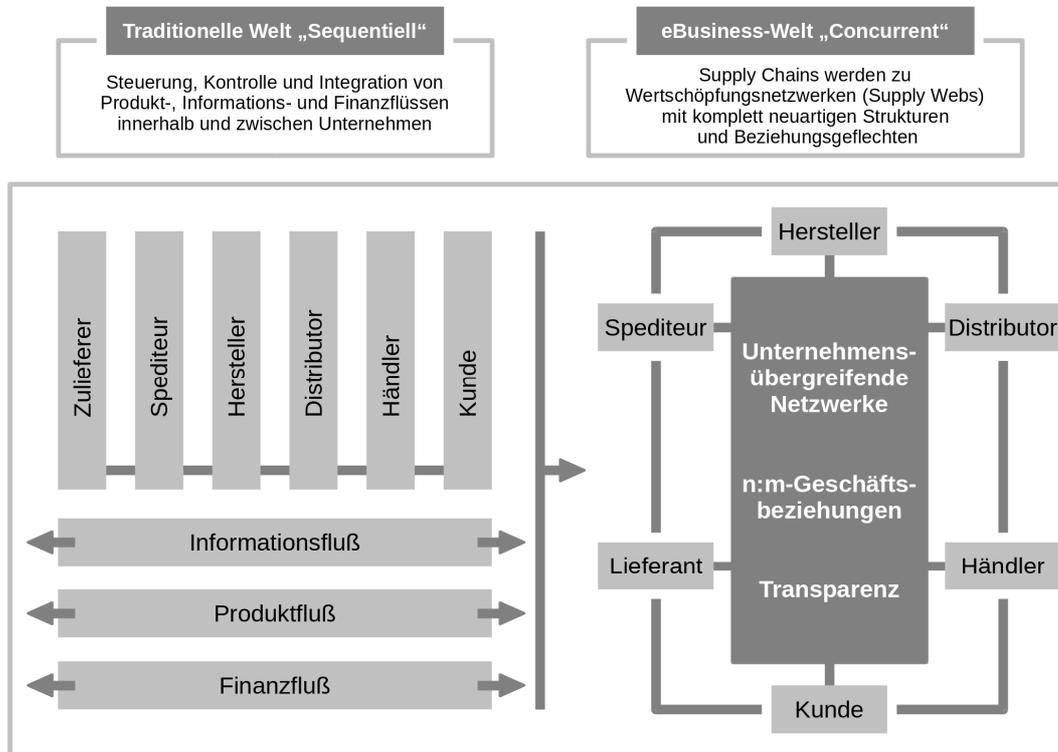


Abbildung 19: Wandel im Supply Chain Management Ansatz²⁷

²⁶ Wortwörtliche Übersetzung: gemeinsame Planung, Prognose und Bestandsführung

²⁷ Quellen: Angelehnt an [Deßloch 2002] sowie KPMG Consulting AG

3.2.5 Supply Chain Modelle

[Mentzer 2001] stellt drei Arten von Supply-Chain-Modellen vor, nämlich

- das *Modell der direkten Supply Chain*, das außer einem Fokalen Unternehmen jeweils nur einen einzigen Lieferanten und Kunden umfasst,
- das *Modell der erweiterten Supply Chain*, das den gesamten Wertschöpfungsprozess und Materialfluss abbildet sowie
- das *Modell der ultimativen Supply Chain*, die das Modell der erweiterten Supply Chain um die Betrachtung von Dienstleistern erweitert, die zwar nicht unmittelbar am Wertschöpfungsprozess teilnehmen, aber doch mittelbar daran beteiligt sind.

Abbildung 20 dient dem Vergleich des Betrachtungsrahmens dieser drei Modellarten. Aus Vereinfachungsgründen werden die Supply-Chain-Modelle in der Abbildung jeweils linear und nicht netzwerkartig dargestellt.

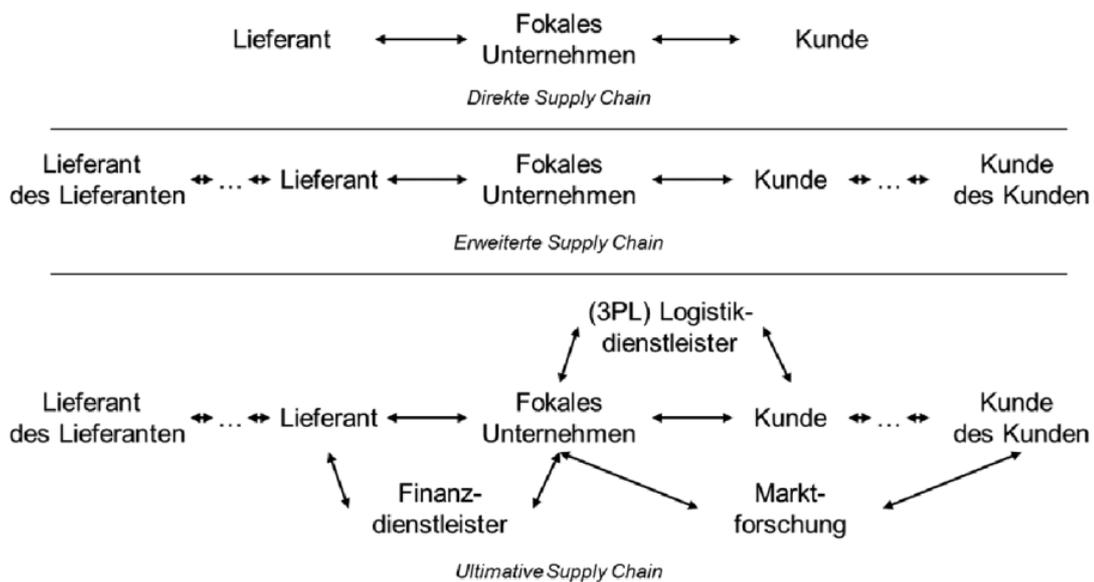


Abbildung 20: Supply Chain Modelle nach [Mentzer et al. 2001]²⁸

[Hohrath 2013] gibt an, dass die in der Literatur verfügbaren Supply Chain Modelle jeweils nur einen begrenzten Ausschnitt eines vollständigen Netzwerks betrachten. Dieser Betrachtungsrahmen orientiert sich in der Regel an der sogenannten *Sichtbarkeitsgrenze* (vgl. [Svensson 2004]), innerhalb derer die operativen Prozesse einer Supply Chain von einem fokalen Unternehmen noch gut gesteuert und/oder überwacht werden können. [Hohrath 2013] gibt an, dass empirische Untersuchungen belegen, dass in der Praxis die Sichtbarkeitsgrenzen sowohl materialstromaufwärts als auch –abwärts lediglich bis zu maximal zwei Stufen von dem fokalen Unternehmen entfernt sind.

Daher erscheint [Hohrath 2013] nach „ ... dem Vergleich einiger in der Literatur verfügbarer Supply-Chain-Modelle [...] eine Beschränkung des Betrachtungsrahmens auf eine Supply Chain mit fünf Stufen für eine empirische Analyse am ehesten geeignet zu sein.“

Supply-Chain-Modelle mit diesem Betrachtungsrahmen umfassen in der Regel eine Sicht vom Lieferanten des Lieferanten bis zum Kunden des Kunden eines fokalen Unternehmens.

²⁸ Quelle: [Hohrath 2013, S. 19]

Beispielsweise nutzen das SCOR-Modell (vgl. [Supply Chain Council 2005]), das Rahmenkonzept zum SCM von [Lambert et al. 1998] und das Supply-Chain-Modell von [Lee 2000] einen entsprechenden Betrachtungsrahmen.

Neben ihrem Rahmenkonzept haben [Lambert et al. 1998] noch ein weiteres, weniger stark vereinfachendes Supply-Chain-Modell entwickelt, das in Abbildung 21 dargestellt ist. Dieses besitzt auf der Versorgungsseite einen von mehreren Lieferanten ausgehenden und auf der Nachfrageseite zu verschiedenen Kunden führenden Materialfluss, der in der Darstellung sogar zum fokalen Unternehmen hin konvergiert und ausgehend von diesem in Richtung der Nachfrager divergiert. Solch eine Struktur ist aber in der Regel nur gegeben, wenn es in einem Wertschöpfungsnetzwerk auch tatsächlich ein fokales Unternehmen gibt. Ansonsten sind auf der mittleren Stufe auch mehrere Unternehmen möglich.

Obwohl das in Abbildung 21 dargestellte Modell die Beziehungen beziehungsweise Verbindungen innerhalb des Netzwerks über mehr als fünf Stufen abbildet, so fällt auf, dass die Verbindungen, die vom fokalen Unternehmen aktiv verwaltet oder überwacht werden können, wiederum nicht über die Sichtbarkeitsgrenze von zwei Stufen hinausgehen.

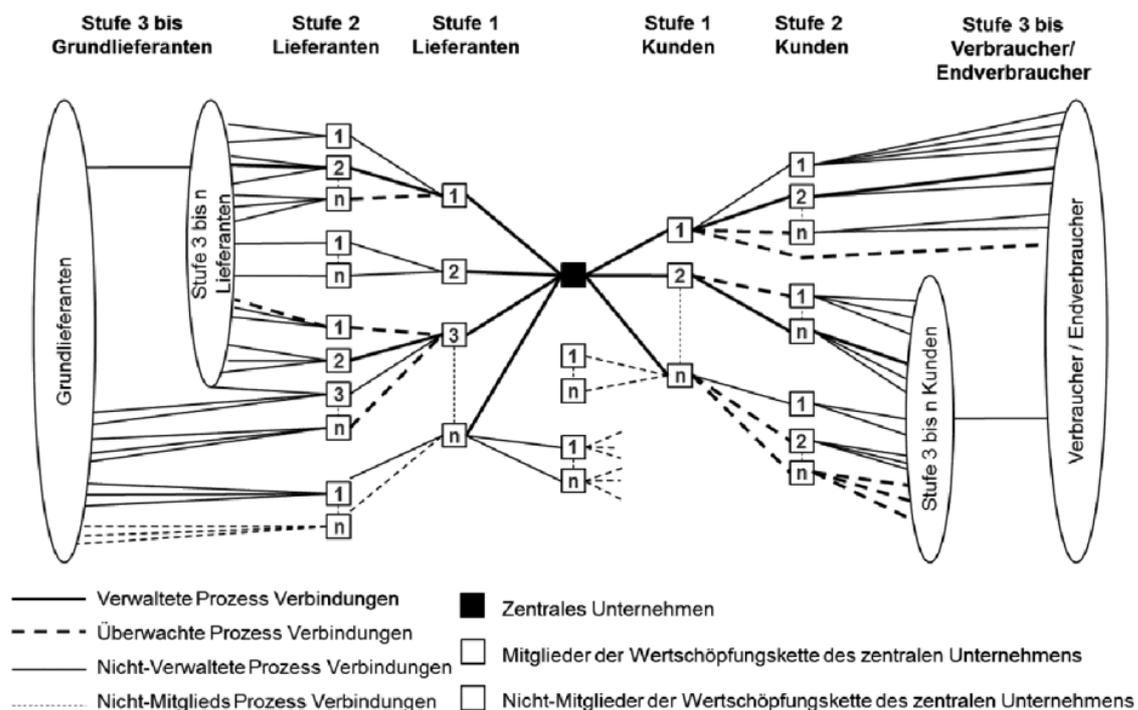


Abbildung 21: Struktur eines Wertschöpfungsnetzwerks nach [Lambert et al. 1998]

Vor diesem Hintergrund wurde das Basis-Supply-Chain-Szenario von MACSIMA so entworfen, dass es fünf Stufen umfasst (siehe Abschnitt 5.2.1). Da es kein fokales Unternehmen enthält, sind die Agenten über die fünf Stufen gleichmäßig verteilt.

Ein weiteres Supply-Chain-Modell ist das von [Bowersox et al. 2002], welches die Entitäten einer Supply Chain in ein fokales Produktionsunternehmen sowie in die Mitglieder von Teilnetzwerken, nämlich eines Versorgungs- sowie eines Distributionsnetzwerks segmentiert (siehe Abbildung 22).

Eine Zusammenfassung der Eigenschaften der angesprochenen Supply-Chain-Modelle ist Tabelle 1 zu entnehmen.

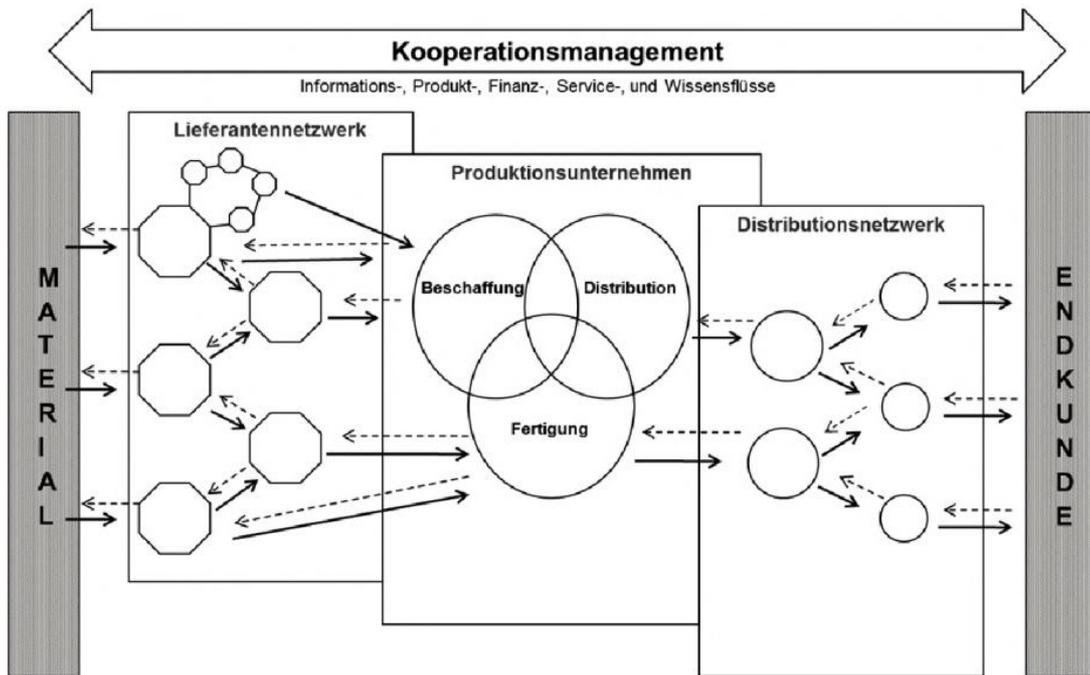


Abbildung 22: Allgemeines Supply-Chain-Modell nach [Bowersox et al. 2002]²⁹

Bezeichnung	Betrachtungsumfang	Struktur	Sichtbarkeitsgrenze	Umfang	Beziehungen
Direkte Supply Chain [Mentzer et al. 2001]	3 Stufen	Linear	Ja	Elemente des Materialflusses	Einfach
Erweiterte Supply Chain [Mentzer et al. 2001]	Rohstoffe bis Fertigwaren	Linear	Nein	Elemente des Materialflusses	Einfach
Ultimative Supply Chain [Mentzer et al. 2001]	Rohstoffe bis Fertigwaren	Linear	Nein	Elemente des Materialflusses, Dienstleister	Einfach
SCOR Modell [Supply Chain Council 2005]	5 Stufen	Linear	Ja	Elemente des Materialflusses	Einfach
Rahmenkonzept [Lambert et al. 1998]	5 Stufen	Linear	Ja	Elemente des Materialflusses	Einfach
Supply Chain Modell [Lee 2000]	5 Stufen	Linear	Ja	Elemente des Materialflusses	Einfach
Allgemeines Supply Chain Modell [Bowersox et al. 2002]	Rohstoffe bis Fertigwaren	Netzwerk	Nein	Elemente des Materialflusses	Einfach
Supply Chain Modell [Lambert et al. 1998]	5 Stufen	Netzwerk	Ja	Elemente des Materialflusses	Differenziert

Tabelle 1: Überblick über verschiedenen Supply-Chain-Modelle³⁰

²⁹ Quelle: [Hohrath 2013, S. 28]

³⁰ Quelle: [Hohrath 2013]

3.3 Grundlagen marktlicher Koordination

In dieser Arbeit wurden bereits in den Abschnitten 2.5.3 und 2.7 Grundlagen marktlicher Koordination behandelt. In Abschnitt 2.5.3 wurde der Begriff *Koordination* erläutert und darauf hingewiesen, dass Koordinationsmechanismen, auch marktlicher Art, eingesetzt werden, um Koordinationsprobleme zu lösen.

Hilfestellung bei der Auswahl und dem Entwurf von geeigneten Koordinationsmechanismen leistet dabei die *Mechanismusdesign-Theorie* (englisch *Coordination Mechanism Design*) [Hehenkamp 2007; Ruß 1997], welche ein Teilgebiet der *Spiel- und Entscheidungstheorie* darstellt.

Weiterhin wurde in 2.5.3 ein Überblick über die von drei von Moulin definierten Koordinations-Modi gegeben, nämlich den *Gerechtigkeits-Modus*, den *Modus direkter Verhandlungen* sowie den *Modus des dezentralen Verhaltens*.

Im Fokus dieser Arbeit steht dabei die Koordination der Agenten mittels des Modus direkter Verhandlungen. Zwischen den MACSIMA-Agenten finden bilaterale Preisverhandlungen statt. Daher werden in diesem Abschnitt die theoretischen Grundlagen bilateraler Verhandlungen erläutert. Das zur Koordination in MACSIMA eingesetzte Verhandlungsprotokoll wird später in Kapitel 5 beschrieben.

[Eymann 2000] und [Sackmann 2003] geben ergänzend zu den Darstellungen in dieser Arbeit einen umfangreichen Überblick über den Markt als Koordinationsverfahren und insbesondere die marktliche Koordination von Software-Agenten.

3.3.1 Das Konzept Markt

Unter einem Markt versteht man einen realen oder virtuellen Bereich, an dem sich Anbieter und Nachfrager von Gütern und Produktionsfaktoren treffen. Übersteigt die Nachfrage das Angebot, so spricht man von einem *Verkäufermarkt*. Im umgekehrten Fall handelt es sich um einen *Käufermarkt*. Von einem *Marktgleichgewicht* spricht man, wenn sich Angebot und Nachfrage die Waage halten. Märkte üben eine ganze Reihe von Funktionen aus. Von besonderer Wichtigkeit sind ihre *Preisbildungs-* sowie ihre *Koordinationsfunktion*.

Marktpreise ergeben sich aus der Relation von Angebot und Nachfrage auf dem Markt und haben eine wichtige Signalfunktion, da ein steigender Preis einen Indikator für die Knappheit eines Gutes darstellt. Dies kann die Hersteller eines knappen Gutes zur Ausweitung ihrer Produktionskapazitäten anregen, sofern sie durch einen Vergleich der Einkaufs- und Verkaufspreise auf den Märkten für ihre Input- und Output-Güter feststellen, dass sie damit Gewinne erwirtschaften können.

Die Volkswirtschaftslehre unterscheidet zwischen *Faktormärkten* wie dem Kapital-, dem Arbeits- und den Immobilienmarkt sowie *Gütermärkten* (für Investitions- bzw. Produktions- und Konsumgüter). Die in dieser Arbeit behandelten Szenarien fokussieren sich dabei auf Gütermärkte (für die bei den MAS-Simulationen gehandelten Güter). Im Vergleich mit Produktionsfaktoren wie z. B. der Arbeitskraft eines Menschen sind Güter homogener und ggf. sogar genormt, so dass sich ihre Eigenschaften und Preise leichter vergleichen lassen. Dadurch sind sie als Gegenstand automatisierter Verhandlungen auf elektronischen Märkten insgesamt besser geeignet.

Unternehmen als auch Konsumenten sind in der Regel gleichzeitig auf mehreren Märkten aktiv. So fragen Produktions- oder Konsumgüterunternehmen die bei ihren Herstellungsprozessen benötigten Produktionsgüter und -faktoren (u. a. Arbeit und Kapital) auf den entsprechenden Märkten nach. Die von ihnen damit produzierten Güter bieten sie auf dem Markt

für Produktionsgüter sowie dem Konsumgütermarkt an. Die Unternehmen treffen sich dort mit den menschlichen Endkonsumenten, die auf dem Konsumgütermarkt als Nachfrager und auf dem Arbeitsmarkt als Anbieter ihrer Arbeitskraft auftreten.

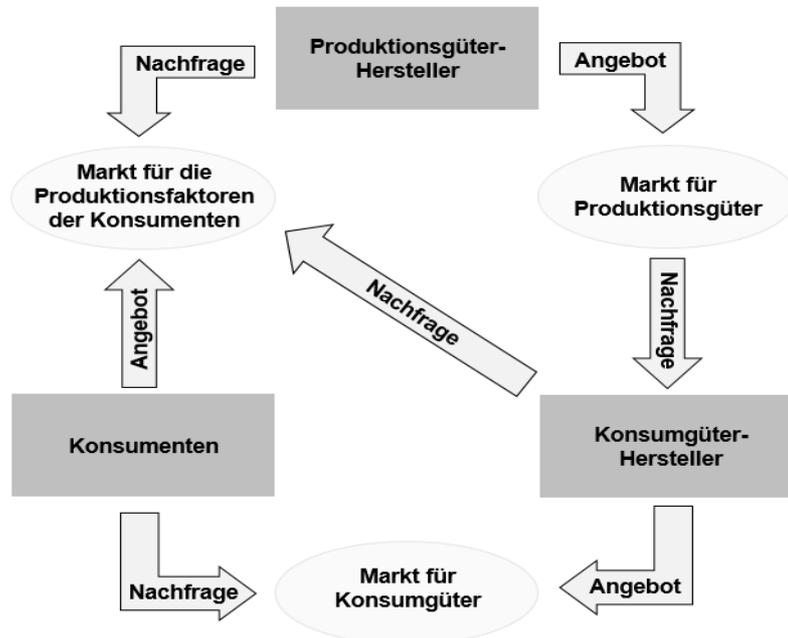


Abbildung 23: Übergreifende Koordinationsfunktion der Märkte

Dieser Mechanismus der Preisvergleiche über Märkte hinweg erlaubt den Wirtschaftssubjekten bei ihren Planungen die Gegenüberstellung von damit verbundenen erforderlichen Ausgaben und erzielbaren Erlösen, was innerhalb der Gesamtwirtschaft zu einer Abstimmung der individuellen Pläne von Nachfragern und Anbietern führt (*Koordinationsfunktion der Märkte*, siehe Abbildung 23).

Damit ein Markt effizient funktionieren kann, müssen die Interaktionen zwischen den in der Regel autonom und eigennützig handelnden Marktteilnehmern mittels marktspezifischer Interaktionsregeln und -protokolle sowie gesetzliche Rahmenbedingungen auf geeignete Weise strukturiert und gesteuert werden.

Grundlegende wirtschaftswissenschaftliche Arbeiten zu Märkten stammen von den Ökonomen Adam Smith und Léon Walras. Smith beschäftigte sich in seinem Werk, „Wohlstand der Nationen“ [Smith 1776] u. a. mit den Auswirkungen von Einfuhrbeschränkungen für ausländische Güter. Auf ihn geht die populäre Metapher der „unsichtbaren Hand des Marktes“ zurück, wobei er diese in seinem Gesamtwerk lediglich dreimal verwendet. Er erkannte, dass Wettbewerb mit vollkommener Konkurrenz, ohne Einmischung durch den Staat eine effiziente Produktion und Verteilung der Güter sicherstellen kann, wenn keiner der Marktteilnehmer über mehr Marktmacht verfügt als die übrigen Teilnehmer. In den allermeisten Fällen kann jedoch nicht von vollkommener Konkurrenz ausgegangen werden, so dass es für die Marktteilnehmer vorteilhaft sein kann, Koalitionen zu bilden.

Léon Walras ist v. a. durch sein *allgemeines Gleichgewichts-Modell* [Walras 1874] bekannt. In diesem führt er den sogenannten Walras-Auktionator ein, der u. a. dazu in der Lage ist, in einem zentralisierten Markt durch Variation der Güterpreise ein sogenanntes *Marktgleichgewicht* zu erreichen, d. h. Gleichgewichtspreise, zu denen das Angebot für jedes gehandelte

Gut gleich der Nachfrage nach ihm ist. Beim dabei vom Auktionator verwendeten Marktmechanismus handelt es sich um ein Auktionsverfahren, das auch als *Tâtonnement-Prozess* bezeichnet.

3.3.2 Die Bildung von Katallaxien

Neben der Möglichkeit zur Organisation und Koordination eines Marktes durch eine zentrale Instanz besteht die Möglichkeit zur Selbstorganisation des Marktes, auch Katallaktik oder Katallaxie (von altgriechisch „katallein“, übersetzt „austauschen“) genannt.

Der Begriff der Katallaxie geht auf den Erzbischof Richard Whately zurück, der ihn 1831 einführte. Er wurde von Hayek [Hayek 1996, S. 122] wieder aufgegriffen, um den Effekt der Genese einer spontanen Ordnung auf einem Markt zu benennen, ohne dass diese von den Marktteilnehmern oder einer zentralen Instanz auf dem Markt bewusst geplant wurde. Nach dem Gabler Wirtschaftslexikon werden in spontaner Ordnung „... *Informationen genutzt, die nur einzelnen Individuen bekannt sind und die sich nicht sinnvoll aggregieren lassen.*“ [Gabler 2015b].

Die Begriffe Katallaxie und Katallaktik werden in der Literatur nur unscharf voneinander abgegrenzt. Während der Begriff Katallaktik mehr die theoretische Lehre von den Auswirkungen direkter und freier Interaktionen zwischen autonom und eigennützig handelnden Marktteilnehmern auf die Entwicklung des Marktgeschehens bezeichnet, wird der Begriff Katallaxie eher für die Bildung spontaner Marktordnungen verwendet. Wie Eymann schreibt, „... *gibt es über die Funktionsweise der Katallaxie nur grundlegende, aber nicht detailliert ausgearbeitete Ansichten.*“ [Eymann 2003].

Auf Märkten, in denen Verhandlungen, Preisbildungsprozesse und sonstige Interaktionen der Akteure indirekt über eine zentrale koordinierende Instanz wie einen Arbitrator oder Auktionator ablaufen, der evtl. sogar noch ergebnisorientiert interveniert, kann keine Selbstorganisation auftreten, die im Sinne von Hayek als spontane Ordnung bzw. Katallaxie bezeichnet werden könnte.

Eine solche „geplante“ Organisation des Marktgeschehens ist laut [Gabler 2015] „...*der spontanen Ordnung in Bezug auf die Koordination großer und komplexer Gesellschaften unterlegen, weil sie viel weniger Wissen verarbeiten könne (nämlich nur das an der Hierarchiespitze vorhandene) und deswegen notwendigerweise nur einen geringen Komplexitätsgrad hervorzubringen in der Lage sei.*“ Daher empfehlen die Vertreter der Katallaktik der Wirtschaftspolitik, auf interventionistische Eingriffe weitgehend zu verzichten und lediglich adäquate ordnungspolitische Rahmenregeln zu setzen.

Dennoch sind geeignete Verhaltensregeln, denen die Elemente einer spontanen Ordnung zu folgen haben, eine wesentliche Voraussetzung für deren Entstehung. Daher sieht nach [Gabler 2015] Hayek eine zentrale Aufgabe für Sozialwissenschaftler darin, „...*die Eigenschaften von Regeln herauszuarbeiten, die eine spontane Ordnung ermöglichen.*“ Denn nicht jede Regelmäßigkeit im Verhalten führt zur Bildung einer spontanen Ordnung bzw. Katallaxie.

Insbesondere der Staat soll einen minimalen Regelrahmen vorgeben, der für die Bildung und den Erhalt einer spontanen Ordnung zwingend notwendig ist. Der Regelrahmen soll kein bestimmtes positives Verhalten vorschreiben, sondern nur ungewünschte Verhaltensweisen sanktionieren, damit ein großer Spielraum an individueller Freiheit für die Elemente der spontanen Ordnung erhalten bleibt, der ihre Bildung und Erhaltung begünstigt.

[Hoppmann 1999, S. 135- 169] nennt für die Bildung von Katallaxien Bedingungen, welche die Fähigkeiten von Agenten betreffen. Einige wesentliche davon bestehen darin,

1. an Märkten teilnehmen zu können, um Informationen zu sammeln und Handlungen (wie z. B. Transaktionen) durchzuführen,
2. ein Einkommen zu erzielen und eine Verknüpfung zwischen Einkommen und Nutzen herzustellen,
3. sich zwischen mehreren Handlungsalternativen entscheiden zu können, indem diese anhand von gesammelten Informationen bewertet werden und
4. aus Erfahrungen lernen zu können.

Maßgebliche Motivation der Agenten für ihre Teilnahme an einem Markt sind ihre Einkommenserwartungen. Hiernach richten sie ihre nutzenorientierten individuellen Ziele aus, wobei ihnen kein Wissen darüber vorliegt, inwieweit diese Ziele und ihr daraus abgeleitetes Handeln zu einem übergeordneten Ziel oder Zweck des Marktes beiträgt.

Anhand der verfügbaren Informationen über das aktuelle Marktpreissystem und die Relationen der Preise der gehandelten Güter ist es den Agenten dennoch möglich, ihr Handeln zu jeder Zeit auf Aktivitäten zu fokussieren, die eine besonders hohe Gewinnmarge versprechen. Dadurch handeln sie einerseits eigennützig, helfen aber andererseits auch dabei, die Wohlfahrt des Gesamtsystems zu steigern, da sie danach streben, mit einem möglichst geringen Mitteleinsatz ebenjene Bedürfnisse anderer Agenten zu befriedigen, für die eine hohe Zahlungsbereitschaft vorliegt. Dies werden Bedürfnisse sein, die entweder sehr dringend sind oder wiederum die nachfragenden Agenten dazu befähigen, in einer anderen marktlichen Interaktion eine hohe Gewinnmarge zu erzielen. Insgesamt führt dies zu einer Selbstkoordination des Gesamtsystems, bei der die zur Verfügung stehenden Ressourcen möglichst nutzenstiftend eingesetzt werden.

Die wesentlichen Dinge, die eine solche Selbstkoordination erst möglich machen, sind dabei das Preissystem beziehungsweise die Preisrelationen der gehandelten Güter sowie die an seine Einkommenserwartung gekoppelte Motivation jedes einzelnen Marktteilnehmers. Dabei repräsentiert das Preissystem des Marktes eine Aggregation des über den ganzen Markt verteilten Wissens über die momentanen Bedürfnisse der Marktteilnehmer. Da sich deren Bedürfnisse dynamisch ändern und auch das Wissen über die Bedürfnisse über das Gesamtsystem verteilt ist, ändert sich auch die emergierende spontane Marktordnung laufend, so dass für das Gesamtsystem eine exakte Bestimmung eines anhaltenden optimalen Zustands nicht möglich.

Nach Hayek ist das Wissen im Markt „seiner Natur nach verstreut und lässt sich keinesfalls sammeln und einer Behörde übermitteln, der die Aufgabe der vorsätzlichen Schaffung von Ordnung übertragen wäre.“ [Hayek 1996, S.88]. Dies sieht er als wesentlichen Grund für den Untergang der sozialistischen Planwirtschaft an [Hayek 1988].

Bei dem in dieser Arbeit vorgestellten und in Kapitel 8 evaluierten MACSIMA-System handelt es sich um ein marktlich koordiniertes Multiagentensystem ohne zentrale Koordinations-Instanz, bei dem die teilnehmenden autonomen und eigennützigen Agenten frei und direkt miteinander interagieren bzw. verhandeln.

Ausgehend von einer suboptimalen Situation ohne ein festgesetztes Preissystem zum Systemstart verhandeln die Agenten und tätigen Transaktionen, die ihren Nutzen steigern. Dabei sind sie zunächst nur mit sehr unvollständigem individuellem Wissen über die Marktgeschehnisse ausgestattet. Im Zeitverlauf können sie jedoch weiteres Wissen aggregieren, indem sie aus ihren Erfolgen und Misserfolgen bei den geführten Verhandlungen lernen und ihre Strategien anpassen. Darüber hinaus besitzen sie die Fähigkeit, Informationen über ihre Erfolge und Verkaufspreise mit anderen koalierenden Agenten auszutauschen und sich an die aktuellen Marktbedingungen zu adaptieren.

Wie im Simulations- und Evaluationsteil der Arbeit aufgezeigt wird, entwickeln sich auf der Basis des nichtdeterministischen Verhaltens der Agenten spontane Marktordnungen, die sich an Mustern wie beispielsweise zwar Börsenkurs-ähnlichen, aber doch durch Streuungen gekennzeichneten Marktpreis-Verläufen für die gehandelten Güter sichtbar werden. Die einzuhaltenden Verhaltensregeln gibt das eingesetzte Verhandlungsprotokoll ein (s. Kapitel 5).

Anhand der in Kapitel 7 vorgestellten Evaluationsmethodik wird die Selbstorganisation bzw. die Katallaxie innerhalb des MACSIMA-Systems untersucht, wobei ein besonderer Fokus auf der Frage liegt, wie sich verschiedene Parametrisierungen der Lern-Komponente der Agenten auf die zu beobachtenden Katallaxien auswirken.

3.3.3 Elektronische Märkte

Elektronische Märkte dienen, wie in Abschnitt 2.7 kurz angesprochen, der Zusammenführung von Angebot und Nachfrage mit Hilfe der Informations- und Kommunikationstechnologie. Sie stellen häufig Teilmärkte bestehender nicht-elektronischer Produkt- und Dienstleistungsmärkte dar, aber grenzen sich von diesen dadurch ab, dass zumindest einige Markttransaktions-Phasen mittels elektronischer Unterstützung effizienter abgewickelt werden können.

Im Folgenden werden die Erläuterungen zu Märkten und elektronischen Märkten aus anderen Teilen dieser Arbeit bzgl. weiterer Gesichtspunkte ergänzt.

3.3.3.1 Arten elektronischer Märkte

Es kann zwischen elektronischen Märkten im engeren und weiteren Sinne sowie zwischen offenen und geschlossen elektronischen Märkten unterschieden werden.

[Schmid 1993] differenziert zwischen elektronischen Märkten im engeren und weiteren Sinne:

- Unter *elektronischen Märkten im engeren Sinne* versteht er Mechanismen des marktmäßigen Tausches, welche alle Phasen der Transaktion Telematik-gestützt abwickeln.
- *Elektronische Märkte im weiteren Sinne* stellen für ihn Informationssysteme dar, die mindestens eine Phase marktmäßig organisierter Leistungskoordination unterstützen.

Darüber hinaus kann man zwischen offenen und geschlossenen elektronischen Märkten unterscheiden:

- *Geschlossene elektronische Märkte* (wie beispielsweise das Börsensystem *XETRA*) werden von einem Betreiber bzw. einer zentralen Instanz, dem sogenannten *Market Maker*, kontrolliert. Dieser erhält über ein zentrales Informationssystem von den angeschlossenen Marktteilnehmer Kauf- und Verkaufsaufträge und entscheidet welche Markttransaktionen vorgenommen werden. Insbesondere entscheidet die zentrale Instanz darüber, welche Akteure am Markt teilnehmen dürfen.
- Im Unterschied dazu gibt es bei *offenen elektronischen Märkten* keine zentrale Instanz mehr, die den Zugang zum Markt kontrolliert, so dass dieser prinzipiell allen interessierten Teilnehmern offen steht. Offene elektronische Märkte werden durch eine öffentlich zugängliche Informationsstruktur realisiert. Da keine zentrale Instanz mehr existiert, werden Angebot und Nachfrage auch nicht mehr zentral koordiniert, so dass die Koordinationsleistung entweder durch die Selbstkoordination aller Teilnehmer erreicht werden muss oder aber die koordinativen und sonstigen Aufgaben

der zentralen Instanz von mehreren ausgezeichneten Entitäten übernommen werden müssen.

Damit die Anbieter und Nachfrager auf elektronischen Märkten miteinander in Kontakt treten können, verfügen diese Märkte häufig über einen Verzeichnisdienst oder einen Katalog, mittels derer die Marktteilnehmer nach geeigneten Partnern suchen können, um mit diesen bzgl. der Preise und Eigenschaften der angebotenen Produkte in Verhandlungen zu treten.

In Abhängigkeit davon, ob ein elektronischer Markt branchenübergreifend oder branchenspezifisch ist, wird laut [Nenninger & Lawrenz 2013, S.290 f.; Kortus-Schultes & Ferfer 2015, S. 126 f.; Müller et al. 2003, S. 349] zudem zwischen *horizontalen* und *vertikalen elektronischen Märkten* unterschieden:

- *Horizontale elektronische Märkte* decken den Bedarf mehrerer Branchen ab und fokussieren auf den Handel bestimmter Produktgruppen.
- *Vertikale elektronische Märkte* decken lediglich den Bedarf einer einzelner Branchen und beschränken den Zugang häufig auf eine geschlossene Benutzergruppe.

Je nachdem, ob ein elektronischer Markt auf die inner- oder überbetriebliche Koordination abzielt, kann weiterhin grob zwischen

- *Innerbetrieblichen elektronischen Märkten* und
- *Überbetrieblichen elektronischen Märkten*

unterschieden werden, wie bereits in Abschnitt 2.7 angesprochen.

Die Art der gehandelten Wirtschaftsgüter bestimmt häufig die Ausprägungsform eines elektronischen Marktes. Auf einigen elektronischen Märkten, auf denen standardisierte Transaktionsobjekte gehandelt werden, stellt der Preis das alleinige Einigungskriterium dar. Für entsprechende Märkte geeignete Software-Agenten sind eher einfach zu konstruieren. Auf anderen elektronischen Märkten müssen auch weitere Leistungs- bzw. Beschreibungsparameter der gehandelten Wirtschaftsgüter berücksichtigt werden. Je mehr davon zu berücksichtigen sind, umso komplexer wird die Aufgabe, auf diesen Märkten agierende Software-Agenten so mit Wissen und Verhandlungs- sowie Entscheidungsfähigkeiten auszustatten, dass sie vollautomatisiert handeln und dabei erfolgreich sein können.

Im Evaluationsteil dieser Arbeit werden Szenarien untersucht, bei denen es sich um offene horizontale elektronische Märkte handelt.

3.3.3.2 Phasen marktlicher Transaktion

Der Tauschprozess für Güter und Leistungen folgt unabhängig vom Tauschobjekt einem einheitlichen Schema, welches sich nach [Schmid 1993] und [Müller 2004] in drei bis vier Phasen unterteilen lässt (s. Abbildung 24).

- In der *Informationsphase*, auch Anbahnungsphase genannt, werden Informationen darüber gesammelt, welche Güter auf dem Markt angeboten werden, welches deren Anbieter und Nachfrager sind, welche Reputation diese besitzen und zu welchen Konditionen (beispielsweise Preise, Lieferbedingungen und -zeiträume) die Güter angeboten werden. Die Informationsphase des Anbieters bzw. Verkäufers setzt dabei häufig etwas zeitversetzt zu der des potenziellen Käufers ein, nachdem ihn dieser kontaktiert hat. Sie kann relativ aufwendig sein, denn oft werden in dieser Phase Informationen zu mehreren als Geschäftspartner in Frage kommenden Marktteilnehmern parallel eingeholt, um daraus die passendsten zu selektieren und mit ihnen in der nächsten Phase Verhandlungen zu führen.

- In der *Vereinbarungsphase* wird anschließend von beiden Seiten entschieden, ob die Transaktion durchgeführt wird. In dieser Phase erfolgen die Verhandlungen über den Kaufpreis und sonstige Konditionen, so dass häufig ein hoher Kommunikationsaufwand resultiert. Die Phase endet mit einem Verhandlungsabbruch oder einem Vertragsabschluss.
- Die dritte Phase, in der die Abwicklung der Transaktion, d. h. der physische Gütertausch, logistisch erfolgt, wird *Abwicklungsphase* genannt. Aus ihr können Folgeprozesse und -geschäfte auf weiteren Märkten, z. B. Märkten für Transportdienstleistungen entstehen, die wiederum die genannten Phasen durchlaufen. Diesbezüglich relevante Veröffentlichungen zu entsprechenden agentenbasierten Märkten werden in Abschnitt 4.2.2.1.2 genannt.
- An die Abwicklungsphase kann sich eine weitere *After-Sales-Phase* anschließen, welche z. B. Service-Leistungen für den Kunden umfasst.

Die in MACSIMA instanziierten Agenten durchlaufen die drei erstgenannten Phasen. Die beschriebenen Phasen und ihr Zusammenspiel sind in Abbildung 24 noch einmal übersichtlich veranschaulicht.

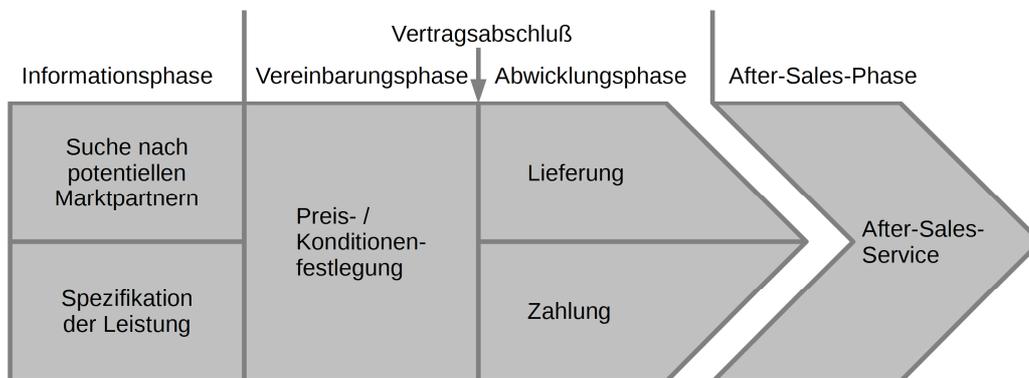


Abbildung 24: Phasen einer marktlichen Transaktion ³¹

3.3.4 Marktliche Koordinationsmechanismen

Der wesentliche Anreiz eines Agenten, mit anderen Agenten zusammenzuarbeiten, besteht in einem von ihm dadurch erwarteten Nutzenzuwachs, den er ohne Kooperation mit anderen nicht erreichen könnte.

Im Rahmen von kooperativen Interaktionen zwischen Akteuren kann ein geeigneter Koordinationsmechanismus zu großen Effizienzsteigerungen führen, wenn es Interdependenzen zwischen mehreren Akteuren gibt (vgl. Abschnitt 2.5.3), durch die es zu Konflikten kommen kann. Zu solchen Interdependenzen kommt es beispielsweise, wenn die zu einem bestimmten Zeitpunkt auf einem Markt nachgefragte Menge einer Ressource die angebotene Menge übersteigt.

Kommt es zu solchen Interdependenzen, so kann ein marktlicher Koordinationsmechanismus bzw. Marktmechanismus dabei helfen, die entstehenden Konflikte aufzulösen. Dabei strebt er in der Regel an, eine Konfliktlösung zu finden, die den Gesamtnutzen aller Agenten maximiert.

³¹ Quelle: In Anlehnung an [Deßloch 2002] und der Vorlesung „Elektronische Märkte“ von Michael Herrmann (TU München)

In MACSIMA wurden als marktliche Koordinationsmechanismen *bilaterale Verhandlungen* und eine Reihe von *Auktionen* integriert, welche von den Agenten zur Koordination ihrer Aktivitäten auf den instanziierten elektronischen Gütermärkten genutzt werden können.

3.3.4.1 Bilaterale Verhandlungen

Wenn sich zwei Marktteilnehmer auf einen Verkaufspreis für ein Gut einigen müssen, handelt es sich um ein Koordinationsproblem, welches im *Modus direkter Verhandlungen* (siehe 2.5.3) ohne die Unterstützung durch eine zentrale Koordinator-Instanz innerhalb einer bilateralen Verhandlung gelöst werden kann.

In der Literatur wird der Koordinationsmechanismus der bilateralen Verhandlung innerhalb des Forschungsbereichs der Verhandlungstheorie³² untersucht, wobei die axiomatische (mit der Verhandlungslösung von Nash) und die strategische Verhandlungstheorie (mit dem Rubinsteinspiel) zu unterscheiden sind.

[Rosenschein & Zlotkin 1994] haben auf der Basis des Rubinsteinspiels ein einfaches bilaterales Verhandlungsprotokoll für Agenten, das sogenannte *Monotonic Concession Protocol*, entwickelt, bei dem sich die Opponenten durch einen Prozess von Angeboten und Gegenangeboten mit schrittweisen Zugeständnissen immer weiter annähern, bis sie eine Einigung erzielen oder die Verhandlung abbrechen.

Die dabei von seinem Opponenten gemachten Vorschläge (englisch *Proposals*) werden von einem Agenten jeweils auf ihren Nutzen hin bewertet. Sofern das Angebot des Opponenten ihm den gleichen oder sogar einen höheren Nutzenzuwachs bietet als sein letztes eigenes Angebot, ist eine Einigung erzielt und der Agent leitet einen Vertragsabschluss ein. Ist dies nicht der Fall, so muss sich der Agent entscheiden, ob er die Verhandlung abbricht oder seinerseits einen Gegenvorschlag macht. Dieser kann entweder darin bestehen, auf dem letzten eigenen Angebot zu beharren und dieses dem Opponenten wieder vorzulegen oder aber es beinhaltet ein Zugeständnis. Solange mindestens einer der Opponenten im Verlauf der Verhandlung regelmäßig Zugeständnisse macht, werden sich die von den Agenten gemachten Angebote irgendwann überschneiden, so dass eine Einigung erreicht und ein Vertragsabschluss getätigt wird. Falls jedoch beide Agenten im Verhandlungsverlauf einen Punkt erreichen, an dem sie zu keinen weiteren Zugeständnissen mehr bereit sind, so entsteht ein Konflikt, der zum Verhandlungsabbruch führt.

Durch geschicktes strategisches Verhalten ist es für die Opponenten dabei oftmals möglich, sich Vorteile zu verschaffen. [Rosenschein & Zlotkin 1994] haben einige Vorschläge für günstige Verhandlungsstrategien entwickelt.

In Abbildung 25 ist der Ablauf bilateraler Verhandlungen übersichtlich dargestellt. Ihre Erstvorschläge beinhalten ihren jeweiligen *Startpreis*, der in der Regel so gewählt ist, dass dessen Durchsetzung in Anbetracht der aktuellen Marktpreise für sie einen überdurchschnittlichen Nutzenzuwachs bedeuten würde. Solange keine Einigung erzielt wurde und die Verhandlung andauert, kommen sie dann ihrem Opponenten schrittweise preislich entgegen. Jeder der Agenten besitzt dabei einen geheimen *Indifferenz-* bzw. *Reservationspreis*. Wird der Indifferenzpreis des Verkäufers unterschritten, so bringt eine Fortführung der Verhandlung diesem keinen Nutzen mehr, so dass er sie abbricht. Gleiches gilt für den Käufer bei einem Überschreiten seines Indifferenzpreises. Das Preisintervall zwischen den Indifferenzpreisen der Opponenten spannt den möglichen *Einigungsraum* bzw. *Abschlussbereich* auf, in dem der Kaufpreis liegen wird, sofern nicht einer der Agenten die Verhandlung (z. B. aufgrund von Ungeduld oder Verärgerung) vorher abbricht.

³² In der englischsprachigen Literatur auch *bargaining theory* genannt.

Das in MACSIMA implementierte bilaterale Verhandlungsprotokoll entspricht dem vorgestellten Protokoll. Der konkrete Ablauf bilateraler Verhandlungen in MACSIMA wird in den Abschnitten 5.3.1 und 5.3.3 weiter vertieft.

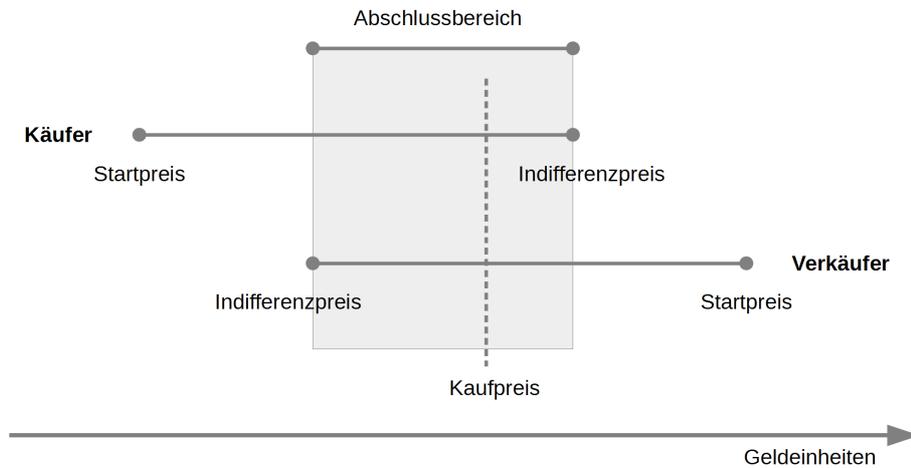


Abbildung 25: Bilaterale Preisverhandlungen³³

An dieser Stelle ist zu bemerken, dass die in Kapitel 5 beschriebenen, originär für bilaterale Verhandlungen entwickelten Konzepte darüber hinausgehend so erweitert wurden, dass MACSIMA-Agenten nicht nur an bilateralen Verhandlungen, sondern auch an Auktionen teilnehmen können. Die Lernfähigkeiten der Agenten werden also mittlerweile auch zur Anpassung von Bietstrategien bei verschiedenen Auktionsformen genutzt.

Die hierzu notwendigen konzeptionellen Erweiterungen und Implementierungsarbeiten wurden von meinem ehemaligen Diplomanden Alexander Walz im Rahmen seines laufenden Promotionsvorhabens an der Universität Stuttgart durchgeführt.

Neben bilateralen Verhandlungen stellt das MACSIMA-System seinen Agenten daher gegenwärtig vier weitere marktliche Koordinationsmechanismen zur Verfügung, nämlich die *Vickrey-Auktion*, die *verdeckte Erstpreis-Auktion*, die *Englische Auktion* sowie die *Holländische Auktion*.

Da diese Auktionsformen in MACSIMA genutzt und auch im Rahmen der in Abschnitt 4.4 vorgestellten holonischen Koordinationsarchitektur eingesetzt werden können, werden sie im anschließenden Abschnitt kurz vorgestellt.

3.3.4.2 Auktionsmechanismen

Bei Auktionen handelt es sich um eine der in ökonomischen Anwendungsszenarien mit am häufigsten verwendeten Koordinationsmechanismen. Dies liegt darin begründet, dass viele Auktionsformen relativ transparent und gut verständlich sind und sich auch ohne übermäßigen Aufwand implementieren und durchführen lassen. Darüber hinaus liegen zu Auktionen umfangreiche Forschungsergebnisse der Mechanismus-Design-Theorie vor, welche die Vor- und Nachteile einer Vielzahl an Auktionsformen in Abhängigkeit von ihrem ökonomischen Einsatzzweck äußerst detailliert darlegen. Wesentliche Vorteile ihres Einsatzes bestehen darin, dass sie in den allermeisten Fällen dazu in der Lage sind, effiziente Ressourcenallokationen bei sehr niedrigem Kommunikationsaufwand und Beibehaltung eines hohen Autonomiegrades zu realisieren.

³³ Quelle: In Anlehnung an [De Paula et al. 2001]

„Auctions provide efficient, distributed and autonomy preserving ways of solving task and resource allocation problems in computational multiagent systems.“ [Sandholm 1999, S. 1]

Bei Auktionen tritt einer der Agenten als Auktionator auf. In dieser Eigenschaft bietet er den übrigen Agenten an, über ihn ein von diesen angebotenes Gut zur Versteigerung auszusprechen. Dafür muss der Anbieter festlegen, welche Auktionsform bei der Versteigerung benutzt werden soll und wann die Auktion zu starten hat. Darüber hinaus muss der Anbieter je nach Auktionsform weitere Angaben, vor allem preislicher Art machen. Bei der prominenten sogenannten Englischen Auktion muss er beispielsweise den Mindest- bzw. Reservationspreis angeben, welcher das Mindestgebot angibt, unter dem die Bieter nicht in die Auktion einsteigen dürfen.

Melden sich daraufhin genügend Interessenten bzw. potenzielle Bieter beim Auktionator, so startet dieser die Auktion und erwartet die Gebote der Interessenten. Gehen Gebote ein, so sorgt der Auktionator für die Einhaltung der durch das Protokoll der Auktionsform vorgegebenen Abläufe und Regeln, gemäß derer er auch bestimmt, welcher der Bieter zu welchem Preis letztlich den Zuschlag erhält.

Auktionen werden im Forschungsgebiet des *Mechanismus-Design-Theorie* [Hehenkamp 2007] zur Klasse der direkten Mechanismen gezählt und werden in [Ruß 1997] detailliert vorgestellt und diskutiert. Kriterien für ihre Auswahl wurden in dieser Arbeit in Abschnitt 2.5.3 erläutert.

Die vielen unterschiedlichen existierenden Auktionsformen lassen sich anhand ihrer spezifischen Auktionsregeln sowie nach der Art der Bewertung der gehandelten Güter oder Leistungen ganz grundsätzlich in die Klassen der *private-value-* und der *common-value-Auktionen* einordnen. In der ersten Klasse besitzt jeder der Bieter unterschiedliche Präferenzen für den Auktionsgegenstand, da jeder von ihnen (z. B. bei einem Kunstgegenstand) eine individuelle, rein *private Bewertung* des gehandelten Guts vornimmt, welche von den Bewertungen anderer Bieter vollkommen unabhängig ist. Im Fall von *common-value-Auktionen* werden hingegen Güter angeboten, die zwar einen objektiven Marktwert besitzen, der aber zum Auktionszeitpunkt nur geschätzt werden kann (z. B. bei Bohr- und Schürfrechten), so dass die Abgabe eines Gebots mit der Unsicherheit behaftet ist, ob dessen Höhe nicht vielleicht höher als der tatsächliche Wert des Auktionsobjekts ist (englisch *Winner`s Curse*).

Weiterhin lassen sich Auktionen auch den ihren Auktionsregeln klassifizieren:

- Bei *progressiven Auktionen*: werden so lange offene Gebote abgegeben, bis nur noch ein Bieter vorhanden ist, welcher den Zuschlag zu seinem letztgenannten Gebot erhält.
- Bei *Auktionen mit verdeckten Geboten* werden diese nicht öffentlich gemacht, sondern dem Auktionator nur verdeckt mitgeteilt. Hierbei gibt es ein- und mehrstufige Prozesse sowie auch unterschiedliche Verfahren zur Bestimmung des Zuschlagspreises, auf die an dieser Stelle nicht weiter eingegangen wird. In vielen Fällen erhält aber auch bei diesen Auktionsformen derjenige Bieter den Zuschlag, der das höchste Gebot abgegeben hat. Welchen Preis dieser zu zahlen hat, hängt von der spezifischen Auktionsform ab.
- Bei *Auktionen über mehrere Güter* erfolgt die gleichzeitige Versteigerung mehrerer Güter.

In [Ruß 1997] wurden folgende Varianten im Detail untersucht:

- Mehrstufige progressive Auktionsformen mit offener Gebotsabgabe:

- *Englische Auktion*: Sie ist die wohl bekannteste Auktionsform, wie sie z. B. bei Kunstauktionen eingesetzt wird. Ausgehend vom Mindestpreis überbieten sich die Teilnehmer gegenseitig, bis ein Höchstpreis erreicht ist, den niemand mehr überbieten will. Zu diesem Höchstpreis erhält der Höchstbietende den Zuschlag.
- *Holländische Auktion*: Ausgehend von einem Höchstpreis verringert der Auktionator in vorher festgelegten Schritten den Angebotspreis, bis einer der Bieter zum aktuellen Preis zuschlägt.
- Einstufige Auktionsformen mit verdeckter Gebotsabgabe:
 - *Verdeckte Ersterpreis-Auktion* (englisch *First-price Sealed-bid Auction*): Bei dieser Auktionsart geben alle Bieter jeweils genau ein verdecktes Gebot ab. Den Zuschlag bekommt der Höchstbietende zu dem in seinem Gebot genannten Preis.
 - *Vickrey-Auktion* (englisch *Second-price Sealed-bid Auction*): Bei dieser Variante der First-Price Sealed-Bid Auktion geben ebenfalls alle Bieter jeweils genau ein verdecktes Gebot ab. Den Zuschlag erhält auch der Höchstbietende. Dieser muss jedoch bei dieser Auktionsform lediglich den Preis des zweithöchsten Gebots zahlen, das von den übrigen Bietern beim Auktionator abgegeben wurde.

In [Ruß 1997] wird erläutert, dass die Englische Auktion und die Vickrey Auktion beide zu einem pareto-effizienten Ergebnis führen. In beiden Auktionsformen besteht die dominante Strategie der Agenten darin, ihren Präferenzen entsprechend Gebote abzugeben, die ihre Präferenzen wahrheitsgemäß widerspiegeln. Die Englische Auktion hat gegenüber der Vickrey Auktion den Nachteil, dass sie umständlicher zu implementieren ist und einen größeren Kommunikationsaufwand verursacht. Letzterer wird durch die Vickrey Auktion extrem minimiert, allerdings hat diese Auktionsform auch einige Nachteile, wie beispielsweise die Manipulierbarkeit durch Bieter-Koalitionen, die Sandholm in seinem Beitrag „*Limitations of the Vickrey Auction in Computational Multiagent Systems*“ [Sandholm 1996b] erläutert.

4 Agenten in ökonomischen Wertschöpfungsstrukturen

4.1 Einleitung und Übersicht

Die Prozesse in Wertschöpfungsnetzwerken beziehungsweise Supply Webs laufen in der Regel verteilt und hochdynamisch ab. Neue Unternehmen (z. B. Lieferanten oder Produzenten) können in solche Netzwerke eintreten, während andere Teilnehmer sie verlassen. Jede Entität in einem Wertschöpfungsnetzwerk kann ein eigenständiges autonomes Unternehmen darstellen. Dies hat zur Folge, dass die Mitglieder individuelle Ziele haben und sich daher nur solange kooperativ verhalten, wie dies in ihrem eigenen Interesse liegt (siehe [Zeng & Sycara 1998]).

Daher sind die beteiligten Entitäten auch häufig nicht bereit, sensible individuelle Informationen zu offenbaren, so dass die für eine effiziente unternehmensübergreifende Planung und Koordination notwendigen Informationen in der Regel nur dezentral verfügbar sind. Eine Planung kann daher nur verteilt und auf der Basis von unvollständiger Information über den Status und die Aktivitäten anderer Teilnehmer erfolgen. Hierdurch und aufgrund der Wechselwirkungen zwischen den Entscheidungen und Interaktionen der autonomen Mitglieder kommt es in Wertschöpfungsnetzwerken häufig zu einer großen Zahl an Koordinationsproblemen.

Die Wertschöpfungsnetzwerk-Domäne kann daher insgesamt als nur partiell beobachtbar, nicht-deterministisch und stark dynamisch charakterisiert werden (vgl. dazu die Umwelt-Charakteristika von Software-Agenten in Abschnitt 2.3.3). Darüber hinaus beinhaltet sie Koordinationsprobleme, für die das traditionelle SCM noch keine ausgereiften Lösungsansätze bietet. Dies liegt daran, dass sich das SCM in der Vergangenheit vornehmlich mit der zentralen Koordination und Steuerung von Prozessen in eher statischen Wertschöpfungsketten beschäftigt hat, deren Mitglieder eine vergleichsweise geringe Entscheidungsautonomie besaßen.

Agenten können auf Änderungen in ihrem Umfeld schnell und flexibel reagieren und sind in der Lage, mit den Anforderungen in einer solchen Domäne gut zurecht zu kommen. Zudem existieren in der MAS-Forschung zahlreiche Ansätze, die sich zur Lösung der in Wertschöpfungsnetzwerken auftretenden Koordinationsprobleme anbieten. Verschiedene Möglichkeiten, mit denen man eine Koordination in MAS erreichen kann, wurden bereits in Abschnitt 2.5.3 erläutert. Ansätze zur marktlichen Koordination in MAS wurden zudem in den Abschnitten 2.7 und 3.3 vertieft behandelt.

Daher erscheint es naheliegend, bei der Entwicklung von Software für das Management von Wertschöpfungsketten oder -netzwerken Agententechnologie einzusetzen. Die operativen Einheiten von Wertschöpfungsnetzwerken können agentifiziert, das heißt als sogenannte *digitale Geschäftsagenten* [Eymann 2003] realisiert werden, um die grundlegenden dynamischen und stochastischen Anforderungen der Wertschöpfungsnetzwerk-Domäne zu erfüllen.

Da intelligente Agenten autonom Aufgaben im Auftrag ihrer Benutzer durchführen, können sie automatisch und dynamisch auf Veränderungen reagieren. Gerade weil die Operationalität von Wertschöpfungsnetzwerken häufig durch viele externe Einflüsse gefährdet ist, ermöglicht die Agentifizierung der Supply Chain Einheiten eine flexible und robuste Operationalität des gesamten Netzwerks.

Auf einer übergreifenden Ebene kann dann eine geeignete *agentenbasierte Koordinationsinfrastruktur* (siehe u. a. [Ruß & Gerber 2000, Ruß & Gerber 2001a; Ruß et al. 2001; Ruß et al. 2003]) realisiert werden, welche *Supply-Web-Entitäten* (wie z.B. Produzenten, Lieferanten, Großhändler, Einzelhändler, usw.) mit gemeinsamen oder zueinander passenden Zielen zusammenbringt und es ihnen ermöglicht ihre, Aktivitäten innerhalb des Netzwerks zu koordinieren.

Dieses Kapitel gibt zunächst einen Überblick über Einsatzszenarien agentenbasierter Koordination in ökonomischen Wertschöpfungsstrukturen und liefert einen Überblick der relevanten Forschungsprojekte, der zugehörigen Literatur sowie der entwickelten Prototypen.

Im Anschluss daran werden in Abschnitt 4.3 holonische Geschäftsagenten und ein holonischer Koordinationsserver-Agent vorgestellt, die für den Einsatz auf elektronischen Märkten und in der Wertschöpfungsnetzwerk-Domäne konzipiert sind. Ihr Design beruht auf den Konzepten der Holonischen Multiagenten-Systeme (H-MAS), die in Abschnitt 2.6 erläutert wurden.

Die holonischen Geschäftsagenten können sich über Auktionsmechanismen, die von einem oder mehreren Koordinationsserver-Agenten zur Verfügung gestellt werden, miteinander marktlich koordinieren. Dabei werden die Funktionsweise sowie das Zusammenspiel von Geschäftsagenten und Koordinationsserver-Agenten ausführlich erläutert. Hierbei zeigt sich, dass die in Kapitel 5 vorgestellten Verhandlungs- und Lern-Funktionalitäten der MACSIMA-Agenten in die vorgestellten, holonisch strukturierten Geschäftsagenten integriert werden können.

4.2 Einsatzszenarien

Im Bereich ökonomischer Wertschöpfungsstrukturen gibt es eine Fülle an Möglichkeiten für den Einsatz von intelligenten Agenten. Agentenbasierte Konzepte bieten sich insbesondere dort an, wo ein Handel mit Ressourcen, Gütern, Dienstleistungen, Kapazitäten und Aufträgen stattfindet. Zwar handelt es sich häufig um Forschungsprototypen, die im Moment nicht in allen Fällen in der praktischen Unternehmensrealität eingesetzt werden, jedoch nimmt die Zahl der in der Praxis kontinuierlich im Realeinsatz befindlichen Systeme kontinuierlich zu. Im Folgenden werden, ohne den Anspruch an Vollständigkeit zu erheben, einige relevante MAS-Forschungsarbeiten vorgestellt.

4.2.1 Supply Chain Management

In der Literatur zu *Managementinformationssystemen (MIS)* ist die Idee, Multiagenten-Systeme (MAS) und *Supply Chain Management (SCM)* miteinander zu verbinden, von vielen Forschern vorgeschlagen worden [Lin et al. 2005; Lin et al. 2000; Nissen 2000; Nissen 2001; Ram & Lui 2005; Sadeh et al. 2003; Sikora & Shaw 1998; Strader et. al. 1998; Yung et al. 2000].

[Grolik et al. 2001] und [Chaib-draa & Müller 2006] geben jeweils einen guten Überblick über die grundlegende Forschungsliteratur, woran sich auch Teile der folgenden Ausführungen orientieren.

Im Projekt *DISPOWEB (Dispositive Supply-Web-Koordination)* [Grolik et al. 2001, S. 143-155] wurden Software-Agenten entwickelt, die insbesondere in den Bereichen Produktionsplanung und der Logistik Entscheidungen treffen. Als Koordinationsmechanismen werden dabei bilaterale und multilaterale Verhandlungen eingesetzt, in denen vor allem optimale Zeitpunkte für Lieferungen ausgehandelt werden. Jedes Mitglied einer Supply Chain kann in DISPOWEB Bestandteil von mehreren Wertschöpfungsnetzwerken sein, die sich gegenseitig überlappen. So ist es möglich, dass ein Lieferant (zum Beispiel ein Reifenhersteller) mehrere produzierende Unternehmen (zum Beispiel Automobilhersteller) beliefert, die zueinander in Konkurrenz stehen. Der Zulieferer kann dabei seine Produktionsplanung und die von ihm angebotenen Lieferzeitpunkte nur sehr schwer zeitgleich im Hinblick auf die unterschiedlichen Wünsche seiner Kunden optimieren. Da sich die Kosten einer Lieferung sowohl bei zu frühem als auch bei zu spätem Liefern erhöhen können, bestimmt in DISPOWEB den von ihm für eine Lieferung geforderten Preis zeitabhängig. Um zu Zeitpunkten liefern zu können, die ihm günstig erscheinen, kann er in Verhandlungen mit seinen Kunden auch temporäre, an bestimmte Lieferzeitpunkte gekoppelte Preissenkungen anbieten. Die dafür erforderliche kontinuierliche, flexible und dezentrale Planung wird agentenbasiert vorgenommen.

[Pippow 2004] beschäftigt sich mit dem Einsatz von Software-Agenten zur Lösung des so genannten Peitscheneffekts (englisch bullwhip effect, whiplash effect) [Lee 1997; Herlyn 2014]. Dabei handelt es sich um das Phänomen, dass sich Nachfrageschwankungen am Ende einer Versorgungskette in Richtung der vorgelagerten Stufen der Kette zu immer größeren Bestellmengenschwankungen aufschaukeln. [Pippow 2004] zeigt anhand von Simulationen, dass mit Software-Agenten, mit denen Transaktionen in Distributionsnetzen automatisiert durchgeführt werden, Transaktionskosten gesenkt und eine verbesserte Koordination erzielt werden können. [Teuteberg 2005] beschäftigt sich mit der Realisierung ubiquitärer Supply Networks auf Basis von Agenten-Technologien.

[Zimmermann 2006] entwickelt einen generischen agentenbasierten Ansatz für das Event Management in Wertschöpfungsnetzwerken, der allen Mitgliedern ereignisbezogen die relevanten Logistikinformationen für ihre verteilten Entscheidungsfindungsprozesse zur Verfügung stellt. Es wird gezeigt, dass dadurch Kosten vermindert und die Performanz von Wertschöpfungsnetzwerken gesteigert werden kann. [Ickerott 2007] stellt ein agentenbasiertes Planungs- und Simulationssystem vor, das dezentrale SCM-Planungsaufgaben unterstützt und dabei die individuellen und teilweise gegensätzlichen Zielvorstellungen der Mitglieder eines Wertschöpfungsnetzes berücksichtigt.

[Castellanos 2012] stellt ein agentenbasiertes Modell für generische Supply-Chain-Simulationen vor. Dieses ermöglicht es, verschiedene Supply-Chain-Strukturen mit unterschiedlichen Komplexitätsgraden und variierenden Strategiedefinitionen zu entwerfen und die damit verbundenen Auswirkungen auf das Systemverhalten und die Performanz der Supply Chains zu simulieren.

Grundlegende Forschungserkenntnisse im Bereich der agentenbasierten SCM-Systeme gehen auf die Arbeiten von Fox und Barbuceanu zurück (siehe [Barbuceanu et al. 1993; Barbuceanu & Fox 1995a; Barbuceanu & Fox 1997]), die die Modellierung von Versorgungsketten als agentenbasierte Netzwerke vorschlagen und beschreiben. In den von ihnen modellierten Supply Chains dienen sogenannte *Informationsagenten* dazu, innerhalb des Wertschöpfungsnetzwerks wichtige Informationen an *Funktionsagenten* zu verteilen. Die Funktionsagenten sind auf verschiedene Aufgabenbereiche (wie z. B. Logistik, Produktion, etc.) spezialisiert und leisten die Ausführung der anfallenden operativen Aufgaben innerhalb der Versorgungskette. Die Agentengesellschaft ist in ihren Arbeiten hierarchisch organisiert, so dass Agenten auf höheren Ebenen auf Agenten auf niedrigeren Ebenen steuernd, kontrollie-

rend und vermittelnd einwirken können. Informationsagenten können dabei auch die Aufgaben eines Intermediärs übernehmen. Um die Kommunikation zwischen den von ihnen modellierten Agenten zu verbessern, entwickelten sie die auf KQML-Sprechakten [Fini et al. 1994] basierende Sprache *COOL (COOrdination Language)* [Barbuceanu & Fox 1995b].

In [Kalakota et al. 1996; Hinkkanen et al. 1997] wird ein MAS zur dynamischen Kontrolle der in einer Versorgungskette ablaufenden Aktivitäten vorgestellt. Zur Repräsentation der Versorgungskette dient dabei ein mathematisches Modell. Die Agentengesellschaft ist hierarchisch organisiert. Die Agenten haben verschiedene Zuständigkeitsbereiche, innerhalb derer sie Aufgaben (sogenannte *Tasks*) übernehmen und ausführen. Dabei entscheiden sie unter Verwendung von Operations-Research-Methoden rational.

Bei *MetaMorph* [Shen & Norrie 1998a; Shen & Norrie 1998b] handelt es sich um eine hybride Architektur für die agentenbasierte Implementierung und Koordination von Wertschöpfungsnetzwerken. Bereits bestehende eigenständige Systeme werden agentifiziert, indem sie in Agenten „gekapselt“ werden. Darüber hinaus werden funktionale Elemente bis hin zur Produktions- beziehungsweise Maschinenebene als Agenten modelliert. Zur Koordination der Aktivitäten kommen sogenannte *Mediator-Agenten* zum Einsatz. Diese unterstützen die Agenten zum einen bei ihrer Suche nach anderen kooperativen Agenten beziehungsweise geeigneten Verhandlungspartnern. Die funktionalen Agenten können sich dann über Verhandlungsmechanismen, die das Contract Net Protocol [Smith 1980] nutzen, direkt miteinander koordinieren. Zum anderen können die Mediator-Agenten auf die Verhandlungs- und Entscheidungsprozesse der funktionalen Agenten auch Einfluss nehmen und deren Aktivitäten hierarchisch koordinieren.

[Swaminathan et al. 1998] verwenden einen ähnlichen Modellierungsansatz für das Design eines flexiblen Frameworks, das im SCM für Simulations- und Entscheidungsunterstützungszwecke eingesetzt werden kann. Sie agentifizieren sogenannte *Struktur-Elemente* von Versorgungsketten (wie z. B. Lieferanten, Produzenten, Vertriebszentren und Händler), die miteinander kommunizieren und sogenannte *Kontroll-Elemente* nutzen können. Die Kontrollelemente unterstützen die Agenten in ihren Entscheidungsprozessen und übernehmen, beispielsweise auch im Hinblick auf die Verteilung von Informationen, koordinative Aufgaben innerhalb der Versorgungskette.

In [Zeng & Sycara 1998] wurde untersucht, welche Rolle elektronische Marktplätze bei der Koordination und Flexibilisierung von Wertschöpfungsnetzwerken spielen können. Das Netzwerk wurde dabei als MAS modelliert, in dem die teilnehmenden Unternehmen von jeweils einem Software-Agenten repräsentiert werden. Dabei wurden insbesondere die auftretenden Koordinationskosten betrachtet.

[Chen et al. 1999a; Chen et al. 1999b] entwickelten einen Koordinationsansatz für das SCM, der auf miteinander verhandelnden Agenten beruht. Wie in den vorgenannten Ansätzen enthält ihr Ansatz sowohl Informationsagenten als auch funktional-operative Agenten, die die Ressourcen des Wertschöpfungsnetzwerks kontrollieren. Wissen über die Ressourcen und die zur Koordination verwendeten Verhandlungsprotokolle liegt in einer gemeinsamen Ontologie vor. Darüber hinaus erlaubt ihnen eine gemeinsame *ACL (Agent Communication Language)* die Kommunikation untereinander.

Parunak konnte mit der *ANTS*-Architektur (*Agent Network for Task Scheduling*) (vgl. [Parunak et al. 1999]) zeigen, dass – analog zu Insektivölkern – durch den Einsatz vieler sehr einfacher Agenten beim Scheduling in Versorgungsketten eine hohe Robustheit erzielt werden kann. Auch mit dem Projekt-*AARIA (Autonomous Agents for Rock Island Arsenal)* [Parunak et al. 1998; Baker et al. 1999] konnte er verdeutlichen, dass Agenten für die Steuerung von Produktionsprozessen sehr gut geeignet sind und deren Flexibilität verbessern können.

Während sich die erstgenannten Forschungsarbeiten von Parunak vor allem auf die Produktionsplanung fokussierten, stellte er im *DASCh*-Projekt (*Dynamical Analysis of Supply Chains*) [Parunak 1998; Parunak 1999] einen Modellierungsansatz vor, bei dem agentenbasierte Wertschöpfungsnetzwerke auf der Basis von drei Agenten-Typen, nämlich *Firmen-Agenten*, *PPIC-Agenten* und *Transport-Agenten* realisiert werden. Die Firmen-Agenten steuern dabei die PPIC-Agenten, die die Produktionsplanungs- und Materialwirtschaftsaufgaben übernehmen.

An der Carnegie Mellon Universität wurde die MAS-Architektur *MASCOT* (*Multi-Agent Supply Chain Coordination Tool*) entwickelt (vgl. [Kjenstad 1998; Sadeh et al. 1999; Sadeh et al. 2001]). Diese hat zum Ziel, die Koordination von Planungs-, Scheduling- und Materialfluss-Aufgaben innerhalb von produktionsorientierten Versorgungsketten zu verbessern und damit deren Agilität zu steigern. Wie in den vorgenannten Arbeiten werden auch in *MASCOT* funktionale Elemente der Versorgungskette agentifiziert. Die Agenten agieren auf mehreren Hierarchieebenen. Auf höheren Ebenen werden strategische und eher langfristige Entscheidungen getroffen, die die globale Planung der gesamten Kette und das interorganisatorische Zusammenspiel ihrer Mitglieder betreffen. Auf den unteren Ebenen werden hingegen taktische und kurzfristige Entscheidungen getroffen, die die aktuellen Produktionsprozesse betreffen. Um eine globale Koordination der Aktivitäten zu gewährleisten, können Agenten übergeordneter Ebenen auf Agenten von niedrigeren Ebenen steuernd einwirken. Neben diesem hierarchischen Koordinationsansatz können sich Agenten, die sich auf der gleichen Hierarchieebene befinden, über den Austausch von Nachrichten aber auch direkt miteinander koordinieren.

Bei der MAS-Simulationsplattform *SWARM* (vgl. [Strader et al. 1998; Lin et al. 1996; Lin 1996]) handelt es sich um ein weiteres System, das sich mit dem Einsatz von Agenten in Wertschöpfungsnetzwerken beschäftigt.

Neben diesen Forschungsarbeiten werden in verschiedenen Veröffentlichungen auch mobile Agenten zur Steuerung von Versorgungsnetzwerken eingesetzt (vgl. [Brugali et al. 1998; Papaioannou & Edwards 1998; Papaioannou & Edwards 2000; Szirbik et al. 1999; Szirbik et al. 2000]).

4.2.2 Elektronische Märkte

Elektronische Märkte dienen der Zusammenführung von Angebot und Nachfrage mit Hilfe der Informations- und Kommunikationstechnologie. Diese ermöglicht eine effiziente Abwicklung von Transaktionen zwischen Anbietern und Nachfragern.

Auf einigen elektronischen Märkten, auf denen standardisierte Transaktionsobjekte gehandelt werden, stellt der Preis das alleinige Einigungskriterium dar – auf anderen müssen auch weitere Leistungs- bzw. Beschreibungsparameter der gehandelten Objekte berücksichtigt werden.

Je nachdem, ob ein elektronischer Markt auf die inner- oder überbetriebliche Koordination abzielt, kann grob zwischen

- *Innerbetrieblichen elektronischen Märkten* und
- *Überbetrieblichen elektronischen Märkten*

unterschieden werden. Die in dem jeweiligen elektronischen Markt implementierten Koordinationsmechanismen determinieren die Qualität der emergierenden Lösungen auf entscheidende Art und Weise.

4.2.2.1 Innerbetriebliche elektronische Märkte

Innerbetriebliche elektronische Märkte zielen auf die effiziente Allokation begrenzter Ressourcen. In einer ganzen Reihe von Forschungsarbeiten wurden MAS-basierte elektronische Märkte für Problemstellungen der Ressourcenallokation in Produktion und Logistik konzipiert und prototypisch realisiert.

4.2.2.1.1 Produktion

Es gibt eine Fülle von Ansätzen für den Einsatz von MAS für die Planung und Steuerung von Produktions-Prozessen.

[Bussmann et al. 2013] stellt mit *DACS (Designing Agent-based Control Systems)* eine Methodologie für den agentenbasierten Entwurf von Fertigungssteuerungssystemen vor, der im Berliner Forschungslaboratorium der DaimlerChrysler AG entwickelt wurde und illustriert diesen anhand einer Reihe von Fallstudien.

Weitere grundlegende Ansätze finden sich zum Beispiel in [Zelewski 1995; Kassel 1996; Corsten & Gössinger 1997a]. Darüber hinaus gibt es Arbeiten zu elektronischen Märkten zur Prozesskoordination in Produktionsnetzwerken [Zelewski 1997] sowie zu *agentenbasierten holonischen Fertigungssystemen* [Fischer 1999].

In der *Produktionsplanung und -steuerung (PPS)* werden agentenbasierte Systeme eingesetzt, da immer kürzere Produktzyklen realisiert und immer mehr Produktvarianten hergestellt werden müssen. Agentenbasierte PPS-Systeme sollen eine hohe Auslastung und einen hohen Durchsatz gewährleisten. Gleichzeitig soll der Herstellungsprozess robust gegenüber Störungen (z. B. Maschinenausfälle), effizient, skalierbar und flexibel sein.

Der Einsatz von Multiagenten-Systemen setzt voraus, dass die Werkstücke flexibel auf unterschiedlichen Maschinen bearbeitet werden können. Jedes Werkstück, jede Maschine und auch die Bestandteile des Transportsystems zwischen den Maschinen können durch einen Software-Agenten oder auch physischen Agenten bzw. Roboter repräsentiert werden.

Agentifizierte Werkstücke können beispielsweise das Contract Net oder verschiedene Auktionsmechanismen (z. B. vgl. Abschnitt 3.2.5) nutzen und agentifizierte Maschinen dazu auffordern, Gebote für die Bearbeitung des Werkstücks abzugeben. Die Maschinen-Agenten bieten nur dann, wenn sie sowohl technisch als auch hinsichtlich zeitlicher und sonstiger Restriktionen dazu in der Lage sind, den nächsten nötigen Produktionsschritt auszuführen. Die Werkstück-Agenten sammeln die Gebote, wählen das Beste aus und fordern dann von Transportsystem-Agenten Gebote zum Transport des Werkstücks an. Nicht alle lokalen Entscheidungen sind optimal, aber global ergibt sich eine deutliche Flexibilisierung und ggf. Verbesserung der Leistung des Gesamtsystems.

Entsprechende Ansätze zur effizienten Auslastung von Maschinenkapazitäten wurden im PPS-Kontext u. a. in [Corsten & Gössinger 1997b; Möhle et al. 1996; Schmidt 2013] verfolgt.

4.2.2.1.2 Logistik

Weiteres Einsatzgebiet für MAS-basierte elektronische Märkte ist die Logistik, insbesondere die Transportplanung und die Allokation von Transportkapazitäten [z.B. Falk et al. 1993; Fischer et al. 1993; Fischer et al. 1998; Gomber et al. 1997; Gomber et al. 2000; Ruß & Vierke 1998a; Ruß & Vierke 1999a; Ruß et al. 1999].

Beispiele sind z. B. das *TELETRUCK*-System [Vierke 2000; Bürckert et al. 1998] oder der MAS-basierte Marktplatz *RoadMapOne*, deren Ziel es war, Nachfrager und Anbieter von

Transportdienstleistungen zusammen zu bringen, um die Nutzung vorhandener Transportkapazitäten zu optimieren und damit Kosten einzusparen.

[Nissen 2006] enthält eine Sammlung verschiedener Ansätze zum Einsatz von Agenten im Geschäftsprozessmanagement. Dazu zählen auch Ansätze, bei denen Software-Agenten zur Koordination von Logistik- und Transportplanungsprozessen eingesetzt werden.

4.2.2.2 Überbetriebliche elektronische Märkte

4.2.2.2.1 Agentenbasierte Verhandlungen und Auktionen

[Sackmann 2003; Sackmann 2013] beschäftigt sich mit dem Einsatz von Software-Agenten, die elektronische Repräsentanten menschlicher Akteure darstellen, auf elektronischen Märkten. Insbesondere untersucht er die dezentrale Koordination der Software-Agenten durch bilaterale Verhandlungen.

In den Arbeiten von Eymann (siehe [Eymann 2000; Eymann 2003; Eymann 2013]), nutzen die Agenten ebenfalls ein bilaterales Verhandlungsprotokoll zur marktlichen Selbstkoordination innerhalb eines Wertschöpfungsnetzwerks.

[Vetter 2006] entwirft ein MAS für die Automatisierung von Verhandlungen in elektronischen Märkten. Die am Markt teilnehmenden Agenten sind in der Lage, komplexe Verhandlungen durchzuführen, in die mehrere Verhandlungsattribute eingehen. Dies bedeutet, dass dabei neben dem Preis eines Produktes auch zum Beispiel Konditionen für dessen Bezahlung und Lieferung berücksichtigt werden. Neben der MAS-Architektur entwirft Vetter auch effiziente Verhandlungsstrategien, die er anhand von Simulationen evaluiert.

[Teuteberg 2007] beschäftigt sich mit der Modellierung und Realisierung agentenbasierter Marktplätze und simuliert darin ablaufende multidimensionale Verhandlungen. [Lang 2008] beschäftigt sich mit der wissensbasierten Automatisierung von Verhandlungen über mehrere variable Leistungsspezifika auf elektronischen Echtzeit-Märkten. Die Strategien, die dabei von den Agenten genutzt werden, sind dabei analog zu MACSIMA (siehe Abschnitt 5.3.2) genetisch codiert.

[Büttner 2010] entwickelt einen agentenbasierten Verhandlungsmechanismus für nur imperfekt beschreibbare Verhandlungsgegenstände und evaluiert diesen anhand von durchgeführten Simulationen. [Brenner et al. 2012, S. 267 ff.] geben einen Überblick über Einsatzszenarien von Agenten im Electronic Commerce. Dabei behandeln sie ausführlich agentenbasierte Marktplätze und die auf ihnen operierenden Agenten.

[Peters 2013] führt eine Analyse des Status Quo elektronischer Märkte durch und entwickelt ein erweitertes Auktionsverfahren sowie Automatisierungskonzepte für die auf elektronischen Märkten handelnden Akteure. Die Arbeit liefert eine recht umfassende Theorie zu automatisierten Marktsystemen, die anhand eines Multiagenten-Systems in verschiedenen simulationsbasierten Fallstudien verifiziert wird. Dabei werden sowohl Monopol- als auch Polypol-Marktszenarien sowie vollständig automatisierte Verhandlungen mit mehreren Verhandlungsparametern untersucht.

[Zarnekow 2013] erstellt Referenzmodelle, die der Integration von Software-Agenten in Kaufprozesse auf elektronischen Märkten dienen.

Im *MAGNET-System (Multi-Agent Negotiation Testbed)* der Universität von Minnesota [Collins et al. 1999; Collins et al. 2000a; Collins et al. 2000b] verwenden Agenten bei ihren Verhandlungen das sogenannte *finite leveled commitment protocol*. Die Agenten repräsentieren dabei entweder Lieferanten oder Bieter.

Die Projekte *Kasbah* [Chavez & Maes 1996; Guttman & Maes 1998, Guttman et al. 1997; Guttman et al. 1998], *Fishmarket* [Noriega 1997; Rodríguez et al. 1998] und *MAGMA* [Tsvetovaty et al. 1997] beschäftigen sich mit der Realisierung agentenbasierter elektronischer Märkte auf der Basis von Auktionen, wobei ihr grundlegender Aufbau sich am Blackboard-Ansatz orientiert, wie er bereits beschrieben wurde.

Bei *Kasbah* werden Verkaufs- und Kauf-Agenten von ihren Benutzern auf einem Marktplatz-Server erzeugt, über den die Agenten miteinander kommunizieren können. Die Agenten durchlaufen die in Abschnitt 3.3.3.2 erläuterten Phasen einer marktlichen Transaktion und erhalten von ihren Benutzern alle dafür notwendigen Informationen. Zu den Parameter-einstellungen, die vorgenommen werden müssen, gehören dabei zum Beispiel Informationen zu den Gütern, die ein Agent kaufen oder verkaufen soll, zu den Mindest- und Maximalpreisen, die dabei für seinen Nutzer noch akzeptabel sind und zu den Zeitpunkten, bis zu denen ein Agent entsprechende Transaktionen durchführen soll. Darüber hinaus kann der Nutzer eine von drei Preisstrategien auswählen, die sein Agent in seinen Verhandlungen zu verfolgen hat. Wählt der Benutzer die Preisstrategie *ängstlich* (englisch *anxious*), so versucht der Agent, eine gewünschte Transaktion möglichst schnell abzuschließen, ohne dem Transaktionspreis eine große Bedeutung beizumessen. Agiert der Agent gemäß der Einstellung *mit kühlem Kopf* (englisch *cool-headed*), so ist er bereit, etwas länger abzuwarten, um einen für ihn günstigen Transaktionspreis zu erzielen. Dabei nutzt er den von seinem Nutzer spezifizierten Zeitraum, der ihm für Verhandlungen zur Verfügung steht, jedoch in der Regel nicht vollständig aus, da er mit einer möglichst großen Wahrscheinlichkeit zu einer Transaktion gelangen will. Bei der Einstellung *sparsam* (englisch *economical*) wartet der Agent hingegen solange, bis er ein sehr gutes Angebot erhält – auch auf die Gefahr hin, dass es zu keiner Transaktion kommt.

Fishmarket realisiert ein MAS, bei dem Markt-Agenten anderen Agenten mehrere Auktionsmechanismen bereitstellen, über die sich die Agenten koordinieren können. Bei dabei auftretenden Problemen, zum Beispiel falls zwei Bieter das gleiche Gebot abgeben, greifen die Markt-Agenten zudem vermittelnd ein.

Bei *MAGMA* (*Minnesota Agent Marketplace Architecture*) verfolgt einen noch umfassenderen Ansatz. Käufer und Verkäufer werden ebenfalls als Agenten modelliert. Die Agenten besitzen ein explizites Verhandlungsmodul, mit dessen Hilfe sie Preisverhandlungen durchführen. Darüber hinaus sieht das System aber auch die Einbindung von Finanz- und Logistikdienstleistern über geeignete Schnittstellen vor.

Sammlungen von aktuellen Ansätze zum Einsatz von Agenten in komplexen automatisierten Verhandlungen finden sich in [Fujita et al. 2015] und [Ito 2008; Ito et al. 2009; Ito et al. 2010; Ito et al. 2011].

Darüber hinaus wird seit dem Jahr 2002 mit der *Agent Trading Competition (TAC)* ein jährlicher Wettbewerb ausgetragen, bei dem Agenten, die auf elektronischen Märkten kaufen und/oder verkaufen und daher *Trading Agents* genannt werden, in verschiedenen Spielszenarien gegeneinander antreten. Eines der Spielszenarien (*SCM-TAC*) fokussiert dabei auf die SCM-Domäne, ein weiteres (*POWER-TAC*) auf den liberalisierten Energiemarkt (siehe <http://tradingagents.eecs.umich.edu/>). Das SCM-TAC-Szenario, das u. a. von der Carnegie Mellon Universität und der Universität von Michigan (Ann Arbor) entworfen wurde, beschäftigt sich dabei analog zum MACSIMA-Basiszenario (siehe Abschnitt 5.2.1) mit einer Supply Chain zur Fertigung von Computersystemen. Das SCM-TAC-Spiel ist in [Arunachalam 2003; Arunachalam 2004] im Detail beschrieben.

Der TAC-Wettbewerb hat zu einer Vielzahl von Forschungsarbeiten geführt, die sich mit dem Design und der Analyse von Trading Agents für elektronische Märkte und den von

ihnen nutzbaren Mechanismen und Strategien beschäftigen (siehe zum Beispiel [Ceppi et al. 2014; Collins et al. 2008; David et al. 2010; David et al. 2012; Liu 2005; Poutré et al. 2006; Hertz 2012; Ketter et al. 2010; Wellman et al. 2007; Wellman 2011]).

4.2.2.2.2 Computergesteuerter Börsen- und Wertpapierhandel

Im Zuge des computergesteuerten Börsenhandels, dem sogenannten *Algorithmic Trading (AT)*, werden intelligente Software-Agenten auch zunehmend im Börsenhandel eingesetzt, um komplexe Investmentstrategien zu entwickeln und diese dann in Millisekunden umzusetzen. In vielen Medienberichten zur US-Immobilienkrise in 2008 sowie zum Zusammenbruch der Investmentbank Lehman Brothers sowie auch zu sonstigen Anomalien auf den Finanzmärkten wird die Intransparenz des Algorithmic Trading für auftretende Probleme verantwortlich gemacht.

[Gomolka 2011] stellt hierzu eine gute Einführung dar. Unter anderem werden reale AT-System-Architekturen vorgestellt und gemäß dem Vergleich von AT-Systemen mit den gängigen Definitionen und Eigenschaften von Agenten (vgl. die Abschnitte 2.3 und 2.4) die Hypothese abgeleitet, dass es sich „... bei den selbständig handelnden Software-Programmen im Algorithmic Trading um Software-Agenten handelt.“

[Gomber 2013; Gomber 2014] stellt mit dem *AMTRAS*-System den Prototypen eines agentenbasierten Handelssystems für Rentenwerte vor und weist nach, dass die Realisierung agentenbasierter Handelssysteme ökonomisch sinnvoll ist.

[Hu & Watt 2014] evaluieren im Rahmen des AT bzw. des *Hochfrequenzhandels (HFT)* Handelsstrategien mit Hilfe von *agentenbasierter Simulation (ABS)*.

[Ponomareva & Calinescu 2012; Ponomareva & Calinescu 2014] stellen ein adaptives Bayes'sches Agentenmodell vor und beleuchten den Zusammenhang zwischen der Frequenz des Treffens von Entscheidungen und der Einbeziehung komplexer Entscheidungskriterien auf der einen Seite und den erzielten Handelserfolgen des Agenten auf der anderen Seite.

4.2.2.2.3 Virtuelle Unternehmen und Unternehmensnetzwerke

Agentenbasierte marktliche Ansätze fanden auf überbetrieblicher Ebene auch beim Aufbau Virtueller Unternehmen [Arnold et al. 1995a; Fischer et al. 1996b; Fischer & Ruß 1998; Fischer & Ruß 1999; Ruß & Vierke 1998a] oder im Rahmen überbetrieblicher Produktionsnetzwerke [Zelewski 1997] Verwendung.

4.3 Holonische Agenten in elektronischen Märkten

Grundlage für den folgenden Abschnitt sind die in den Veröffentlichungen [Ruß & Gerber 2000; Ruß & Gerber 2001a; Ruß & Gerber 2001b; Ruß & Gerber 2001c; Ruß et al. 2001; Ruß et al. 2002; Ruß et al. 2003] vorgestellten konzeptionellen Ideen.

Eine praktische Implementierung und auch eine Evaluation der mit den beschriebenen holonischen Agenten erzielbaren Koordinations-Leistungen wurden im Rahmen von betreuten Diplomarbeiten [Deßloch 2002; Puin 2004] durchgeführt. Im Folgenden wird nur eine kurze Zusammenfassung wesentlicher Aspekte gegeben. Im Detail sind die Konzepte in den genannten Veröffentlichungen und Diplomarbeiten nachlesbar.

Die nachfolgenden beschriebenen Agenten-Typen spannen eine holonisch organisierte Koordinationsinfrastruktur auf, die der effizienten marktlichen Koordination von Geschäftsagenten mittels verschiedener Koordinationsmechanismen dient. Sie verfügt über einen Ver-

zeichnisdienst, in den auch Katalog-, Ontologie- und weitere Interoperabilitäts-Services integriert sein können. Über diesen können in Frage kommende Geschäftspartner beziehungsweise Geschäftsagenten und Koordinationsserver identifiziert werden.

Darüber hinausgehend beruht sie auf den Konzepten der *Holonischen Multiagenten-Systeme (H-MAS)*, die in Abschnitt 2.6 vorgestellt wurden und baut auf den folgenden beiden, holonisch strukturierten Agenten-Typen auf:

- *Holonischen Geschäftsagenten*, die auf elektronischen Märkten und in Wertschöpfungsnetzwerken agieren und sich marktlich koordinieren wollen und
- einem *holonischen Koordinationsserver*, der Koordinationsmechanismen (wie beispielsweise Auktionen) anbietet und durchführt, über die die Geschäftsagenten miteinander interagieren und sich koordinieren können. Hiervon kann es auch mehrere Instanzierungen geben.

Im Folgenden werden die wesentlichen Aspekte des Aufbaus, der Arbeitsweise und des Zusammenspiels von holonischen Geschäftsagenten und holonischen Koordinationsserver-Agenten kurz erläutert. Darüber hinaus wird deutlich gemacht, dass die in Kapitel 5 vorgestellten Verhandlungs- und Lern-Funktionalitäten nahtlos in holonische Geschäftsagenten integrierbar sind.

4.3.1 Holonische Geschäftsagenten

Holonische Geschäftsagenten, im Kontext der Wertschöpfungsnetzwerk-Domäne und der betreuten Diplomarbeit [Puin 2004] auch *generische Supply-Chain-Agenten (SCGA)* genannt, können auf einem elektronischen Markt beziehungsweise innerhalb eines Wertschöpfungsnetzwerks aus verschiedenen Rollen heraus agieren.

Beispielsweise können sie als Händler, Produzent, Käufer oder Verkäufer eines oder mehrerer Güter auftreten und handeln. Aus diesen unterschiedlichen Rollen ergeben sich verschiedene Aufgaben, die oftmals in weitere Teilaufgaben zerfallen und häufig auch parallel zueinander durchgeführt werden müssen. Zwischen den Aufgaben und Teilaufgaben bestehen dabei in der Regel komplexe Abhängigkeiten, da sie oftmals auf die gleichen Ressourcen (z. B. Geldmittel) zugreifen. Diese Interdependenzen müssen daher bei der Aufgabebearbeitung von einem Geschäftsagenten auf geeignete Weise berücksichtigt und koordiniert werden.

Dabei kann es hilfreich sein, den Geschäftsagenten holonisch zu strukturieren. Hierzu stehen einige konzeptionelle Alternativen zur Auswahl (siehe Abschnitt 2.6). Für die betrachtete Anwendungsdomäne scheint es jedoch sinnvoll zu sein, den Geschäftsagenten als Kopf-gesteuerten holonischen Verbund mit einer hierarchisch organisierten Struktur zu realisieren, um die Komplexität, die bei der Koordinierung mehrerer interdependenter Aufgaben entsteht, möglichst effektiv herunter zu brechen und zu verteilen.

So könnte ein holonischer Geschäftsagent (beispielsweise in der Rolle als Produzent) über eine Komponente beziehungsweise ein Sub-Holon verfügen, an das er zu koordinierende Aufgaben delegieren und deren Status kontrollieren kann. Ein entsprechendes Sub-Holon wird im Folgenden *Koordinationsagent* (englisch *Coordination Agent*) genannt.

4.3.1.1 Koordinationsagent

Erhält der Koordinationsagent als Sub-Holon des Geschäftsagenten beziehungsweise Supply-Chain-Agenten von dessen Kopf den Auftrag, eine komplexe Aufgabe zu erfüllen, so wird er diese teilautonom ausführen.

Dies bedeutet, die Aufgabe und ihre Teilaufgaben mit der vom Kopf des Holons zur Verfügung gestellten Ressourcen-Ausstattung so lange autonom zu bearbeiten, bis die Aufgabe gelöst ist oder Konflikte auftreten, für deren Lösung der Sub-Holon der Weisung, Unterstützung oder Koordination des Kopfes oder anderer Teilstrukturen des Geschäftsagenten-Holons bedarf.

Um die ihm vom Kopf des Geschäftsagenten gesetzten Ziele effizient zu erfüllen, verfügt der Koordinationsagent über eine rekursiv aufgebaute holonische Agentenhierarchie, wie sie in Abbildung 26 dargestellt ist.

Agiert der holonische Geschäftsagent auf dem elektronischen Markt als Produzent, so sind vom Koordinationsagenten Beschaffungs- und Verkaufs-Aufgaben und damit verbundene Teilaufgaben zu erfüllen. Oftmals lassen sich dabei übergeordnete komplexe Aufgaben in Aufgabenbündel (englisch *Bundles*) zergliedern. Entsprechende Aufgabenbündel können zum Beispiel aus Beschaffungs-Aufträgen für mehrere im Produktionsprozess benötigte Input-Güter bestehen oder aber aus Transportaufträgen für verkaufte Produkte.

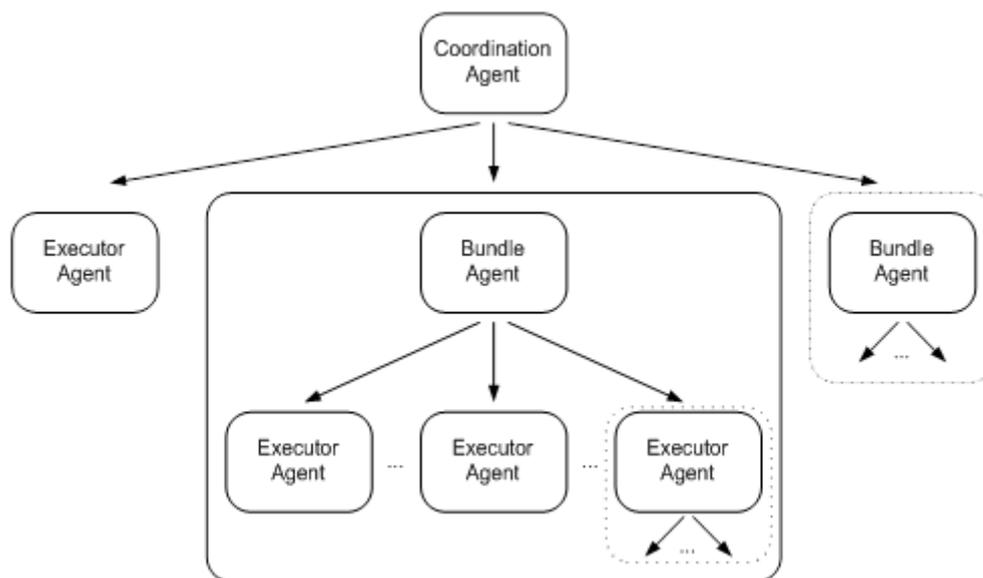


Abbildung 26: Der holonische Koordinationsagent³⁴

4.3.1.2 Bündelagenten

Ist ein Aufgabenbündel zu erledigen, so kann der Koordinationsagent einen wiederum teilautonomen *Bündelagenten* (englisch *Bundle Agent*) damit beauftragen.

Hierzu kann der Bündelagent entweder vom Koordinationsagenten neu kreiert bzw. gestartet werden. Oder ein bereits bestehender Bündelagent mit freien und für die Erledigung der Aufgaben geeigneten Kapazitäten wird vom Koordinationsagenten als Sub-Holon in seine hierarchische Struktur integriert.

Die Bündelagenten können wiederum als hierarchisch organisiertes Holon aufgebaut sein. Je nach dem Komplexitätsgrad der gebündelten Aufgaben kreieren sie entweder weitere Bündelagenten oder aber sogenannte *Ausführungsagenten* (englisch *Executor Agents*).

³⁴ Quelle: [Ruß & Gerber 2001c]

4.3.1.3 Ausführungsagenten

Ausführungsagenten dienen der Erledigung einzelner, nicht mehr sinnvoll auf mehrere untergeordnete Agenten verteilter Aufgaben. Eine solche Aufgabe könnte beispielsweise in einem Beschaffungsauftrag für ein bestimmtes Gut bestehen, inklusive der Teilnahme an einer Auktion oder einer Verhandlung, in der das gesuchte Gut angeboten beziehungsweise verkauft wird.

Ein Ausführungsagent könnte dabei beispielsweise darin spezialisiert sein, für die Beschaffung besonderer Güter auf bestimmten Märkten zu ständig zu sein. Wobei es je nach den Eigenschaften der Umwelt des Geschäftsagenten auch sinnvoll sein könnte, mit einer entsprechenden Differenzierung bereits auf der Ebene der Bündelagenten zu beginnen.

Darüber hinaus kann ein Ausführungsagent auf die Teilnahme an bestimmten Koordinationsmechanismen, insbesondere Auktions- und Verhandlungsmechanismen (siehe u. a. [Vetter 2006; Teuteberg 2007; Büttner 2010; Sackmann 2013]), spezialisiert sein.

Ein entsprechender Agent könnte also wie ein MACSIMA-Agent auf bilaterale Verhandlungen (siehe Abschnitt 5.3) spezialisiert sein und auch über ähnliche Lernfähigkeiten und Adaptionsprozesse verfügen (siehe Abschnitt 5.4). Daher sind die in Kapitel 5 erarbeiteten und in Kapitel 8 simulativ evaluierten Konzepte auf die Architektur der Ausführungsagenten übertragbar.

Bietentscheidungen können dann in Abhängigkeit von dem Auktionsmechanismus, den Bietstrategien des Agenten und den dabei von ihm gemachten Erfahrungen getroffen und adaptiert sowie optimiert werden.

4.3.1.4 Informations- und Kontrollfluss

Der Kopf des holonischen Geschäftsagenten erteilt dessen Sub-Holonen Aufträge und Befehle und teilt auch die zu deren Erfüllung notwendigen Ressourcen zu. Von oben nach unten gerichtet (englisch *top-down*) findet daher im Wesentlichen ein Steuerungs- und Kontrollfluss statt.

Von unten nach oben gerichtet (englisch *bottom-up*) dominiert umgekehrt ein Informationsfluss, da die untergeordneten Sub-Strukturen Informationen über den Status der sich in Bearbeitung befindlichen Aufgaben direkt oder indirekt an den Kopf des Holons weitergeben.

Informationen, die von untergeordneten Ebenen nach oben propagiert werden, können von den Bündel- und dem Koordinationsagenten dazu genutzt werden, um die Zuteilung von Aufgaben und Ressourcen an die ihnen untergeordneten Agenten dynamisch anzupassen. Um dies zielgerichtet tun und die Zuteilung ausbalancieren zu können, verfügen die Koordinations- und Bündelagenten über eine Prioritäts-Relation.

Beispielsweise handelt es sich bei dem Beschaffungsauftrag für ein Gut um eine Aufgabe, zur deren Erfüllung die Teilnahme an Verhandlungen oder Auktionen und damit auch finanzielle Ressourcen erforderlich sind. Daher stehen mehrere Beschaffungsaufträge in Bezug auf die nur beschränkt verfügbaren Geldmittel in einem Konflikt zueinander, der durch die Zuweisung von Prioritäten aufgelöst werden kann. Sofern sich die Prioritäten auf den oberen Ebenen der Hierarchie ändern, wird die Zuteilung von Aufgaben und Ressourcen an die Agenten auf den unteren Hierarchieebenen neu angepasst. Die Zuteilung von Aufgaben und Ressourcen erfolgt dabei ausschließlich „top-down“. Die Agenten auf den nachfolgenden Hierarchieebenen verfahren mit den ihnen zugeteilten Aufgaben und Ressourcen analog hierzu.

Der Koordinationsagent und die Bündelagenten versuchen, die unteren Bearbeitungsebenen zu koordinieren und die dort entstehenden Probleme nach Möglichkeit eigenständig, das heißt ohne weitere Einwirkung der jeweils übergeordneten Ebene, zu lösen. Von tieferen Ebenen kommende Informationen geben sie gefiltert und aggregiert nach oben weiter.

Wenn dies alle Hierarchieebenen des holonischen Geschäftsagenten auf geeignete Art und Weise tun, kann Komplexität reduziert und eine Informationsüberflutung auf den oberen, eher strategisch planenden und kontrollierenden Ebenen des Holons vermieden werden.

Ist für den Produktionsprozess des Geschäftsagenten die Beschaffung mehrerer Input-Güter erforderlich, so erhält der Koordinationsagent den Auftrag, ein entsprechendes Bündel an Beschaffungsaufträgen zu erledigen. Er beauftragt damit einen oder mehrere Bündelagenten, die die Aufträge in Teilaufträge und Unteraufgaben zerlegen und diese dann schließlich spezialisierten Ausführungsagenten zuweisen, die sie dann operativ erledigen.

4.3.1.5 Interaktionen auf elektronischen Märkten

Wie die in Kapitel 5 beschriebenen MACSIMA-Agenten streben auch die Ausführungsagenten danach, auf elektronischen Marktplätzen (vergleiche Abschnitt 3.3.3) mit anderen Agenten zu interagieren, um zum Beispiel Kauf- und Verkaufsaufträge zu erledigen.

Ein elektronischer Marktplatz stellt seinen Marktteilnehmern in der Regel einen Verzeichnisdienst zur Verfügung über den geeignete Partner für bilaterale Verhandlungen oder aber Auktionshäuser mit Auktionen für nachgefragte Güter (vergleiche Abschnitt 3.3.4) identifiziert und selektiert werden können. In einer holonischen Koordinationsinfrastruktur wird ein solcher Verzeichnisdienst als holonisch strukturierter *Agent Name Server (ANS)* realisiert, der wie in Abschnitt 5.2.3 beschrieben gegebenenfalls auch die Funktionalitäten eines *Interoperabilitäts-Service-Agenten (ISA)* bieten kann.

Da der Ablauf bilateraler Verhandlungen in MACSIMA in Kapitel 5 detailliert beschrieben wird, wird im Folgenden erläutert, wie die Teilnahme eines Geschäftsagenten beziehungsweise SCGA-Agenten und seiner Ausführungsagenten an vom holonischen Koordinationsserver realisierten Auktionen abläuft.

Im Falle einer holonischen Koordinationsinfrastruktur sucht der Ausführungsagent des SCGA über den ANS nach Auktionshäusern, die auf von ihm gesuchte Güter oder von ihm präferierte Auktions-Mechanismen spezialisiert sein können. Sie können aber auch, ähnlich wie die der Koordinationsagent des SCGA mehrere Auktionsformen oder Güter-Kategorien bündeln.

Entsprechende Auktionshäuser werden innerhalb der Koordinationsinfrastruktur auch *Koordinationsserver* genannt. Ihre Schnittstelle zur Außenwelt wird über einen *Coordination-Matchmaker-Agenten (CMM)* realisiert. Entsprechende agentenbasierte Koordinationsserver stellen wie der SCGA und dessen Koordinationsagent ein Holon dar und steuern und kontrollieren einen oder mehrere *Koordinationsmechanismusagenten* beziehungsweise *Coordination Mechanism Agents (CMech)*, die wiederum alle Auktionen steuern und kontrollieren, die mit einem bestimmten Auktionsprotokoll durchgeführt werden. Die einzelnen Auktionen werden von *Koordinationsausführungsagenten* beziehungsweise *Coordination Execution Agents (CX)* durchgeführt.

Hat ein Supply-Chain-Agent über den Agent Name Server einen geeigneten CMM-Agenten identifiziert, so fragt er bei diesem bezüglich einer laufenden oder bald startenden Auktion für das benötigte Gut nach. Je nach Realisierung des CMM-Agenten stellt dieser entweder lediglich eine Matchmaking-Funktion zur Verfügung, über die er die Anfrage an geeignete,

unter Umständen auch relativ weitgehend autonom agierende CX-Agenten weiterleitet, die dann direkt mit dem SCGA interagieren.

In diesem Fall wäre der Koordinationsserver-Holon als eine Gruppe autonomer kooperierender Agenten (vergleiche 2.6.3.1) realisiert, die daher auf eine Hierarchiestufe, wie sie von CMech-Agenten realisiert wird, verzichten kann.

Oder aber der CMM ist analog zum SCGA streng hierarchisch strukturiert, so dass der weitere Informations- beziehungsweise Kommunikationsfluss zwischen CX- und SCGA-Agent von ihm selbst oder (zur Reduktion von Komplexität) von ihm untergeordneten CMech-Agenten kontrolliert wird. Der beschriebene Ablauf, der aufgrund der Flexibilität der zugrundeliegenden holonischen Konzepte Freiheitsgrade aufweist, ist in Abbildung 27 veranschaulicht.

Ein Freiheitsgrad besteht beispielsweise darin, dass auch im Fall eines hierarchisch strukturierten CMM-Holons die direkte Kommunikation zwischen seinen Sub-Holonen (also CMech- oder CX-Agenten) und Geschäftsagenten (oder ihren Ausführungsagenten) möglich ist, wenn dadurch Kommunikationsflaschenhalse vermieden werden können. Dieser Aspekt wird in der Folge noch genauer beleuchtet.

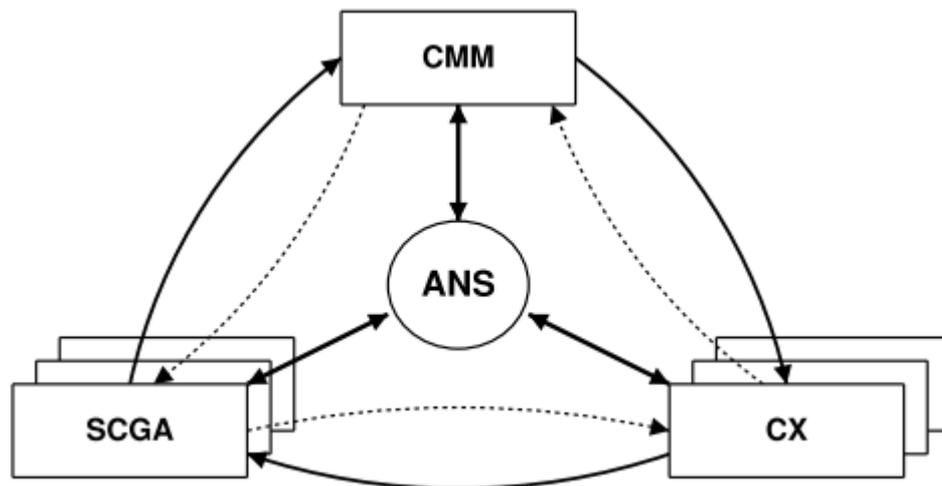


Abbildung 27: H-MAS basierte Koordinations-Infrastruktur für elektronische Märkte³⁵

Im folgenden Abschnitt wird beschrieben, wie ein hierarchisch organisierter holonischer Koordinationsserver aufgebaut ist und arbeitet.

4.3.2 Der holonische Koordinationsserver

Der Koordinationsserver stellt eine generische Plattform dar, die den Geschäftsagenten beziehungsweise Supply-Chain-Agenten verschiedene Koordinationsmechanismen zur Verfügung stellt. Die Teilnahme an diesen Koordinationsmechanismen erlaubt es den Agenten, sich dezentral zu koordinieren.

Eine besondere Art von Koordinationsmechanismen (vergleiche [Hehenkamp 2007; Hülsmann 2008; Vorobeychik 2008; Kolesnewa 2009]) sind Auktionen, die sich gut für die dezentrale marktliche Koordination auf elektronischen Märkten beziehungsweise von Unternehmens- oder Produktionsnetzwerken eignen (siehe [Weinhardt & Gomber 1995; Schimmel & Zelewski 1996; Zelewski 1997; Zelewski 1998; Ruß 1997; Ruß & Vierke 1998b; Ruß & Vierke 1999b; van Dinther 2007; Hertz 2012; Schmidt 2013]).

³⁵ Quelle: [Puin 2004]

Der holonische Koordinationsserver ist agentifiziert und kann mit den Geschäftsagenten kommunizieren. Seine Organisationsstruktur nutzt in der Regel drei Hierarchieebenen, auf denen jeweils unterschiedlich spezialisierte Agenten-Typen operieren. Die Koordinationsmechanismen, die beispielsweise mit verschiedenen Auktionsformen instanziiert werden können, sind in die auf der zweiten oder dritten Hierarchieebene operierenden Agenten-Typen eingebettet.

In [Deßloch 2002] und [Puin 2004] wurden als Auktionsformen beispielsweise die *Englische Auktion*, die *Holländische Auktion*, die *Vickrey Auktion* [Vickrey 1961], die *First-Price Sealed-Bid Auktion*, die *Double Auction* und das *Contract Net Protocol* (vgl. Kapitel 3) implementiert und genutzt.

Die grundlegende holonische Struktur des Koordinationsservers wird durch Abbildung 28 veranschaulicht. Die nächsten Abschnitte dienen der Beschreibung der drei Hierarchieebenen sowie der auf ihnen agierenden Agenten-Typen mit ihren jeweiligen Funktionalitäten.

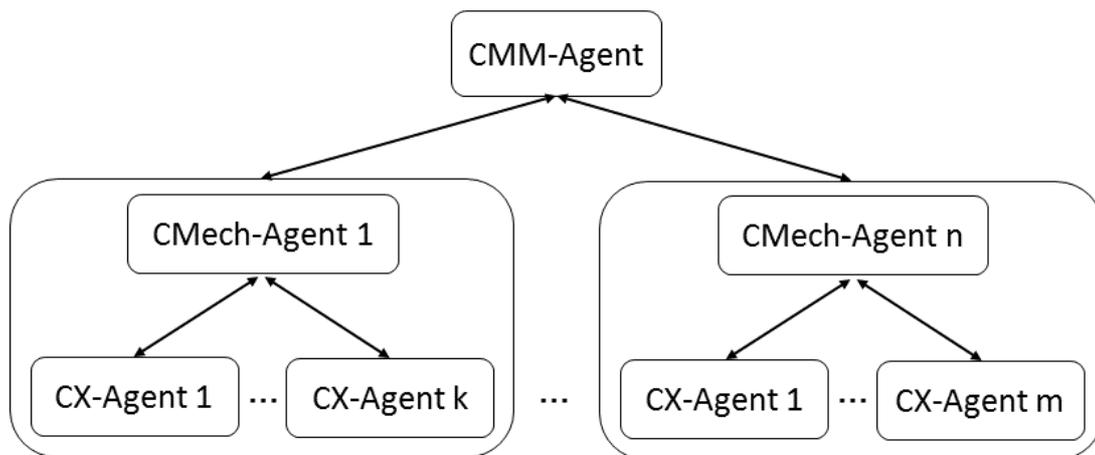


Abbildung 28: Der holonische Koordinationsserver³⁶

4.3.2.1 Koordinations-Matchmaker-Agent

Auf der obersten Hierarchiestufe des Koordinationsservers befindet sich der *Koordinations-Matchmaker-Agent* (CMM). Er steuert die Interaktionen des Koordinationsservers mit den Geschäftsagenten und verarbeitet deren Anfragen und Aufträge. Handelt es sich bei den angebotenen Koordinationsmechanismen um Auktionen, so sammelt der CMM Informationen zu allen momentan laufenden und demnächst startenden Auktionen. Diese werden von den unteren Hierarchieebenen des Koordinationsservers in geeigneter Weise aufbereitet und an den CMM übermittelt. In der Regel sind dem CMM zu jeder Auktion

- der verwendete Auktionsmechanismus,
- das in der Auktion angebotene Gut,
- dessen Menge sowie
- der Start- und Endzeitpunkt,
- der Startpreis,
- der aktuelle Gebotspreis und

³⁶ Quelle: In Anlehnung an [Ruß & Gerber 2001c]

- sonstige auktionsspezifische Details, zum Beispiel einen vom Verkäufer festgesetzten Mindestschritt, um den sich neue Gebote vom aktuellen Gebotspreis unterscheiden müssen,

bekannt. Darüber hinaus weiß der CMM, welche der ihm untergeordneten Agenten die Auktionen jeweils kontrollieren und ausführen.

Kontrolle und Ausführung werden dabei normalerweise auf zwei weitere Hierarchieebenen verteilt, auf denen zwei Agenten-Typen, nämlich die *Koordinationsmechanismusagenten (CMech)* und die *Koordinationsausführungsagenten (CX)* operieren.

4.3.2.2 Koordinationsmechanismus-Agenten

In der Regel befinden sich auf der zweiten Hierarchieebene des Koordinationsservers die CMech-Agenten. Jeder CMech-Agent realisiert ein Auktionsmechanismus-Holon, in dem normalerweise alle Aktivitäten des Koordinationsservers, die sich auf eine bestimmte Koordinations- oder Auktionsform beziehen, gebündelt, kontrolliert und gesteuert werden. Der Informations- und Kontrollfluss erfolgt weitgehend analog zu den oben genannten Bündelagenten der Geschäftsagenten.

Damit der Koordinationsserver eine Vielzahl an parallel ablaufenden Auktionen performant abarbeiten kann, führen die CMech-Agenten die Auktionen üblicherweise nicht selbst aus. Sondern sie besitzen die Fähigkeit, spezielle CX-Agenten mit der operativen Durchführung der Auktionen zu beauftragen. Dabei senden sie entweder an bereits gestartete CX-Agenten, die gerade keine Auktion durchführen, neue Auktionsstart-Aufträge oder sie starten neue CX-Agenten, falls die vorhandenen Kapazitäten bereits ausgelastet sind.

Dabei sind die CMech-Agenten in der Lage, die CX-Agenten und die ihnen zugewiesenen Auktions-Durchführungsaufträge auf mehrere Berechnungs-Plattformen innerhalb eines Netzwerks zu verteilen und ihr Zusammenspiel zu kontrollieren und zu steuern. Auf diese Weise sorgen die CMech-Agenten für eine dynamische Anpassung der Rechen- und Datenhaltungs-Kapazitäten sowie für eine flexible Verteilung der Rechenlast, so dass Ressourcen-Engpässe vermieden und auch bei hoher Aufgabenlast ein stabiles Performanzniveau gewährleistet werden kann.

Will man den Koordinationsserver mit einer flacheren Hierarchie realisieren, so kann unter Umständen auch auf die CMech-Ebene verzichtet werden, indem die Kontroll- und Steuerungsfunktionalitäten der CMech-Agenten in den CMM integriert werden.

4.3.2.3 Koordinationsausführungsagenten

Auf der dritten Hierarchieebene befinden sich normalerweise die CX-Agenten, die die Auktionen operativ durchführen und an die sie kontrollierende Hierarchieebene berichten. Je nach der Art ihrer Realisierung können sie dazu befähigt sein, nicht nur eine, sondern mehrere Auktionen der gleichen Auktionsform parallel zueinander durchzuführen.

Die CX-Agenten geben Informationen über den Status ihrer Auktionen, ihre Aufgabenlast und ihre Versorgung mit Ressourcen an die ihnen übergeordneten Ebenen des Koordinationsserver-Holons weiter. Nach der Beendigung einer Auktion geben sie statistische Informationen zum Verlauf der Auktion sowie Daten zum Auktionsgewinner und dem Endgebot nach oben weiter. Auf Basis dieser Informationen steuern die übergeordneten CMech- und CMM-Agenten dann die Verteilung von Aufgaben und Ressourcen innerhalb des Holons. Stehen für einen CX-Agenten momentan keine weiteren Aufgaben zur Erledigung an, so

kann er von dem ihn kontrollierenden Agenten eingefroren oder terminiert werden. Der Lebenszyklus der CX-Agenten kann also im Gegensatz zu den CMM- und CMech-Agenten beschränkt sein, da sie von diesen dynamisch erschaffen und beendet werden können.

4.3.2.4 Interaktionen und Prozesse

Der CMM stellt den Kopf des Koordinationsserver-Holons dar, repräsentiert den Holon nach außen und dient für die Interaktionen mit den Geschäftsagenten als primäre Kommunikations-Schnittstelle. Seine Datenbasis wird anhand der Status- und Statistikinformationen, die er von den CMech- und CX-Agenten erhält, ständig aktualisiert.

Wollen Geschäftsagenten Auktionen starten, so leitet er die Erfüllung dieser Aufgaben in die Wege und sorgt im Zusammenspiel mit den CMech-Agenten dafür, dass die dafür notwendigen Prozesse und Agenten gestartet und Ressourcen bereitgestellt werden.

Gehen von den Geschäftsagenten Suchanfragen nach passenden Auktionen ein, so beantwortet er diese auf der Grundlage der in seiner Datenbasis verfügbaren Informationen mit einer Liste von Auktionen, die den Suchkriterien entsprechen.

Bei einer streng hierarchischen Organisation des Holons kommunizieren die Geschäftsagenten mit dem Koordinationsserver-Holon ausschließlich über dessen Kopf beziehungsweise den CMM. Es ist allerdings möglich, dass die Anzahl der dort ein- und ausgehenden und zu verarbeitenden Nachrichten ein kritisches Maß erreicht, bei dem der CMM zu einem Kommunikationsflaschenhals wird, durch den die Performanz des Holons gefährdet wird. Das Weiterreichen vieler Nachrichten, insbesondere einer Vielzahl an Geboten und auktionsspezifischer Status-Updates kann darüber hinaus zu hohen internen Kommunikationskosten und einem unnötigen Verbrauch an Ressourcen führen.

Daher wurde in den Implementierungen des Koordinationsservers (vgl. [DeBloch 2002, Puin 2004]) die Möglichkeit geschaffen, dass Geschäftsagenten und CX-Agenten auch direkt miteinander kommunizieren können.

Will ein Geschäftsagent eine bestimmte Auktion starten oder an ihr teilnehmen, so teilt der CMM dem Geschäftsagenten die Adresse des zuständigen CX-Agenten mit bzw. stellt einen direkten Kontakt zwischen diesen Agenten her. Spezifische Details zur Auktion kann ein Geschäftsagent dann auch unmittelbar beim CX-Agenten erfragen.

Weitere Interaktionen zur Laufzeit der Auktion, insbesondere Gebotsabgaben und Status-Updates, können dann ebenfalls auf direkte Art und Weise zwischen den CX- und Geschäftsagenten und weitgehend unabhängig vom CMM und dem zuständigen CMech-Agenten erfolgen. Dies führt dazu, dass die CX-Agenten zur Laufzeit einer durchgeführten Auktion innerhalb des Holons einen höheren Autonomiegrad aufweisen.

In diesem Fall muss nicht jedes Detail zum Auktionsverlauf an die übergeordnete Hierarchieebene weiter gegeben werden, so dass der Informationsfluss nach oben stärker gefiltert bzw. auf aggregierte Art und Weise gestaltet werden kann. Dies reduziert die Kommunikationskosten und den Ressourcenverbrauch innerhalb des Holons.

Zusätzlich hat dies den Vorteil, dass die Robustheit des Koordinationsservers gegenüber Störungen und Ausfällen auf der CMech- und CMM-Ebene verbessert wird. Treten Störungen auf der CX-Ebene auf, so können die CMech-Agenten entweder direkt eingreifen, wenn die Status-Updates eines CX-Agenten ausbleiben. Oder aber sie reagieren auf entsprechende Beschwerdemeldungen der Geschäftsagenten an den CMM.

Nach der Beendigung einer Auktion informieren die CX-Agenten sowohl deren Teilnehmer als auch die übergeordneten Hierarchieebenen über das Auktionsergebnis. An die CMech-

und CX-Agenten werden darüber hinaus auch Statistikdaten zum Auktionsverlauf übermittelt, die für statistische Auswertungen genutzt werden können.

Die typischen Interaktionen, die bei der Durchführung einer Auktion auftreten, werden durch das Interaktionsdiagramm in Abbildung 29 veranschaulicht.

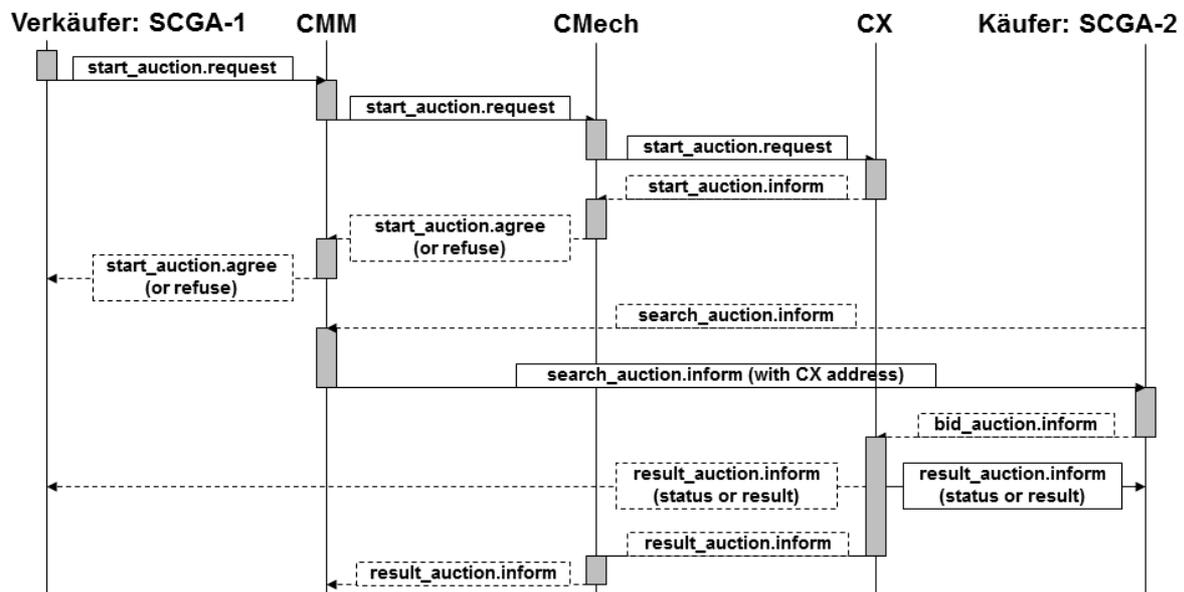


Abbildung 29: Interaktionsdiagramm zum Ablauf einer Auktion

In Abbildung 30 wird der Prozess, den der CMM beim Eintreffen eines Auktionsauftrags von einem Geschäftsagenten X abarbeitet, anhand einer ereignisgesteuerten Prozesskette (EPK) veranschaulicht.

Die dargestellte EPK bezieht sich auf eine flache Realisierungsform des Koordinationssservers mit nur zwei Hierarchieebenen (gemäß Abbildung 27), die von den Agenten-Typen CMM und CX gebildet werden. Die von den CMech-Agenten realisierte Hierarchiestufe wird dabei ausgespart. Die EPK wurde im Rahmen der betreuten Diplomarbeit [Puin 2004] entwickelt und in der Implementierung umgesetzt.

Sobald der Auftrag zum Starten einer Auktion eingeht, wird der zuständige CX-Agent darüber informiert und mit der Ausführung beauftragt. Anschließend werden alle Geschäftsagenten, für die die gestartete Auktion gemäß ihrer Suchkriterien von Interesse ist, vom CMM über den Start der Auktion informiert.

Ein Geschäftsagent, der über den Start der Auktion informiert wurde und an ihr teilnehmen sowie ein Gebot abgeben will, durchläuft einen Prozess, wie er in Abbildung 31 skizziert ist. Im dargestellten EPK startet der Geschäftsagent eine Holländische Auktion (englisch *Dutch Auction*). Es könnte aber auch eine andere Auktionsform sein.

In Abbildung 31 sucht ein Geschäftsagent gezielt nach einer Auktion, die eine der drei Auktionsformen Vickrey Auktion, First-Price Sealed-Bid (FPSB) oder Contract Net Protocol (CNP) verwendet.

Agent: CMM

Role: Receiving an auction request

Plan: newAuction

Action: A blackboard entry is created, noting that a new auction is launched. Agent CX is informed to execute the auction. All agents looking for a matching auction are informed.

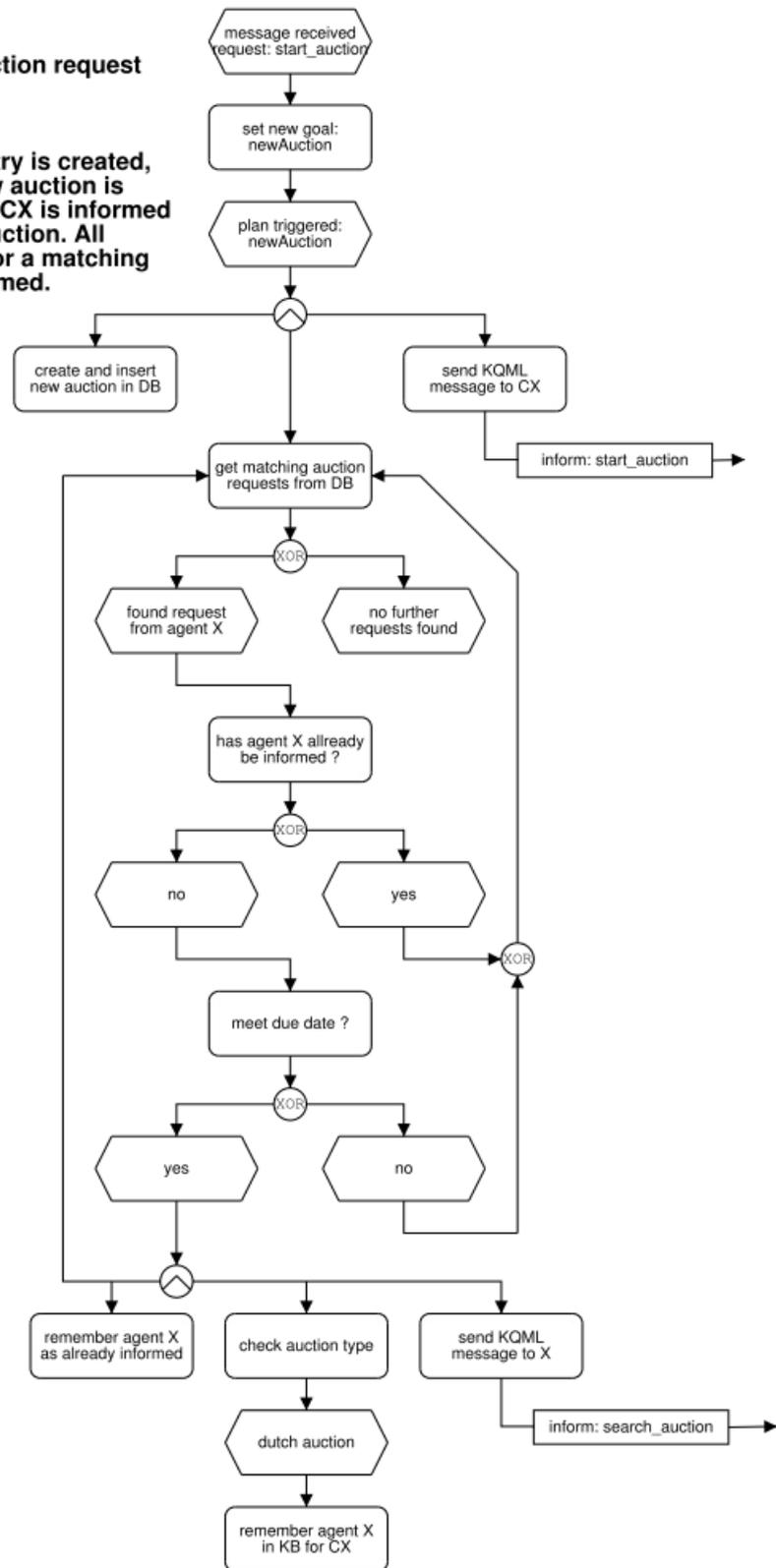


Abbildung 30: EPK zum Start einer Auktion durch den CMM³⁷

³⁷ Quelle: [Puin 2004] (betreute Diplomarbeit)

Sobald er vom CMM (oder wie in Abbildung 31 vom CX) über den Start beziehungsweise das Vorhandensein einer passenden Auktion informiert wurde, entscheidet der Geschäftsagent darüber, ob er ein Gebot beim CX-Agenten abgibt und berechnet dessen Höhe anhand seiner individuellen Prioritäten und seiner Bietstrategie.

Agent: SCGenericAgent
Role: Auction searcher
Plan: newAuction
Action: The agent is informed by CX about a new matching auction. Now the agent waits until the best time to make a bid, decides if he should make one and finally sends the bid.

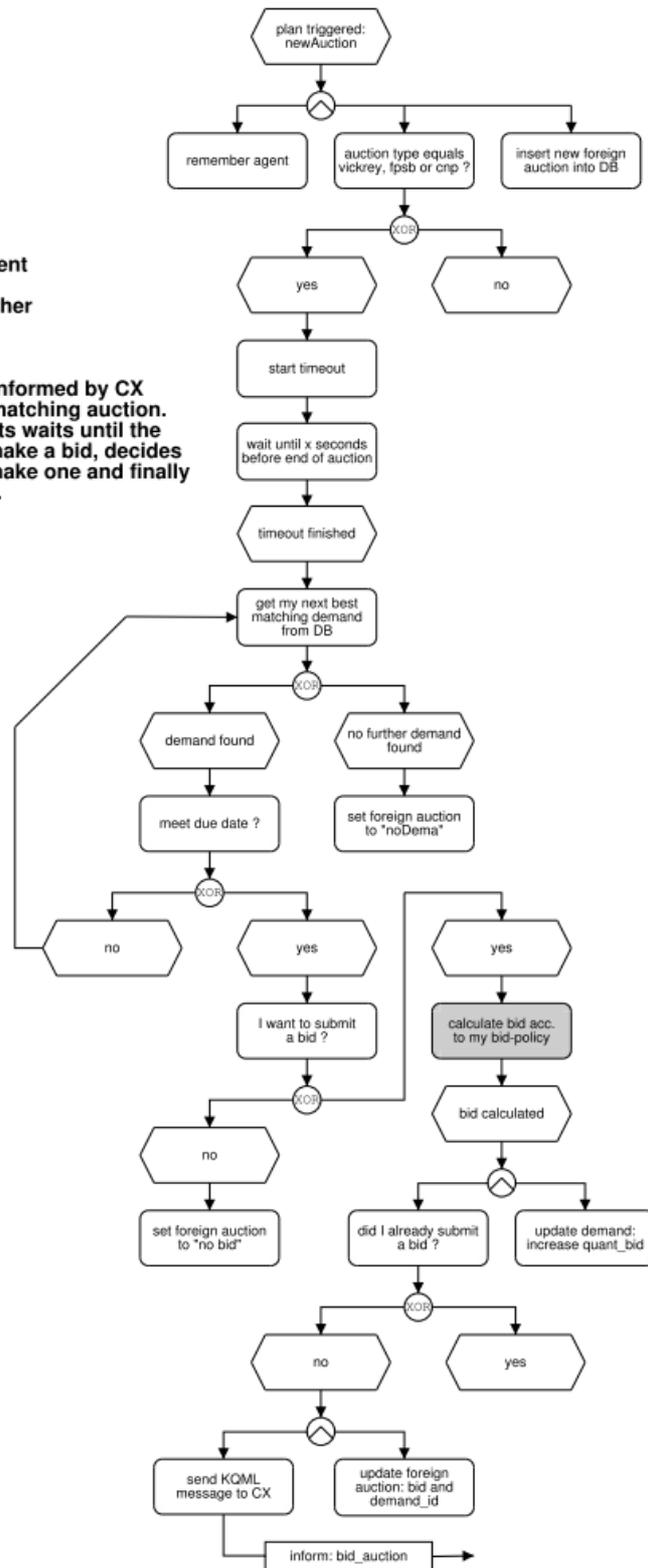


Abbildung 31: EPK zur Abgabe eines ersten Gebots durch einen Geschäftsagenten³⁸

³⁸ Quelle: [Puin 2004] (betreute Diplomarbeit)

In der folgenden Abbildung 32 wird der Prozess dargestellt, den ein CX-Agent durchläuft, wenn er eine Auktion mit verdeckter Gebotsabgabe (beispielsweise eine First-Price Sealed-Bid Auktion oder eine Vickrey Auktion) durchführt beziehungsweise abschließt, nachdem die Auktion ihren definierten Endzeitpunkt erreicht hat.

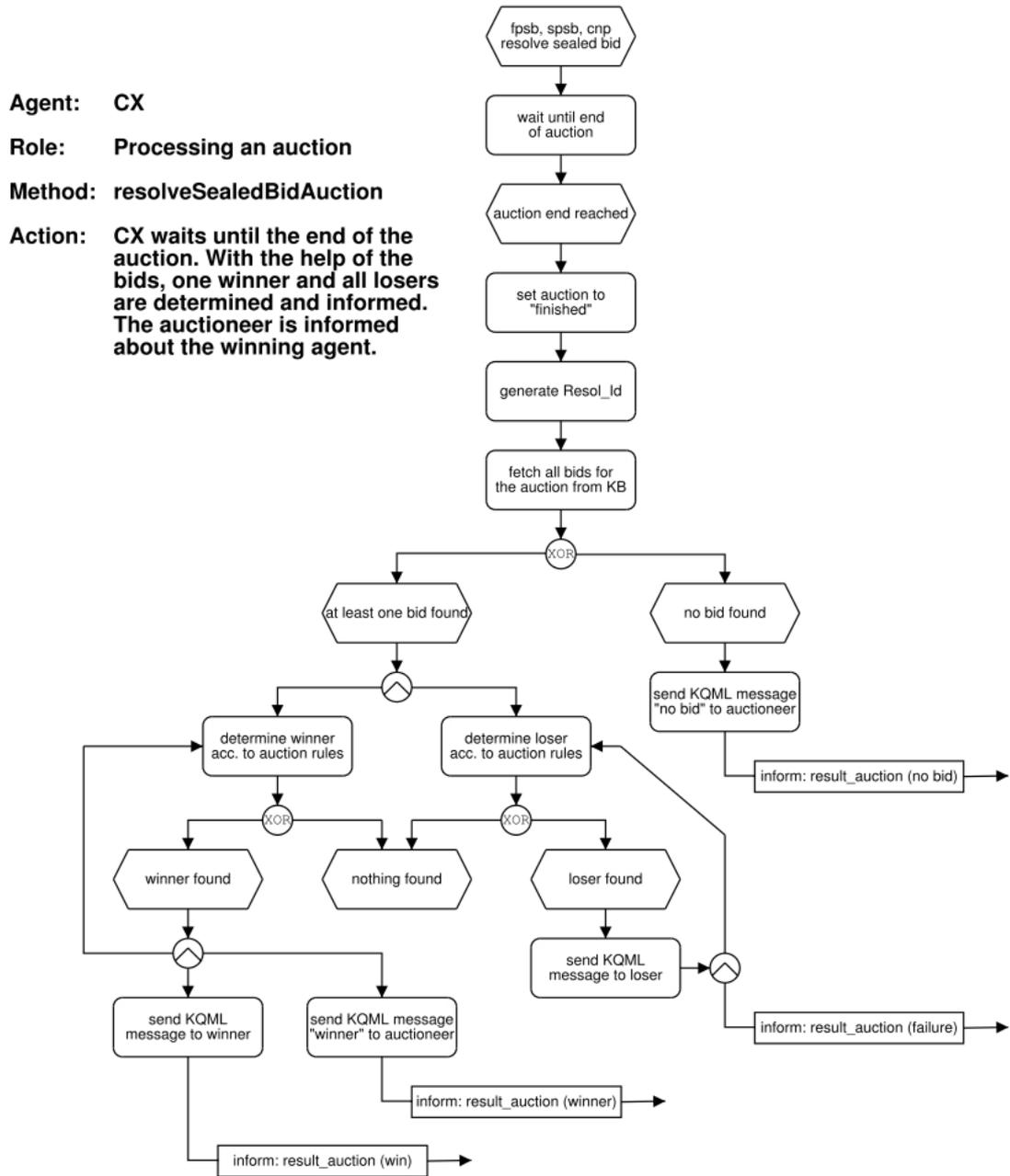


Abbildung 32: Abschluss einer Auktion durch den CX-Agenten³⁹

Kann am Ende der Auktion anhand der Auktionsregeln ein Gewinnergebot ermittelt werden, so werden der Käufer- sowie der Verkäuferagent (bzw. Auktionator) hiervon durch den CX-Agenten informiert, wie es Abbildung 32 zeigt.

³⁹ Quelle: [Puin 2004] (betreute Diplomarbeit)

5 MACSIMA: Ein MAS für die marktliche Koordination von Wertschöpfungsnetzwerken

5.1 Einleitung und Übersicht

Es existiert eine Vielzahl von Forschungsarbeiten, die sich mit dem Einsatz von intelligenten Agenten und MAS in ökonomischen Anwendungsszenarien beschäftigen. Dazu gehören zum Beispiel Anwendungen von MAS im Supply Management (siehe u. a. [Beleva 2007; Chaib-draa & Müller 2006; Ickerott 2007; Castellanos 2012; Dangelmaier et al. 2013; Pip-pow 2013; Teuteberg 2005; Zimmermann 2006]). MAS werden aber auch in Produktion und Logistik (siehe u. a. [Busmann et al. 2013; Marik et al 2009; Marik et al. 2011; Marik et al. 2013]) und auf elektronischen Märkten (siehe u. a. [Büttner 2010; Peters 2013; Sackmann 2013; Teuteberg 2007; Vetter 2006]) eingesetzt. Die für diese Arbeit relevanten Anwendungsszenarien von MAS werden im Detail in Abschnitt 4.2 vorgestellt.

Dennoch liegen bislang nur wenig abgesicherte Ergebnisse dazu vor, welche katallaktischen⁴⁰ Selbstkoordinationsleistungen sich ergeben, wenn der Warenaustausch in B2B-Wertschöpfungsnetzwerken nicht über eine zentrale Instanz (wie zum Beispiel einen Auktionator), sondern von einer größeren Zahl eigennütziger Geschäftsagenten über einen bilateralen Verhandlungsmechanismus vollständig dezentral koordiniert wird.

Wesentliche Beiträge zur Untersuchung von Fragestellungen dieser Art konnte Professor Dr. Torsten Eymann leisten, da sich ein bedeutender Teil seiner Forschungen mit der katallaktischen Selbstkoordination in entsprechenden Wertschöpfungsszenarien beschäftigt. Seine umfangreichen Forschungsarbeiten zu dem genannten Themenkomplex haben sowohl national als auch international Beachtung gefunden und wurden mit dem „Forschungspreis Technische Kommunikation“ der Alcatel-Lucent Stiftung ausgezeichnet.

In den zentralen Arbeiten von Professor Dr. Torsten Eymann (siehe [Eymann 2000; Eymann 2003; Eymann 2013]), die in dem AVALANCHE-System implementiert wurden, nutzen die Agenten zur marktlichen Selbstkoordination innerhalb eines Wertschöpfungsnetzes ein bilaterales Verhandlungsprotokoll, in dessen Verlauf sie auf der Basis von verschiedenen Strategieparametern Entscheidungen treffen. Eine einfache Anpassung ihrer Strategieparameter an sich ändernde Marktbedingungen ist ihnen anhand eines einfachen Adaptionverfahrens (siehe [Smith & Taylor 1998]) möglich.

Zur Beantwortung weiterer Fragestellungen zur Co-Evolution von Preisverhandlungsstrategien und damit einhergehenden katallaktischen Selbstorganisationsprozessen innerhalb von Wertschöpfungsnetzen wurde die Multiagenten-Simulationsumgebung *MACSIMA (Multi Agent Supply Chain SIMulation Framework)* realisiert.

Beiträge zur Entwicklung des MACSIMA-Systems wurden im Rahmen der betreuten Diplomarbeiten [Zenner 2004; Graeber 2006; Walz 2007] geleistet. Mit dem System erzielte Forschungsergebnisse wurden auf mehreren Konferenzen und Workshops präsentiert und veröffentlicht (siehe [Ruß & Walz 2009a; Ruß & Walz 2009b; Ruß & Walz 2009c; Ruß &

⁴⁰ Der Begriff der „Katallaktik“ beziehungsweise „Katallaxie“ wird in Abschnitt 3.3.2 erläutert.

Walz 2009d; Ruß & Walz 2012]). Darüber hinaus wurde in den auf den *ATOP* Workshops der Jahre 2009 und 2010 basierenden *Lecture Notes in Business Information Processing (LNBIP 98)* des Springer-Verlags ein Buchkapitel zu MACSIMA veröffentlicht (s. [Ruß & Walz 2012]).

Wesentliche Erweiterungen, die in der Multiagenten-Simulationsumgebung MACSIMA erstmals in dieser Kombination umgesetzt wurden, sind die Ausstattung der Agenten mit:

- einem adaptiven, evolutionär lernenden Verhandlungsmodul, das es den Agenten ermöglicht, aus in Verhandlungen gemachten Erfahrungen zu lernen und so ihre Verhandlungsstrategien an die sich im Netzwerk dynamisch und nichtdeterministisch ändernden Rahmenbedingungen anzupassen.

Die verwendeten *evolutionären Algorithmen (EA)* verwenden dabei viele verschiedene genetische Operatoren der Phasen *Selektion*, *Rekombination* und *Mutation* eines EA (siehe Abschnitt 5.4.3) sowie zwei unterschiedliche Ersetzungsschemata. Einführungen zu EA⁴¹ finden sich in [Davis 1991; Nissen 2013; Pohlheim 2013; Weicker 2007; Bibel et al. 2013; Weicker 2015].

Neben aus der Literatur bekannten genetischen Operatoren, die in MACSIMA implementiert wurden, wurde mit der *Deterministic Minority Selection (DMS)* eine neuartige Selektions-Methode entwickelt und in drei leicht unterschiedlichen Varianten (DMS, DMS2 und DMS3) implementiert. Die Entstehung der DMS-Varianten hatte den Zweck, die Adaption der Agenten zielgerichteter zu gestalten, ohne dabei zu viel an der im Gen-Pool gespeicherten Information zu verlieren (siehe auch Abschnitt 5.4.3.2).

Für jeden der genetischen Operatoren sind dabei über die GUI des MACSIMA-System jeweils mehrere Methoden einstellbar und fein parametrisierbar. Dabei können die Lernfähigkeiten verschiedener Agentengruppen innerhalb eines Simulationsszenarios auch individuell unterschiedlich parametrisiert werden.

- mehreren neuen und in das Gesamtkonzept integrierten Fitness-Berechnungsmethoden zur nutzenorientierten Steuerung der Adaptionprozesse der Agenten.
- sozialen Fähigkeiten zum Austausch von auf dem Markt gemachten Erfahrungen und Preisinformationen innerhalb von Gruppen koalierender Agenten.

Dadurch kann die Evolution der Verhandlungsstrategie eines Agenten nicht nur allein durch seine eigenen Erfahrungen geleitet, sondern zudem auch durch die Einbeziehung der Erfahrungen anderer Agenten beeinflusst werden, so dass sie zielgerichteter und effizienter erfolgen kann. Auch hierzu sind umfangreiche Möglichkeiten der Feineinstellung gegeben. Insbesondere kann der Austausch von Informationen zwischen Agenten wesentlicher detaillierter eingestellt werden.

- In MACSIMA können Agentengruppen definiert werden, mit denen sich ein Agent austauscht, wohingegen im AVALANCHE-System lediglich bei Benutzung des *STDEA-Verfahrens* (siehe Abschnitt 5.4.6 sowie [Smith & Taylor 1998]) bei jedem Verkaufsabschluss eine Information an einen einzigen, zufällig ausgewählten Agenten des Szenarios gesendet werden konnte.

⁴¹ Dawid erläutert in [Dawid 1995] den Einsatz von GA in ökonomischen Systemen und fokussiert in [Dawid 1999] auf die EA-gestützte Realisierung von adaptivem Lernen. [Vogt 2010] beschäftigt sich mit dem Einsatz von EA in einer Supply Chain. [Dasgupta & Michalewicz 1997] behandeln die Nutzung von EA in ingenieurwissenschaftlichen Anwendungen. [Klügl 2004; Smith et al. 1999] setzen EA zur Adaption in MAS ein.

- In MACSIMA können nicht nur Gruppen von miteinander Informationen austauschenden Agenten bestimmt werden, sondern diese können auch mehrere *Informationsaustausch-Modi* nutzen (siehe Abschnitt 5.4.1.3). Für Agenten innerhalb einer Gruppe gibt es diesbezüglich drei Parametrisierungsmöglichkeiten. Bei der Einstellung *NoExchange (N)* versenden sie überhaupt keine Informationen an andere Agenten. Wird die Einstellung *Everybody (E)* verwendet, so versenden sie standardmäßig Informationen an alle Agenten im MAS. Ist der Parameter *In-Group (G)* gesetzt, so senden die Agenten Informationen ausschließlich innerhalb ihrer eigenen Agentengruppe.
- Darüber hinaus kann das Ausmaß des Informationsaustauschs für das gesamte MAS inklusive all seiner Gruppen anhand einer Informationsaustausch-Quote global definiert werden. Dadurch kann feinjustiert werden, mit welchem Prozentsatz der Mitglieder seiner Gruppe ein Agent in welcher Häufigkeit Erfahrungswerte austauscht (siehe Abschnitt 5.4.1.3).
- Zusätzlich kann zwischen dem Austausch von Informationen und deren Nutzung differenziert werden (siehe Abschnitt 5.4.1.4). So können auch Agenten modelliert werden, die anderen altruistisch Informationen zukommen lassen, aber selbst nicht von den Informationen anderer profitieren können beziehungsweise dies nicht wollen. Letzteres könnte beispielsweise der Fall sein, wenn ein Agent die Korrektheit der übermittelten Informationen anzweifelt, da er entweder den die Informationen sendenden Agenten misstraut oder die Informationsübermittlung den Inhalt der Information verfälschen könnte.
- der Möglichkeit, die eigenen (internen) und/oder die von anderen Agenten (das heißt von extern) stammenden aktuellen Erfahrungen und Marktpreisinformationen in zwei voneinander getrennten Gen-Pools für den Einkauf und Verkauf zu verwalten. Jeder der Agenten kann dann seine Lernfähigkeiten auf beide Gen-Pools separiert voneinander anwenden und dadurch seine Verhandlungsstrategien für den Einkauf und Verkauf getrennt voneinander adaptieren beziehungsweise optimieren.
- korrigierenden Elementen für den Einsatz von in der Form von Genotypen spezifizierten Verhandlungsstrategien, wenn sich das erlernte strategische Verhalten eines Agenten in eine Richtung entwickelt, die seinen wirtschaftlichen Erfolg gefährdet. Hierdurch können Phasen, in denen ein Agent durch ungeschicktes Verhalten keine Abschlüsse erzielen kann, minimiert und die Handelsfrequenz sowie der Gesamtdurchsatz des Gesamtsystems nutzenfördernd gesteigert werden. Hierzu zählen
 - das *Highest-Fitness-First-Verfahren* (siehe Abschnitt 5.4.5.1),
 - die Nutzung des Verfahrens der *Ticket-Limitierung* (siehe Abschnitt 5.4.5.2) sowie
 - ein *Preisautomat* (siehe Abschnitt 5.3.3.7), der, falls ein Agent die aktuelle Marktsituation beziehungsweise die momentan am Markt akzeptierten Güterpreise falsch einschätzt, dessen Preisvorstellungen auf sinnvolle Weise schrittweise an die aktuellen Gegebenheiten anpasst. Die Evaluation dieser Agentenkomponente in Kapitel 8 zeigt, dass hierdurch die Relation von erzielten Verkaufsabschlüssen zu Verhandlungsabbrüchen häufig signifikant verbessert werden kann.

Neben den genannten Erweiterungen enthält das MACSIMA-System alle wesentlichen Konzepte und Funktionalitäten des an der Universität Freiburg entwickelten AVALANCHE-Systems (vergleiche [Eymann & Padovan 1999; Eymann 2000]).

Die Entwicklung von MACSIMA orientierte sich anfangs an den Quellen des Multiagenten-Systems B2B-OS, das aus der Entwicklung von AVALANCHE hervorgegangen ist und ebenfalls an der Universität Freiburg entwickelt wurde. Die entsprechenden JAVA-Quellen wurden in [Eymann 2003] auf einer dem Buch beiliegenden CD-ROM veröffentlicht. Zudem ist unter <http://b2bos.sourceforge.net> ein Download der Software möglich. Bei B2B-OS handelt es sich um eine freie Software, die unter den Bedingungen der GNU General Public License modifiziert werden kann.

Darauf aufbauend wurden bei der Entwicklung von MACSIMA so umfangreiche Überarbeitungen und Erweiterungen vorgenommen, dass es sich bei MACSIMA um ein Software-System handelt, das sich strukturell und vom Funktionsumfang her in weiten Bereichen von B2B-OS differenziert. Dennoch enthält das MACSIMA-System auch einige Pakete und Klassen, die nur mit geringen Modifikationen von B2B-OS übernommen wurden (zum Beispiel die Klassen zur Realisierung des STDEA-Verfahrens). Dadurch wurde sichergestellt, dass bestimmte Eigenschaften des AVALANCHE-Systems in MACSIMA unverändert erhalten blieben.

Werden die in MACSIMA realisierten Erweiterungen abgeschaltet, so zeigen die MACSIMA-Agenten ein zu den Agenten in AVALANCHE vergleichbares Verhalten. Es können auch die gleichen katallaktischen Selbstorganisations-Effekte beobachtet werden, was in Abschnitt 8.2 des Evaluationsteils dieser Arbeit in einem qualitativen Vergleich noch einmal verdeutlicht wird. Werden die Erweiterungen von den Agenten hingegen genutzt, so zeigt das MACSIMA-System in Bezug auf seine Selbstkoordinationsleistungen wesentliche Verbesserungspotenziale, was in Kapitel 8 vertieft dargelegt wird.

MACSIMA ist mit den genannten Eigenschaften über ein GUI bedienbar, mit dem Simulationsszenarien für die verschiedensten Fragestellungen leicht entworfen werden können, wie in Kapitel 7 erläutert wird. Hierdurch erlaubt es MACSIMA, den Einfluss von unterschiedlichen Parametrisierungen (insbesondere für das Verhandlungsmodul und die Lernprozesse der Geschäftsagenten) auf die Koordinationseffizienz und die Gesamtperformanz von Wertschöpfungsnetzwerken zu simulieren und zu untersuchen.

Zudem besteht ein wesentlicher Vorteil der MACSIMA-Simulationsumgebung darin, dass sie umfangreiche Möglichkeiten des Szenario-Topologie-Designs zur Verfügung stellt. So können mit MACSIMA beinahe beliebige Topologien beziehungsweise Strukturen von Wertschöpfungsnetzwerken konstruiert und mit einer Vielzahl von intelligenten Agenten instanziiert werden. Auf diese Weise können von einem Experimentator oder MAS-Entwickler grundlegende Design-Aspekte für den Aufbau und die Koordination von agentenbasierten Wertschöpfungsnetzwerken untersucht werden und so Fehlentscheidungen vermieden werden, die bei einer Instanziierung in der realen Welt zu einer suboptimalen Performanz mit entsprechenden wirtschaftlichen Konsequenzen führen würden.

Ein weiterer wichtiger Beitrag besteht darin, die Analyse des Agentenverhaltens nicht nur qualitativ, sondern darüber hinaus auch quantitativ durchführen zu können, wie es in den Kapiteln 6 und 7 im Detail erläutert wird. Die hierfür entwickelte Evaluationsmethodik bietet durchdachte Konzepte, die so in den dem Autor bekannten Veröffentlichungen zu dem bearbeiteten Themenkomplex nicht zu finden waren. So kann die Auswertung und statistische Analyse der Simulationsergebnisse (unter anderem bzgl. der Kennzahlen Gesamtgewinn, Gesamtumsatz, Anzahl erfolgreicher oder abgebrochener Verhandlungen etc.) für einzelne Agenten (Mikro-Ebene), Agenten-Gruppen (Meso-Ebene) und für das Gesamtsystem (Makro-Ebene) durchgeführt werden. Damit kann vorher analysiert werden welche Auswirkungen unterschiedliche Szenario-Parametrisierungen für die gesamte Supply Chain, einzelne Gruppen aber auch jeden einzelnen Agenten haben.

Die Möglichkeit zur Parametrisierung der Agenten-individuellen Lernmechanismen und der Art des Informationsaustauschs gestattet es, die Effekte verschiedener Lernmechanismen in Kombination mit verschiedenen Informationsaustausch-Modi unter denselben externen Einflüssen und Beschränkungen in einem Wertschöpfungsnetzwerk zu untersuchen. Zusätzlich ist der Experimentator im Stande, auf jeder Ebene eines Wertschöpfungsnetzwerks kooperierende Gruppen aus mehreren Agenten zu definieren und auf diese Weise die Effekte der Koalitionsbildung zwischen mehreren Agenten zu untersuchen. Entsprechende Analysere-sultate können anschließend dazu genutzt werden, geeignete Empfehlungen für das Design agentenbasierter Wertschöpfungsnetzwerke zu geben.

Eine wesentliche mit MACSIMA bearbeitete Forschungsfrage war, ob sich die oben genannten Erweiterungen auf die Selbstkoordinationsfähigkeit und die Performanz eines agenten-basierten Wertschöpfungsnetzes positiv beziehungsweise effizienz- und nutzensteigernd auswirken – sowohl isoliert betrachtet, als auch in Ihrer Gesamtheit. Für die detaillierte Beantwortung dieser Frage wird auf den Evaluationsteil in Kapitel 8 verwiesen. Es kann jedoch an dieser Stelle bemerkt werden, dass durch die Kombination der vorgenannten Erweiterungen und die Feinjustierung des verwendeten Lernverfahrens die Systemperformanz in Bezug auf die Kennzahlen Gesamtumsatz und –gewinn bis hin zum ungefähr Zehnfachen gesteigert werden konnte, was für eine signifikant verbesserte Selbstkoordinationsleistung der Agenten spricht.

Im Folgenden werden die Simulationsumgebung MACSIMA und die ihr zugrunde liegenden Konzepte inklusive des eingesetzten Verhandlungsprotokolls, des von den Agenten genutzten adaptiven Verhandlungsmoduls und der für die Realisierung ihrer auf evolutionären Algorithmen basierenden Lernfähigkeiten verwendeten Komponenten beschrieben.

Es wird erläutert, welche Möglichkeiten zur Definition von zu simulierenden Szenarien zur Verfügung stehen. Zudem wird die Systemarchitektur von MACSIMA mit der Realisierung des Marktplatzes und den instanziierten Agenten-Typen beschrieben.

Anschließend wird die Realisierung der Verhandlungen, das Verhandlungsprotokoll sowie die darauf basierenden Verhandlungsstrategien beschrieben.

Darauf folgt die Beschreibung des adaptiven Verhandlungsmoduls, mit dem die Agenten ihre Verhandlungsstrategien im Verlauf einer Simulation an die sich nichtdeterministisch und dynamisch ändernden Marktumstände anpassen.

Das Kapitel schließt mit einer detaillierten Beschreibung der Komponenten, Methoden und Verfahren sowie den für diese gegebenen Einstellmöglichkeiten, über die MACSIMA verfügt. Auf dieser Grundlage können die Lernfähigkeiten verschiedener Agenten und ihre zyklischen Adaptionsprozesse mit einer Vielzahl an Gestaltungsmöglichkeiten realisiert und vielfältige Simulationsszenarien definiert werden.

5.2 Die Architektur von MACSIMA

Die Multiagenten-Simulationsumgebung MACSIMA ist in Java implementiert. Mit ihr können Wertschöpfungsnetzwerk-Szenarien definiert und simuliert werden, deren Aufbau im folgenden Abschnitt erläutert wird.

Zur Instanziierung entsprechender Szenarien bietet MACSIMA eine Grundmenge generischer Typen von Geschäftsagenten an, die in Abschnitt 5.2.2 beschrieben werden.

Diese Geschäftsagenten verfügen sowohl über die Fähigkeit, an bilateralen Verhandlungen auf einem elektronischen Markt teilzunehmen, als auch über Lernfähigkeiten, die es ihnen ermöglichen, aus ihren Verhandlungserfolgen und -misserfolgen zu lernen.

Wie der elektronische Markt im MACSIMA-System realisiert ist, wird in Abschnitt 5.2.3 erläutert. Der Kontrollzyklus der Geschäftsagenten wird in Abschnitt 5.2.4 dargelegt.

5.2.1 Aufbau von Simulationsszenarien

MACSIMA erlaubt das Design von Wertschöpfungsnetzwerk-Topologien, in denen jede Wertschöpfungsstufe mit einer frei wählbaren Anzahl von autonomen, eigennützig handelnden Geschäftsagenten instanziiert werden kann.

MACSIMA ermöglicht es, Simulationsszenarien mit hohen Freiheitsgraden zu definieren bzw. auf sehr detaillierte Weise zu parametrisieren. Der Benutzer ist völlig frei darin, sich für

- a) eine bestimmte Struktur des Topologie-Graphen,
- b) eine bestimmte Zahl von Wertschöpfungsstufen und
- c) eine bestimmte Zahl von konkurrierenden Agenten auf jeder Stufe

zu entscheiden und dadurch auch sehr komplexe und umfangreiche Netzwerke zu definieren, indem er die vorhandenen generischen Bausteine miteinander kombiniert (Beispiel siehe Abbildung 33).

Ein zu simulierendes Wertschöpfungsnetz lässt sich auf Basis von vier generischen Arten von Geschäftsagenten modellieren, nämlich *Ressourcen-Agenten* R_i , *Produzenten-Agenten* P_i , *Händler-Agenten* T_i und *Konsumenten-Agenten* C_i (siehe Abschnitt 5.2.2). Diese Agenten

- verkaufen und liefern *Güter* G_i (beziehungsweise Ressourcen) oder
- handeln mit diesen Gütern oder
- kaufen (*Input-*)*Güter* und produzieren aus ihnen neue (*Output-*)*Güter* (beziehungsweise *Produkte*), die sie weiter verkaufen oder
- konsumieren die Endprodukte des Netzwerks als *Konsumgüter*.

Die Output-Güter einer Stufe stellen jeweils die Input-Güter der jeweils nächsten Stufe dar. Die Geschäftsagenten einer Stufe stehen sowohl beim Kauf von Input-Gütern einer vorgelagerten Wertschöpfungsstufe als auch beim Verkauf von Output-Gütern an eine nachgelagerte Stufe zueinander in Konkurrenz.

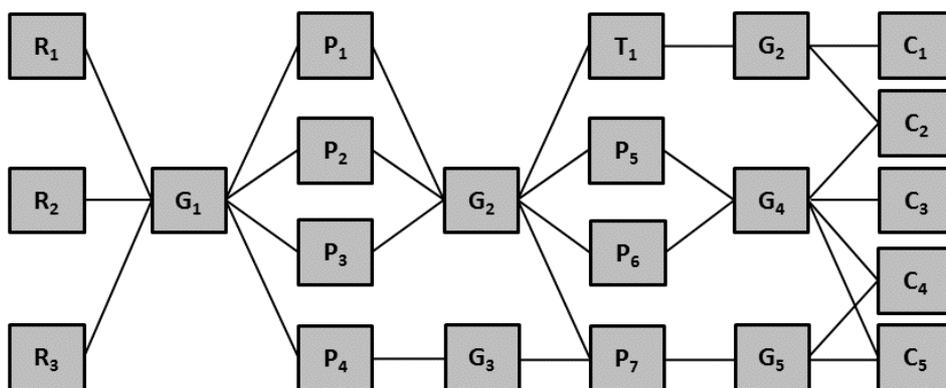


Abbildung 33: MACSIMA-Simulationsszenario mit generischen Bausteinen⁴²

⁴² Quelle: In Anlehnung an [Graeber 2006] (betreute Diplomarbeit)

Die genannten generischen Agenten-Typen werden in Abschnitt 5.2.2 genauer beschrieben. Ressourcen und Konsumgüter stellen aus ökonomischer Sicht besondere Arten von Gütern dar, müssen aber für die Zwecke dieser Arbeit nicht gesondert behandelt werden.

Die für den Evaluationsteil dieser Arbeit durchgeführten Simulationenläufe haben sich auf verschiedene Instanzierungen eines fünfstufigen Wertschöpfungsnetzwerks für die Computer-Fertigung mit linearem Güterdurchfluss konzentriert, wie sie in Abbildung 34 skizziert ist.

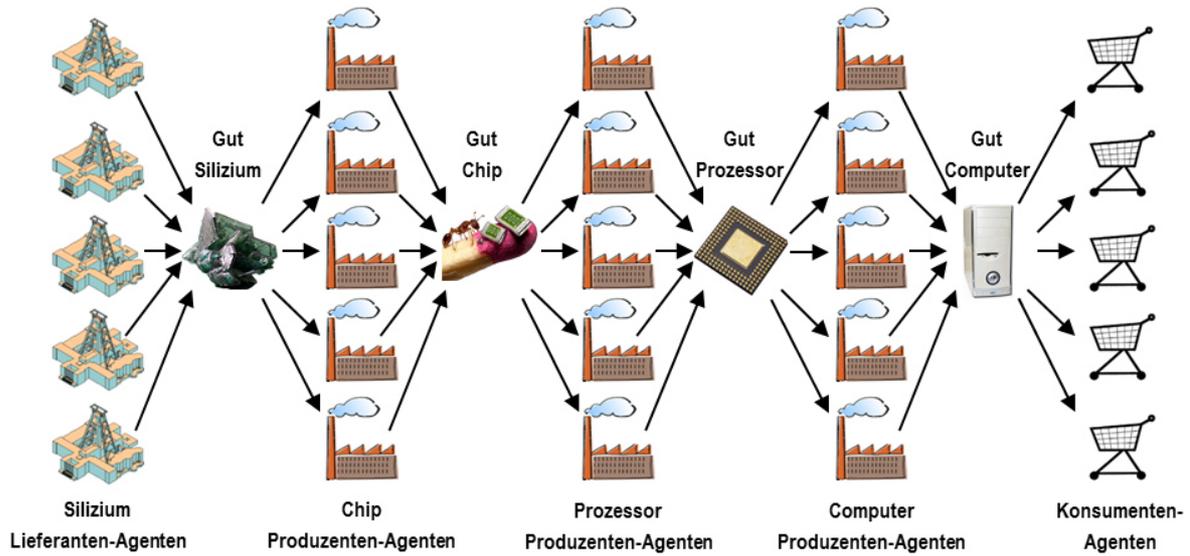


Abbildung 34: Basis-Topologie von Simulationsszenarien für Wertschöpfungsnetzwerke in der Anwendungsdomäne Computer⁴³

Silizium wird als Rohstoff an Agenten verkauft, die Chips produzieren. Diese Chip-Produzenten verkaufen ihre Chips als Zwischenprodukt an Prozessor-Produzenten weiter. Die von diesen hergestellten Prozessoren werden wiederum von Computer-Herstellern nachgefragt. Die produzierten Computer werden in der Basis-Topologie als Endprodukt der Wertschöpfungskette von den Konsumenten erworben und daher nicht weiter im System gehandelt.

File									
Graph	Goods	Suppliers	Deposits	Branches	Strategies	StrategyPools	EvolutionaryAlgorithms	Learning	Start
Simulation rounds:		300000	Start Single Simulation		Simulation and Exit		Directory Simulation		from 1 to 4
Agent	Quantity	GroupID	Information Exchange Type	StrategyPool		Learning			
SupplierSilizium	10	1	InGroup	Random Acq 0,05-0,95		Ext-P3-Pn20-RWS			
DepositChip	10	2	InGroup	Random Acq 0,05-0,95		Ext-P3-Pn20-RWS			
DepositProzessor	10	3	InGroup	Random Acq 0,05-0,95		Ext-P3-Pn20-RWS			
DepositComputer	10	4	InGroup	Random Acq 0,05-0,95		Ext-P3-Pn20-RWS			
BranchComputer	10	5	InGroup	Random Acq 0,05-0,95		Ext-P3-Pn20-RWS			
Add					Remove				

Abbildung 35: Einstellung von Agentengruppen mit der jeweiligen Anzahl von Agenten und deren Informationsaustausch- sowie Lern-Parametrisierung

⁴³ Quelle: [Ruß & Walz 2012] (eigene Veröffentlichung)

Die Wertschöpfung im Wertschöpfungsnetzwerk erfolgt sequentiell und vollkommen dezentral, so dass weder der Silicium-Lieferant weiß, was und wann welcher Chip-Produzent produziert und daher Silicium nachfragt, noch ein Prozessor-Produzent über die Prozesse und Angebotszeitpunkte der Chip-Produzenten Kenntnis hat.

Zum Zeitpunkt des Simulationsstarts werden zunächst nur die in der Definition des zu simulierenden Szenarios spezifizierten Agenten mit den von ihnen nachgefragten und produzierungsbeziehungsweise angebotenen Ressourcen und Gütern instanziiert.

Gemäß der Art und Anzahl der vom Systemdesigner gewünschten Wertschöpfungsstufen werden dabei Agenten verschiedenen Typs instanziiert. Der Designer kann darüber hinaus auswählen, wie viele Instanzen jedes Agenten-Typs im System aktiv sein sollen (siehe Abbildung 35).

Da nur ein Ausschnitt aus der realen Welt betrachtet werden kann, muss der Experimentator zum Simulationsstart zudem spezifizieren, welche Ressourcen in welcher Frequenz von den Ressourcen-Agenten an das Wertschöpfungsnetz lieferbar sind. Näheres zu den Agenten und ihren Versorgungs- und Produktionsbeziehungsweise Nachfragefunktionen wird in Abschnitt 5.2.2 erläutert.

Da bezüglich der Ressourcen und Güter zum Zeitpunkt des Systemstarts noch kein Handel stattgefunden hat, existiert für diese noch kein Preissystem, das sich durch die marktliche Selbstkoordination der Agenten hätte bilden können. Daher enthält ein zu simulierendes Szenario für alle handelbaren Güter (und Ressourcen) neben absoluten Preisunter- und -obergrenzen auch jeweils initiale Marktpreise, an denen sich die Agenten bei ihren ersten Verhandlungen orientieren können. Diese initialen virtuellen Marktpreise (sogenannte *average*-Preise) und Preisschranken aus der Szenario-Definition werden beim Start eines Agenten in dessen Datenstruktur *PriceDistribution* hinterlegt, die in Abschnitt 5.3.3.1 im Detail erläutert wird.

MACSIMA bietet für entsprechende Einstellungen eine grafische Benutzerschnittstelle (GUI), welche nicht nur die Spezifikation von Netzwerk-Topologien, Güter-Arten und Preisinformationen und -schranken vereinfacht, sondern es auch ermöglicht, die Lernfähigkeiten jedes einzelnen Agenten, der in einem Simulationsszenario instanziiert wird, individuell zu parametrisieren. Die instanziierten Agenten lassen sich verschiedenen Gruppen zuordnen und ihren Informationsaustausch sowie ihre Lernfähigkeiten feinjustieren.

Die grafische Benutzerschnittstelle, die in Abschnitt 6.2 ausführlich behandelt wird, bietet eine Vielzahl von Ansichten, um auf diese Einstellmöglichkeiten, zum Beispiel den Graphen des Wertschöpfungsnetzwerks, die nachgefragten und angebotenen Güter, die Verhandlungsstrategien und die Lernfähigkeiten der Agenten (initiale Strategie-Pools, verfügbare evolutionäre Algorithmen, etc.) zuzugreifen und die entsprechenden Parametereinstellungen vorzunehmen.

Tabelle 2 liefert einen Überblick über die wichtigsten Parametrisierungsmöglichkeiten für MACSIMA-Simulationsszenarien. Da es sich um eine Auswahl von Einstellmöglichkeiten handelt, ist dieser Überblick nicht vollständig. Weitere Detail-Einstellungen, die vorgenommen werden können, sind aus den drei Tabellen ersichtlich, die in Abschnitt 6.3 die Notation für die Spezifikation der MACSIMA-Simulationsszenarien zusammenfassen (siehe Tabelle 5, Tabelle 6 und Tabelle 7). Alle in der Notation definierten Parameter lassen sich einstellen und feinjustieren. Eine detaillierte Beschreibung der MACSIMA-Simulationsumgebung und weitere Informationen zu den einstellbaren Simulationsparametern sowie zu den meisten Simulationsläufen zugrundeliegenden Basis-Topologie finden sich in Kapitel 6.2. Die Einstellungen, die zur Parametrisierung der Lernfähigkeiten der Agenten vorgenommen werden können, werden in Abschnitt 5.4 ausführlich beschrieben.

Parametrisierung der Topologie	
<ul style="list-style-type: none"> - Spezifikation der Güter und der Stufen des Wertschöpfungsnetzwerks - Spezifikation der Agenten-Typen sowie der Agentengruppen und der Anzahl ihrer Mitglieder - Modellierung der Wertschöpfungsstruktur (Vernetzung und Angebots- und Nachfragesituation) 	
Parameter für das Marktpreisystem	
Preisinformationen für die Güter	<ul style="list-style-type: none"> - Absolute untere Preisgrenze - Absolute obere Preisgrenze - Durchschnittlicher Marktpreis
Parameter der Agenten	
Agenten-Typen	<ul style="list-style-type: none"> - Ressourcen-Agent - Produzenten-Agent - Händler-Agent - Konsumenten-Agent
Parameter der Verhandlungsstrategie	<ul style="list-style-type: none"> - Acquisitiveness - Satisfaction - Delta_Change - Weight_Memory - Delta_Jump - Reputation
Sonstige Parameter	<ul style="list-style-type: none"> - Produktionsfunktionen - Versorgungs- und Konsumfunktionen - Gewinnvorstellungen - Agenten-Attribute
Parameter des Informationsaustausches	
Informationsaustausch-Modus	<ul style="list-style-type: none"> - No Exchange: kein Informationsaustausch - InGroup: Informationsaustausch innerhalb von Agenten-Gruppen - Everybody: Austausch mit allen Agenten des Szenarios <p>Eine detaillierte Beschreibung der Informationsaustausch-Modi findet sich in Abschnitt 5.4.1.3.</p>
Informationsaustausch-Quote	<p>Für eine Agentengruppe G ist (bei den Informationsaustausch-Modi <i>InGroup</i> und <i>Everybody</i>) einstellbar,</p> <ul style="list-style-type: none"> - in wieviel Prozent F der Fälle eines erfolgreichen Verhandlungsabschlusses ein Mitglied der Gruppe überhaupt dazu bereit ist, Informationen darüber als sogenanntes <i>Plumage</i> (siehe Abschnitt 5.4.1) an die übrigen Gruppenmitglieder weiterzuleiten und - an wieviel Prozent P der übrigen Gruppenmitglieder das Mitglied in diesem Fall dann tatsächlich ein <i>Plumage</i> sendet. <p>Bei der Einstellung $100P100F$ sendet jedes Gruppenmitglied bei jedem seiner Abschlüsse an alle übrigen Gruppenmitglieder ein <i>Plumage</i>.</p> <p>Detaillierte Erläuterungen zur Informationsaustausch-Quote finden sich in Abschnitt 5.4.1.3.</p>
Parameter des Lernprozesses	
Lern-Modus	<ul style="list-style-type: none"> - Internes Lernen - Externes Lernen - Gemischtes Lernen - NoLearning: Lernen ist ausgeschaltet - Smith98: Dezentrales Adaptionsverfahren von [Smith & Taylor1998]
Evolutionäres Lernen	<ul style="list-style-type: none"> - 1-Pool- oder 2-Pool-Modus - Fitness-Berechnungsmethode - Genetische Operatoren <ul style="list-style-type: none"> ▪ Selektion ▪ Rekombination ▪ Mutation ▪ Ersetzungsschema - Definierte Adaptionsverfahren <ul style="list-style-type: none"> ▪ STDEA-Verfahren - Spezialverfahren: <ul style="list-style-type: none"> ▪ Highest-Fitness-First ▪ Ticket-Limitierung

Tabelle 2: Wesentliche Parameter für die Definition von Simulationsszenarien

5.2.2 Generische Agenten-Typen

Die Agenten eines mit MACSIMA simulierten Wertschöpfungsnetzwerks stellen die Teilnehmer eines elektronischen Marktes dar, da sie mit anderen Agenten autonom über die Preise nachgefragter und angebotener Güter verhandeln und sich auf dieser Grundlage marktlich selbst koordinieren. Wie der elektronische Markt realisiert ist, wird im folgenden Abschnitt 5.2.3 erläutert.

Die Agenten versuchen, als eigennützige, ökonomisch handelnde, *nutzenbasierte Agenten* (siehe Abschnitt 2.4.4) ihren eigenen Nutzen zu maximieren, welcher in Geldeinheiten beziehungsweise anhand ihres Gewinns gemessen wird. Jeder Agent verfügt über eine Anfangsausstattung an Geld, die er versucht zu vermehren. Die einzigen Möglichkeiten, Geld zu verdienen, bestehen für einen Agenten auf einer der mittleren Stufen des Wertschöpfungsnetzwerks im Handel oder in der Produktion und dem anschließenden Verkauf von Gütern.

Daher streben die Agenten danach, eine möglichst große Gewinnmarge zu erzielen. Ihr Gewinn ergibt sich, indem von den Verkaufspreisen ihrer Output-Güter deren Herstellungskosten abgezogen werden. Hierbei setzen sich die Herstellungskosten aus den Input-Kosten, das heißt der Summe der Einkaufspreise der für die Produktion benötigten Input-Güter, und den Produktionskosten zusammen. Der einstellbare Parameter *ProfitWish* spezifiziert dabei die Höhe der mindestens gewünschten prozentualen Gewinnmarge, die ein Agent als Verkäufer in einer Verhandlung zu erzielen versucht. Indem der Verkäufer den Wert von *ProfitWish* prozentual auf seine Herstellungskosten für ein Gut aufschlägt, ermittelt er seinen Mindestbeziehungsweise Reservationspreis RVP_{VK} (siehe auch Aktivität 6.a in Abschnitt 5.3.3.6).

Kann ein Verkäufer seinen Reservationspreis RVP_{VK} in einer Verhandlung nicht mehr realisieren, so bricht er die Verhandlung in der Regel ab – es sei denn, der in Abschnitt 5.3.3.7 beschriebene Preisautomat hat den Mindestpreis, den der Verkäufer in der Verhandlung erzielen will, vorher korrigierend auf einen Wert abgesenkt, der unterhalb von RVP_{VK} liegt.

Den Agenten werden zudem pro Simulationsrunde vom Marktplatz konstante Fixkosten (*Cost of life*) abgezogen, was als eine Teilnahmegebühr am Markt plus weiterer Fixkosten angesehen werden kann. Diese Kosten können eingestellt werden und innerhalb jeder Agentengruppe unterschiedlich sein. Sollte ein Agent diese Fixkosten nicht bezahlen können, so wird er insolvent und muss den Marktplatz verlassen. Die einzige Ausnahme sind die Konsumenten-Agenten, die über keine im System generierten Einnahmen verfügen, da sie per Definition nichts verkaufen. Durch den Abzug der *Cost of life* ist eine Strategie des Nichtstuns nicht erfolgreich. Die Agenten werden durch diese Benutzungsgebühr gezwungen, Transaktionen durchzuführen und ihr Vermögen durch erfolgreiches Verhandeln zu mehren.

Ein MACSIMA-Agent zeichnet sich durch einen eindeutigen Namen (*agent_name*) aus und repräsentiert die Instanziierung einer der folgenden generischen Agenten-Typen, welche wir im Folgenden allgemein als Geschäfts-Agenten bezeichnen:

1. *Produzenten-Agenten* (P_i) dienen der Instanziierung der mittleren Stufen einer Wertschöpfungskette. Sie stellen die Produzenten und Zwischenhändler der Wertschöpfungskette dar. Als solche kaufen sie von Ressourcen-Agenten Rohmaterialien oder von anderen Produzenten-Agenten Zwischenprodukte ein, die sie gemäß ihrer eigenen Produktionsfunktion als Input-Güter benötigen.

Ihr in der Produktionsfunktion definierter Produktionsprozess, für den über die GUI Produktionskosten und die zur Produktion benötigte Zeit eingestellt werden müssen, darf erst starten, wenn sie alle notwendigen spezifizierten Input-Güter erworben haben.

Anschließend können sie gemäß ihrer Produktionsfunktion ihre Output-Güter bzw. Produkte produzieren, wobei die definierten Produktionskosten dem Produzenten-Agenten von seinem Finanzguthaben abgezogen werden.

Die Produktionsfunktion der Produzenten-Agenten kann in Bezug auf Input- und Output-Güter sowie Produktionszeiten und –kosten pro Agentengruppe (zum Beispiel für Chip- und Prozessor-Produzenten, aber auch für zwei Gruppen von Chip-Produzenten) individuell unterschiedlich parametrisiert werden. Die produzierten Output-Güter werden anschließend über den Marktplatz anderen Produzenten-Agenten oder Konsumenten-Agenten als Zwischen- oder Endprodukte zum Kauf angeboten.

Aufgrund der Wertschöpfung, welche durch die Verarbeitung der Input-Güter in ihrem Produktionsprozess entsteht, streben sie eine Gewinnmarge (sogenannter *ProfitWish*) an, die sie als Verkäufer in den Verhandlungen mit den Agenten auf der nachgelagerten Wertschöpfungsstufe durchzusetzen versuchen.

2. *Händler-Agenten* (T_i) können auf Basis der Produzenten-Agenten modelliert werden, denn sie stellen den Spezialfall eines Produzenten dar, dessen Produktionsfunktion nur einen Zwischenlagerungsprozess beinhaltet. Ihr Output-Gut ist daher mit ihrem Input-Gut identisch. Da auch der Handel mit Produkten einen Mehrwert stiften kann, verfügen Händler-Agenten ebenfalls über einen *ProfitWish*. In den im Evaluationsteil der Arbeit untersuchten Simulationsszenarien wurde dieser Typus nicht verwendet, da der reine Handel oder die Spekulation mit Gütern nicht in ihrem Fokus stand.
3. *Ressourcen-Agenten* (R_i) stehen am Anfang der Wertschöpfungskette und versorgen das Netzwerk gemäß einer einstellbaren Versorgungsfunktion mit initialen Ressourcen.

Im Prinzip stellen sie in ihren Fähigkeiten beschränkte Produzenten-Agenten dar. Im Unterschied zu diesen weisen sie vereinfachte Produktionsfunktion auf, die Versorgungsfunktion. Zudem sind sie in ihren marktlichen Interaktionsfähigkeiten auf den Verkauf von Ressourcen beschränkt.

Die Versorgungsfunktion kann so parametrisiert werden, dass Rohstoffe in regelmäßigen Abständen verfügbar gemacht und auf dem Markt anderen Agenten angeboten werden. Die Zeitintervalle, innerhalb derer für einen Ressourcen-Agenten Rohstoffe verfügbar werden, wird fest mit der Größe *OutputFrequency* eingestellt. Dadurch wird das pro Zeiteinheit verfügbare Angebot an Ressourcen am Markt limitiert.

Ressourcen-Agenten kaufen Ressourcen selbst nicht von anderen Agenten ein, so dass kein Geldzufluss bzw. spekulatives Verhalten durch günstiges Einkaufen und teureres Verkaufen bzw. die Lagerung von Rohstoffen möglich ist. Zudem besitzen sie auch keine Prozesse zur Verarbeitung von Rohstoffen zu Zwischen- und Endprodukten.

4. *Konsumenten-Agenten* (C_i) stehen am Ende des Wertschöpfungsnetzwerks und kaufen Güter (sogenannte *Konsumgüter* beziehungsweise *Endprodukte*) von vorgelagerten Wertschöpfungsstufen, das heißt den Ressourcen- oder Produzenten-Agenten. Sie agieren gemäß einer Verbrauchsfunktion, die für bestimmte von ihnen nachgefragte Güter sowohl eine Nachfragefrequenz (das heißt ihren Verbrauch pro Zeiteinheit) als auch ihre maximale Zahlungsbereitschaft angibt.

Das heißt, sie besitzen eine genaue Preisvorstellung darüber, was sie bereit sind, für ein Gut als maximalen Preis zu bezahlen. Es steht ihnen stets genügend Finanzmittel zur Verfügung, um ihre Nachfrage am Markt zu befriedigen. In ihren marktlichen Interaktionsfähigkeiten sind sie auf den Kauf von Gütern beschränkt.

Die Geschäftsagenten führen über ihre Güter, ihre Käufe und Verkäufe sowie über ihren Geldbestand Buch, da sie Zu- und Abgänge von Gütern und Geld verarbeiten müssen.

MACSIMA-Agenten sind proaktiv und kommunizieren untereinander mittels Nachrichten. Zur gezielten Kontaktaufnahme benutzen sie den Verzeichnisdienst des Marktplatzes. Damit besitzen sie die grundlegend notwendigen sozialen Fähigkeiten zur Durchführung von Verhandlungen. Zudem können sie Mitglied von Agentengruppen sein, innerhalb derer ein Austausch von Informationen über erfolgreiche Verhandlungsstrategien und damit erzielte Verkaufspreise erfolgt. Aufbauend auf diesen Fähigkeiten besitzen sie Methoden zum Verhandeln und zur Bestimmung und Anpassung ihrer Ziele.

5.2.3 Realisierung des elektronischen Marktes

Das MACSIMA-System realisiert einen elektronischen Marktplatz und stellt den Agenten einen Verzeichnisdienst bereit, der von einer Instanz der Klasse *MarketAdmin* bereitgestellt wird. Dieser ist ein relativ einfach implementierter Dienst der MACSIMA-Simulationsumgebung. Jedoch wurde ein Konzept entwickelt, wie dieser Verzeichnisdienst bei einem Einsatz in der realen Welt als *Interoperabilitäts-Service-Agent (ISA)* realisiert werden kann, der Directory-, Mediator- und Ontologie-Dienste kapselt und diese den Geschäftsagenten zur Verfügung stellt (siehe [Ruß & Walz 2012; Ruß & Walz 2009] sowie Abbildung 36).

Ein Mediator-Dienst bestünde beispielsweise darin, ein Verhandlungsprotokoll mit einer zugehörigen Ontologie anzubieten, welches die Agenten bei ihren Verhandlungen benutzen können, falls unter den Agenten keine Klarheit darüber herrschen sollte, welches Verhandlungsprotokoll genutzt werden soll, welches in Szenarien mit einer großen Anzahl heterogener Agenten sinnvoll ist.

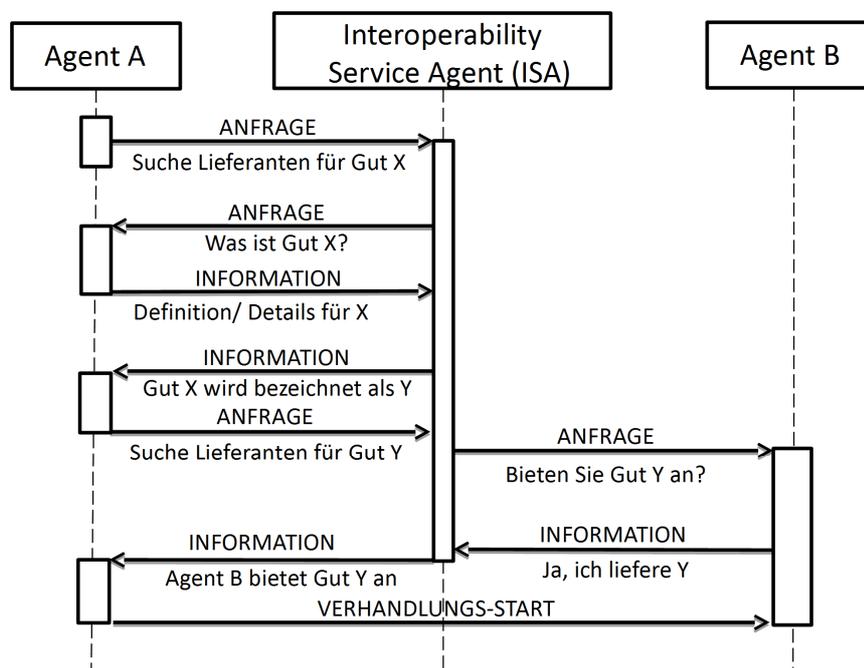


Abbildung 36: Matchmaking und Ontologie-Abbildung über den ISA-Agenten⁴⁴

Unabhängig voneinander entworfene heterogene Agenten können intern unterschiedliche Begrifflichkeiten und Repräsentationen für Güter und Dienste verwenden. Daher kann es

⁴⁴ Quelle: [Ruß & Walz 2009] (eigene Veröffentlichung)

vorkommen, dass der Name eines Gutes nicht ausreichend ist, um dieses eindeutig zu identifizieren und ein Matchmaking zwischen heterogenen Agenten mit sich überschneidenden Zielen zu realisieren.

Für diesen Fall kann der ISA mehrere Ontologien für Input- und Output-Güter verwalten, um unterschiedliche Ontologien der heterogenen Agenten aufeinander abbilden zu können und so die Interoperabilität der Agenten zu gewährleisten. In Abbildung 36 wird der über den ISA in diesem Fall stattfindende Nachrichtenaustausch skizziert.

Der in MACSIMA integrierte Verzeichnisdienst erlaubt es allen in MACSIMA instanziierten Geschäftsagenten, andere Agenten mit sich überschneidenden Zielen zu finden (zum Beispiel Agenten, die ein Gut kaufen wollen, die ein anderer Agent verkaufen will). Der Verzeichnisdienst verwaltet hierzu alle Agenten innerhalb eines Simulationsszenarios mit ihren eindeutigen Namen, Adressen, ihrem Agenten-Typ sowie den von ihnen angebotenen und nachgefragten Gütern. Die Agenten melden sich und die von ihnen angebotenen Güter hierfür beim Verzeichnisdienst an. Der Verzeichnisdienst besitzt aber keine Katalogfunktion, da er keine Informationen darüber besitzt, zu welchem Preis die Agenten die angebotenen Güter verkaufen oder kaufen wollen.

Um herauszufinden, ob eine gegenseitig vorteilhafte Transaktion ausgeführt werden kann, kann jeder Agent auf der Basis der über den MarketAdmin-Verzeichnisdienst abrufbaren Informationen einen anderen Agenten auf dem Marktplatz auswählen und kontaktieren, um eine bilaterale Verhandlung aufzunehmen. Erst zum Start einer solchen Verhandlung beginnen die Verhandlungspartner damit, Preisvorschläge direkt miteinander auszutauschen. Der Ablauf einer solchen Verhandlung in MACSIMA wird in Abschnitt 5.3.3 im Detail beschrieben.

In [De Paula et al. 2001] werden zwei Arten von Attributen differenziert, anhand derer gehandelte Güter spezifiziert werden können. Dabei handelt es sich zum einen um *Produktattribute*, die die innerhalb einer Verhandlung unveränderbaren Eigenschaften eines Gutes spezifizieren. Dabei kann es sich um objektive Eigenschaften wie beispielsweise die physische Beschaffenheit eines Gutes und dessen Qualität handeln. Darüber hinaus kann es sich hierbei aber auch um Eigenschaften wie zum Beispiel die Zugehörigkeit zu einer Marke oder den Hersteller beziehungsweise Anbieter des Gutes handeln, die bei der subjektiven Bewertung eines Gutes durch einen Nachfrager häufig eine Rolle spielen.

Zum anderen handelt es sich um *Verhandlungsattribute* (englisch *deal attributes*), bezüglich derer die Teilnehmer an einer Verhandlung Konzessionen machen können. Solche Verhandlungsattribute können beispielsweise der Preis sowie die Liefer- und Zahlungskonditionen für ein Gut sein.

Besitzt ein Gut mehrere Verhandlungsattribute, bezüglich derer sich die Verhandlungspartner einigen müssen, so resultiert daraus eine *mehrdimensionale Verhandlung*, in der die Ausprägungen der verschiedenen Attribute gewichtet beziehungsweise gegeneinander abgewogen werden müssen, um den Nutzen von Angeboten beziehungsweise möglichen Verhandlungslösungen bewerten und miteinander vergleichen zu können. Häufig lassen sich die Entscheidungskriterien und Präferenzen, nach denen menschliche Akteure in solchen multidimensionalen Verhandlungen handeln, nur schwer vollständig erheben und damit formalisieren. Dieser Umstand erschwert es, solche Verhandlungen zu automatisieren und autonome Agenten mit ihrer Durchführung zu beauftragen (vergleiche auch [Eymann 2000, S. 110; Ickerott 2007, S. 115 ff.]).

Bei den in MACSIMA gehandelten Gütern handelt es sich um sogenannte *homogene Güter*, die innerhalb einer Gutsklasse gleiche standardisierte Eigenschaften aufweisen und leicht austauschbar sind. Die Kaufentscheidungen ihrer Nachfrager hängen allein von ihrem Preis

ab, der somit das alleinige Verhandlungsattribut darstellt. Bei den von den MACSIMA-Agenten durchgeführten bilateralen Verhandlungen handelt es sich daher um sogenannte *eindimensionale Verhandlungen*. Spielen bei den Kaufentscheidungen auch subjektive Präferenzen der Nachfrager wie zum Beispiel die Vorliebe für eine bestimmte Marke oder einen Anbieter eine Rolle (wie beispielweise bei Automobilen), so spricht man von *heterogenen Gütern*.

Das Ziel jedes auf dem Marktplatz handelnden Agenten ist es, seinen eigenen Nutzen zu maximieren, wie es in Abschnitt 5.2.2 dargelegt wurde.

5.2.4 Kontrollzyklus eines MACSIMA-Agenten

Jeder Agent ist vollständig autonom und verfolgt das Ziel der Nutzen- bzw. Gewinnmaximierung. Agiert der Agent auf dem Marktplatz als Produzent, so folgen seine Handlungen einem Kontrollfluss, der in Abbildung 37 dargestellt ist.

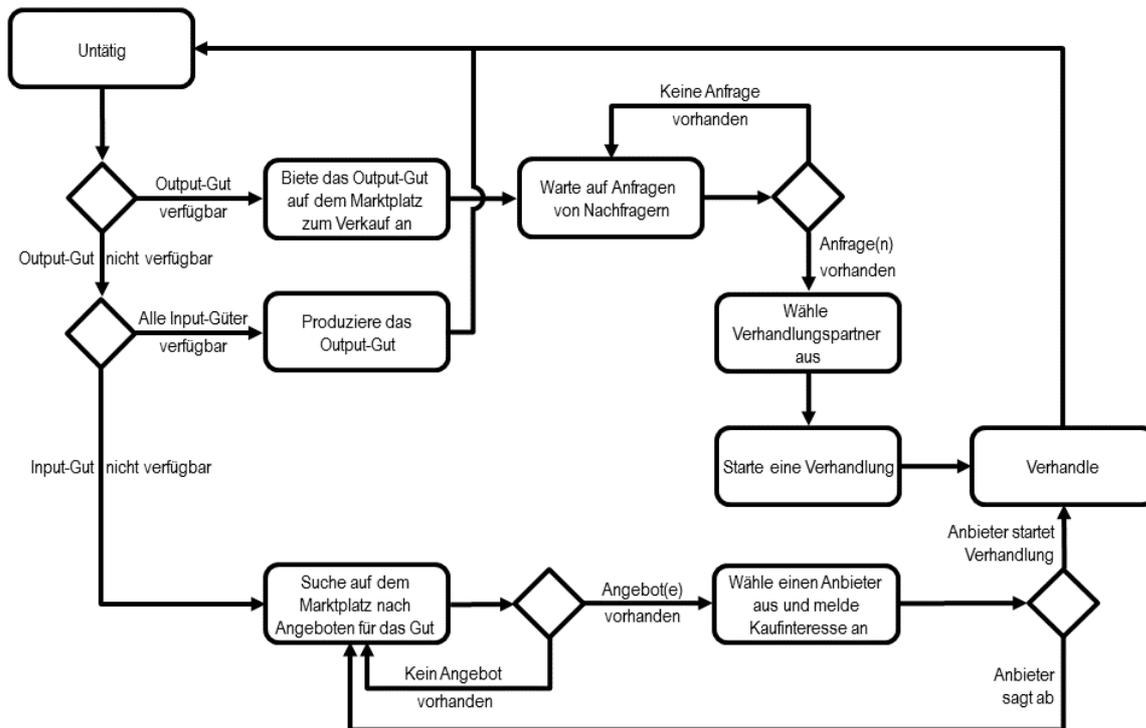


Abbildung 37: Kontrollzyklus eines MACSIMA-Produzenten-Agenten

Der in Abbildung 37 veranschaulichte Kontrollzyklus wird im Folgenden erläutert:

1. Der Produzenten-Agent startet seinen Kontrollfluss in einem Zustand „Untätig“. Wann immer er sich in diesem Ausgangszustand befindet, überprüft er als erstes, ob ein Output-Gut verfügbar ist, das er verkaufen kann.
2. Ist dies der Fall, so veröffentlicht er über den MarketAdmin-Verzeichnisdienst auf dem Marktplatz, dass er das Output-Gut zum Verkauf anbietet. Anschließend wartet der Verkäufer darauf, dass ihn Nachfrager kontaktieren, die am Kauf des von ihm angebotenen Gutes interessiert sind. Aus den Anfragen, die bei ihm eingehen, wählt der Agent einen Nachfrager aus, mit dem er eine Verhandlung startet. Nach dem erfolgreichen Abschluss oder dem Abbruch dieser Verhandlung geht er wieder in den Ausgangszustand zurück.
3. Wenn der Agent kein Output-Gut anzubieten hat, so überprüft er, ob er über alle Input-Güter verfügt, die er für die Produktion des Output-Gutes benötigt. Sind alle erforderlichen Input-Güter vorhanden, so produziert er das Output-Gut. Dabei bleibt der Agent

in der Anzahl der Runden, die sein Produktionsprozess gemäß seiner Definition andauert, ansonsten untätig. Nach dem Ende des Produktionsprozesses kehrt der Agent in seinen Ausgangszustand zurück.

4. Sollten nicht alle für den Produktionsprozess erforderlichen Input-Güter beschafft worden beziehungsweise verfügbar sein, so sucht der Agent über den MarketAdmin-Verzeichnisdienst des Marktplatzes nach Anbietern, die dort für eines der fehlenden Input-Güter einen Verkaufswunsch veröffentlicht haben. Findet der Agent passende Angebote, so wählt er einen der Anbieter aus und meldet sein Kaufinteresse an. Sagt ihm der Anbieter ab, so sucht der Agent als Nachfrager auf dem Marktplatz nach anderen Anbietern. Startet der Anbieter hingegen mit ihm eine Verhandlung über den Preis des von ihm nachgefragten Gutes, so nimmt der Agent an dieser Verhandlung teil. Der Ablauf einer solchen Verhandlung wird in Abschnitt 5.3 im Detail erläutert. Nach dem Verhandlungsende geht der Agent in seinen Ausgangszustand zurück.

Die Berechnung des Verhandlungsintervalls von Käufer und Verkäufer wird in Abschnitt 5.3.3.2 veranschaulicht (siehe Abbildung 39). Gelingt es den Agenten, in einer Verhandlung eine Einigung über den Verkaufspreis eines Gutes erzielen, so führen sie für dieses Gut eine Transaktion durch. Über die sich aus den Transaktionen ergebenden Geldbestände der einzelnen Agenten wird Buch geführt.

Falls ein Agent nicht als Produzenten-Agent, sondern als Ressourcen- oder Konsumenten-Agent instanziiert wird, verkürzt sich für ihn der in Abbildung 37 dargestellte Kontrollzyklus entsprechend. Zum Beispiel sind bei einem Ressourcen-Agenten die für den Einkauf von Gütern relevanten Methoden nicht implementiert.

5.3 Verhandlungen in MACSIMA

In MACSIMA können zwei Agenten direkt miteinander verhandeln, indem sie ein *bilaterales Verhandlungsprotokoll* nutzen, das in Abschnitt 5.3.1 erläutert wird. Damit ein Agent im Namen eines Prinzipals bilaterale Verhandlungen auf einem elektronischen Markt durchführen kann, benötigt er nach [Preist 1998] und [Kraus 1997, S. 79-97]:

- eine Spezifikation der Güter oder Dienste, die er handeln soll,
- eine Spezifikation des Zieles, welches der Agent für seinen Besitzer erreichen soll sowie
- eine Verhandlungsstrategie, die mindestens genauso effektiv ist wie diejenige eines qualifizierten Menschen in der gleichen Situation.

Damit die Agenten in MACSIMA interagieren und bilaterale Verhandlungen durchführen können, benötigen sie zunächst ein Verhandlungsprotokoll mit einer definierten Menge von wählbaren Sprech- beziehungsweise Verhandlungsakten. Zudem brauchen sie eine Verhandlungsstrategie, mit deren Hilfe sie im Verhandlungsverlauf zwischen den zur Verfügung stehenden Verhandlungsakten wählen können.

5.3.1 Verhandlungsprotokoll

In dem in MACSIMA implementierten und in Abbildung 38 dargestellten bilateralen Verhandlungsprotokoll sind alle Agenten mit denselben Handlungsalternativen ausgestattet, welche sich aus Pruitt's *Strategic Choice Model* (siehe [Pruitt 1981; Eymann 2000, S. 123 ff.] ableiten.

Pruitt hat von Menschen in den Bereichen Politik und Wirtschaft geführte Verhandlungen analysiert und festgestellt, dass zwei menschliche Verhandlungspartner in jedem Schritt einer Verhandlung abwechselnd je eine Aktion aus einer Menge von fünf elementaren Strategien auswählen, die im Folgenden in Anlehnung an [Eymann 2000, S. 123] erläutert werden:

1. *Einseitige Konzession*: Der Verhandlungspartner, der an der Reihe ist, gibt nach und macht einen neuen Verhandlungsvorschlag, in dem er seine Verhandlungsposition der Position seines Verhandlungspartners unilateral annähert.
2. *Kompetitives Verhalten*: Der Verhandlungspartner, der an der Reihe ist, gibt nicht nach und hält an seinem letzten Verhandlungsvorschlag fest. Er tut dies in der Hoffnung, dass dieses Verhalten seinen Verhandlungspartner zu einer Konzession veranlasst. Um seinen Opponenten aktiv zu einer Konzession zu bewegen, kann es sein, dass er zusätzlich argumentativ und möglicherweise sogar drohend auf ihn einwirkt.
3. *Koordinatives Verhalten*: Die Verhandlungspartner versuchen, auf kooperative und vertrauensvolle Weise gemeinsam zu einer Kompromisslösung zu gelangen. Dabei ist es möglich, dass sie einen dritten Partner darum bitten, vermittelnd tätig zu werden.
4. *Abwarten* (beziehungsweise *Nichtstun*): Der Verhandlungspartner, der an der Reihe ist, stellt seine Verhandlungsaktivitäten zunächst ein und lässt seinen Opponenten (im Unterschied zur elementaren Strategie des kompetitiven Verhaltens) darüber im Unklaren, ob er die Verhandlung fortführen will. Er wartet ab, ob er sein Gegenüber hierdurch nach einer bestimmten Zeitdauer zu einer Aktion, beispielsweise einer Konzession, provozieren kann und hält sich die Möglichkeit offen, die Verhandlung fortzuführen.

Die Strategieoption des Abwartens haben MACSIMA-Agenten nicht (siehe unten). In jeder Verhandlungsrunde senden sie an ihren Verhandlungspartner eine Nachricht, die einem der unten beschriebenen Verhandlungsakte ACCEPT, PROPOSE oder REJECT entspricht.

5. *Verhandlungsabbruch*: Der Verhandlungsteilnehmer, der an der Reihe ist, kann zu der Entscheidung kommen, dass er die Verhandlung nicht mehr fortführen will. In diesem Fall bricht er die Verhandlung einseitig ab, indem er seinem Verhandlungspartner diese Entscheidung unmissverständlich mitteilt.

Wie [Eymann 2000; S. 115] in Anlehnung an [De Paula 2001] ausgeführt hat, besitzt ein intelligenter Geschäftsagent in einer Verhandlungsrunde drei grundsätzliche Aktionsalternativen, auf die die oben erläuterten Strategien reduziert werden können. Daher beschränkt sich MACSIMA zur Realisierung von Verhandlungen zwischen den Geschäftsagenten auf die Implementierung der folgenden Verhandlungs- beziehungsweise Sprechakte:

1. *Akzeptieren (ACCEPT)*: Indem ein Agent seinem Verhandlungspartner ein ACCEPT sendet, gibt er diesem zu verstehen, dass er dessen letztgemachtes Angebot akzeptiert. Anschließend erfolgt die Transaktion des Gutes gegen die Zahlung eines Geldbetrags.
2. *Vorschlag machen (PROPOSE)*: PROPOSE-Nachrichten erfüllen in verschiedenen Verhandlungssituationen (siehe hierzu den Ablauf einer bilateralen Verhandlung in Abschnitt 5.3.3) unterschiedliche Zwecke:
 - a) Beim Start einer Verhandlung (siehe Abschnitt 5.3.3.3) dient eine PROPOSE-Nachricht dazu, dem Verhandlungspartner gegebenenfalls ein erstes Angebot mit einem initialen Preisvorschlag zu unterbreiten.
 - b) In einer laufenden Verhandlung teilt ein Agent seinem Verhandlungspartner dadurch mit, dass er dessen letztes Angebot nicht akzeptiert und ihm stattdessen ein Gegenangebot unterbreitet (siehe Abschnitt 5.3.3.4). Dieses kann darin bestehen,

keine weitere Konzession zu machen und auf dem letzten eigenen Preisvorschlag zu beharren oder dem Opponenten mit einem Gegenangebot preislich entgegen zu kommen.

3. *Abbrechen (REJECT)*: Durch Senden einer REJECT-Nachricht bricht ein Agent die laufende Verhandlung ab. Zum Abbruch der Verhandlung reicht es aus, wenn einer der Opponenten eine REJECT-Nachricht sendet.

Von den MACSIMA-Agenten wird ein bilaterales Verhandlungsprotokoll verwendet, das auf diesen drei Verhandlungsakten aufbaut und in Abbildung 38 in der Form eines UML-Sequenzdiagramms dargestellt ist.

Dabei handelt es sich um ein sogenanntes *monotones Konzessionsprotokoll* (englisch *Monotonic Concession Protocol*, siehe [Rosenschein & Zlotkin 1994]), das in Abschnitt 3.3.4.1 erläutert wird. Bei diesem nähern sich die Preisvorstellungen von Käufer und Verkäufer ausgehend von ihren Startpreisen (siehe Abschnitt 5.3.3) durch eine Folge von Angeboten und Gegenangeboten mit schrittweisen preislichen Zugeständnissen beziehungsweise Konzessionen sukzessive immer weiter an.

Dieser Verhandlungsprozess dauert so lange, bis die Agenten eine Einigung über einen Abschluss- beziehungsweise Kaufpreis erzielen oder einer von ihnen irgendwann die Verhandlung abbricht. Konnten die Agenten einen beiderseits akzeptablen Preis aushandeln und die Verhandlung somit erfolgreich abschließen, so führen sie die Transaktion aus, verbuchen diese und kontaktieren danach in Abhängigkeit von ihrem Kontrollzyklus den Verzeichnisdienst, um ein anderes Gut zu kaufen oder zu verkaufen. Endet die Verhandlung mit einem Verhandlungsabbruch, so suchen die Agenten über den Verzeichnisdienst nach neuen Verhandlungspartnern, die sich für den Kauf oder Verkauf des gleichen Gutes anbieten.

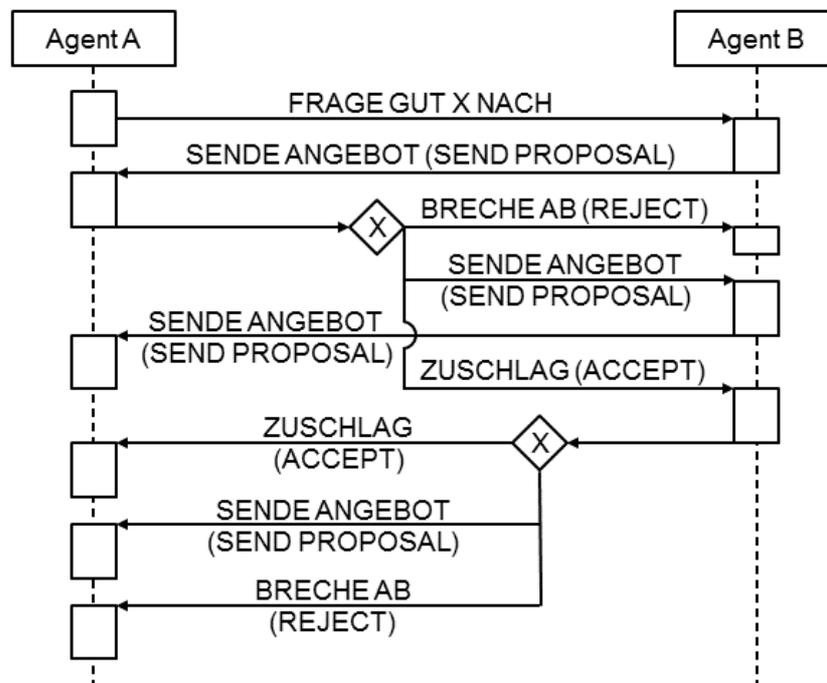


Abbildung 38: Bilaterales Preisverhandlungsprotokoll⁴⁵

Da jeder der beiden Verhandlungspartner die individuellen Preisvorstellungen und insbesondere den Indifferenzpreis seines Opponenten nicht kennt, befinden sich beide in einer Entscheidungssituation mit unvollständiger Information. In dieser gibt ihnen die Nutzung des

⁴⁵ Quelle: [Ruß & Walz 2012] (eigene Veröffentlichung)

Verhandlungsprotokolls die Möglichkeit, einen gegenseitig akzeptablen Abschlusspreis auszuhandeln beziehungsweise eine Kompromisslösung zu erzielen.

5.3.2 Parameter der Verhandlungsstrategie

Das Verhandlungsprotokoll und die zur Verfügung stehenden Sprechakte bilden die Grundlage für die automatisierte Durchführung von Verhandlungen. Innerhalb einer Verhandlung strebt ein Agent danach, seinen individuellen Nutzen zu erhöhen beziehungsweise ein bestimmtes Nutzenziel zu erreichen. Agiert der Agent als Verkäufer, so kann er zum Beispiel danach streben, beim Verkauf eines Gutes eine bestimmte Gewinnmarge zu realisieren.

Um seinen Nutzen durch ein geschicktes Verhalten zu erhöhen, verfolgt jeder Agent eine Verhandlungsstrategie, die durch sechs *Strategieparameter* bestimmt wird (siehe auch [Eymann 2000, S. 156 ff.; Walz 2007, S. 55 ff.]):

1. *Acquisitiveness (AQ)*
2. *Delta_Change (DC)*
3. *Delta_Jump (DJ)*
4. *Satisfaction (SF)*
5. *Weight_Memory (WM)*
6. *Reputation (RP)*

Einige der Parameter beeinflussen den Agenten bei der Auswahl des Verhandlungsaktes, den er Agent als nächstes ausführt, falls ihm im Rahmen des Verhandlungsprotokolls mehrere Alternativen zur Verfügung stehen. Auf Basis des Parameters AQ entscheidet der Agent zum Beispiel darüber, ob er seinem Opponenten eine preisliche Konzession anbietet oder dies nicht tut und zunächst weiter auf seiner Verhandlungsposition beharrt.

Andere Parameter (wie DC, DJ und WM) beeinflussen hingegen maßgeblich die Preisentscheidungen eines Agenten, das heißt wie hoch der Agent innerhalb einer Verhandlung seinen Mindest- und Maximalpreis ansetzt und wie groß preisliche Konzessionen ausfallen können. Details hierzu finden sich auch in den Abschnitten 5.3.3.2 und 5.3.3.3, in denen beschrieben wird, wie ein Agent seinen Mindest- und Maximalpreis sowie das Verhandlungsintervall bestimmt, innerhalb dessen für ihn akzeptable Abschlusspreise für ein Gut liegen.

Die Strategieparameter nehmen reelle Werte aus dem Intervall $[0; 1]$ an. In der aktuellen Implementierung wurde ihr Wertebereich allerdings auf das Intervall $[0,05; 0,95]$ beschränkt, um zu verhindern, dass ein Agent ein unflexibles und extremes Verhalten annehmen kann. Die Werte der Parameter einer Verhandlungsstrategie werden von einem Agenten in einem sogenannten *Genotyp* gespeichert, einer Datenstruktur, die für die Verarbeitung durch einen evolutionären Algorithmus geeignet ist. Die Strategieparameter werden daher auch *Gene* genannt. Ein Agent besitzt in der Regel mehrere Genotypen, die von ihm in seinen Verhandlungen abwechselnd eingesetzt und mit der Zeit mittels eines evolutionären Algorithmus an sich ändernde Umwelt- beziehungsweise Marktbedingungen adaptiert werden. Die Details hierzu werden in Abschnitt 5.4 und insbesondere bei der Beschreibung des adaptiven Verhandlungsmoduls der MACSIMA-Agenten in Abschnitt 5.4.1 beschrieben.

Im Folgenden wird für die einzelnen Parameter beziehungsweise Gene in Anlehnung an [Eymann 2000, S. 156 ff.] erläutert, wie sie das Verhalten eines Agenten beeinflussen. Im Zusammenspiel mit der Agentenlogik führen die Parameter dazu, dass die Agenten ein komplexes und nichtdeterministisches strategisches Verhandlungsverhalten zeigen.

Der Wert des Gens *Acquisitiveness* (*AQ*) (übersetzt: Gewinnsucht) eines Agenten bestimmt die Wahrscheinlichkeit dafür, dass der Agent im Verlauf der Verhandlung bei einem PROPOSE-Verhandlungsakt seinem Opponenten eine einseitige preisliche Konzession anbietet. Ob er dies tatsächlich tut, wird anhand einer stochastischen Probe gegen den AQ-Wert entschieden, dem sogenannten *Acquisitiveness-Check*. Fällt dabei ein vom Agenten für eine Zufallsvariable ermittelter Wert größer als der Wert des Gens AQ aus, so bietet der Agent eine Konzession an. Ansonsten tut er dies nicht und wiederholt sein vorhergehendes Angebot.

Wäre der Wert des Gens AQ gleich 0, so würde der Agent, nachdem er seinen Startpreis geboten hat, auf jedes Angebot des Opponenten mit einer preislichen Konzession antworten, bis sein Mindestpreis erreicht oder die Verhandlung mit einem erfolgreichen Abschluss oder einem Abbruch beendet wird. Als Käufer würde er also sein Angebot im Rahmen seines Verhandlungsintervalls (siehe Abschnitt 5.3.3.2) stetig erhöhen und als Verkäufer stetig senken. Bei einem AQ-Wert von 1 würde der Agent hingegen ein kompetitives Verhalten zeigen und seinem Opponenten gegenüber während des gesamten Verhandlungsverlaufs keine einzige Konzession machen.

Mit Hilfe des Gens *Delta_Change* (*DC*) bestimmt ein Agent im Normalfall seine individuelle Konzessionsschrittweite *PriceStep*. Hiervon weichen nur bestimmte Ausnahmefälle ab, in denen die Konzessionsschrittweite von einem sogenannten *Preisautomaten* festgesetzt wird (siehe Abschnitt 5.3.3.7).

Die individuelle Konzessionsschrittweite *PriceStep* eines Agenten gibt an, wie hoch sein preisliches Entgegenkommen ausfällt, wenn er sich dazu entschließt, auf ein Angebot mit einem Gegenangebot zu reagieren, das eine Konzession beinhaltet. Die Aktivitäten, die ein Agent ausführt, wenn er mit einem Gegenangebot reagiert, werden im Detail in Abschnitt 5.3.3.4 beschrieben.

Ein als Käufer auftretender Agent *K* und ein als Verkäufer agierender Agent *VK* berechnen normalerweise ihre individuellen Konzessionsschrittweiten *PriceStep_K* und *PriceStep_{VK}*, sobald ihnen das Startgebot ihres jeweiligen Opponenten bekannt ist. Als Basis dient ihnen hierbei zum einen die Preisdifferenz zwischen ihrem eigenen Startpreis und dem Startpreis ihres Verhandlungspartners. Zum anderen nutzen sie hierfür auch den Wert ihres Gens DC, der in dem jeweils von ihnen in der Verhandlung eingesetzten Genotyp enthalten ist:

$$PriceStep_K = (|Startgebot_{VK} - Startgebot_K|) * Delta_Change_K$$

$$PriceStep_{VK} = (|Startgebot_{VK} - Startgebot_K|) * Delta_Change_{VK}$$

Der Wert des Gens *Delta_Change_K* gibt einen prozentualen Anteil am Abstand zwischen den beiden Startgeboten an, um welchen der Agent *K* dem anderen Agenten in einem einzelnen Gegenangebot beziehungsweise Konzessionsschritt preislich entgegen kommen kann.

Die Konzessionsschrittweite wird von jedem Agenten (auch bei ihrer Festsetzung mit Unterstützung des Preisautomaten) individuell für eine einzelne Verhandlung bestimmt und bleibt während der Dauer dieser Verhandlung unverändert. Eine Reduktion der Schrittweite, wenn im Verlauf der Verhandlung eine Annäherung an die Mindest- oder Maximalpreisvorstellung eines Agenten erfolgt, wird nicht durchgeführt.

Der Umstand, dass beide Opponenten die Preisdifferenz der Startgebote als Ausgangsbasis für die Berechnung ihrer individuellen Schrittweiten nutzen, sorgt dafür, dass die Unterschiede in den von ihnen gemachten Preisschritten ausschließlich auf den unterschiedlichen Ausprägungen ihrer DC-Werte beruhen und keiner der Verhandlungsteilnehmer benachteiligt wird.

Würde statt der Preisdifferenz die Höhe des eigenen Startgebots als Berechnungsbasis dienen, so würde sich bei identischen DC-Werten für den Verkäufer wegen seines üblicherweise höheren Startgebotes in der Regel eine größere Schrittweite ergeben als für den Käufer. Dies würde dazu führen, dass der Verkäufer in der Regel höhere Konzessionen eingehen würde als der Käufer. Dadurch wäre der Verkäufer dann benachteiligt.

Wenn ein Agent A an einer neuen Verhandlung für ein Gut teilnimmt, hilft ihm das Gen *Delta_Jump* (DJ) bei der Bestimmung seines Verhandlungsintervalls (siehe Abschnitt 5.3.3.2), das seinen auf diese Verhandlung bezogenen preislichen Spielraum festlegt. Sowohl als Käufer als auch als Verkäufer berechnet der Agent seinen Mindestpreis LLP_A (englisch *Lower Limit Price*) und seinen Maximalpreis ULP_A (englisch *Upper Limit Price*) für das Gut, die er in der neuen Verhandlung verwenden will.

Im Normalfall entspricht dabei der LLP_A dem $oLLP_A$ (engl. *original Lower Limit Price*) und der ULP_A dem $oULP_A$ (engl. *original Upper Limit Price*). Die Werte von $oLLP_A$ und $oULP_A$ werden anhand der folgenden Formeln berechnet:

$$oLLP_A = \text{weighted_Average} * (1 - \text{Delta_Jump}_A)$$

$$oULP_A = \text{weighted_Average} * (1 + \text{Delta_Jump}_A)$$

Agiert der Agent A als Käufer, so spezifiziert in der Regel $LLP_A = oLLP_A$ seinen Startpreis, den er angibt, wenn er in einer Verhandlung ein initiales Startgebot macht. Handelt der Agent hingegen als Verkäufer, so wird die Höhe seines Startgebots im Regelfall von $ULP_A = oULP_A$ bestimmt.

In die Berechnung von $oLLP_A$ und $oULP_A$ geht dabei neben dem Wert von DJ auch der (individuell vom Agenten berechnete) gewichtete Durchschnittspreis *weighted_Average* ($wAVG$) für das Gut ein. Die Berechnung von $wAVG$ ist dabei vom Gen *Weight_Memory* abhängig und wird daher in diesem Abschnitt weiter unten bei der Beschreibung der Verwendung von *Weight_Memory* erläutert.

In bestimmten Fällen werden die Preisgrenzen LLP_A und ULP_A und damit das Verhandlungsintervall des Agenten vor dem Start einer Verhandlung auch noch etwas angepasst (siehe Abschnitt 5.3.3.2), so dass sie nicht mehr den ursprünglichen, im Normalfall verwendeten Werten von $oLLP_A$ und $oULP_A$ entsprechen. Durch die Anpassung von LLP_A wird beispielsweise sichergestellt, dass der Mindestpreis, den ein Verkäufer für ein Gut ansetzt, dessen Herstellungskosten deckt. In Ausnahmefällen, in denen das Verhandlungsintervall möglicherweise nicht ausreichend an die aktuellen Marktpreise angepasst ist, kann die Modifikation beziehungsweise Aktualisierung der Preisgrenzen LLP_A und ULP_A auch von dem in Abschnitt 5.3.3.7 beschriebenen Preisautomaten übernommen werden.

Das Gen *Satisfaction* (SF) (übersetzt: Zufriedenheit) dient dem Agenten dazu, zu entscheiden, ob er bei einem Gegenangebot seines Verhandlungspartners, in dem dieser auf seiner Verhandlungsposition beharrt, die Verhandlung fortführt oder darauf mit einem Verhandlungsabbruch reagiert (siehe Abschnitt 5.3.3.5). In diesem Fall (und nur dann) führt der Agent einen sogenannten Satisfaction-Check durch. Dabei handelt es sich um eine stochastische Probe gegen den Wert des Gens SF , der gewissermaßen das Niveau der Zufriedenheit des Agenten widerspiegelt. Ermittelt der Agent für eine Zufallsvariable einen Wert, der größer als SF ausfällt, so bricht der Agent die Verhandlung ab (siehe Abschnitt 5.3.3.5). Je geringer der aktuelle Wert des Gens SF ausfällt beziehungsweise je weniger zufrieden der Agent momentan ist, umso höher ist die Wahrscheinlichkeit, dass er eine Verhandlung abbricht, wenn sein Opponent ihm in seinem aktuellen Gegenangebot keine Konzession anbietet. Weist das Gen SF den Wert 0,2 beziehungsweise zwanzig Prozent auf, so beträgt jedes

Mal, wenn dieser Fall während einer Verhandlung auftritt, die Abbruchwahrscheinlichkeit ($1 - SF$) beziehungsweise achtzig Prozent.

Nähert sich sein Opponent in seinem aktuellen Gegenangebot hingegen an, so führt der Agent keinen Satisfaction-Check durch und bricht die Verhandlung auch nicht ab. Stattdessen reagiert er darauf seinerseits mit einem Gegenangebot (siehe Abschnitt 5.3.3.4). In diesem Fall wird der Wert des Gens SF vom Agenten daher nicht benötigt, um über seinen nächsten Verhandlungsschritt zu entscheiden.

Das Gen *Weight_Memory* (WM) wird von einem Agenten dazu verwendet, für jedes von ihm gehandelte Gut einen individuellen gewichteten Durchschnittspreis *weighted_Average* ($wAVG$) zu berechnen. Der Agent ermittelt den aktualisierten Wert $wAVG_{t+1}$ nach jedem erfolgreichen Verhandlungsabschluss für ein Gut anhand des ausgehandelten Verkaufspreises (*lastAgreementPrice*) sowie der bis zum Zeitpunkt t des Abschlusses verwendeten Werte von WM und *weighted_Average_t* anhand der Formel:

$$weighted_Average_{t+1} = (lastAgreementPrice * WM) + weighted_Average_t * (1 - WM)$$

Den Wert von *lastAgreementPrice* entnimmt der Agent seiner in Abschnitt 5.3.3.1 beschriebenen Datenstruktur *PriceDistribution*. Und da diese Berechnung unmittelbar nach dem Abschluss erfolgt, wird der dabei verwendete WM -Wert dem Genotyp entnommen, der in der erfolgreich abgeschlossenen Verhandlung eingesetzt wurde.

Der Wert des Gens WM gibt an, wie stark der neu erzielte Abschlusspreis und der sich aus vorhergehenden Abschlüssen ergebene *weighted_Average_t* in die Ermittlung des aktualisierten gewichteten Durchschnittspreises *weighted_Average_{t+1}* eingehen. Nimmt WM einen Wert an, der 0,5 übersteigt, so wird der bisherige Durchschnittspreis dabei weniger stark berücksichtigt als der aktuelle Abschlusspreis. Da der Wert des Gens WM durch den Lernprozess verändert werden kann (zum Beispiel durch den genetischen Operator der Mutation, siehe Abschnitt 5.4.3.4), ist ein Agent in der Lage, bei der Berechnung des *weighted_Average* die Gewichtung von alten und neuen Erfahrungen im Laufe seiner Lernzyklen selbständig anzupassen.

Den für ein Gut aktualisierten Wert von *weighted_Average* speichert ein Agent in seiner Datenstruktur *PriceDistribution* (siehe Abschnitt 5.3.3.1), so dass er ihn modifizieren beziehungsweise aktualisieren und bei der Anpassung und Prüfung von Preisgrenzen verwenden kann. Nimmt ein Agent als Käufer an einer Verhandlung teil, so überprüft er beispielsweise das Startgebot des Verkäufers darauf, ob dieses den Wert von *weighted_Average* um mehr als achtzig Prozent übersteigt. Ist dies der Fall, so fasst der Käufer dies als sogenanntes Wucherangebot auf und bricht die Verhandlung direkt einseitig ab. Weitere Erläuterungen hierzu finden sich in Abschnitt 5.3.3.5.

Das Gen *Reputation* (RP) wird in der aktuellen Implementierung von MACSIMA nur dann verwendet, wenn sich die Agenten miteinander über Auktionsmechanismen koordinieren. Nimmt ein Agent beispielsweise an einer Englischen Auktion teil (siehe Abschnitt 3.3.4.2), so entscheidet in bestimmten Fällen eine stochastische Probe gegen dieses Gen mit darüber, welchen grundsätzlichen Charakter seine Bietstrategie haben soll.

Der Charakter der Bietstrategie, der *normal*, *aggressiv* oder *kompetitiv* sein kann, bestimmt tendenziell, wann ein Agent als Bieter in einer Auktion dazu bereit ist, das bisherige Höchstgebot eines Konkurrenten zu überbieten. Zudem determiniert der Charakter der Bietstrategie auch, ob er in diesem Fall das Höchstgebot nur um einen vom Auktionator festgelegten Mindestdschritt erhöht oder eventuell deutlich darüber hinaus geht, um seine Konkurrenten von weiteren preistreibenden Geboten abzuhalten.

Bei bilateralen Verhandlungen wird dieses Gen in der aktuellen Implementierung von den Agenten nicht verwendet. Es könnte aber bei zukünftigen Erweiterungen dazu eingesetzt werden, um weitere Aspekte des Agentenverhaltens (wie zum Beispiel die Zuverlässigkeit eines Agenten) zu modellieren und zu parametrisieren.

Zum Beispiel könnte der Wert des Gens RP die Wahrscheinlichkeit dafür angeben, dass ein Agent nach einem erfolgreichen Verhandlungsabschluss auch tatsächlich zügig zahlt oder liefert. In der aktuellen Implementierung tun die Agenten dies immer zuverlässig.

Die MACSIMA-Agenten sind mit einem adaptiven Verhandlungsmodul ausgestattet, das es ihnen ermöglicht, ihre in Genotypen der Form $\langle AQ, DC, DJ, SF, WM, RP \rangle$ codierten Verhandlungsstrategien mit Methoden des evolutionären Lernens an sich dynamisch ändernde Umwelt- beziehungsweise Marktbedingungen anzupassen (siehe Abschnitt 5.4).

5.3.3 Ablauf einer bilateralen Verhandlung

In MACSIMA können sich die Agenten im Verlauf einer bilateralen Verhandlung unter anderem in den folgenden fünf wesentlichen Verhandlungssituationen befinden:

1. Bestimmung des Verhandlungsintervalls
2. Start einer Verhandlung
3. Reaktion mit Gegenangeboten
4. Reaktion mit Verhandlungsabbrüchen
5. Erfolgreicher Verhandlungsabschluss

In einer laufenden Verhandlung wiederholt sich die Verhandlungssituation 3 (alternierend zwischen den Verhandlungspartnern) so lange, bis

- entweder einer der beiden Agenten feststellt, dass das Gegenangebot des Opponenten in einem von ihm preislich akzeptierbaren Rahmen liegt (sogenannter *CrossedOffers-Fall*) und dieses daher akzeptiert und dem Opponenten eine ACCEPT-Nachricht sendet oder aber
- einer der Verhandlungspartner einen Satisfaction-Check durchführt, der negativ ausfällt. In diesem Fall ist der Agent mit dem Verhandlungsverlauf unzufrieden und bricht die Verhandlung daher ab.

In den oben genannten Verhandlungssituationen, die im Verlauf einer Verhandlung in MACSIMA auftreten, greifen die Agenten bei ihren Berechnungen auf die Werte verschiedener Variablen zu, die in der Datenstruktur *PriceDistribution* (PD) verwaltet werden. Daher wird im Folgenden zunächst diese Datenstruktur erläutert. Anschließend werden in den darauffolgenden Abschnitten die Verhandlungssituationen und die Aktivitäten, die die Agenten in ihnen ausführen, im Detail erläutert.

5.3.3.1 Die Datenstruktur PriceDistribution

Die Datenstruktur PriceDistribution hat die Aufgabe, den Agenten mit seinen Marktpreisvorstellungen für ein Gut inklusive der damit verbundenen preislichen Limitierungen zu versorgen. In die PriceDistribution fließen daher die Beobachtungen des Agenten über erzielte Transaktionen mit den dazu gehörenden Abschlusspreisen ein.

Der Agent verwaltet für jedes von ihm gehandelte (das heißt gekaufte oder verkaufte) Gut jeweils eine PD. Er nutzt die in ihr enthaltenen Informationen bei der Berechnung seines Verhandlungsintervalls und seiner Preisvorschläge, die er im Verlauf einer Verhandlung macht.

Bei Start des MACSIMA-Systems beziehungsweise der Agenten muss jeder Agent für jedes von ihm gehandelte Gut eine *PriceDistribution* initialisieren. Dabei werden ihre Variablen gemäß der Beschreibung in Tabelle 3 belegt.

Nr.	Variable	Initialisierung (übergebener Wert)
1	<i>average</i>	Für den Simulationsstart erforderlicher virtueller Markt- bzw. Durchschnittspreis für ein Gut, an dem sich der Agent vor seinem ersten Abschluss orientiert. (aus der Szenario-Definition)
2	<i>absoluteLowerLimitPrice (aLLP)</i>	Absolute untere Preisgrenze für das Gut in einem Simulationslauf (aus der Szenario-Definition)
3	<i>absoluteUpperLimitPrice (aULP)</i>	Absolute obere Preisgrenze für das Gut in einem Simulationslauf (aus der Szenario-Definition)
4	<i>weighted_Average (wAVG)</i>	Diese Variable wird mit dem Wert der Variablen <i>average</i> initialisiert. Nach jedem erfolgreichen Verhandlungsabschluss wird sie auf Basis des Abschlusspreises und des Wertes des Gens WM aktualisiert. Dabei wird der Wert des Gens WM dem Genotyp entnommen, der in der Verhandlung eingesetzt wurde.
5	<i>LowerLimitPrice (LLP)</i>	Der aktuelle individuelle Mindestpreis für das Gut, den der Agent für eine Verhandlung festlegt. Der <i>LLP</i> wird aktualisiert, wenn <ul style="list-style-type: none"> - der Agent eine neue Verhandlungsstrategie einsetzt und dafür seinen Genotyp wechselt oder - der in Abschnitt 5.3.3.7 beschriebene Preisautomat korrigierend eingreift.
6	<i>UpperLimitPrice (ULP)</i>	Der aktuelle individuelle Maximalpreis für das Gut, den der Agent für eine Verhandlung festlegt. Der <i>ULP</i> wird aktualisiert, wenn <ul style="list-style-type: none"> - der Agent einen neuen Genotyp einsetzt oder - der Preisautomat korrigierend eingreift.
7	<i>lastAgreementPrice</i>	Der letzte Abschlusspreis, zu dem der Agent das Gut verkauft oder gekauft hat. Dieser wird mit dem Wert -1 initialisiert.

Tabelle 3: Die Variablen der Datenstruktur *PriceDistribution*

Für die Initialisierung der Variablen mit den Nummern 1 bis 3 werden dabei Parameterwerte verwendet, die bei der Szenario-Definition festgelegt wurden und vom Agenten im Verlauf der Simulation nicht geändert werden.

In der Folge aktualisieren die Agenten nach jeder erfolgreich abgeschlossenen Verhandlung für ein Gut ihre jeweilige *PriceDistribution* für das Gut. Dies wird in Abschnitt 5.3.3.6 erläutert.

5.3.3.2 Bestimmung des Verhandlungsintervalls

Die Anbahnung einer Verhandlung erfolgt gemäß dem in Abschnitt 5.2.4 erläuterten und in Abbildung 37 dargestellten Kontrollzyklus eines MACSIMA-Agenten.

Nimmt ein Agent A an einer Verhandlung über den Verkauf eines bestimmten Gutes teil, so bestimmt er vor dem eigentlichen Verhandlungsstart, das heißt, bevor er oder sein Opponent ein erstes Gebot abgibt, zunächst intern seinen jeweiligen Mindestpreis LLP_A (engl. *Lower Limit Price*) und seinem Maximalpreis ULP_A (engl. *Upper Limit Price*) für das Gut. Diese beiden von ihm individuell ermittelten Preise bestimmen die sogenannten *Preisgrenzen* seines Verhandlungsintervalls, das seinen preislichen Verhandlungsspielraum mit den für ihn akzeptablen Abschlusspreisen für ein Gut angibt.

In Abschnitt 5.3.2 wurde darauf hingewiesen, dass hierbei LLP_A und ULP_A in der Regel zunächst auf die beiden Werte $oLLP_A$ und $oULP_A$ gesetzt und danach gegebenenfalls weiter angepasst werden, so dass sie schließlich die Werte LLP'_A und ULP'_A annehmen (siehe auch Abbildung 39).

So kann es beispielsweise vorkommen, dass ein Verkäufer VK statt seines $oLLP_{VK}$ einen etwas höheren Wert LLP'_{VK} als Mindestpreis verwendet, der seine Input- und Produktionskosten deckt und eine Gewinnmarge enthält, falls der $oLLP_{VK}$ dafür zu niedrig angesetzt sein sollte (siehe hierzu auch die Erläuterungen zur Verhandlungssituation 6 a) in Abschnitt 5.3.3.6).

Analog hierzu ist es auch möglich, dass der Käufer K des Gutes seinen $oULP_K$ auf einen Wert ULP'_{VK} absenkt. Dies ist sinnvoll, wenn sich sein $oULP_K$ in einem Bereich bewegt, der es dem Agenten zum Beispiel nicht gestattet, ein von ihm mit dem gekauften Gut hergestelltes Produkt den Agenten auf einer nachfolgenden Wertschöpfungsstufe zu einem konkurrenzfähigen Preis anzubieten (siehe hierzu auch die Erläuterungen zur Verhandlungssituation 6 b) in Abschnitt 5.3.3.6).

Darüber hinaus kann in bestimmten Ausnahmefällen, bei denen von einem Agenten über eine längere Folge von Verhandlungen kein erfolgreicher Abschluss erzielt werden konnte, auch sein Preisautomat Anpassungen vornehmen. Dessen Funktionsweise wird in Abschnitt 5.3.3.7 erläutert. Nach der Anpassung ihrer Preisgrenzen LLP_K und ULP_K sowie von LLP_{VK} und ULP_{VK} speichern Käufer und Verkäufer die von ihnen aktualisierten Werte in ihrer jeweiligen PD. Die sich für sie ergebenden Verhandlungsintervalle sind in Abbildung 39 dargestellt. Aus der Schnittmenge der aktualisierten Preisgrenzen beziehungsweise ihrer Verhandlungsintervalle ergibt sich für einen Käufer K und einen Verkäufer VK dann ein eindimensionaler Einigungsraum, der in Abbildung 39 in Dunkelgrau dargestellt ist.

Abbildung 39 verdeutlicht, dass ein initiales Verhandlungsintervall durch entsprechende Preisanpassungen oder vom Preisautomaten veränderte Preise (siehe Abschnitt 5.3.3.7) deutlich eingeschränkt werden kann. Unter Umständen bleibt vom anfänglich gegebenen theoretischen Einigungsbereich $[oLLP_{VK}; oULP_K]$ (hellgrau markiert) nur das kleinere Intervall $[LLP'_{VK}; ULP'_K]$ (dunkelgrau markiert) übrig.

Ein kleiner Einigungsbereich kann den Vorteil haben, dass die Agenten schneller eine Einigung erzielen. Im Extremfall kann es aber vorkommen, dass Käufer und Verkäufer ihre Mindest- und Maximalpreise so anpassen, dass der ULP_K des Käufers schließlich kleiner als der LLP_{VK} des Verkäufers wird und kein Einigungsbereich mehr vorhanden ist.

Sowohl Käufer als auch Verkäufer besitzen auch absolute Preisunter- ($aLLP$) und -obergrenzen ($aULP$), die bei der Definition der zu simulierenden Szenarien festgesetzt werden. Die Agenten können diese Preisgrenzen ihrer PD für das Gut entnehmen.

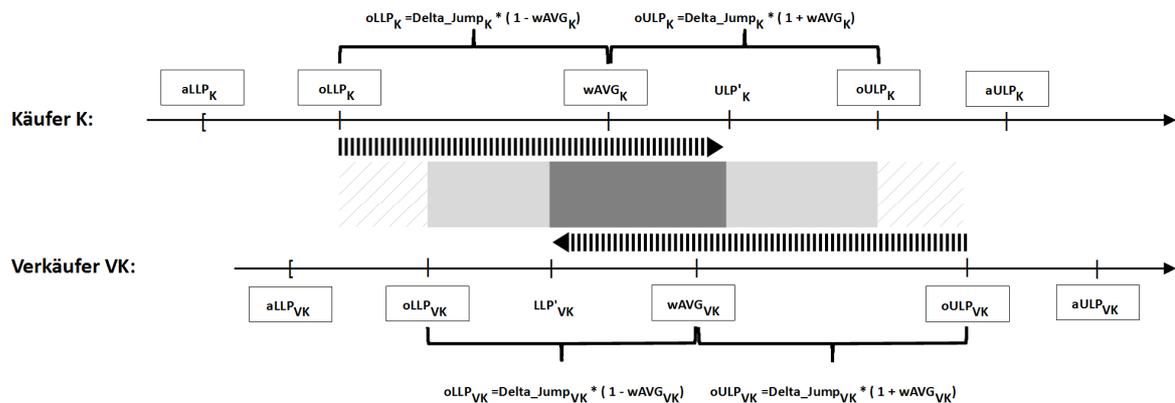


Abbildung 39: Verhandlungsintervall von Käufer und Verkäufer

Die aktuellen Werte der Preise LLP_K , ULP_K , LLP_{VK} und ULP_{VK} dürfen nicht über die absoluten Preisgrenzen hinausgehen. Falls sich zum Beispiel der $wAVG$ eines Agenten für ein Gut einer dieser absoluten Grenzen nähert, so kann es vorkommen, dass eine der Preisgrenzen verletzt wird. In diesem Fall wird der LLP auf den $aLLP$ oder der ULP auf den $aULP$ gesetzt.

5.3.3.3 Start einer Verhandlung

Agiert der Agent in einer Verhandlung in der Rolle des Käufers, so beginnt er die Verhandlung mit einem Startgebot, dessen Höhe in der Regel seinem Mindestpreis LLP_K entspricht. Und sein Maximalpreis ULP_K (engl. *Upper Limit Price*) entspricht seiner maximalen Zahlungsbereitschaft, das heißt dem Preis, den er maximal für das Gut zu bieten bereit ist.

Nimmt der Agent hingegen als Verkäufer an der Verhandlung teil, so beginnt er die Verhandlung mit einem Startgebot, das in der Höhe in der Regel seinem Maximalpreis ULP_{VK} entspricht. Und sein Mindestpreis LLP_{VK} entspricht dem Preis, unterhalb dessen er nicht bereit ist, das Gut zu verkaufen.

Nachdem die Agenten ihre Mindest- und Maximalpreise und damit die Höhe ihrer initialen Gebote berechnet haben, kann der tatsächliche Start der Verhandlung erfolgen. Vor dem ersten Verhandlungsakt wird dabei noch durch einen fairen Münzwurf entschieden, ob dieser vom Käufer oder vom Verkäufer ausgeführt wird. Der erste Verhandlungsakt besteht dann jeweils darin, dass der Agent, auf den die Wahl fällt, dem anderen sein Startgebot übermittelt beziehungsweise einen initialen Preisvorschlag unterbreitet.

Fällt die Wahl auf den Agenten mit der Käufer-Rolle, so sendet dieser mittels einer PROPOSE-Nachricht mit dem ProposalType ASK seinen LLP_K als Startgebot an den als Verkäufer agierenden Agenten. Ist hingegen der als Verkäufer handelnde Agent als erster am Zug, so sendet dieser seinem Verhandlungspartner seinen ULP_{VK} als Startgebot (mittels einer PROPOSE-Nachricht mit dem ProposalType BID).

Der faire Münzwurf gewährleistet, dass jeder der Verhandlungspartner die gleiche Chance darauf besitzt, in der Verhandlung den ersten Preisvorschlag machen zu dürfen. Eine Chancengleichheit beziehungsweise Fairness ist an dieser Stelle wichtig, da der Verhandlungsteilnehmer, der seinen Startpreis als erster offenbart, hierdurch einen strategischen Nachteil haben kann. Dies ist möglich, wenn die zum Verhandlungsstart noch geheimen Startpreise der Verhandlungspartner bei ihrer Offenbarung für beide Akteure wechselseitig direkt ak-

zeptabel sind. In diesem Fall ist der Käufer bereit ist, ein Startgebot in Höhe von LLP_K abzugeben, das über dem vom Verkäufer maximal angestrebten Verkaufspreis ULP_{VK} liegt, den dieser selbst als Startgebot nennen würde.

Eröffnet der Verkäufer unter diesen Vorbedingungen die Verhandlung mit einem Startgebot in Höhe von ULP_{VK} , so wird der Käufer dieses direkt akzeptieren. Umgekehrt würde der Verkäufer aber auch ein Startgebot des Käufers in Höhe von LLP_K mit $LLP_K > ULP_{VK}$ sofort akzeptieren. Falls der Verkäufer in der Verhandlung das erste Gebot macht, verliert er dadurch also einen Umsatz in Höhe von $LLP_K - ULP_{VK}$.

5.3.3.4 Reaktion mit Gegenangeboten (PROPOSE-Fälle)

Im Folgenden werden die Aktivitäten eines Agenten K erläutert, der auf das aktuelle Angebot seines Verhandlungspartners VK in Höhe von $currentPrice_{VK}$ mit einem Gegenangebot $newPrice_K$ reagiert (sogenannte PROPOSE-Fälle). Falls der Agent K , der gerade am Zug ist, von seinem Opponenten VK ein Angebot erhalten hat,

- in dem der Opponent eine preisliche Konzession macht oder
- in dem der Opponent auf seinem Preis beharrt und keine Konzession anbietet, aber der von Agent K durchgeführte *Satisfaction-Check* (siehe Abschnitt 5.3.2) kein REJECT nach sich zieht,

so berechnet der Agent K ein Gegenangebot (PROPOSE).

Ein Satisfaction-Check bedeutet, dass der Agent eine stochastische Probe gegen den Wert des Gens Satisfaction beziehungsweise SF vornimmt, um zu entscheiden, ob er einen Verhandlungsabbruch vornimmt. Würfelt der Agent also eine Zufallszahl, die größer als SF ausfällt, so bricht der Agent die Verhandlung ab (siehe Abschnitt 5.3.3.5).

Da es sich nicht um ein PROPOSE zum Unterbreiten eines initialen Startgebotes handelt, signalisiert der Agent K mit einem PROPOSE im Kontext dieser Verhandlungssituation seinem Verhandlungspartner beziehungsweise Opponenten, dass er

- dessen aktuellen Preisvorschlag nicht akzeptiert und
- ihm stattdessen ein Gegenangebot mit dem Preisvorschlag $newPrice_K$ macht. Dieses Gegenangebot kann in der Änderung seines letzten Preisvorschlags $oldPrice_K$, aber auch im Beharren auf diesem bestehen.

Ob der der Agent K in seinem Gegenangebot tatsächlich eine preisliche Konzession anbietet, entscheidet dieser mit einer stochastischen Probe gegen den Wert seines Gens AQ beziehungsweise einem Acquisitiveness-Check, der in Abschnitt 5.3.2 erläutert wurde. Entscheidet sich der Agent K zu einem Gegengebot mit einer Konzession, so ändert er seinen letzten Preisvorschlag $oldPrice_K$ um eine Schrittweite $PriceStep_K$, mit der er seinem Verhandlungspartner VK preislich entgegen kommt.

Ein entsprechender Preisschritt wird von jedem der Agenten K und VK auf Basis des aktuellen Wertes ihres eigenen DC-Gens individuell bestimmt, nachdem ihnen ihr jeweiliger Opponent sein Startgebot mitgeteilt hat. Wie die Agenten K und VK ihre Konzessionsschrittweiten $PriceStep_K$ und $PriceStep_{VK}$ berechnen, wird in Abschnitt 5.3.2 genau erläutert.

Die Berechnung eines Gegenangebots, das eine preisliche Konzession beinhaltet, erfolgt von dem Agenten, der als Käufer oder Verkäufer gerade am Zug ist, folgendermaßen:

1. PROPOSE-Fall 1 – Konzession des als Käufer handelnden Agenten:

Bei einer Konzession seitens des Käufers bewegt sich der Käufer auf den Verkäufer zu, indem er den Gebotspreis um einen Preis-Schritt mit der Schrittweite $PriceStep_K$ erhöht. Dabei gilt:

$$newPrice_K = \text{Min}(oldPrice_K + PriceStep_K, ULP_K, aULP_K, currentPrice_{VK})$$

Also wird berücksichtigt, dass der neue vom Käufer gebotene Preis $newPrice_K$ weder über dem ULP_K noch über dem $aULP_K$ oder dem aktuell vom Verkäufer beziehungsweise Opponenten verlangten Preis liegt.

2. PROPOSE-Fall 2 – Konzession des als Verkäufer handelnden Agenten:

Bei einer Konzession seitens des Verkäufers bewegt sich der Verkäufer auf den Käufer zu, indem er den verlangten Gebotspreis um einen Preisschritt $PriceStep$ vermindert. Dabei gilt:

$$newPrice_{VK} = \text{Max}(aLLP_{VK}, LLP_{VK}, oldPrice_{VK} - PriceStep_{VK}, opponentsCurrentPrice)$$

Also wird sichergestellt, dass der neue Preis $newPrice_{VK}$ weder unter seinem $aLLP_{VK}$ noch unter seinem LLP_{VK} liegt und der neue Preis nicht niedriger als der aktuell vom Käufer beziehungsweise Opponenten gebotene Preis ist.

5.3.3.5 Reaktion mit Verhandlungsabbrüchen (REJECT-Fälle)

Hat der Opponent ein Gebot gemacht, so wird dieses vom Agenten zunächst darauf geprüft, ob es sich um ein preislich unangemessenes Angebot beziehungsweise ein Wucherangebot handelt. In diesem Fall reagiert der Agent mit einem Verhandlungsabbruch (siehe REJECT-Fall 1 unten).

Darüber hinaus kann der Agent auch mit einem Verhandlungsabbruch reagieren, falls sein Opponent in seinem letzten Angebot keine Konzession macht und auf seinem letzten Preis beharrt. In diesem Fall führt der Agent einen Satisfaction-Check durch, der einen Verhandlungsabbruch nach sich ziehen kann (siehe REJECT-Fall 2 unten).

Falls der Agent sich zu einem Verhandlungsabbruch entscheidet, so bricht er die Verhandlung mit seinem Verhandlungspartner ab, indem er diesem eine REJECT-Nachricht sendet. Nach dem Senden oder Empfangen einer REJECT-Nachricht kehrt ein Agent in den Wartezustand seines Kontrollzyklus (siehe Abbildung 37 in Abschnitt 5.2.4) zurück. Die beiden REJECT-Fälle werden im Folgenden erläutert:

1. REJECT-Fall 1 – Wucher-Fall:

- a) In MACSIMA prüft ein Agent jedes Gebot (das heißt ein PROPOSE mit einem Preisvorschlag), das er von seinem Verhandlungspartner erhält, auf „Wucher“.
 - Der Verkäufer klassifiziert ein Gebot des Käufers als Wucher, wenn dieses unterhalb von $wAVG_{VK} * 0,2$ liegt.
 - Der Käufer klassifiziert ein Gebot des Verkäufers als Wucher, wenn dieses oberhalb von $wAVG_K * 1,8$ liegt.
- b) Wird der Preisvorschlag des Verhandlungspartners als Wucher-Angebot klassifiziert, so sendet der Agent seinem Opponenten eine REJECT-Nachricht.

2. REJECT-Fall 2 – nach negativem Satisfaction-Check:

Ein Agent führt einen Satisfaction-Check durch, wenn er von seinem Verhandlungspartner einen Preisvorschlag erhält, der von dessen letztem Angebot nicht abweicht. Führt dieser zu einem Verhandlungsabbruch, so wird eine REJECT-Nachricht (mit dem Attribut "rejected because of unsatisfaction") gesendet.

5.3.3.6 Erfolgreicher Verhandlungsabschluss (ACCEPT-Fälle)

Es können zwei Konstellationen unterschieden werden, die zu einem erfolgreichen Abschluss einer Verhandlung führen:

1. ACCEPT-Fall 1 – Direkte Überschneidung der Startpreise:

Ein Agent A hat bereits einen Startpreis berechnet, aber diesen noch nicht durch ein Startgebot an seinen Verhandlungspartner B übermittelt. Er hat aber schon das erste Gebot des Verhandlungspartners in Höhe von $Startgebot_B$ empfangen und stellt fest, dass sich der Preisvorschlag $Startgebot_B$ mit seinem eigenen Startpreis überschneidet. Dies bedeutet, dass entweder

- a) $Startgebot_B \leq Startpreis_A$ gilt, falls Agent A der Käufer ist oder
- b) $Startgebot_B \geq Startpreis_A$, falls Agent A der Verkäufer ist.

In beiden Fällen akzeptiert Agent A direkt das jeweilige Startgebot von Agent B .

2. ACCEPT-Fall 2 – Einigung im Verhandlungsverlauf:

Ein Agent A empfängt ein Gebot seines Verhandlungspartners B , das einen Preisvorschlag in Höhe des Wertes $currentPrice_B$ enthält. Agent A akzeptiert diesen Preisvorschlag, falls er ausgehend von seinem eigenen letzten Preisvorschlag $oldPrice_A$ zu einer Konzession in Höhe des Preisschritts $PriceStep$ bereit ist und dies zu einem Preisvorschlag $newPrice_A$ führen würde, der sich mit $currentPrice_B$ überschneidet.

In beiden ACCEPT-Fällen arbeitet der Agent A den sogenannten *crossedOffers-Fall* ab, der zu einem erfolgreichen Verhandlungsabschluss und den folgenden Aktivitäten der Agenten führt:

1. Agent A leitet eine Transaktion zum Abschlusspreis $currentPrice_B$ ein, indem er seinen $newPrice_A$ auf den Wert von $currentPrice_B$ setzt und seinem Opponenten B eine ACCEPT-Nachricht mit $newPrice_A$ sendet.
2. Beide Agenten beziehungsweise Käufer und Verkäufer setzen in ihrer PD für das Gut die Variable $lastAgreementPrice$ jeweils auf den Wert von $currentPrice_B$.
3. Käufer und Verkäufer aktualisieren den Wert der Variablen $wAVG$ in ihrer jeweiligen PD für das gehandelte Gut. Dies tun sie unter Verwendung des Wertes des Gens WM , der in dem Genotyp enthalten ist, mit dem der erfolgreiche Verhandlungsabschluss erzielt wurde:

$$wAVG_K = (lastAgreementPrice * WM_K) + wAVG_K * (1 - WM_K)$$

$$wAVG_{VK} = (lastAgreementPrice * WM_{VK}) + wAVG_{VK} * (1 - WM_{VK})$$

4. Beide Agenten wechseln den Genotyp, mit dem die Transaktion erreicht wurde, gegen einen neuen Genotyp aus, den sie aus dem Gen-Pool auswählen. Die Auswahl des Genotyps erfolgt entweder gemäß der Reihenfolge in der Pool-Datenstruktur, die sich bei der Generierung des Pools zufällig (in Abhängigkeit von den beim Lernen verwendeten genetischen Operatoren, siehe Abschnitt 5.4.3) ergeben hat. Oder sie erfolgt auf Basis des in Abschnitt 5.4.5.1 vorgestellten *Highest-Fitness-First-Verfahrens*.
5. Darauffolgend aktualisieren Käufer und Verkäufer die Variablen LLP_K und ULP_K sowie LLP_{VK} und ULP_{VK} und speichern die aktualisierten Werte in ihrer PD für das gehandelte Gut. Dies tun sie unter Verwendung des Wertes des Gens $Delta_Jump$, der in dem Genotyp enthalten ist, den der jeweilige als Käufer oder Verkäufer handelnde Agent in der nächsten Verhandlung einsetzen wird.

$$LLP_K = wAVG_K * (1 - Delta_Jump_K) \text{ und } ULP_K = wAVG_K * (1 + Delta_Jump_K)$$

$$LLP_{VK} = wAVG_{VK} * (1 - Delta_Jump_{VK}) \text{ und } ULP_{VK} = wAVG_{VK} * (1 + Delta_Jump_{VK})$$

6. Abschließend nehmen der Verkäufer und der Käufer vor einem neuerlichen Verhandlungsstart noch folgende Berechnungen vor, die gegebenenfalls zur einer weiteren Aktualisierung beziehungsweise Anpassung des LLP_{VK} und des ULP_K in der jeweiligen PD führen können:

- a) Der Verkäufer berechnet für das von ihm zu verkaufende Gut einen zu LLP_{VK} alternativen Mindest- beziehungsweise Reservationspreis RVP_{VK} mit

$$RVP_{VK} = (Input\text{-}Kosten_{VK} + Produktionskosten_{VK}) * (1 + ProfitWish_{VK})$$

Dabei wurde der Wert der Variable *ProfitWish* in der GUI eingestellt, so dass er in der Szenario-Definition (in der entsprechenden .sce-Datei) hinterlegt ist. Dieser Wert spezifiziert die Höhe der gewünschten Gewinnmarge, die ein Agent als Verkäufer in einer Preisverhandlung für ein Gut zu erzielen versucht. Die Standardeinstellung für die Variable *ProfitWish* beträgt zehn Prozent.

Falls der alternative Mindestpreis RVP_{VK} größer als der in der PD gespeicherte Wert LLP_{VK} ausfällt, so erhöht der als Verkäufer agierende Agent seinen LLP_{VK} auf den Wert von RVP_{VK} . So stellt der Verkäufer sicher, dass der von ihm verwendete Wert von LLP_{VK} kostendeckend ist und zusätzlich die von ihm gewünschte Gewinnmarge beinhaltet.

- b) Analog hierzu kann der Käufer des Gutes auf der Basis seiner Verhandlungserfahrungen wissen, dass er für das Gut nicht den vollen $oULP_K$ bezahlen kann. Denn er muss seine Zahlungsbereitschaft ULP_K für ein Input-Gut davon abhängig machen, zu welchem Preis er ein von ihm mit dem Input-Gut produziertes Gut auf dem Markt absetzen kann. Fällt der Marktpreis für das von ihm produzierte Gut, so muss er auch seine Zahlungsbereitschaft für das Input-Gut auf einen Wert ULP'_K absenken.
- c) In weiteren Ausnahmefällen kann auch der Preisautomat Anpassungen von LLP_{VK} und des ULP_K (sowie von ULP_K und LLP_K) vornehmen. Die Funktionsweise des Preisautomaten wird im folgenden Abschnitt 5.3.3.7 beschrieben.

5.3.3.7 Preisautomat

Ein Agent befindet sich in der Rolle des Käufers oder Verkäufers eines Gutes und nimmt an einer Verhandlung in der entsprechenden Rolle teil. Sein Verhandlungsmodul wählt für den Agenten eine Verhandlungsstrategie aus, indem es im 2-Pool-Modus je nach seiner Rolle entweder einen Genotyp aus seinem Buy-Pool oder aus seinem Sell-Pool selektiert (siehe Abschnitt 5.4.1.2).

Handelt der Agent in der Rolle des Verkäufers, so bestimmt sein ULP (beziehungsweise der ULP_{VK}) sein Startgebot und sein LLP_{VK} seinen Mindest- beziehungsweise Indifferenz- oder Reservationspreis. Umgekehrt gibt der LLP_K das Startgebot des Käufers und ULP_K dessen Indifferenz- beziehungsweise Maximalpreis an.

Der *Preisautomat* überwacht zunächst, wie lange der Agent mit seiner aktuellen Verhandlungsstrategie keinen erfolgreichen Verhandlungsabschluss erreicht (Überwachungs- beziehungsweise *Monitoring-Funktion*). Dauert die Erfolglosigkeit der Verhandlungsstrategie länger an, so kann der Preisautomat zum Verhandlungsstart Anpassungen an den Preisvorstellungen des Agenten vornehmen. Denn es ist möglich, dass die Erfolglosigkeit damit zusammenhängt, dass der Agent die aktuelle Marktsituation beziehungsweise die momentan am Markt akzeptierten Güterpreise falsch einschätzt. In diesem Fall könnte es sein, dass es

aufgrund eines ungünstig gesetzten Startpreises, eines zu niedrig angesetzten Maximalpreises oder eines zu hohen Mindestpreises zu keiner Einigung zwischen den Verhandlungspartnern kommt.

Befindet sich der Agent in der Rolle des Verkäufers, so kann der Automat seinen Startpreis ULP_{VK} und seinen Mindestpreis LLP_{VK} senken. Befindet sich der Agent in der Rolle des Käufers, so kann der Automat seinen Startpreis LLP_K und seinen Maximalpreis ULP_K erhöhen. Hieraus ergeben sich die vier grundlegenden Anwendungsfälle

- Senkung des Startpreises ULP_{VK} des Verkäufers,
- Senkung des Mindestpreises LLP_{VK} des Verkäufers,
- Erhöhung des Startpreises LLP_K des Käufers und
- Erhöhung des Maximalpreises ULP_K des Käufers,

die in den untenstehenden Abschnitten erläutert werden.

Jeder der vier grundlegenden Anwendungsfälle gliedert sich in sechs Phasen. Diese werden erreicht, wenn bestimmte Grenzwerte für die Anzahl an aufeinanderfolgenden Verhandlungsabbrüchen (REJECTS) erreicht beziehungsweise überschritten werden. Dazu verwaltet der Automat für das Zählen der fortlaufenden REJECTS, die der Agent seit seiner letzten erfolgreichen Verhandlung erhalten hat, eine entsprechende Variable.

Die Vorbedingungen der sechs Phasen sind jeweils die folgenden:

1. if (*Anzahl aufeinanderfolgender Verhandlungsabbrüche* ≤ 5) {Phase 1}
2. else if ($5 < \textit{Anzahl aufeinanderfolgender Verhandlungsabbrüche} \leq 10$) {Phase 2}
3. else if ($10 < \textit{Anzahl aufeinanderfolgender Verhandlungsabbrüche} \leq 15$) {Phase 3}
4. else if ($15 < \textit{Anzahl aufeinanderfolgender Verhandlungsabbrüche} \leq 25$) {Phase 4}
5. else if ($25 < \textit{Anzahl aufeinanderfolgender Verhandlungsabbrüche} \leq 35$) {Phase 5}
6. else {Phase 6}

In Phase 1, das heißt bei bis zu fünf Verhandlungsabbrüchen, greift der Automat in keinem der Anwendungsfälle ein und lässt ULP_{VK} , LLP_{VK} , ULP_K und LLP_K jeweils unverändert.

Die Phasen 2 bis 5 der Anwendungsfälle sind die entscheidenden des Automaten. Sie versuchen, abzuschätzen, ob fortlaufende Verhandlungsmisserfolge eines Genotyps auf einen ungünstig gewählten Start-, Mindest- oder Maximalpreis des Agenten zurückzuführen sind. Ist dies wahrscheinlich, so nimmt der Automat eine schrittweise Anpassung der entsprechenden Preise an die aktuellen Marktgegebenheiten vor. Mit zunehmender Anzahl an aufeinanderfolgenden Verhandlungsabbrüchen greift der Preisautomat in den Phasen 2 bis 5 dabei immer stärker ein. Befindet sich der Agent in der Rolle des Verkäufers, so kann der Automat zum Beispiel in Phase 5 des Anwendungsfalls *Senkung des Mindestpreises LLP_{VK} des Verkäufers* sogar eine Art „Schlussverkauf“ einleiten.

Führt das schrittweise Eingreifen des Preisautomaten in mehr als 35 aufeinanderfolgenden Verhandlungen nicht zum Erfolg, so tritt in allen Anwendungsfällen schließlich die Phase 6 ein. Diese ist für alle Anwendungsfälle identisch und sorgt dafür, dass die bislang nicht erfolgreiche Verhandlungsstrategie beziehungsweise der aktuell verwendete Genotyp gegen einen anderen Genotypen aus dem Buy- oder Sell-Pool ausgetauscht wird. Sobald in einer Phase der Anwendungsfälle ein Verkaufsabschluss erzielt wird, wird der Preisautomat wieder in seinen Ausgangszustand versetzt und in Phase 1 neu gestartet. Solange kein Abschluss erzielt wurde, läuft der Automat in jedem der Anwendungsfälle bis Phase 6 weiter.

5.3.3.7.1 Kennzahlen des Preisautomaten

In den folgenden Abschnitten werden die Phasen 2 bis 5 der verschiedenen Anwendungsfälle beschrieben. In allen Anwendungsfällen berechnet der Automat einen *meanOpponentQuotient*. Für die letzten 10 abgebrochenen Verhandlungen merkt sich jeder Agent die letzten darin abgegebenen Gebote von sich selbst und seinem Opponenten beziehungsweise von Verkäufer und Käufer und berechnet daraus jeweils die Kennzahl

$$\text{OpponentQuotient} = \frac{\text{letztes_Gebot}_{VK}}{\text{letztes_Gebot}_K}$$

Der *meanOpponentQuotient* stellt das arithmetische Mittel dieser zehn Opponenten-Quotienten dar. Je weiter die Preisangebote in den letzten abgebrochenen Verhandlungen auseinander lagen, umso größer fällt also der *meanOpponentQuotient* aus.

Darüber hinaus berechnet der Automat aus der Anzahl an Verhandlungsabbrüchen, die der Käufer selbst initiiert hat und denen, die die jeweilige Gegenseite vorgenommen hat, einen Quotienten

$$\text{Abbruchanteil} = \frac{\text{Anzahl selbst abgebrochener Verhandlungen}}{\text{Gesamtzahl der abgebrochenen Verhandlungen}}$$

Gilt *Abbruchanteil* $< 0,5$, so brechen die Verhandlungspartner in Verhandlungen, bei denen der Agent seine aktuelle Verhandlungsstrategie nutzt, öfter ab als der Agent selbst. Darüber hinaus wird der Quotient

$$\text{Verhandlungsdauer} = \frac{\text{Anzahl empfangener Nachrichten}}{\text{Anzahl an Verhandlungsabbrüchen}}$$

bestimmt, der angibt, wie lange (beziehungsweise kurz) eine durchschnittliche Verhandlung dauert.

Beide Quotienten stellen einen Indikator dafür dar, dass eine Anpassung der Preisschranken des Agenten seine Chancen auf einen Abschluss erhöhen könnte. Je geringer der *Abbruchanteil* bei wechselnden Verhandlungspartnern ausfällt, umso mehr deutet dies darauf hin, dass die Preisgrenzen *LLP* und *ULP* des Agenten sich nicht ausreichend an dem aktuellen Marktpreis für das Gut orientieren. Eine geringe Verhandlungsdauer deutet ebenfalls darauf hin, wobei dies vor allem vermuten lässt, dass in der Verkäufer-Rolle der Startpreis *ULP_{VK}* beziehungsweise in der Käufer-Rolle der Startpreis *LLP_K* ungünstig gewählt sein könnte.

Je kleiner die Quotienten *Abbruchanteil* und *Verhandlungsdauer* ausfallen, umso größer wird der Wert der Variablen *stronger*, die bei der Berechnung eines Preisschritts *PriceStep* in den Phasen 2 und 3 der Anwendungsfälle (siehe unten) Verwendung findet. Dieser Preisschritt wird jeweils zu Beginn von Phase 2 und Phase 3 bestimmt und dann für alle Preisadjustierungen innerhalb der jeweiligen Phase beibehalten.

In Abhängigkeit von den Werten der Quotienten liegt *stronger* in Phase 2 im Intervall $[1,0; 1,2]$ und nimmt den maximalen Wert von 1,2 an, wenn die Bedingungen *Abbruchanteil* $< 0,5$ und *Verhandlungsdauer* ≤ 3 gleichzeitig erfüllt sind. Ist nur eine dieser Bedingungen erfüllt, nimmt *stronger* den Wert von 1,1 an. Ansonsten wird *stronger* mit dem Wert 1,0 initialisiert.

In Phase 3 liegt der Wert der Variablen *stronger* in Abhängigkeit von den Werten der Quotienten *Abbruchanteil* und *Verhandlungsdauer* im erweiterten Intervall $[1,0; 1,3]$. Ihren maximalen Wert von 1,3 erreicht sie bereits in Phase 3, wenn die Bedingungen *Abbruchanteil* $< 0,75$ und *Verhandlungsdauer* ≤ 4 gleichzeitig erfüllt sind. Gilt nur eine dieser Bedingungen, so wird sie in Phase 3 mit dem Wert 1,2 belegt. Ansonsten erhält sie

den Wert 1,0. Dadurch sind die Bedingungen, unter denen ein größerer Preisschritt möglich ist, in Phase 3 schwächer gestaltet als in Phase 2.

Oft stehen dem Preisautomaten auch in Form von Plumages externe Preisinformationen über das Gut zur Verfügung, über dessen Verkauf der Agent gerade verhandelt. Ist dies der Fall, so kann der Preisautomat je nach Anwendungsfall verschiedene Kennzahlen berechnen und nutzen. Die dazu notwendigen Informationen entnimmt er den Plumages, die der Agent von anderen Agenten empfangen hat. Der Informationsgehalt eines Plumage wird in Abschnitt 5.4.1 im Detail beschrieben. Jedes Plumage enthält neben der Verhandlungsstrategie beziehungsweise dem Genotyp und seiner Fitness-Bewertung (siehe Abschnitt 5.4.2) unter anderem auch die folgenden für den Preisautomaten wesentlichen Informationen:

- den Wert *roundsForCurrentGenotype*, der die Anzahl der Simulationsrunden angibt, die der Genotyp bis zum Erreichen des Abschlusses benötigt hat,
- den Namen *goodName* des Gutes, das verkauft wurde,
- den Verkaufspreis *currentPrice*, der vereinbart wurde,
- den Wert *weighted_Average*, das heißt den gewichteten Durchschnittspreis für das Gut, den der das Plumage sendende Agent beobachtet hat sowie
- die Werte *minPrice* und *maxPrice*, die das kleinste und das größte Preisangebot angeben, die für das Gut im Verhandlungsverlauf abgegeben wurden. Bei *minPrice* handelt es sich dabei um den Startpreis des Käufers und bei *maxPrice* um den Startpreis des Verkäufers.

Auf dieser Grundlage berechnet der Preisautomat aus den Informationen der letzten dreißig Plumages, die der Agent von anderen Agenten empfangen hat und die sich auf den Verkauf des entsprechenden Gutes beziehen, die folgenden Kennzahlen:

- den (gleitenden) Mittelwert *gmExternalPriceData* der aus den Plumages bekannten Verkaufspreise des Gutes.
- den (gleitenden) Mittelwert *gmExternalMaxPriceData* der aus den Plumages bekannten Startpreise, mit denen Verkäufer des Gutes die Verhandlung begonnen haben. Die Startpreise der Verkäufer entsprechen dabei den *maxPrice*-Daten der Plumages.
- den (gleitenden) Mittelwert *gmExternalMinPriceData* der aus den Plumages bekannten Startpreise, mit denen Käufer des Gutes die Verhandlung begonnen haben. Die Startpreise der Käufer entsprechen dabei den *minPrice*-Daten der Plumages.

5.3.3.7.2 Senkung des Startpreises ULP_{VK} des Verkäufers

Ein Grund für häufige Abbrüche könnte der ULP_{VK} sein, also die obere Preisschranke, mit der der Verkäufer in eine Verhandlung startet. Diese wird vom Verkäufer als Startpreis genannt. Ist dieser ULP_{VK} relativ hoch angesetzt, so kann es sein, dass die Gegenseite die Verhandlungen direkt abbricht, weil sie diesen Preis als zu hoch ansieht. Dies könnte zum Beispiel der Fall sein, wenn der ULP_{VK} die derzeit am Markt üblichen Preise deutlich übersteigt.

Ausgehend von einem hohen ULP_{VK} muss ein Verkäufer gegebenenfalls auch mehrere Konzessionen eingehen beziehungsweise den von ihm geforderten Verkaufspreis mehrmals senken, bis er eine Einigung mit dem Käufer erreichen kann. Dies lässt die Abbruchwahrscheinlichkeit ansteigen, weil es in einer länger dauernden Verhandlung häufiger dazu kommen

kann, dass einer der Verhandlungspartner in einer Verhandlungsrunde keine Konzession eingeht. In diesem Fall ist es möglich, dass der Verhandlungsopponent darauf nach der Durchführung einen Satisfaction-Checks mit einem Verhandlungsabbruch reagiert.

Das grundlegende Interesse des Verkäufers besteht darin, zumindest keinen Verlust und am besten einen Gewinn zu erzielen. Er bietet dem potenziellen Käufer am Beginn einer Verhandlung daher einen Startpreis ULP_{VK} an, der in der Regel mindestens seinem Indifferenzbeziehungswise Reservationspreis RVP_{VK} (siehe Aktivität 6 a) in Abschnitt 5.3.3.6) entspricht. Liegt der Reservationspreis RVP_{VK} oberhalb von $oULP_{VK}$, so startet der Verkäufer die Verhandlung mit $ULP_{VK} = RVP_{VK}$, ansonsten mit $ULP_{VK} = oULP_{VK}$.

Wie der Startpreis ULP_{VK} des Verkäufers in den verschiedenen Phasen des Anwendungsfalls schrittweise gesenkt werden kann, ist in Abbildung 40 skizziert.

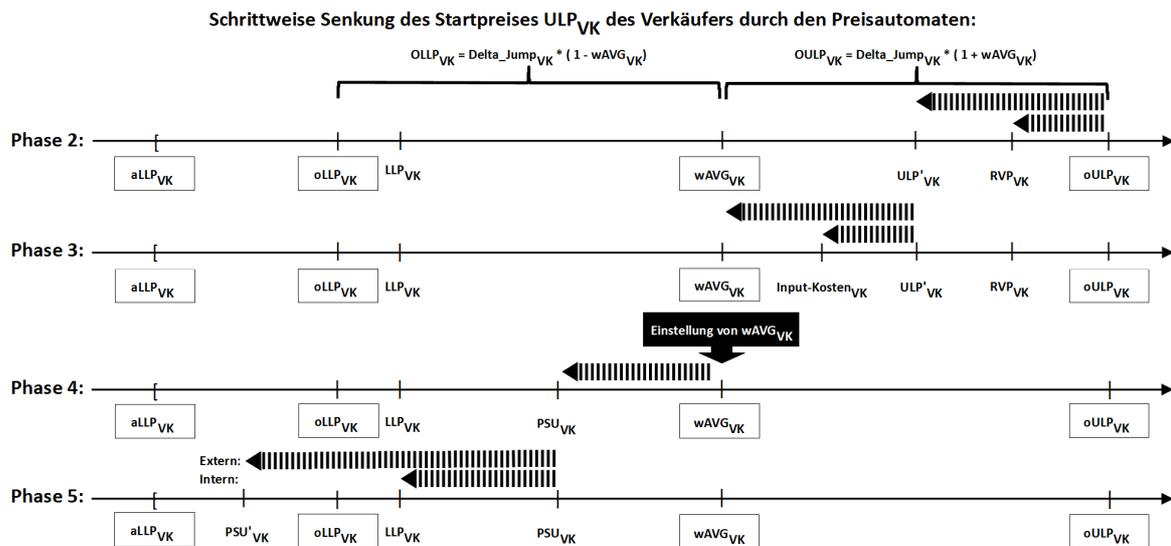


Abbildung 40: Die Phasen des Preisautomaten für den Startpreis des Verkäufers

In Phase 2 wird auf Basis der Differenz zwischen dem Startpreis ULP_{VK} und dem $wAVG_{VK}$ ein Preisschritt berechnet, mit dem der Startpreis im Verlauf von Phase 2 bei jedem neuerlichen Verhandlungsabbruch schrittweise weiter in Richtung des $wAVG_{VK}$ gesenkt wird. Sollte dabei $wAVG_{VK} \leq RVP_{VK}$ gelten, so kann der Startpreis im Verlauf von Phase 2 nur bis maximal zum RVP_{VK} gesenkt werden, ansonsten noch weiter bis hin zum $wAVG_{VK}$. Die Motivation für diese Einschränkung besteht darin, dass in Phase 2 nur eine Anpassung des Startpreises erfolgen soll, die noch die Chance beinhaltet, einen Gewinn zu realisieren.

Der ULP_{VK} wird bei jedem Durchlauf von Phase 2 jeweils um einen Preisschritt $PriceStep$ gesenkt und aktualisiert. Dabei wird der Preisschritt anhand der Formel

$$PriceStep = \frac{ULP_{VK} - wAVG_{VK}}{10} * meanOpponentQuotient * stronger$$

berechnet. Je weiter die Preisangebote in den letzten zehn abgebrochenen Verhandlungen auseinander lagen, umso größer fallen der $meanOpponentQuotient$ und damit auch der in Phase 2 gegebenenfalls zu tätige Preisschritt aus. Je kleiner die Quotienten $Abbruchanteil$ und $Verhandlungsdauer$ ausfallen, umso größer wird der Wert der Variablen $stronger$, die in Abhängigkeit von den Werten der Quotienten im Intervall $[1,0; 1,2]$ liegt, und damit der Preisschritt $PriceStep$.

Aufgrund der Senkungsschritte, die für die Preisschranke ULP_{VK} in Phase 2 durchgeführt werden, wird am Ende von Phase 2 ein ULP'_{VK} erreicht, der noch oberhalb vom $wAVG_{VK}$ liegt.

Phase 3 verläuft im Wesentlichen analog zu Phase 2. Jedoch geht Phase 3 nicht von ULP_{VK} sondern vom ULP'_{VK} aus. Statt auf Basis der Differenz ($ULP_{VK} - wAVG_{VK}$) wird der Preisschritt in Phase 3 daher auf der Grundlage der Differenz ($ULP'_{VK} - wAVG_{VK}$) berechnet. Zudem wird der Startpreis in Phase 3 nur bis maximal zur Höhe der *Input-Kosten* $_{VK}$ herabgesetzt, falls diese den $wAVG_{VK}$ übersteigen. Im Unterschied zu Phase 2 wird also nicht mehr versucht, unbedingt einen Gewinn zu erzielen oder die Produktionskosten zu decken. Ansonsten kann in Phase 3 maximal der $wAVG_{VK}$ erreicht werden. Zudem liegt in Phase 3 der Wert der Variablen *stronger* in Abhängigkeit von den Werten der Quotienten *Abbruchanteil* und *Verhandlungsdauer* im erweiterten Intervall [1,0; 1,3] (siehe Abschnitt 5.3.3.7.1).

Falls am Ende von Phase 3 die Preisgrenze $wAVG_{VK}$ noch nicht erreicht wurde, so wird zu Beginn von Phase 4 der zu verwendende Startpreis ULP_{VK} auf den Wert von $wAVG_{VK}$ aktualisiert. Danach berechnet der Verkäufer in Phase 4 eine Preisschranke PSU_{VK} , bis zu der der ULP_{VK} im Fall 4 maximal gesenkt werden darf.

Hat der Verkäufer keine externen Preisinformationen von anderen Agenten empfangen, so wird der Unterfall *Phase-4-intern* abgearbeitet, in dem

$$PSU_{VK} = \frac{LLP_{VK} + ULP_{VK}}{2}$$

gilt. In der Regel entspricht der LLP_{VK} dabei dem $oLLP_{VK}$. Da bei der Bestimmung des LLP_{VK} aber die tatsächlichen Input-Kosten berücksichtigt werden, kann er den $oLLP_{VK}$ auch übersteigen. Unabhängig davon wird der Startpreis ULP_{VK} in diesem Unterfall schrittweise bis maximal zu dem Wert von PSU_{VK} vermindert.

Stehen jedoch externe Preisinformationen zur Verfügung, so greift der Unterfall *Phase-4-extern*. In diesem setzt der Preisautomat die Preisschranke PSU_{VK} auf den Wert $gmExternalMaxPriceData$, das heißt den Mittelwert der letzten dreißig für das Gut bekannten Startpreise anderer Verkäufer. Dann senkt der Preisautomat den Startpreis ULP_{VK} schrittweise nur bis maximal zum Wert von PSU_{VK} ab.

Der Preisschritt *PriceStep*, um den der aktuell gültige Startpreis des Verkäufers jeweils gesenkt wird, wird dabei sowohl in *Phase-4-intern* als auch in *Phase-4-extern* anhand der folgenden Formel berechnet:

$$PriceStep = \frac{ULP_{VK} - PSU_{VK}}{10} * meanOpponentQuotient$$

In Phase 5 wird ebenfalls zwischen einem internen und externen Unterfall unterschieden. In *Phase-5-intern* sind keine externen Preisinformationen verfügbar und der Startpreis wird ausgehend von dem Startpreis, der in Phase 4 erreicht wurde bis zu einem Wert gesenkt, der fünf Prozent über LLP_{VK} liegt.

Sind Informationen über die Verkaufspreise des Gutes verfügbar, die in anderen Verhandlungen erzielt werden konnten, so wird aus den dreißig letzten bekannten Verkaufspreisen der Mittelwert $gmExternalPriceData$ berechnet und als Preisschranke PSU'_{VK} verwendet. Falls $aLLP_{VK} \leq PSU'_{VK}$ gilt, so senkt der Automat im Unterfall *Phase-5-extern* den Mindestpreis schrittweise nur bis maximal zum Wert von PSU'_{VK} ab, ansonsten nur bis zu einem Preis, der fünf Prozent über LLP_{VK} liegt.

Der in den beiden Unterfällen jeweils gültige *PriceStep* wird analog zu Phase 4 bestimmt:

$$PriceStep_{intern} = \frac{ULP_{VK} - LLP_{VK}}{5} * meanOpponentQuotient$$

$$PriceStep_{extern} = \frac{ULP_{VK} - PSU'_{VK}}{5} * meanOpponentQuotient$$

5.3.3.7.3 Senkung des Mindestpreises LLP_{VK} des Verkäufers

Der Mindestpreis des Verkäufers LLP_{VK} wird erst am Verhandlungsende relevant. Bevor der Verkäufer seinen LLP_{VK} senkt, wird daher überprüft, ob es wahrscheinlich ist, dass eine Senkung des geforderten Mindestpreises für einen erfolgreichen Verhandlungsabschluss notwendig ist. Die Variable *lowerBoundReached* zählt, wie oft eine Verhandlung vom Opponenten beim Erreichen von LLP_{VK} abgebrochen wurde. LLP_{VK} wird in den folgenden Phasen des Anwendungsfalls daher nur gesenkt, wenn vorhergehende Verhandlungen beim Erreichen des Mindestpreises hinreichend oft abgebrochen wurden. In der Regel ist dies der Fall, wenn $lowerBoundReached \geq 3$ gilt.

Das Niveau des Mindestpreises LLP_{VK} , den ein Verkäufer für ein Gut verlangt, kann dafür verantwortlich sein, dass viele Verhandlungen von potenziellen Käufern abgebrochen werden. So kommt es vor, dass ein Käufer eine Konzession wünscht, der Verkäufer aber keinen weiteren Preissenkungsschritt durchführt, da er damit seinen Mindestpreis LLP_{VK} , unter dem er nicht zu einem Verkauf bereit ist, unterschreiten würde. Ein Grund dafür könnte sein, dass der LLP_{VK} verhältnismäßig hoch angesetzt ist, da der Verkäufer die Input-Güter teuer eingekauft hat. Der LLP_{VK} kann daher vom Preisautomaten in den Phasen 2 bis 5 dieses Anwendungsfalls schrittweise gesenkt werden. Welche Anpassungen des Mindestpreises in den verschiedenen Phasen vorgenommen werden können, ist in Abbildung 41 veranschaulicht.

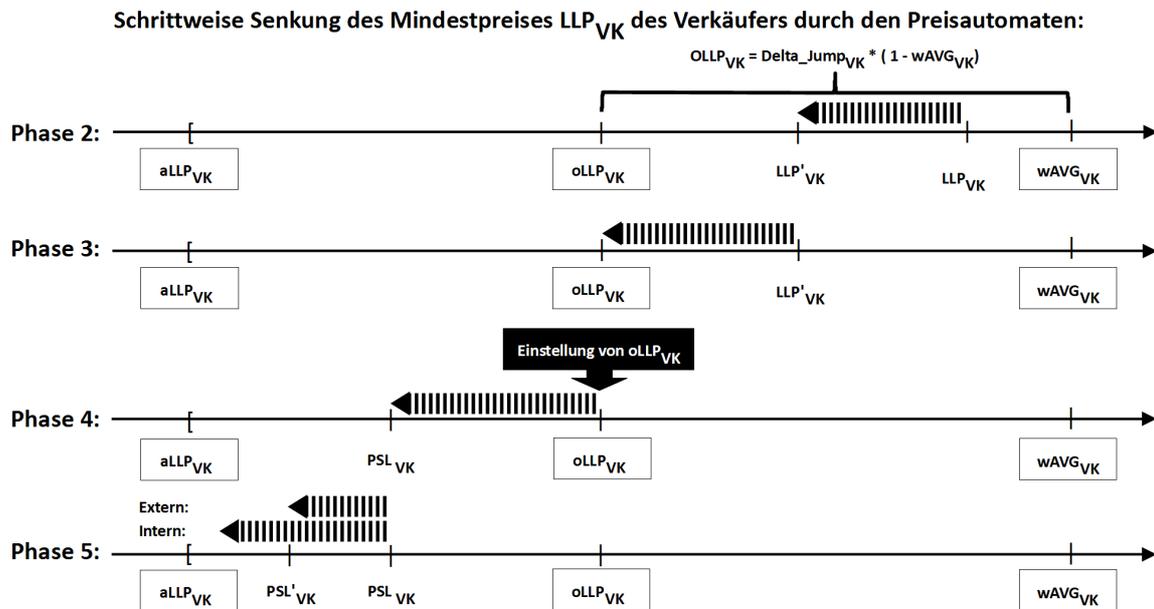


Abbildung 41: Die Phasen des Preisautomaten für den Mindestpreis des Verkäufers

In den Phasen 2 und 3 wird der LLP_{VK} , solange er oberhalb des $oLLP_{VK}$ liegt, schrittweise in Richtung des $oLLP_{VK}$ gesenkt, aber nicht darüber hinaus. Wenn der LLP_{VK} gesenkt wird, so erfolgt dies bei jedem Durchlauf von Phase 2 jeweils um einen Preisschritt *PriceStep*, der anhand der folgenden Formel berechnet wird:

$$PriceStep = \frac{LLP_{VK} - oLLP_{VK}}{10} * meanOpponentQuotient * stronger$$

Je geringer die Quotienten *Abbruchanteil* und *Verhandlungsdauer* ausfallen, umso größer wird der Wert der Variablen *stronger*, die im Intervall $[1,0; 1,2]$ liegt (siehe Abschnitt 5.3.3.7.1). Aufgrund von Senkungsschritten, die für die Preisschranke LLP_{VK} in Phase 2 durchgeführt werden, wird am Ende von Phase 2 ein LLP'_{VK} erreicht.

Phase 3 verläuft analog zu Phase 2, nur geht Phase 3 nicht von LLP_{VK} sondern vom LLP'_{VK} aus. Statt auf Basis der Differenz $(LLP_{VK} - oLLP_{VK})$ wird der Preisschritt in Phase 3 daher auf

der Grundlage der Differenz ($LLP'_{VK} - oLLP_{VK}$) berechnet. Außerdem liegt in Phase 3 der Wert der Variablen *stronger* in Abhängigkeit von den Werten der Quotienten *Abbruchanteil* und *Verhandlungsdauer* im erweiterten Intervall $[1,0; 1,3]$ (siehe Abschnitt 5.3.3.7.1).

Zu Beginn von Phase 4 stellt der LLP_{VK} zunächst die Preisschranke $oLLP_{VK}$ ein, falls diese nicht schon in Phase 3 erreicht wurde. Danach berechnet der Verkäufer eine neue Preisschranke PSL_{VK} . Hat der Verkäufer keine externen Preisinformationen von anderen Agenten empfangen, so wird der Unterfall *Phase-4-intern* abgearbeitet, in dem

$$PSL_{VK} = \frac{oLLP_{VK} + aLLP_{VK}}{2}$$

gilt. Stehen jedoch externe Preisinformationen zur Verfügung, so greift der Unterfall *Phase-4-extern*. In diesem entspricht PSL_{VK} im Unterschied zum Unterfall *Phase-4-intern* dem Mittelwert $gmExternalPriceData$ der letzten dreißig Verkaufspreise für das Gut, die in den vom Verkäufer empfangenen Plumages (siehe Abschnitt 5.4.1) gespeichert waren.

Der Preisschritt *PriceStep*, um den der aktuell gültige Mindestpreis des Verkäufers jeweils gesenkt wird, wird dabei sowohl in *Phase-4-intern* als auch in *Phase-4-extern* anhand der folgenden Formel berechnet:

$$PriceStep = \frac{LLP_{VK} - PSL_{VK}}{10} * meanOpponentQuotient$$

Innerhalb von Phase 4 führt der Verkäufer anschließend eine schrittweise Reduktion seines Mindestpreises in Richtung und bis maximal PSL_{VK} durch.

Stehen in Phase 5 keine externen Preisinformationen zur Verfügung, so senkt der Preisautomat im Unterfall *Phase-5-intern* den Mindestpreis des Verkäufers ausgehend von dem in Phase 4 erreichten aktuellen Wert weiter schrittweise in Richtung des absoluten Mindestpreises $aLLP_{VK}$.

Stehen hingegen externe Preisinformationen zur Verfügung, so arbeitet der Preisautomat den Unterfall *Phase-5-extern* ab. Aus den dreißig letzten Startpreisen von Käufern, die dem Verkäufer aus empfangenen Plumages bekannt sind, berechnet der Preisautomat den Durchschnitt $gmExternalMinPriceData$ und setzt PSL'_{VK} auf diesen Wert. Falls die Bedingung $aLLP_{VK} \leq PSL'_{VK}$ gilt, so senkt der Automat in *Phase-5-extern* den Mindestpreis nur bis maximal zum Wert von PSL'_{VK} ab, ansonsten ebenfalls bis hin zu $aLLP_{VK}$.

Der in den beiden Unterfällen jeweils gültige *PriceStep* wird analog zu Phase 4 bestimmt:

$$PriceStep_{intern} = \frac{LLP_{VK} - aLLP_{VK}}{5} * meanOpponentQuotient$$

$$PriceStep_{extern} = \frac{LLP_{VK} - PSL'_{VK}}{5} * meanOpponentQuotient$$

5.3.3.7.4 Erhöhung des Startpreises LLP_K des Käufers

Ist ein Agent in der Rolle als Käufer bei Verhandlungen nicht erfolgreich, so könnte dies auch daran liegen, dass er seine Verhandlungen mit einem zu geringen Startpreis beziehungsweise Erstgebot beginnt.

Zum einen könnte dies einen Verkäufer dazu bringen, die Verhandlung direkt abzubrechen, wenn dieses zu weit von den derzeit am Markt erzielbaren Preisen entfernt ist. Zum anderen führt ein geringes Einstiegsgebot im Durchschnitt zu etwas längeren Verhandlungen, bei denen die Gefahr erhöht ist, dass einer der Verhandlungspartner aus mangelnder Geduld nach einem Satisfaction-Check abbricht.

Wie der Startpreis des Käufers LLP_K in den Phasen 2 bis 5 schrittweise immer stärker erhöht wird und wie die verschiedenen Phasen aufeinander aufbauen, ist in Abbildung 42 skizziert.

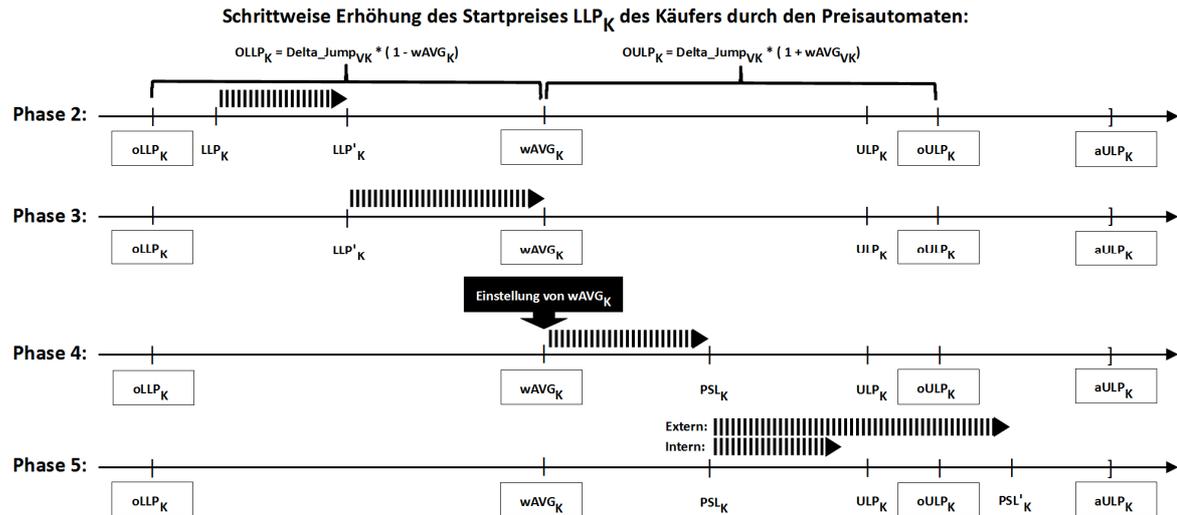


Abbildung 42: Die Phasen des Preisautomaten für den Startpreis des Käufers

In den Phasen 2 und 3 wird der LLP_K , solange er unterhalb des $wAVG_K$ liegt, schrittweise in Richtung des $wAVG_K$ erhöht, aber nicht darüber hinaus. Der aktuelle LLP_K wird bei jedem Durchlauf von Phase 2 jeweils um einen Preisschritt erhöht, der nach der Formel

$$PriceStep = \frac{wAVG_K - LLP_K}{10} * meanOpponentQuotient * stronger$$

berechnet wird. Die Berechnung des *meanOpponentQuotient* und die Bestimmung der Variablen *stronger* erfolgt wie in den vorhergehend erläuterten Anwendungsfällen. Aufgrund der Erhöhungsschritte, die für die Preisschranke LLP_K in Phase 2 durchgeführt werden, wird am Ende von Phase 2 ein LLP'_K erreicht.

Phase 3 verläuft im Wesentlichen analog zu Phase 2, jedoch geht Phase 3 nicht von LLP_K sondern vom LLP'_K aus. Statt auf Basis der Differenz ($wAVG_K - LLP_K$) wird der Preisschritt in Phase 3 auf der Grundlage der Differenz ($wAVG_K - LLP'_K$) berechnet.

Zu Beginn von Phase 4 setzt der Käufer seinen LLP_K zunächst auf den Wert von $wAVG_K$, falls dieser nicht schon in Phase 3 erreicht wurde. Danach berechnet der Käufer eine neue Preisschranke PSL_K . Hat der Verkäufer keine externen Preisinformationen von anderen Agenten empfangen, so wird der Unterfall *Phase-4-intern* abgearbeitet, in dem

$$PSL_K = \frac{ULP_K + LLP_K}{2}$$

gilt. Stehen jedoch externe Preisinformationen zur Verfügung, so greift der Unterfall *Phase-4-extern*. In diesem entspricht PSL_K im Unterschied zum Unterfall *Phase-4-intern* dem gleitenden Mittelwert *gmExternalMinPriceData*, der sich aus den Informationen über die Startpreise anderer Käufer in jeweils erfolgreich abgeschlossenen Verhandlungen ergibt, die in den letzten dreißig vom Käufer empfangenen Plumages (siehe Abschnitt 5.4.1) enthalten sind.

Innerhalb von Phase 4 führt der Käufer anschließend eine schrittweise Erhöhung seines aktuellen Startpreises bis maximal zum Wert von PSL_K durch. Der Preisschritt, um den der aktuell gültige Mindest- beziehungsweise Startpreis des Käufers jeweils erhöht wird, wird dabei sowohl in *Phase-4-intern* als auch in *Phase-4-extern* anhand der folgenden Formel berechnet:

$$PriceStep = \frac{PSL_K - LLP_K}{10} * meanOpponentQuotient$$

Als Ergebnis wird am Ende von Phase 4 ein Startpreis LLP_K erreicht, der PSL_K einstellt oder knapp darunter liegt.

In Phase 5 wird ebenfalls zwischen einem internen und externen Unterfall unterschieden. Stehen in Phase 5 keine externen Preisinformationen zur Verfügung, so erhöht der Preisautomat im Unterfall *Phase-5-intern* den Startpreis des Verkäufers ausgehend von dem in Phase 4 erreichten Wert LLP_K weiter schrittweise bis hin zu einem Maximalwert, der fünf Prozent unterhalb des aktuellen ULP_K liegt.

Stehen hingegen externe Preisinformationen zur Verfügung, so arbeitet der Preisautomat den Unterfall *Phase-5-extern* ab. Sind Informationen über die letzten dreißig Verkaufspreise verfügbar, zu denen andere Käufer das gewünschte Gut kaufen konnten, so wird aus diesen in *Phase-5-extern* der (gleitende) Mittelwert $gmExternalPriceData$ berechnet und als Preis-schranke PSL'_K verwendet. Anschließend erhöht der Automat den Startpreis LLP_K schrittweise bis maximal zum Wert von PSL'_K .

Der in den beiden Unterfällen jeweils gültige *PriceStep* wird analog zu Phase 4 bestimmt:

$$PriceStep_{intern} = \frac{ULP_K - LLP_K}{5} * meanOpponentQuotient$$

$$PriceStep_{extern} = \frac{PSL'_K - LLP_K}{5} * meanOpponentQuotient$$

5.3.3.7.5 Erhöhung des Maximalpreises ULP_K des Käufers

In diesem Anwendungsfall wird der Maximalpreis des Käufers ULP_K in den Phasen 2 bis 5 schrittweise immer stärker bis zu einem Maximalwert erhöht, der allerhöchstens fünf Prozent unterhalb von $aULP_K$ liegen darf. Wie die verschiedenen Phasen aufeinander aufbauen, ist in Abbildung 43 skizziert.

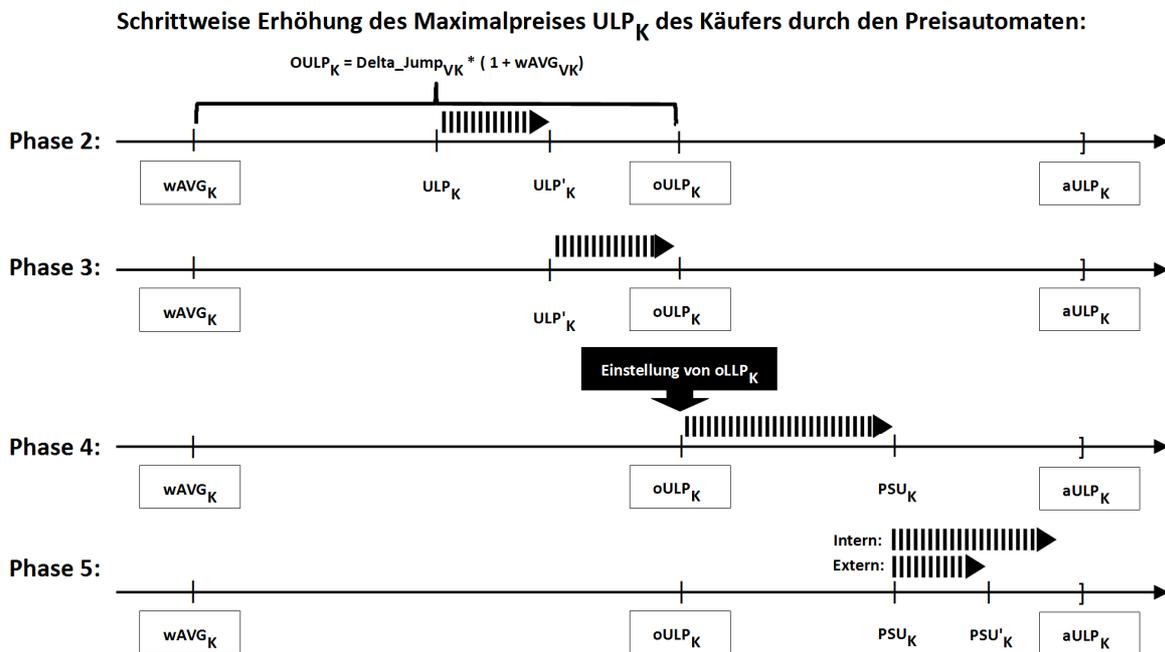


Abbildung 43: Die Phasen des Preisautomaten für den Maximalpreis des Käufers

Es ist dabei möglich, dass der Käufer seinen ULP_K für ein Input-Gut auf einen Wert unterhalb von $oULP_K$ vermindert hat. Dies kann notwendig sein, wenn der Käufer Kenntnis davon hat, dass er sein Output-Gut, für dessen Produktion er das Input-Gut benötigt, unter Berücksichtigung der aktuellen Marktpreise wahrscheinlich nicht zu einem Preis verkaufen kann, der für das Input-Gut einen Einkaufspreis in Höhe von $oULP_K$ rechtfertigt.

Mit der Variablen *upperBoundReached* wird gezählt, wie oft der Käufer bei den Verhandlungen, die er auf Basis seiner aktuellen Verhandlungsstrategie geführt hat, seine maximale Zahlungsbereitschaft ausgeschöpft beziehungsweise seinen Maximalpreis ULP_K geboten hat und dennoch keinen Abschluss tätigen konnte. Ein hoher Wert von *upperBoundReached* stellt daher einen Indikator dafür dar, dass der Käufer seine Chance auf einen Verhandlungserfolg durch die Erhöhung seines ULP_K verbessern könnte.

Der ULP_K wird daher vom Automaten in den folgenden Phasen 2 bis 5 nur dann erhöht, wenn $upperBoundReached \geq 3$ gilt. Ansonsten erfolgt in den jeweils maximal fünf Durchläufen dieser Phasen keine Anpassung des ULP_K .

In den Phasen 2 und 3 wird der ULP_K , solange er unterhalb des $oULP_K$ liegt, durch mehrere Preisanpassungsschritte in Richtung des $oULP_K$ erhöht, aber nicht darüber hinaus. Sind die Vorbedingungen für eine Erhöhung des ULP_K gegeben, so wird dieser bei jedem Durchlauf von Phase 2 jeweils um einen Preisschritt *PriceStep* erhöht und aktualisiert. Dabei wird der Preisschritt anhand der folgenden Formel berechnet:

$$PriceStep = \frac{oULP_K - ULP_K}{10} * meanOpponentQuotient * stronger$$

Die Berechnung des *meanOpponentQuotient* und die Bestimmung der Variablen *stronger* erfolgt wie in den vorhergehend erläuterten Anwendungsfällen. Aufgrund der Erhöhungsschritte, die für die Preisschranke ULP_K in Phase 2 durchgeführt werden, wird am Ende von Phase 2 ein ULP'_K erreicht. Phase 3 verläuft im Wesentlichen analog zu Phase 2, jedoch geht Phase 3 nicht von ULP_K sondern vom ULP'_K aus. Außerdem wird der Preisschritt in Phase 3 nicht auf Basis der Differenz ($oULP_K - ULP_K$), sondern auf der Grundlage der Differenz ($oULP_K - ULP'_K$) berechnet.

Zu Beginn von Phase 4 setzt der Käufer seinen ULP_K zunächst auf die Preisschranke $oULP_K$, falls diese nicht schon in Phase 3 erreicht wurde. Danach berechnet der Käufer eine Preisschranke PSU_K .

Hat der Käufer von koalierenden Agenten keine externen, in Plumages (siehe Abschnitt 5.4.1) enthaltenen Preisinformationen empfangen, so wird der Unterfall *Phase-4-intern* abgearbeitet, in dem folgende Preisschranke gilt:

$$PSU_K = \frac{aULP_K + oULP_K}{2}$$

Stehen jedoch externe Preisinformationen zur Verfügung, so greift der Unterfall *Phase-4-extern*. In diesem entspricht PSU_K im Unterschied zum Unterfall *Phase-4-intern* dem gleitenden Mittelwert *gmExternalPriceData*, der dem Mittelwert der letzten dreißig Verkaufspreise entspricht, den andere Käufer dem Agenten in den von ihnen gesendeten Plumages mitgeteilt haben. Liegt *gmExternalPriceData* zwischen $oULP_K$ und $aULP_K$, so deutet dies die Notwendigkeit an, den eigenen Maximalpreis weiter zu erhöhen, weil die durchschnittlich am Markt erzielbaren Verkaufspreise höher als der $oULP_K$ sind.

Innerhalb von Phase 4 führt der Käufer anschließend eine schrittweise Erhöhung seines aktuellen Maximalpreises ULP_K bis maximal zum Wert von PSU_K durch. Der Preisschritt *PriceStep*, um den der aktuell gültige Mindestpreis des Verkäufers jeweils gesenkt wird, wird dabei sowohl in *Phase-4-intern* als auch in *Phase-4-extern* anhand der folgenden Formel berechnet:

$$PriceStep = \frac{PSU_K - oULP_K}{10} * meanOpponentQuotient$$

Als Ergebnis wird am Ende von Phase 4 ein Maximalpreis ULP_K erreicht, der den jeweils gültigen PSU_K einstellt oder knapp darunter liegt.

In Phase 5 wird ebenfalls zwischen einem internen und externen Unterfall unterschieden. Stehen in Phase 5 keine externen Preisinformationen zur Verfügung, so kann der Preisautomat im Unterfall *Phase-5-intern* den Startpreis des Verkäufers ausgehend von dem in Phase 4 erreichten Maximalpreis weiter schrittweise bis hin zu einem Preis erhöhen, der maximal fünf Prozent unterhalb des $aULP_K$ liegen darf.

Stehen hingegen externe Preisinformationen zur Verfügung, so arbeitet der Preisautomat den Unterfall *Phase-5-extern* ab. Sind Informationen über die letzten dreißig Startpreise verfügbar, mit denen Verkäufer des gewünschten Gutes eine Verhandlung begonnen haben, so wird aus diesen in *Phase-5-extern* der (gleitende) Mittelwert $gmExternalMaxPriceData$ berechnet und als Preisschranke PSU'_K verwendet. Anschließend erhöht der Automat den Startpreis schrittweise nur bis maximal zum Wert von PSU'_K . Der in den beiden Unterfällen jeweils gültige *PriceStep* wird analog zu Phase 4 bestimmt:

$$PriceStep_{intern} = \frac{aULP_K - PSU_K}{5} * meanOpponentQuotient$$

$$PriceStep_{extern} = \frac{PSU'_K - PSU_K}{5} * meanOpponentQuotient$$

5.4 Lernen in MACSIMA

In MACSIMA können Agenten ihre Verhandlungsstrategien mit der Hilfe von *evolutionären Algorithmen (EA)* an sich dynamisch ändernde Marktbedingungen adaptieren. Einführungen zu EA⁴⁶ finden sich in [Davis 1991; Nissen 2013; Pohlheim 2013; Weicker 2007; Bibel et al. 2013; Weicker 2015]. MACSIMA stellt dem Designer eines Agenten oder eines MAS eine Art Baukasten für die Realisierung eines EA zur Verfügung, das heißt eine Menge von genetischen Operatoren und weitere Einstellmöglichkeiten, mit denen für die Agenten in einem Simulationsszenario individuell unterschiedliche Instanziierungen eines EA definiert werden können.

Dabei kann der Designer beziehungsweise Experimentator bei den genetischen Operatoren der *Selektion*, *Rekombination* und *Mutation* jeweils zwischen mehreren Varianten⁴⁷ wählen (siehe Abschnitt 5.4.3). Auf der Grundlage dieser Gestaltungsmöglichkeiten kann untersucht werden, welche Auswirkungen der Einsatz von unterschiedlichen Varianten der genetischen Operatoren auf die Performanz von Agenten und die Koordinationsleistung innerhalb eines MAS beziehungsweise eines Agenten-basierten elektronischen Marktes hat.

In diesem Abschnitt werden daher die verschiedenen von MACSIMA angebotenen Varianten der genetischen Operatoren vorgestellt. Neben den aus der Literatur bekannten genetischen Operatoren, die in MACSIMA implementiert wurden, wurde mit der *Deterministic Minority Selection (DMS)* eine neuartige Selektions-Methode entwickelt und in drei leicht unterschiedlichen Varianten (DMS, DMS2 und DMS3) implementiert (siehe Abschnitt 5.4.3.2). Außerdem werden zwei unterschiedliche Ersetzungsschemata beschrieben, anhand derer in der letzten Phase eines EA entschieden wird, welcher Anteil des Gen-Pools eines Agenten durch die mit den genetischen Operatoren erzeugten Nachkommen ersetzt wird.

Daneben werden auch mehrere Fitness-Bewertungsmethoden beschrieben, die Genotypen und durch diese codierten Verhandlungsstrategien ein Gütemaß zuordnen und damit eine

⁴⁶ Dawid erläutert in [Dawid 1995] den Einsatz von GA in ökonomischen Systemen und fokussiert in [Dawid 1999] auf die EA-gestützte Realisierung von adaptivem Lernen. [Vogt 2010] beschäftigt sich mit dem Einsatz von EA in einer Supply Chain. [Dasgupta & Michalewicz 1997] behandeln die Nutzung von EA in ingenieurwissenschaftlichen Anwendungen. [Klügl 2004; Smith et al. 1999] setzen EA zur Adaption in MAS ein.

⁴⁷ Einen Überblick über die gängigen Varianten findet man in [Weicker 2015] und [Pohlmann 2013].

wesentliche Grundlage für die zielgerichtete Selektion von Eltern-Genotypen durch einen EA schaffen.

Die Auswahl und EA-basierte Adaption der von den Agenten in Verhandlungen einsetzbaren Verhandlungsstrategien wird von einem Verhandlungsmodul übernommen, mit dem jeder MACSIMA-Agent ausgestattet ist. Für das Verhandlungsmodul eines Agenten sind verschiedene Lern-, Informationsaustausch- und Informationsnutzungs-Modi einstellbar, die in Abschnitt 5.4.1 erläutert werden.

5.4.1 Adaptives Verhandlungsmodul der Agenten

Das Verhandlungsmodul eines MACSIMA-Agenten übernimmt die Auswahl und Adaption der vom Agenten in seinen Verhandlungen einsetzbaren Verhandlungsstrategien, die jeweils von einem Genotyp spezifiziert werden. Die Menge der Genotypen, aus denen der Agent zu einem bestimmten Zeitpunkt eine in einer Verhandlung einzusetzende Verhandlungsstrategie auswählen kann, wird *Pool* beziehungsweise *Gen-Pool* oder *Strategie-Pool* genannt.

Wenn ein Agent eine Verhandlung erfolgreich abgeschlossen hat (oder wenn das Verfahren der Ticket-Limitierung oder der Preisautomat dies erzwingt; siehe die Abschnitte 5.3.3.7 und 5.4.5.2), so wählt er vor dem Start einer neuen bilateralen Verhandlung einen neuen Genotyp aus seinem Pool an Genotypen aus, der die Verhandlungsstrategie spezifiziert, die der Agent während der neuen Verhandlung verfolgt.

Dann verhandelt er unter Verwendung dieses Genotyps, bis er mit seinem Opponenten eine Einigung über den Verkaufspreis des Gutes beziehungsweise einen erfolgreichen Verhandlungsabschluss erzielt oder es zu einem Verhandlungsabbruch kommt.

Nach einem erfolgreichen Verhandlungsabschluss berechnen beide Agenten für den von ihnen verwendeten Genotyp in Abhängigkeit vom Verhandlungsergebnis und der von ihnen verwendeten Fitness-Bewertungsmethode (siehe Abschnitt 5.4.2) aus ihrer individuellen Perspektive jeweils einen *Fitness-Wert*.

Anschließend erzeugt der Agent ein sogenanntes *Plumage*, in dem er neben dem verwendeten Genotyp auch den für diesen berechneten Fitness-Wert und weitere, die Verhandlungssituation charakterisierende Informationen speichert. Dabei handelt es sich um eine Datenstruktur, bei der die folgenden Variablen belegt werden:

- *goodName*: Name des Gutes, für das die Verhandlung erfolgreich abgeschlossen wurde.
- *SenderGenes*: Der in der Verhandlung eingesetzte Genotyp, der die Werte der in Abschnitt 5.3.2 erläuterten Gene beziehungsweise Verhandlungsstrategie-Parameter enthält.
- *plumageForSellPool*: Boolesche Variable, die angibt, ob der Genotyp vom Agenten beim Einkauf oder dem Verkauf des Gutes eingesetzt wurde beziehungsweise das Plumage in der Sell-Population oder der Buy-Population gespeichert werden soll.
- *fitness*: Fitness-Wert, der dem Genotyp auf der Grundlage des Verhandlungsergebnisses gemäß der vom Agenten verwendeten Fitness-Bewertungsmethode (siehe Abschnitt 5.4.2) zugeordnet wurde.
- *roundsForCurrentGenotype*: Anzahl der Simulationsrunden, in denen der Agent den Genotypen eingesetzt hat, bis er mit dessen Verhandlungsstrategie einen Abschluss tätigen konnte.
- Preisinformationen, die von den Fitness-Bewertungsmethoden (siehe Abschnitt 5.4.2)

und vom Preisautomaten (siehe Abschnitt 5.3.3.7) für Berechnungen genutzt werden können. Hierzu gehören die Variablen *currentPrice*, *weighted_Average*, *basicPrice*, *maxPrice*, *minPrice* und *mediumPrice*, deren Bedeutung in den entsprechenden Abschnitten, insbesondere in Abschnitt 5.4.2, erläutert wird.

Die generierten Plumages werden in einer Datenstruktur gesammelt, die *Population* genannt wird (siehe Abbildung 44). Die Größe des Pools und der Population können für das Verhandlungsmodul jedes Agenten flexibel eingestellt werden (siehe Abschnitt 6.3).

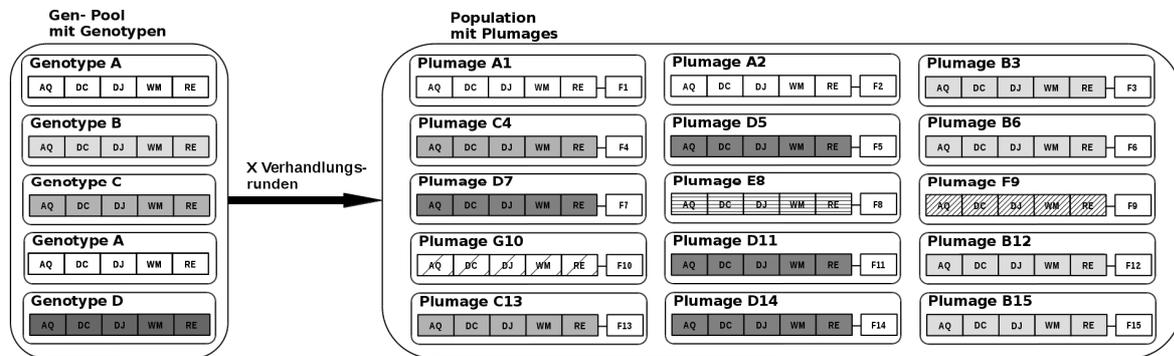


Abbildung 44: Pool und Population eines Agenten⁴⁸

Bei der Definition eines Simulationsszenarios legt der Szenario- beziehungsweise Agenten-Designer in MACSIMA für die Agenten jeder Agentengruppe fest, ob ihr Verhandlungsmodul beim Verhandeln und Lernen einen 1-Pool-Modus oder einen 2-Pool-Modus nutzt. Die entsprechenden Modi werden in den folgenden beiden Abschnitten erläutert.

5.4.1.1 1-Pool-Modus

Im 1-Pool-Modus verfügt ein Agent nur über einen Pool und eine Population. Daher wählt er unabhängig davon, ob er an einer Verhandlung als Käufer oder Verkäufer teilnimmt, den in einer Verhandlung einzusetzenden Genotyp immer aus dem gleichen Pool aus. Ebenso speichert er auch alle Plumages, die er selbst erzeugt hat oder die ihm gemäß der Einstellungen zu Informationsaustausch und –nutzung (siehe die Abschnitte 5.4.1.3 und 5.4.1.4) zur Verfügung stehen, in einer einzigen Population. Wenn die Zahl der in der Population enthaltenen Plumages einen als *Populations-Größe* eingestellten Schwellenwert erreicht, starten die Agenten ihren Lernprozess, der durch den für sie spezifizierten EA realisiert wird.

Abbildung 45 illustriert das Zusammenspiel zwischen den Prozessen, die im Verhandlungsmodul im 1-Pool-Modus bei der Durchführung von Verhandlungen, beim Lernen und beim Informationsaustausch ablaufen. Tabelle 2 gibt einen Überblick über die wesentlichen Einstellungen, anhand derer der Lernprozess parametrisiert werden kann.

Wenn ein Agent die Genotypen beziehungsweise Verhandlungsstrategien, die er beim Kauf und Verkauf von Gütern einsetzt, in einem einzelnen gemeinsamen Pool verwaltet und die aus ihnen erzeugten Plumages in einer einzigen Population speichert, so kommt es zu einer Durchmischung der Gene der Genotypen, wenn der EA des Agenten startet und seine Phasen durchläuft (siehe Abschnitt 5.4.3). Diese Vermischung des Gen-Materials von Genotypen, die unterschiedliche Einsatzzwecke aufweisen, kann die Zielgerichtetheit des Adaptionsprozesses eines Agenten gefährden. Denn ein Agent, der in einer Verhandlung die Rolle des Verkäufers annimmt, hat in der Regel das Ziel, für das von ihm angebotene Gut einen möglichst hohen Verkaufspreis zu erzielen. Seine in Genotypen codierten Verhandlungsstrategien, die er als Verkäufer einsetzt, und der für die Genotypen mit dem EA durchgeführte

⁴⁸ Fortentwicklung einer Grafik aus [Zenner 2004, S. 122]

Adaptionsprozess sollten daher auf dieses Ziel ausgerichtet sein.

Analog hierzu strebt ein Agent in der Rolle des Käufers danach, einen möglichst geringen Kaufpreis für ein Gut auszuhandeln. Daher sollte ein Genotyp, den ein Agent als Käufer einsetzt, so adaptiert werden, dass mit seiner Hilfe möglichst ein unter den aktuell gegebenen Marktbedingungen als günstig einstuftbarer Kaufpreis ausgehandelt werden kann.

Da ein Produzenten-Agent in MACSIMA auf den mittleren Stufen der Wertschöpfungskette operiert, agiert er abwechselnd in beiden Rollen. Denn er verhandelt sowohl mit Agenten der ihm vorgelagerten Wertschöpfungsstufe über den Einkaufspreis von Input-Gütern, als auch mit Agenten der ihm nachgelagerten Wertschöpfungsstufe über den Verkaufspreis seines Output-Gutes.

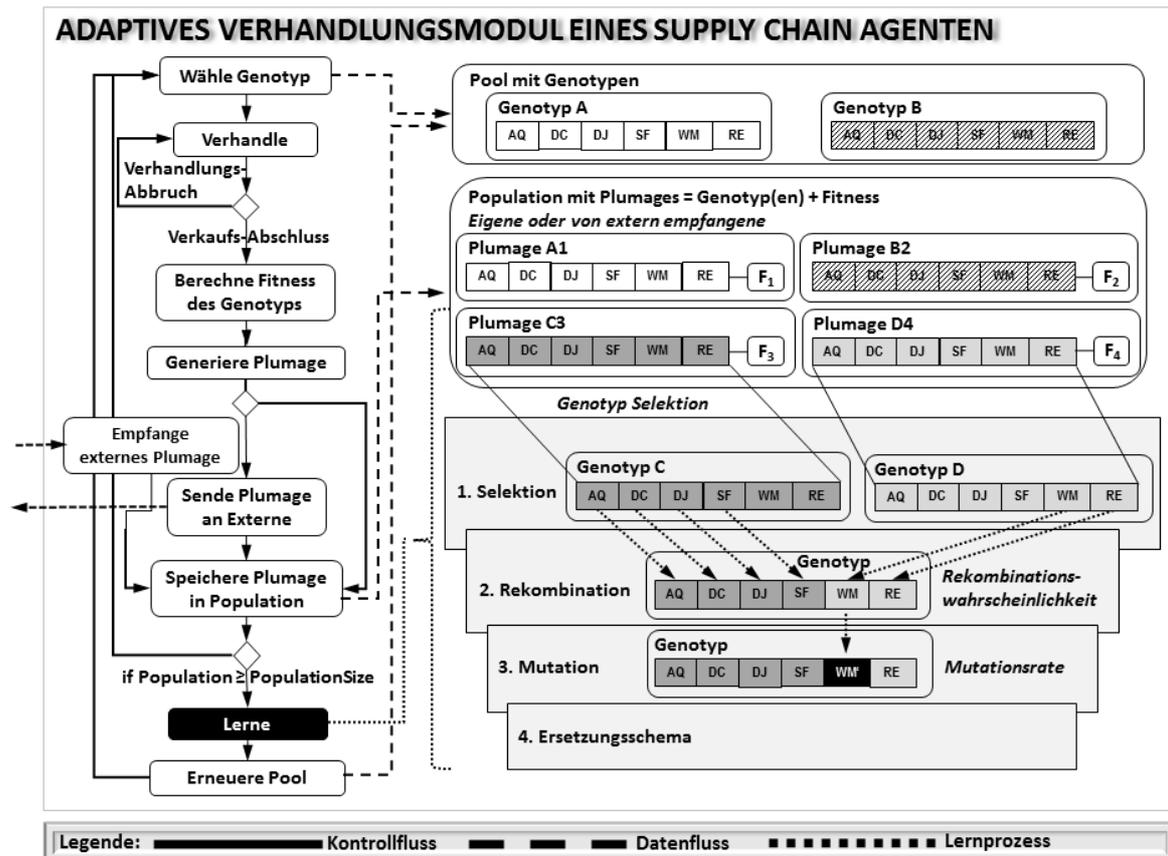


Abbildung 45: Im Verhandlungsmodul ablaufende Prozesse im 1-Pool-Modus⁴⁹

Vor diesem Hintergrund stellt sich die Frage, ob die Adaptionfähigkeit der Produzenten-Agenten gesteigert werden kann, indem das Konzept der sogenannten *Pool-Separierung* eingeführt wird. Dabei werden Genotypen und die sich aus ihrer Verwendung ergebenden Plumages gemäß ihres Einsatzzweckes, der sich auf den Kauf oder Einkauf von Gütern beziehen kann, getrennt voneinander verwaltet und vom EA adaptiert.

Im Evaluationsteil dieser Arbeit wird in Abschnitt 8.4.5 gezeigt, dass die Performanz eines agentenbasierten Wertschöpfungsnetzwerks davon profitiert, wenn alle an ihm partizipierenden Agenten in ihrem Verhandlungsmodul einen sogenannten *2-Pool-Modus* verwenden, mit dem in MACSIMA eine entsprechende Pool-Separierung realisiert wird.

⁴⁹ Quelle: Eigene Darstellung, veröffentlicht in [Ruß & Walz 2012].

5.4.1.2 2-Pool-Modus mit Pool-Separierung

Der 2-Pool-Modus, der für das Verhandlungsmodul eines MACSIMA-Agenten aktiviert werden kann, erlaubt es dem Agenten, die von ihm verwendeten Verhandlungsstrategien zwei disjunkten Klassen, nämlich Strategien für den Einkauf und den Verkauf, zuzuordnen. Dadurch kann der Agent diese gemäß ihres Einsatzzweckes getrennt voneinander einsetzen, verwalten und adaptieren.

Ist der 2-Pool-Modus eines Agenten aktiviert, so verwaltet sein Verhandlungsmodul die Genotypen, die er als Käufer in Verhandlungen für ein Gut einsetzt, in einem speziellen Pool, dem sogenannten *Buy-Pool*. Genotypen, die er als Verkäufer einsetzt, verwaltet das Verhandlungsmodul in einem davon getrennten Pool, dem sogenannten *Sell-Pool*.

Die Plumages, die mit den in diesen Pools enthaltenen Genotypen generiert (oder von anderen Agenten empfangen) werden, werden ebenfalls in voneinander getrennten Populationen gespeichert. Stammt der Genotyp des Plumages aus dem *Buy-Pool*, so wird der Wert der im Plumage enthaltenen Booleschen Variable *plumageForSellPool* auf *false* gesetzt und das Plumage der *Buy-Population* hinzugefügt. Ansonsten wird bei der Erzeugung des Plumages der Wert der Variablen *plumageForSellPool* auf *true* gesetzt und das Plumage in der *Sell-Population* gespeichert.

Der 2-Pool-Modus ist insbesondere für Produzenten-Agenten sinnvoll, da diese sowohl Input-Güter kaufen als auch ein Output-Gut verkaufen und daher in Verhandlungen abwechselnd als Käufer und Verkäufer agieren. Durch die für den Einkauf und den Verkauf separierte Nutzung der Datenstrukturen Pool und Population können die Produzenten-Agenten ihre durch die Genotypen codierten Verhandlungsstrategien bezogen auf ihren spezifischen Einsatzzweck verwenden und mit ihrem EA an die gegenwärtigen Marktbedingungen adaptieren.

Ob ein Agent Plumages, die er in Abhängigkeit von den Einstellungen für Informationsaustausch und -nutzung (siehe die Abschnitte 5.4.1.3 und 5.4.1.4) von anderen Agenten gesendet bekommt, im 2-Pool-Modus in seine Sell-Population oder Buy-Population einfügt, entscheidet er auf Basis des im jeweiligen Plumage enthaltenen Wertes der Booleschen Variable *plumageForSellPool*, den das Plumage enthält. Ist ihr Wert *true*, so wird das empfangene Plumage im 2-Pool-Modus in der *Sell-Population* gespeichert, ansonsten in der *Buy-Population*. Im 1-Pool-Modus ist der Wert dieser Variablen irrelevant, da alle Plumages in einer einzigen Population gespeichert werden.

Im 2-Pool-Modus sind neben den Pools und den Populationen auch die Lernzyklen für die beiden Populationen voneinander separiert. Das Verhandlungsmodul prüft (analog zum 1-Pool-Modus) jede der beiden Populationen jeweils einzeln darauf, ob die Anzahl der in ihr gesammelten Plumages einen Schwellenwert erreicht, der durch den Parameter *Populations-Größe* (siehe Abschnitt 6.3) spezifiziert ist. Ist dies zum Beispiel für die Buy-Population der Fall, so startet der Lernprozess beziehungsweise der EA, der dann exklusiv auf der Buy-Population operiert und als Ergebnis schließlich den Pool mit neu adaptierten Genotypen befüllt. Die andere Population, in diesem Fall die Sell-Population, bleibt davon unberührt. Der durch den EA realisierte Lernprozess startet für sie erst dann, wenn auch sie die entsprechende Populations-Größe erreicht. Im Wesentlichen werden die in Abbildung 45 für ein im 1-Pool-Modus arbeitendes Verhandlungsmodul illustrierten Prozesse für die Realisierung des 2-Pool-Modus also dupliziert.

Die in Abbildung 46 und Abbildung 47 dargestellten UML-Diagramme stellen die Prozesse eines im 2-Pool-Modus arbeitenden Verhandlungsmoduls eines Produzenten-Agenten dar. Die Abläufe für Ressourcen- und Konsumenten-Agenten, die den 2-Pool-Modus verwenden, gleichen weitgehend den Prozessen im 1-Pool-Modus, da diese Agenten-Typen in den von

ihnen geführten Verhandlungen entweder auf die Verkäufer- oder die Käufer-Rolle beschränkt sind. Daher operieren sie entweder nur auf den Datenstrukturen für den Kauf (das heißt dem Buy-Pool und der Buy-Population) oder für den Verkauf (das heißt dem Sell-Pool und der Sell-Population) und beschränken ihre Aktivitäten jeweils auf die linke oder rechte Seite der in Abbildung 46 und Abbildung 47 veranschaulichten Abläufe.

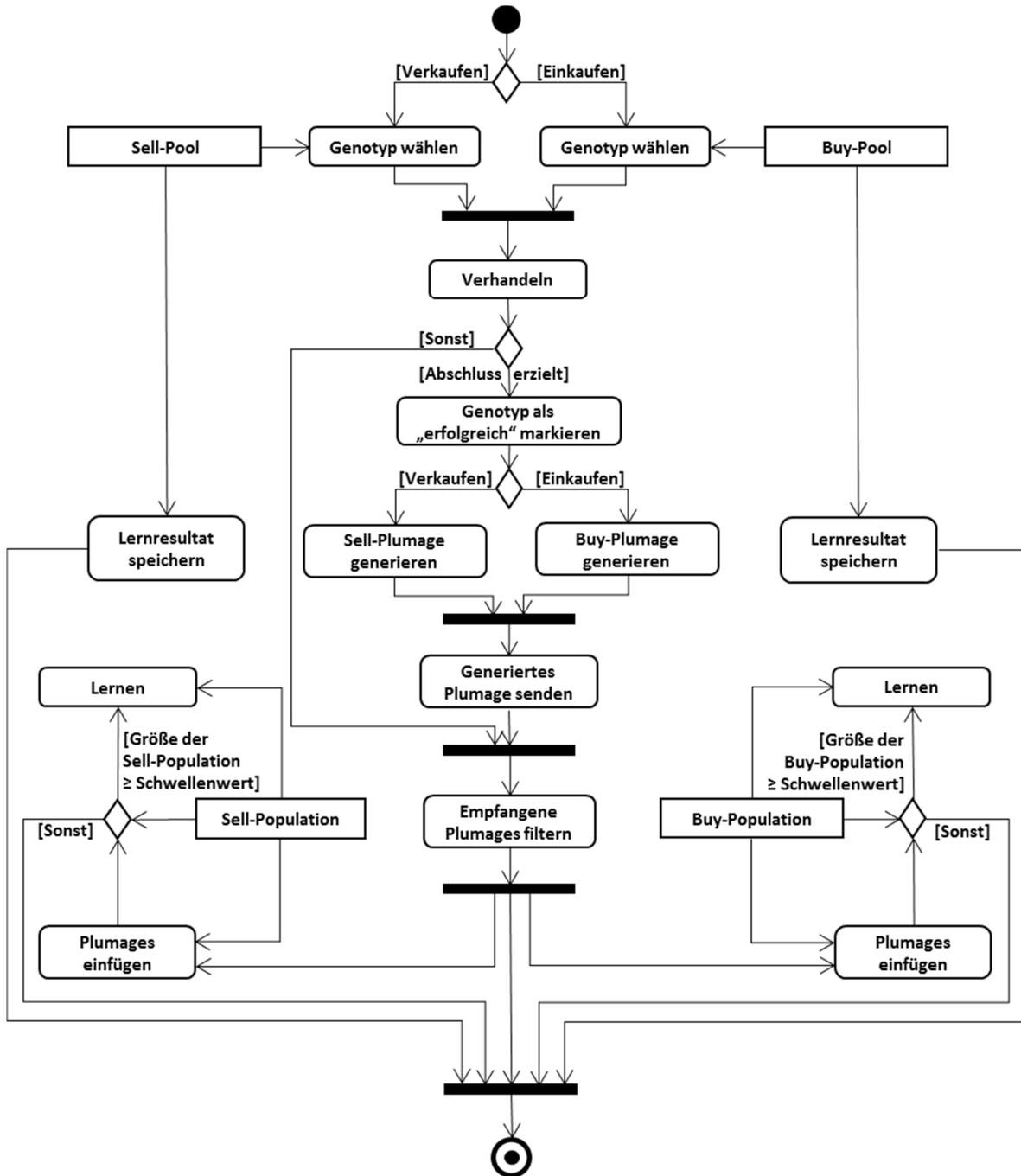


Abbildung 46: UML-Aktivitätsdiagramm für den 2-Pool-Modus⁵⁰

Abbildung 46 illustriert anhand eines UML-Aktivitätsdiagramms den Prozess, der innerhalb des Verhandlungsmoduls im 2-Pool-Modus bei der Durchführung einer Verhandlung durchlaufen wird. Damit detailliert Abbildung 46 den linken Teil von Abbildung 45.

⁵⁰ Quellen: In Anlehnung an [Ruß & Walz 2012] (eigene Veröffentlichung) sowie [Stedter 2010, S. 36] (be- treute Diplomarbeit) überarbeitet.

In Abhängigkeit davon, ob der Agent ein Gut verkaufen oder kaufen will, wählt er vor dem Start einer Verhandlung entweder aus dem *Sell-Pool* oder dem *Buy-Pool* einen Genotyp aus. Die durch den Genotyp spezifizierte Verhandlungsstrategie verwendet er in der nachfolgenden Verhandlung, die entweder erfolgreich, das heißt mit einem Abschluss (engl. *Deal*), oder mit einem Verhandlungsabbruch enden kann.

Endet die Verhandlung erfolgreich, so ordnet der Agent dem von ihm eingesetzten Genotyp anhand einer Fitness-Bewertungsmethode einen Fitness-Wert (siehe Abschnitt 5.4.2) zu und generiert ein *Plumage* (siehe oben), das den Genotyp und den für diesen berechneten Fitness-Wert enthält. Wurde der Genotyp beim Verkauf eines Gutes eingesetzt, so vermerkt er dies im *Plumage*, indem er dessen Variable *plumageForSellPool* auf *true* setzt und dieses als sogenanntes *Sell-Plumage* in der *Sell-Population* speichert. In der *Sell-Population* werden dabei nur die nach einem Verkauf generierten *Plumages* gesammelt, deren in der Variablen *SenderGenes* gespeicherter Genotyp aus dem *Sell-Pool* stammt.

Hat der Agent den Genotyp hingegen bei einem Einkauf erfolgreich eingesetzt, so speichert er das intern erzeugte *Plumage*, dessen Variable *plumageForSellPool* er auf *false* gesetzt hat, als sogenanntes *Buy-Plumage* in seiner *Buy-Population*. In dieser werden ausschließlich *Plumages* gespeichert, deren in der Variablen *SenderGenes* gespeicherter Genotyp aus dem *Buy-Pool* stammt.

Anschließend entscheidet der Agent in Abhängigkeit von seinen Einstellungen für den Informationsaustausch (siehe Abschnitt 5.4.1.3), ob er das soeben erzeugte *Plumage* auch an andere Agenten sendet und wie viele Agenten er dabei gegebenenfalls adressiert.

Darauffolgend überprüft das Verhandlungsmodul, ob der Agent eventuell selbst von anderen Agenten (also extern erzeugte) *Plumages* empfangen hat. Ist dies der Fall, so filtert er diese nach *Sell-* und *Buy-Plumages* und fügt sie je nach der Belegung ihres *plumageForSellPool*-Wertes in seine *Sell-* oder *Buy-Population* ein.

Wurde die *Sell-Population* oder die *Buy-Population* hierdurch so weit aufgefüllt, dass sie den jeweiligen Schwellenwert erreicht oder übersteigt, der für den Agenten als erlaubte Populations-Größe eingestellt ist, so startet der Lernprozess des Agenten. Dieser liefert als Resultat einen vom EA auf der Grundlage der *Sell-Population* oder *Buy-Population* adaptierten *Sell-Pool* oder *Buy-Pool*.

Damit ein Agent, dem in seinem *Buy-* und *Sell-Pool* in der Regel mehrere alternative Genotypen zur Verfügung stehen, in seinen Verhandlungen nicht immer den gleichen Genotypen beziehungsweise die gleiche Strategie verwendet, besitzt er die Fähigkeit, auch zwischen zwei Lernzyklen seine Verhandlungsstrategie für den Einkauf oder Verkauf zu wechseln. In der gegenwärtigen Implementierung wechselt ein Agent dabei seine Verhandlungsstrategie normalerweise nach einem Verhandlungserfolg, damit seine Strategie in der darauffolgenden Verhandlung gegebenenfalls weniger leicht berechenbar ist.

Das in Abbildung 47 illustrierte UML-Aktivitätsdiagramm stellt den Prozess dar, mit dem das Verhandlungsmodul des Agenten einen Genotyp nach einem erfolgreichen Einsatz gegen einen anderen austauscht, der in der nachfolgenden Verhandlung eingesetzt werden soll.

Zunächst überprüft das Verhandlungsmodul, welcher Einsatzzweck vorliegt, das heißt, ob der Agent mit seiner Hilfe ein bestimmtes Input-Gut kaufen oder ein bestimmtes Output-Gut verkaufen will. Da in beiden Fällen im Prinzip die gleichen Aktivitäten folgen, wird im Folgenden der Fall erläutert, in dem der Agent eine neue Verhandlung führen will, die dem Einkauf eines Input-Gutes dient.

Dabei stellt das Verhandlungsmodul zunächst fest, ob es mit dem letzten zu diesem Zweck eingesetzten Genotyp zu einem erfolgreichen Verhandlungsabschluss kam. Dieser Genotyp

wurde für seinen Einsatz vorher einem sogenannten *temporären Buy-Pool* entnommen, der mit Kopien der im Buy-Pool enthaltenen Genotypen aufgefüllt wurde.

Verlief der letzte Einsatz des Genotyps nicht erfolgreich, so wird der dargestellte Prozess nicht weiter durchlaufen, das heißt, der eingesetzte Genotyp nicht ausgetauscht und auch in der neu zu beginnenden Verhandlung als Einkaufsstrategie eingesetzt. Die einzigen Ausnahmen davon sind gegeben, wenn entweder das Verfahren der Ticket-Limitierung (siehe Abschnitt 5.4.5.2) für den Agenten aktiviert ist oder der Preisautomat (siehe Abschnitt 5.3.3.7) eingreift. Sowohl das Verfahren der Ticket-Limitierung als auch der Preisautomat können den Austausch eines Genotyps erzwingen, falls mit diesem über einen zu längeren Zeitraum hinweg keine Verhandlung erfolgreich abgeschlossen werden konnte.

Verlief der letzte Einsatz des Genotyps hingegen erfolgreich, so wird der eingesetzte Genotyp aus dem temporären Buy-Pool gelöscht. Ist der temporäre Buy-Pool anschließend leer, so wird er mit Kopien der im Buy-Pool des Agenten enthaltenen Genotypen neu aufgefüllt.

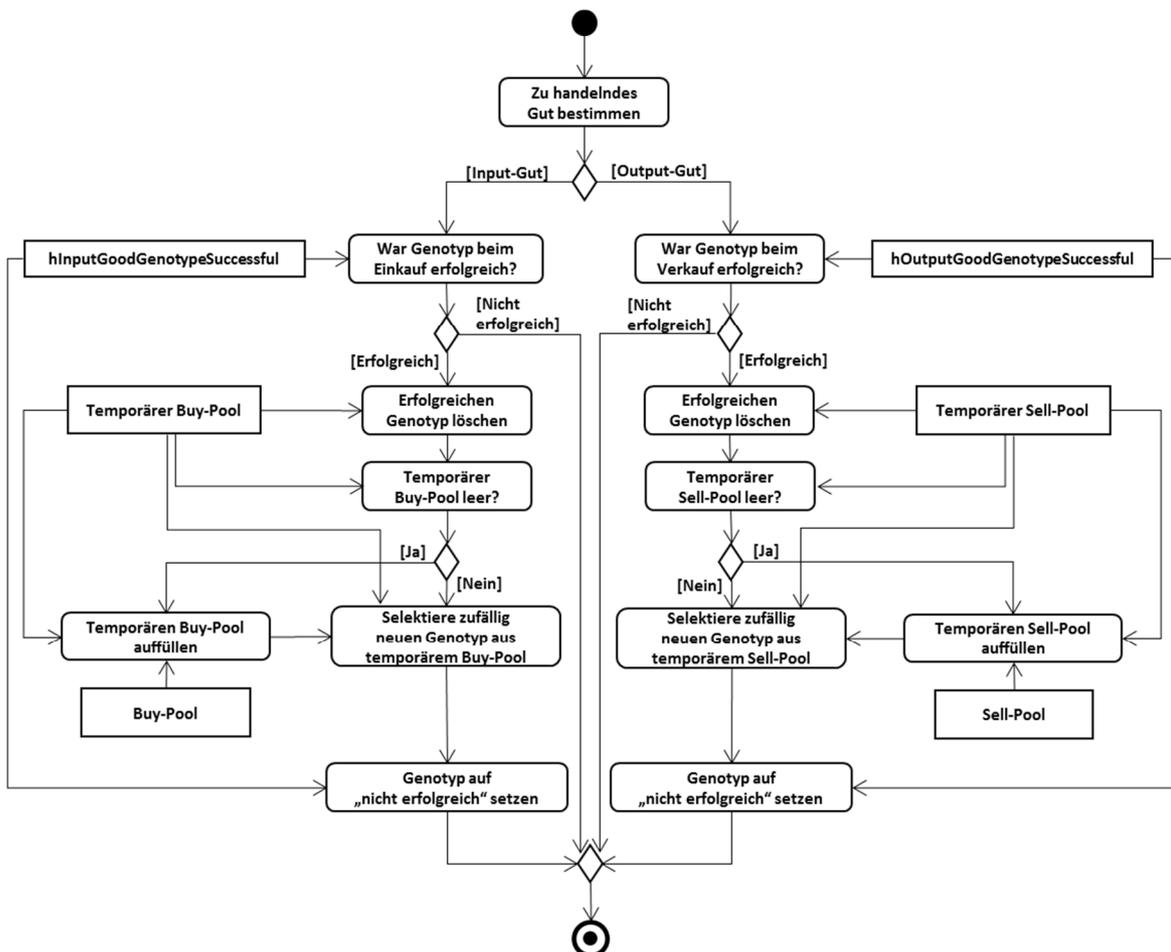


Abbildung 47: Austausch des Genotyps im 2-Pool-Modus⁵¹

Ist im temporären Buy-Pool noch mindestens ein Genotyp enthalten, so wird aus seinen noch vorhandenen Elementen ein Genotyp selektiert und in der neu zu beginnenden Verhandlung eingesetzt. Für den neu selektierten Genotyp wird in der *Hashtable*-Datenstruktur *hInputGoodGenotypeSuccessful* vermerkt, dass er noch nicht erfolgreich eingesetzt wurde. Wird mit ihm später ein erfolgreicher Verhandlungsabschluss erzielt, wird der entsprechende Vermerk auf „erfolgreich“ aktualisiert (siehe Abbildung 46).

⁵¹ Quellen: In Anlehnung an [Ruß & Walz 2012] (eigene Veröffentlichung) sowie [Stuedter 2010] (betreute Diplomarbeit) überarbeitet.

Über die dargestellten Aktivitäten hinaus wird für den neu selektierten Genotyp auch seine sogenannte *Startrunde* protokolliert, das heißt die Simulationsrunde, in der mit ihm erstmals eine Verhandlung gestartet wird. Der selektierte Genotyp wird nun (möglicherweise wiederholt) so lange in den vom Agenten zur Beschaffung (oder zum Verkauf) des Gutes geführten Verhandlungen eingesetzt, bis mit ihm ein Abschluss erzielt wird und damit der Einsatzzweck des Genotyps erfüllt ist.

In diesem Fall wird für ihn seine sogenannte *Abschlussrunde* protokolliert, das heißt die Simulationsrunde, in der der Abschluss zustande kam. Anschließend wird der Wert der Variablen *roundsForCurrentGenotype* bestimmt:

$$\text{roundsForCurrentGenotype} = \text{Abschlussrunde}(\text{Genotyp}) - \text{Startrunde}(\text{Genotyp})$$

Der Wert der Variablen *roundsForCurrentGenotype* entspricht somit der Anzahl der Simulationsrunden, in denen der Agent den selektierten Genotypen eingesetzt hat, bis mit dessen Hilfe ein Abschluss erzielt werden konnte. Dieser Wert ist für einige Fitness-Bewertungsmethoden von Relevanz (siehe Abschnitt 5.4.2).

Die entsprechenden Aktivitäten werden so lange wiederholt ausgeführt, bis die Größe der Buy-Population (oder der Sell-Population) auf den für den Start des EA erforderlichen Schwellenwert angewachsen ist und der Lernprozess gestartet wird (siehe Abbildung 48).

5.4.1.3 Informationsaustausch-Einstellungen

MACSIMA-Agenten können Informationen über Verhandlungsstrategien, die in einer bestimmten Verhandlungssituation zu einem Verhandlungserfolg geführt haben, miteinander austauschen, indem sie die von ihnen intern generierten Plumages an andere Agenten senden und von anderen Agenten (das heißt extern) generierte und versendete Plumages empfangen.

Wie oben ausgeführt wurde, enthält ein Plumege Informationen über den die Verhandlungsstrategie bestimmenden Genotyp, dessen Einsatzzweck, die für ihn ermittelte Fitness-Bewertung sowie verschiedene Preisinformationen, die die Verhandlungssituation charakterisieren. Hierzu gehören unter anderem die Startpreise der Verhandlungspartner, der Einigungs- beziehungsweise Abschlusspreis sowie der (gewichtete) durchschnittliche Marktpreis für das gehandelte Gut, den der das Plumege generierende Agent nach jedem erfolgreichen Verhandlungsabschluss auf Basis des Abschlusspreises und des Gens *Weight_Memory* aktualisiert (siehe Abschnitt 5.3.2).

Nachdem ein Agent ein Plumege generiert hat, entscheidet die vorgenommene Einstellung für die Art beziehungsweise den *Modus des Informationsaustauschs* darüber, ob er das soeben erzeugte Plumege überhaupt an andere Agenten sendet (siehe Abbildung 46). Bei der Definition eines Szenarios stehen dabei die Informationsaustausch-Modi *NoExchange* (*N*), *Everybody* (*Ev*), und *InGroup* (*G*) zur Auswahl:

1. Ist für einen Agenten der Informationsaustausch-Modus *NoExchange* eingestellt, so sendet dieser unter keinen Umständen ein von ihm erzeugtes Plumege an einen anderen Agenten.
2. Operiert ein Agent im Informationsaustausch-Modus *InGroup*, so kann dieser die von ihm erzeugten Plumages ausschließlich an Agenten senden, die sich mit ihm in der gleichen Agenten-Gruppe befinden. Die Zuordnung von Agenten zu Agenten-Gruppen wird bei der Definition des Szenarios festgelegt, wobei ein Agent jeweils nur Mitglied einer Gruppe sein darf. Entsprechende Gruppen können beispielsweise aus allen Agenten des gleichen Typs (beziehungsweise einer Wertschöpfungsstufe) oder aus Teilmengen von diesen bestehen.

3. Ist für einen Agenten der Modus *Everybody* eingestellt, so ist es diesem gestattet, die von ihm generierten Plumages an alle Agenten des Szenarios zu senden. Im Prinzip werden dabei alle Agenten des Szenarios so behandelt, als ob für sie der Modus *InGroup* aktiviert wäre und sie alle der gleichen Agenten-Gruppe angehören würden. In Abhängigkeit vom eingestellten Grad des Informationsaustauschs, dessen Parametrisierung unten erläutert wird, ist bei diesem Modus einstellbar, dass ein Agent jedes von ihm generierte Plumage an alle anderen Agenten des Szenarios sendet.

Zusätzlich zu diesen grundlegenden Einstellungen für den Modus des Informationsaustauschs bietet MACSIMA auch weitere Einstellungen für die *Quote des Informationsaustauschs*. Anhand dieser Einstellungen kann für Agenten, die in den Modi *InGroup* und *Everybody* operieren, feinjustiert werden, an wieviel Prozent der Agenten ihrer Gruppe sie nach wieviel Prozent (P) ihrer *Abschluss-Fälle* (F) ein intern generiertes Plumage senden.

Wird für die Quote des Informationsaustauschs keine spezifische Einstellung vorgenommen, so verwenden die Agenten die MACSIMA-StandardEinstellung 100P100F. Diese bedeutet, dass die Agenten nach jedem erfolgreichen Verhandlungsabschluss an alle beziehungsweise 100 Prozent der Agenten in ihrer Gruppe das von ihnen auf Basis des Abschlusses generierte Plumage senden.

Eine Reduktion dieser StandardEinstellung ist in den meisten Szenarien sinnvoll und realitätsnah, da ein auf einem Markt agierendes Unternehmen in der Regel nicht dazu in der Lage ist, eine große Flut an Informationen zu verarbeiten und bei seinen Entscheidungen zu berücksichtigen.

Die Möglichkeit zur Einstellung der Informationsaustausch-Quote wurde geschaffen, nachdem bei der Analyse der Simulationsergebnisse festgestellt wurde, dass das Risiko besteht, dass die Gen-Pools der Agenten bei einer hohen Informationsaustausch-Quote innerhalb eines kurzen Zeitraums an genetischer Vielfalt verlieren. Denn bei einer hohen Quote werden bestimmte Genotypen, die zu schnell zu vielen Abschlüssen führen, besonders häufig mit anderen Agenten ausgetauscht. Sind zusätzlich die Lern-Parametrisierungen der Agenten so gestaltet, dass sie die Genotypen aus empfangenen Plumages berücksichtigen und sie diese nur wenig verändern, so kann es innerhalb von Agenten-Gruppen rasch zu einer Dominanz bestimmter Gen-Werte beziehungsweise Genotypen kommen. Dadurch kann die Adaptionsfähigkeit der Agentengesellschaft für den weiteren Simulationsverlauf eingeschränkt werden.

In diesem Abschnitt wurden die verschiedenen einstellbaren Arten beziehungsweise Modi des Informationsaustauschs und die Möglichkeiten zu ihrer Feinjustierung beschrieben. Im folgenden Abschnitt werden die Einstellungen erläutert, die spezifizieren, auf welche Art die Agenten die an sie gesendeten Informationen beziehungsweise Plumages beim Lernen beziehungsweise zu ihrer Adaption nutzen können.

5.4.1.4 Informationsnutzungs-Modus

Neben den im vorhergehenden Abschnitt erläuterten Optionen für den Austausch von Informationen beziehungsweise Plumages wird für einen MACSIMA-Agenten in der Szenario-Definition auch ein *Modus der Informationsnutzung* eingestellt. Der eingestellte Modus legt fest, ob ein Agent ausschließlich selbst (intern) oder von anderen Agenten (extern) erzeugte Plumages oder (unabhängig von ihrer Herkunft) alle ihm bekannten Plumages in seine Populationen aufnimmt. Dadurch wird bestimmt, in welchem Maß ein Agent die Erfahrungswerte beziehungsweise Plumages anderer Agenten überhaupt bei seinem evolutionären Lernen nutzt.

Bei der Definition eines Szenarios stehen dabei als Optionen die Informationsnutzungs-Modi *internes Lernen (INT)*, *externes Lernen (EXT)* und *gemischtes Lernen (MIX)* zur Verfügung:

1. Ist für einen Agenten der Informationsnutzungs-Modus *internes Lernen* aktiviert, so nimmt er ausschließlich die von ihm selbst erzeugten Plumages in seine Population(en) auf. Extern generierte Plumages, die er von anderen Agenten empfängt, ignoriert er vollständig, indem er diese keiner von ihm verwalteten Population hinzufügt. In diesem Fall lernt der Agent also ausschließlich auf Basis der von ihm selbst gemachten Erfahrungen. Dieser Modus ist relevant, da in einem realen Wertschöpfungsszenario die Korrektheit von empfangenen Informationen nicht garantiert werden kann. Daher ist es möglich, dass sich Mitglieder eines Wertschöpfungsnetzwerks ganz bewusst dafür entscheiden, von anderen mitgeteilte Erfahrungswerte in ihrem Lernprozess nicht zu berücksichtigen, da diese bezüglich ihres Wahrheitsgehalts nicht überprüfbar sind und daher unsichere Informationen darstellen.
2. Ist der Informationsaustausch-Modus eines Agenten auf *externes Lernen* eingestellt, so nimmt er ausschließlich extern generierte Plumages, die er von anderen Agenten erhält, in seine Population(en) auf. Dies bedeutet, dass sein EA exklusiv auf externen Plumages beziehungsweise Erfahrungswerten operiert. In Abhängigkeit von seinen Informationsaustausch-Einstellungen (siehe oben) sendet der Agent zwar von ihm selbst generierte Plumages möglicherweise auch an andere Agenten, aber er fügt diese selbst keiner von ihm verwalteten Population hinzu.
3. Der Informationsaustausch-Modus *gemischtes Lernen* stellt eine Kombination der beiden vorgenannten Modi dar. In diesem Modus nimmt ein Agent sowohl von ihm intern erzeugte als auch extern generierte Plumages, die er von anderen Agenten empfängt, in seine Population(en) auf. Daher nutzt er in seinem Lernprozess neben seinen eigenen auch die Erfahrungen von anderen Agenten. Da der Agent in diesem Modus seine Population(en) mit Plumages auffüllen kann, die aus mehreren Quellen stammen, wächst der Inhalt beziehungsweise die Größe seiner Population(en) bei sonst gleichen Einstellungen für den Informationsaustausch und die Gruppenzugehörigkeit in der Regel schneller an als in den beiden vorgenannten Modi. Hierdurch erreicht eine von dem Agenten verwaltete Population normalerweise schneller ihre maximal erlaubte Größe, was zu einer höheren Frequenz der Lernzyklen beziehungsweise Adaptionsprozesse des Agenten führt.

5.4.2 Fitness-Bewertungsmethoden

MACSIMA-Agenten, deren Lernfähigkeiten aktiviert sind, durchlaufen einen zyklischen Adaptionsprozess, der einen evolutionären Algorithmus beinhaltet und in Abbildung 48 dargestellt ist.

Unmittelbar nach dem erfolgreichen Abschluss einer Verhandlung berechnen sowohl der Verkäufer als auch der Käufer für den von ihnen bei der Verhandlung jeweils eingesetzten Genotyp einen sogenannten Fitness-Wert. Dieser stellt ein Maß für die Güte des Genotyps beziehungsweise der in ihm (durch die sechs in Abschnitt 5.3.2 erläuterten Gen-Werte) codierten Verhandlungsstrategie dar. Anschließend speichern beide Agenten den Genotyp zusammen mit dem für ihn ermittelten Fitness-Wert in einem Plumage und speichern dieses (im 2-Pool-Modus) in ihrer jeweiligen Buy- oder Sell-Population für das Gut, das verkauft wurde.

Die erste Phase des EA, das heißt die Selektion (siehe die Abschnitte 5.4.3.1 und 5.4.3.2), nutzt dann die zugewiesenen Fitness-Bewertungen, um aus der Population diejenigen Genotypen auszuwählen, aus denen durch Rekombination und Mutation (siehe die Abschnitte 5.4.3.3 und 5.4.3.4) neue Genotypen beziehungsweise Nachkommen erzeugt werden sollen. Anschließend wird der alte Gen-Pool unter Beachtung der Ersetzungsschema-Einstellungen durch einen neuen Gen-Pool ersetzt.

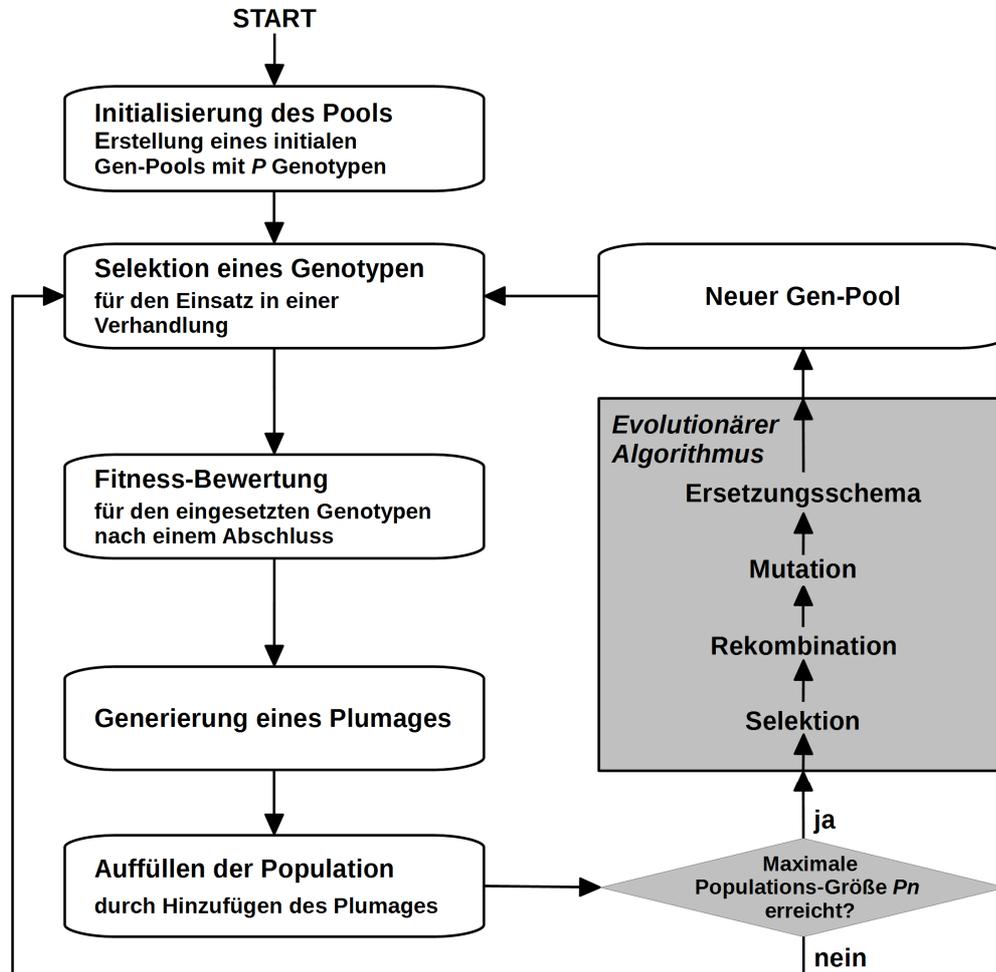


Abbildung 48: Adaptionzyklus eines MACSIMA-Agenten

Der in MACSIMA realisierte EA beinhaltet die typischen Phasen eines *einfachen evolutionären Algorithmus* (siehe [Pohlmann 2013, S. 9]). Darüber hinaus beinhaltet der Adaptionzyklus mit der Fitness-Bewertung auch ein Element eines *erweiterten evolutionären Algorithmus* (siehe [Pohlmann 2013, S. 10]). Aus Gründen der Vereinfachung wurde in MACSIMA auf zusätzliche Erweiterungen wie zum Beispiel *regionale Populationsmodelle* mit konkurrierenden *Unterpopulationen* verzichtet (vergleiche [Pohlmann 2013, S. 87 ff.]).

Die von den Agenten im Adaptionzyklus verwendete Fitness-Bewertungsmethode schafft eine für die Arbeit des EA wesentliche Grundlage. Von ihrer Eignung für die Anwendungsdomäne (beziehungsweise die Problem-Klasse) ist abhängig, wie zielgerichtet der Lern- beziehungsweise Adaptionprozess abläuft.

Im Folgenden werden daher verschiedene Fitness-Bewertungsmethoden vorgestellt, die von den Geschäftsagenten alternativ zueinander eingesetzt werden können. Ihre jeweilige Eignung für die Anwendungsdomäne wird im Evaluationsteil der Arbeit in Abschnitt 8.4.2 simulativ untersucht.

5.4.2.1 Die PmA-Methode

Bei der *PmA-Methode* (engl. *Price minus Average Method*) bestimmt ein Käufer K die Fitness eines Genotyps, indem er den in der gerade abgeschlossenen Verhandlung für ein Gut ausgehandelten Verkaufspreis (*currentPrice*) von seinem *weighted_Average* für das Gut, subtrahiert:

$$fitness_{PmA}(K) = weighted_Average_K - currentPrice$$

Beim *weighted_Average* handelt es sich um den vom Käufer auf Basis der Ergebnisse früherer Verhandlungen berechneten gewichteten Durchschnittspreis für das Gut. Die Berechnung des *weighted_Average* hängt vom Wert des Gens WM ab und wird in Abschnitt 5.3.2 erläutert.

Eine Fitness-Bewertung auf der Grundlage der oben genannten Formel ist aber nur für einen Käufer sinnvoll. Denn für diesen ist es vorteilhaft, wenn es ihm gelingt, ein Gut zu einem Preis zu erwerben, der sich unterhalb des von ihm beobachteten durchschnittlichen Marktpreis *weighted_Average* befindet. Liegt der vereinbarte Kaufpreis darüber, so ist es wahrscheinlich, dass der Käufer mehr als den marktüblichen Preis gezahlt hat und er das Gut mit einer anderen Verhandlungsstrategie günstiger bekommen hätte. In diesem Fall ergibt sich daher anhand der oben angegebenen Formel für den eingesetzten Genotyp ein negativer Fitness-Wert.

Daher berechnet nur der Käufer eines Gutes die Fitness seiner Verhandlungsstrategie auf die oben dargestellte Weise. Der Verkäufer VK berechnet sie gerade umgekehrt, da für ihn das Verhandlungsergebnis und damit die Güte der eingesetzten Verhandlungsstrategie umso positiver zu bewerten ist, je höher der mit dem Käufer ausgehandelte Verkaufspreis über dem von ihm als marktüblich angenommenen Preis liegt. Der Verkäufer berechnet die Fitness seiner Strategieparameter-Kombination daher folgendermaßen:

$$fitness_{PmA}(VK) = currentPrice - weighted_Average_{VK}$$

Dabei aktualisiert jeder der Agenten den *weighted_Average* für ein Gut nach jedem erfolgreichen Verhandlungsabschluss (siehe Abschnitt 5.3.2).

Die PmA-Methode berücksichtigt nicht, wie lange ein Genotyp erfolglos eingesetzt wurde, bis mit Hilfe der in ihm codierten Verhandlungsstrategie der Abschluss erzielt werden konnte. Sondern für sie ist nur entscheidend, wie stark der ausgehandelte Abschlusspreis *currentPrice* zugunsten des Agenten von dem von ihm beobachteten üblichen Marktpreis *weighted_Average* abweicht – unabhängig davon, wie lange er auf diesen Abschluss gewartet hat und wie viele Verhandlungen aufgrund seiner möglicherweise „harten“ Verhandlungsstrategie abgebrochen wurden.

Im Unterschied dazu berücksichtigen die vier folgenden Fitness-Berechnungsmethoden die Anzahl der Simulationsrunden, in denen ein Genotyp beziehungsweise eine Verhandlungsstrategie eingesetzt wurde, bis mit ihr ein erfolgreicher Verhandlungsabschluss erzielt werden konnte.

5.4.2.2 Die AvP-Methode

Die *AvP-Methode* (engl. *Average Price Method*) geht von dem von der PmA-Methode berechneten Fitness-Wert aus und dividiert diesen durch den vom jeweiligen Agenten berechneten *weighted_Average* und die Anzahl der Simulationsrunden (*roundsForCurrentGenotype*), in denen der Agent den Genotypen eingesetzt hat, bis er mit dessen Verhandlungsstrategie einen Abschluss tätigen konnte.

$$fitness_{AvP}(K) = \frac{weighted_Average_K - currentPrice}{weighted_Average_K * roundsForCurrentGenotype_K}$$

Ausgehend von der Berechnungsvorschrift für den Käufer K ergibt sich die Fitness-Bewertungsfunktion des Verkäufers VK , indem im Term, der im Zähler steht, die Variablen vertauscht werden:

$$fitness_{AvP}(VK) = \frac{currentPrice - weighted_Average_{VK}}{weighted_Average_{VK} * roundsForCurrentGenotype_{VK}}$$

Auch bei den folgenden beiden Fitness-Bewertungsmethoden ergibt sich die vom Verkäufer anzuwendende Formel aus der vom Käufer verwendeten Formel, indem die Variablen im Zähler vertauscht werden. Daher wird bei der Beschreibung der BaP- und der MdP-Methode jeweils auf die Darstellung der Formel für den Verkäufer verzichtet.

5.4.2.3 Die BaP-Methode

Im Unterschied zur AvP-Methode berechnet die *BaP-Methode* (engl. *Basic Price Method*) die Fitness eines Genotyps nicht auf der Grundlage eines gewichteten Durchschnittspreises *weighted_Average*, in den die Erfahrungen des Agenten eingehen, sondern unter Verwendung eines festen Basispreises *basicPrice*, der vom Designer des Simulationsszenarios bei der Definition des Szenarios eingestellt wird und im Simulationsverlauf unverändert bleibt. Der Basispreis kann daher als ein vom Designer gewünschter durchschnittlicher Marktpreis für das Gut interpretiert werden. Die Berechnungsvorschrift für den Käufer ist

$$fitness_{BaP}(K) = \frac{basicPrice - currentPrice}{basicPrice * roundsForCurrentGenotype_K}$$

Vom *basicPrice* wird der vereinbarte Verkaufspreis *currentPrice* abgezogen und in Relation zu der Anzahl an Simulationsrunden *roundsForCurrentGenotype* gesetzt, die es (auch über mehrere Verhandlungen hinweg) gedauert hat, bis die Verwendung des Genotyps zu einem erfolgreichen Verhandlungsabschluss geführt hat.

5.4.2.4 Die MdP-Methode

Die *MdP-Methode* (engl. *Medium Price Method*) nutzt den Mittelwert *mediumPrice* der Startgebote von Käufer und Verkäufer. Sie bestimmt die Fitness eines Genotyps anhand der Formel

$$fitness_{MdP}(K) = \frac{mediumPrice - currentPrice}{mediumPrice * roundsForCurrentGenotype_K}$$

$$\text{mit } mediumPrice = \frac{Startgebot_K + Startgebot_{VK}}{2}$$

Wie zuvor bei der AvP- und der BaP-Methode wird auch bei dieser Methode die Anzahl an Simulationsrunden, die der eingesetzte Genotyp bis zu einem erfolgreichen Verhandlungsabschluss benötigt hat, bei der Bestimmung seines Fitness-Werts berücksichtigt.

5.4.2.5 Die RMF-Methode

Die *RMF-Methode* (engl. *Relative Medium Price Fraction Method*) stellt die aufwendigste Fitness-Berechnungsmethode dar. Sie benutzt sowohl die Startgebote von Käufer und Verkäufer als auch den aus diesen Startgeboten (wie bei der MdP-Methode) gemittelten *mediumPrice*. Die Berechnungsvorschrift für den Käufer ist

$$fitness_{RMF}(K) = \frac{maxPrice - currentPrice}{(mediumPrice - minPrice) * 2 * roundsForCurrentGenotype_K}$$

Der Verkäufer berechnet die Fitness folgendermaßen:

$$fitness_{RMF}(VK) = \frac{currentPrice - minPrice}{(maxPrice - mediumPrice) * 2 * roundsForCurrentGenotype_{VK}}$$

Der in den Formeln genannte *maxPrice* entspricht dabei dem höheren der beiden Startgebote und der *minPrice* dem niedrigeren. Es gehen also bei der RMF-Methode mehr Informationen in die Fitness-Berechnung ein als bei den vorgenannten Fitness-Berechnungsmethoden.

5.4.3 Genetische Operatoren

5.4.3.1 Selektion: Standardmethoden

Aus einer größeren Menge an Genotypen, die in der Population (in der Form von Plumages) enthalten sind, muss auf geeignete Weise eine Teilmenge ausgewählt werden, die an die weiteren Phasen des EA übergeben wird. Denn das Gen-Material dieser selektierten Eltern-Genotypen bildet die Grundlage für die Kind-Genotypen, die vom EA generiert und anschließend von der Pool-Datenstruktur des Verhandlungsmoduls übernommen werden. Aus dieser Pool-Datenstruktur, die der EA in seiner letzten Phase mit von ihm neu erzeugten beziehungsweise adaptierten Genotypen befüllt, werden dann im Verlauf der Simulation Genotypen ausgewählt, die jeweils die in den Verhandlungen eingesetzten Verhandlungsstrategien bestimmen.

Binäre Wettkampfsituation (BiC)

Bei der *Binary Competition (BiC)* handelt es sich um eine relativ unkomplizierte Selektions-Methode. Bei ihr werden aus der Population des Agenten jeweils zwei Plumages zufällig entnommen. Anschließend wählt die BiC-Methode denjenigen Genotyp aus, der in dem Plumege enthalten ist, das den höheren Fitness-Wert aufweist. Dieser Genotyp wird in eine neue Datenstruktur, den sogenannten Interims-Pool, kopiert. Der Interims-Pool verfügt dabei über die gleiche Anzahl an Plätzen beziehungsweise Individuen wie der Strategie-Pool. Das korrespondierende Plumege, das den ausgewählten Genotyp enthält, wird aus der Population gelöscht. Dieser Vorgang wird solange wiederholt, bis alle Plätze im Interims-Pool belegt sind.

Roulette-Wheel-Selektion (RWS)

Bei der *Roulette Wheel Selection (RWS)* handelt es sich um eine Selektions-Methode, bei der die Auswahl von Genotypen probabilistisch erfolgt. Dabei ist die Wahrscheinlichkeit für die Auswahl eines Genotyps proportional zu dem ihm in den Plumages der Population zugeordneten Fitness-Werten. Jedem in einem Plumege enthaltenen Genotyp wird ein Sektor auf einem Roulette-Rad zugeordnet. Die Größe des ihm zugeordneten Sektors ergibt sich dabei aus dem im Plumege vermerkten Fitness-Wert und den in den übrigen Plumages gespeicherten Fitness-Werten anhand der folgenden Formel:

$$\alpha = 360 * \frac{f(I_k)}{\sum_{n=1}^N f(I_n)} \text{ , mit}$$

α : Winkel, der dem k -ten Genotyp zugeordnet wird

$f(I_k)$: Fitness des k -ten Genotyps

Ist ein bestimmter Genotyp in mehreren Plumages enthalten, so werden ihm mehrere Sektoren zugeordnet. Das Ergebnis dieses Prozesses kann ein Roulette-Rad mit einer Struktur sein, wie sie beispielsweise in Abbildung 49 dargestellt ist.

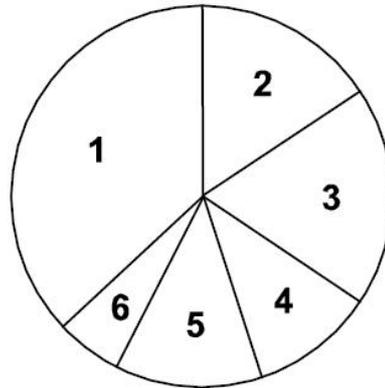


Abbildung 49: Beispiel für ein Roulette-Rad beim RWS-Verfahren⁵²

Sind im Interims-Pool P Plätze zu vergeben, so werden die P Genotypen, mit denen diese Plätze belegt werden, selektiert, indem das Roulette-Rad P -mal gedreht wird.

Alle Genotypen, denen ein Sektor zugeordnet ist, haben die Chance, vererbt zu werden. Im Beispiel besitzt der Genotyp 1 den größten Sektor auf dem Roulette-Rad und damit auch die höchste Auswahlwahrscheinlichkeit. Aber auch der Genotyp 6, welcher die insgesamt schlechtesten Fitness-Werte aufweist und daher nur den kleinsten Sektor beanspruchen kann, besitzt die Chance, zur Fortpflanzung ausgewählt zu werden. Dies führt dazu, dass einem Agenten bei der Befüllung seines Interims-Pools mittels der RWS-Methode eine relativ große genetische Vielfalt erhalten bleiben sollte.

Zwar besteht das Risiko, dass sich beim Einsatz der RWS-Methode auch Genotypen mit schlechten Fitness-Bewertungen fortpflanzen. Aber die Evaluationsergebnisse weisen darauf hin, dass dies im Anwendungsszenario in Maßen vorteilhaft sein kann, da sich die die RWS-Methode verwendenden Agenten besonders gut an Marktveränderungen adaptieren können.

Deterministische Auswahl (DtS)

Für jeden Genotyp beziehungsweise für jedes Individuum I_k einer Population der Größe N wird beim DtS-Verfahren anhand seiner Fitness ein Erwartungswert gemäß der folgenden Formel ermittelt:

$$\text{Erwartungswert } E(I_k) = f(I_k) * \frac{N}{\sum_{n=1}^N f(I_n)}$$

Bei diesem Selektionsverfahren werden die guten Individuen bevorzugt. Damit steigt das Risiko, an genetischer Vielfalt zu verlieren.

Deterministische Durchschnittsauswahl (DtA)

Die in der Population gespeicherten Plumages können den gleichen Genotyp mehrmals enthalten. Denn es ist möglich, dass dieser in mehreren Verhandlungen erfolgreich eingesetzt wurde und ihm so mehrere unterschiedliche Fitnesswerte zugeordnet wurden. Dabei kann es vorkommen, dass ein Genotyp in den meisten der auf seiner Basis durchgeführten Verhandlungen eher schlechte Fitnesswerte erzielt, aber im Einzelfall ausnahmsweise eine besonders

⁵² Quelle: [Zenner 2004] (betreute Diplomarbeit)

hohe Fitness generiert, da sich beispielsweise sein Verhandlungspartner ungeschickt verhalten hat.

Wenn wie in den vorgenannten Selektions-Methoden die Auswahl von Genotypen für die Rekombinationsphase auf Basis ihrer individuellen Fitnesswerte erfolgen würde, könnte sich solch ein Genotyp, der nur im Einzelfall zu guten Verhandlungsergebnissen geführt hat, dennoch überproportional oft fortpflanzen beziehungsweise seine Gene vererben.

Dies wäre dann nicht der Güte der durch den Genotyp codierten Verhandlungsstrategie zuzuschreiben, sondern dem Zufall. Um dies abzumildern, werden von der DtA-Methode nicht einzelne Fitnesswerte, sondern Durchschnittsfitness-Werte betrachtet.

Dafür wird von der DtA-Methode als erstes überprüft, wie viele m unterschiedliche Genotypen in den n Plumages einer an die Methode übergebenen Population vorkommen. Angenommen, ein Genotyp k ist in p Plumages der Population enthalten, so wird für diesen Genotyp eine Durchschnittsfitness berechnet, die dem Durchschnitt aller p in den n Plumages gespeicherten individuellen Fitnessbewertungen für diesen Genotyp entspricht.

Anschließend wird ein neues Plumage erzeugt, das den Genotyp k und dessen Durchschnittsfitness df beinhaltet und in einer Hilfspopulation gespeichert wird. Die Hilfspopulation wird mit $m \leq n$ entsprechenden Plumages gefüllt.

Diese neue Hilfspopulation mit den Durchschnittsfitness-Werten wird anhand der m Durchschnittsfitness-Werte sortiert. Die Durchschnittsfitness-Werte werden skaliert, indem der kleinste Durchschnittsfitness-Wert von allen anderen Durchschnittsfitness-Werten abgezogen wird. Der kleinste resultierende skalierte Durchschnittsfitness-Wert (sdf -Wert) ist somit Null.

Für alle m Genotypen I_k wird im Anschluss ein Erwartungswert für die Anzahl an Nachkommen berechnet. Dies erfolgt anhand der Formel:

$$\text{Erwartungswert } E(I_k) = \frac{sdf(I_k) * P}{\sum_{i=1} sdf(I_j)}$$

mit

$sdf(I_k)$ = skalierte Durchschnittsfitness des Genotyps beziehungsweise Individuums I_k in der Hilfspopulation und

P ist gleich der Poolgröße des Ergebnis-pools und

Die P freien Plätze im Ergebnis-Pool werden zunächst anhand des Vorkommaanteils der Erwartungswerte $E(I_k)$ der Genotypen in der Hilfspopulation vergeben. Dazu wird die Hilfspopulation anhand der Erwartungswerte sortiert und dessen Element $E(I_s)$ mit dem höchsten Erwartungswert einmal in den Ergebnispool der DtA-Methode kopiert. Dabei wird dessen Erwartungswert $E(I_s)$ um eins vermindert.

Anschließend wird die Hilfspopulation neu sortiert und wiederum das Element mit dem dann größten Erwartungswert in den Ergebnispool kopiert und dessen Erwartungswert wiederum um eins vermindert.

Sind die Vorkommaanteile aufgebraucht, aber noch freie Plätze im Ergebnispool vorhanden, so wird mit den Nachkommaanteilen fortgefahren. Dieses Vorgehen wird solange wiederholt, bis der Ergebnispool der DtA-Methode, der an die folgende Rekombinationsmethode übergeben wird, vollständig gefüllt ist.

Deterministische Durchschnittsauswahl mit Löschen des schlechtesten Individuums (DwS)

Dies verläuft analog zu dem vorherigen Verfahren. Allerdings wird der letzte Platz im Pool an einen Durchschnittsgenotyp vergeben, welcher durch Mittelwertbildung der einzelnen Gene der bereits in der neuen Population enthaltenen Genotypen erzeugt wird.

5.4.3.2 Deterministic Minority Selection

Im Folgenden wird die Selektions-Methode *Deterministic Minority Selection (DMS)* vorgestellt, die im Rahmen dieser Arbeit neu entwickelt und in MACSIMA erstmalig implementiert und evaluiert wurde. Der Abschnitt beginnt mit der Erläuterung der grundlegenden DMS-Variante, *DMS* genannt. Anschließend werden zwei weitere Varianten der DMS-Methode, nämlich *DMS2* und *DMS3* vorgestellt.

Die prinzipielle Idee, die allen drei DMS-Varianten zugrunde liegt, besteht darin, eine Quote einzuführen, die dafür sorgt, dass auch das Gen-Material von Genotypen, die aktuell (in dem vom EA betrachteten Zeitraum seit seinem letzten Durchlauf) keine besonders hohen durchschnittlichen Fitness-Bewertungen erzielen konnten, in gewissem Umfang in die Rekombinationsphase eingeht.

Die festgesetzte DMS-Quote bestimmt daher, wie viele Plätze im Ergebnis-Pool der Selektion für Plumages reserviert werden, die von den DMS-Varianten mittels der unten beschriebenen speziellen Zuteilungsprozesse besetzt werden.

In den Zuteilungsprozessen wird dabei auch Gen-Material von unterdurchschnittlich performanten⁵³ Genotypen in den Ergebnis-Pool der Selektion übertragen. Dies hat zum Zweck, die Vielfalt des im Pool befindlichen Gen-Materials länger zu erhalten.

Da sich die Umgebung der Agenten bzw. die Marktgegebenheiten dynamisch ändern, besteht die Möglichkeit, dass Gen-Material, das in der Vergangenheit zu keinen Verhandlungsergebnissen mit hoher Fitness-Bewertung geführt hat, dennoch bei zukünftig erforderlichen Anpassungen der Verhandlungsstrategie hilfreich ist.

Die DMS-Quote ist in der derzeitigen Implementierung für alle DMS-Varianten auf 25 Prozent der Pool-Größe des Agenten festgelegt. Neben der prozentualen Quote ist für die DMS-Methoden zusätzlich auch eine absolute Mindestquote von zwei Genotypen beziehungsweise Plumages festgelegt.

Daher werden maximal 75 Prozent der Genotypen beziehungsweise Plumages, die in die anschließende Rekombinationsphase eingehen, gemäß der Höhe ihrer aktuell bekannten Fitnesswerte selektiert, wobei dies analog zur oben beschriebenen DtA-Methode erfolgt.

Die übrigen mindestens 25 Prozent der freien Plätze im Ergebnis-Pool der Selektion werden durch die verschiedenen DMS-Methoden gemäß spezieller, leicht unterschiedlicher Ergebnis-Pool-Zuteilungsprozesse besetzt, die unten beschrieben werden.

Die drei DMS-Varianten benutzen die gleichen Datenstrukturen und sind auch in ihrem Ablauf zueinander sehr ähnlich. Sie unterscheiden sich nur in den letzten Prozessschritten ihrer jeweiligen Zuteilungsprozesse. Der allen Varianten gemeinsame grundlegende Ablauf wird im Folgenden beschrieben.

⁵³ Die Fitnessbewertung eines Genotyps wird bezogen auf eine einzelne Verhandlung durchgeführt. Daher kann es vorkommen, dass ein Genotyp, der im Prinzip eine überdurchschnittlich gute Verhandlungsstrategie darstellt, im Einzelfall trotzdem eine nur unterdurchschnittliche Fitnessbewertung erhält.

Hauptprozess der Deterministic Minority Selection

In Abhängigkeit von der Relation zwischen Pool- und Populationsgröße kann es häufiger vorkommen, dass ein Genotyp, dessen Verwendung in dem vom evolutionären Algorithmus betrachteten Simulationszeitraum zu einem Verkaufsabschluss geführt hat, mehr als einmal erfolgreich war. Damit ist es möglich, dass der Population mehrfach Plumages hinzugefügt wurden, die den gleichen Genotyp enthalten.

Daher werden wie beim DtA-Verfahren zunächst die in der Population befindlichen Plumages auf gemeinsame Genotypen überprüft und in Teilmengen separiert, deren Elemente jeweils den gleichen Genotyp aufweisen. Anschließend wird für jeden Genotyp anhand der in seinen Plumage-Instanzen enthaltenen Fitnessbewertungen die von ihm durchschnittlich erzielte Fitness df berechnet und zusammen mit dem Genotyp in einer Liste gespeichert, die gemäß der ermittelten Durchschnittsfitness-Werte sortiert wird.

Der kleinste in der Liste enthaltene Durchschnittsfitness-Wert wird anschließend von sich selbst sowie vom jeweiligen Durchschnittsfitness-Wert aller übrigen Listenelemente abgezogen, um die Durchschnittsfitness-Werte zu skalieren. Diese Skalierung erfolgt, um möglicherweise vorhandene negative Durchschnittsfitness-Werte zu eliminieren. Anschließend wird die Summe der so skalierten Durchschnittsfitness-Werte ermittelt. Auf Basis der skalierten Durchschnittsfitness-Werte, ihrer Gesamtsumme und der Anzahl der Elemente der Liste kann für alle Plumages in der Liste ein Erwartungswert berechnet werden.

Im Anschluss daran werden diese Plumages analog zum DtA-Verfahren gemäß ihrer Erwartungswerte zur Befüllung des Ergebnis-Pools selektiert, bis nur noch die durch die DMS-Quote reservierten Plätze frei sind. Danach werden die reservierten Plätze mittels der speziellen Zuteilungsprozesse der DMS-Varianten besetzt, die weiter unten beschrieben werden. Zusammengefasst gliedert sich der grundlegende Ablauf der Deterministic Minority Selection in die folgenden wesentlichen Schritte:

1. Mindestens zwei (oder 25 Prozent) der Plätze des Ergebnis-Pools der Selektion werden für Plumages reserviert, die am Ende des Ablaufs der DMS-Methoden mittels spezifischer Prozesse der DMS-Varianten besetzt werden.
2. Für alle in der Ausgangspopulation enthaltenen Genotypen wird ihre jeweilige Durchschnittsfitness (df) berechnet. Anschließend wird auf Basis der Genotypen und ihrer df -Werte eine sortierte Plumage-Liste erzeugt.
3. Für die Elemente dieser Plumage-Liste werden ihre Erwartungswerte (E) berechnet (beziehungsweise der jeweilige Anteil ihres skalierten df -Werts an der Summe aller skalierten df -Werte). Es wird eine gemäß dieser Erwartungswerte sortierte Plumage-Liste erzeugt.
4. Aus dieser anhand der Erwartungswerte sortierten Plumage-Liste werden analog zum DtA-Verfahren Elemente selektiert, bis nur noch die gemäß der DMS-Quote reservierten Plätze frei sind.
5. Die zu diesem Zeitpunkt noch freien Plätze werden zunächst nach dem Vorkommen-Anteil des Erwartungswertes besetzt. Sofern danach noch unreservierte Plätze im Ergebnis-Pool der Selektion frei sind, erfolgt die Besetzung anhand des Nachkommenanteils.
6. Selektierte Elemente werden in die Liste *Selektiert* aufgenommen. Nicht selektierte werden in der Liste *NichtSelektiert* gespeichert.

Anschließend werden drei neue Durchschnitts-Plumages gebildet. Hierzu wird über alle in den jeweiligen Datenstrukturen enthaltenen Gen- und Fitnesswerten jeweils das arithmetische Mittel gebildet, nämlich

- aus allen Plumages der *Ausgangspopulation* das Durchschnitts-Plumage A,
- aus allen Plumages der Liste *Selektiert* das Durchschnitts-Plumage S und
- aus allen Plumages der Liste *NichtSelektiert* das Durchschnitts-Plumage NS.

Die Durchschnitts-Plumages A, S und NS bestehen daher jeweils aus einer Kombination von Durchschnitts-Genotyp und Durchschnittsfitness.

Damit ist der allen Varianten gemeinsame DMS-Ablauf abgeschlossen. Im Folgenden werden ihre spezifischen, sie voneinander differenzierenden Zuteilungsprozesse beschrieben, mit denen sie die noch unbesetzten, gemäß der DMS-Quote reservierten Plätze im Ergebnis-Pool der Selektion auffüllen.

Zuteilungsprozesse der DMS-Varianten

Die nach dem DMS-Hauptprozess verbleibenden freien Plätze im Ergebnis-Pool der Selektion werden von den Varianten DMS, DMS2 und DMS3 anhand leicht verschiedener Zuteilungsprozesse besetzt, die folgendermaßen ablaufen:

- Bei DMS werden die noch freien Plätze im Ergebnis-Pool besetzt, indem aus einem „Lostopf“ mit den Durchschnitts-Plumages A, S und NS (der *Ausgangspopulation* sowie der Listen *Selektiert* und *NichtSelektiert*, s. o.) wiederholt jeweils ein Element gezogen wird, bis der Ergebnis-Pool gefüllt ist. Gezogene Elemente werden wieder in den Lostopf zurückgelegt.
- Bei DMS2 werden freie Plätze im Ergebnis-Pool schrittweise besetzt, wobei die einzelnen Schritte in der folgenden Reihenfolge ablaufen:
 - 1) Übernahme aus der Liste *NichtSelektiert* (sofern sie nicht leer ist) das Element mit dem höchsten Durchschnittsfitness-Wert in den Ergebnis-Pool. Entferne das übernommene Element aus der Liste *NichtSelektiert*.
 - 2) Ziehe, solange die Liste *NichtSelektiert* nicht leer ist, zufällig ein weiteres Element (mit beliebigem Fitness-Wert) aus der Liste und übernehme es in den Ergebnis-Pool. Entferne das übernommene Element aus der Liste *NichtSelektiert*.
 - 3) Weitere freie Plätze im Ergebnis-Pool werden besetzt, indem wie bei DMS aus den Durchschnitts-Plumages A, S und NS wiederholt jeweils ein Element zufällig gezogen wird, bis der Ergebnis-Pool der Selektion vollständig gefüllt ist.
- Bei DMS3 werden freie Plätze im Ergebnis-Pool, solange diese noch vorhanden sind, schrittweise aufgefüllt, wobei die folgende Schrittreihenfolge beachtet wird:
 - 1) Füge das Durchschnitts-Plumage NS der Listen-Elemente von *NichtSelektiert* zum Ergebnis-Pool hinzu (sofern vorhanden).
 - 2) Sofern die Liste *NichtSelektiert* mindestens ein Element enthält, so ermittle das Element mit dem höchsten Durchschnittsfitness-Wert aus der Liste *NichtSelektiert*, entferne dieses aus der Liste und übernehme es in den Ergebnis-Pool.
 - 3) Füge das Durchschnitts-Plumage S zum Ergebnis-Pool hinzu (sofern vorhanden).
 - 4) Füge das Durchschnitts-Plumage A zum Ergebnis-Pool hinzu (sofern vorhanden).

- 5) Ziehe, solange die Liste *NichtSelektiert* nicht leer ist, zufällig ein weiteres Element (mit beliebigem Fitness-Wert) aus der Liste und übernehme es in den Ergebnis-Pool. Entferne das übernommene Element aus der Liste *NichtSelektiert*.
- 6) Sollten dann noch freie Plätze im Ergebnis-Pool existieren, so werden diese besetzt, indem wie bei DMS aus den Durchschnitts-Plumages A, S und NS wiederholt jeweils ein Element zufällig gezogen und in den Ergebnis-Pool übernommen wird, bis dieser vollständig gefüllt ist.

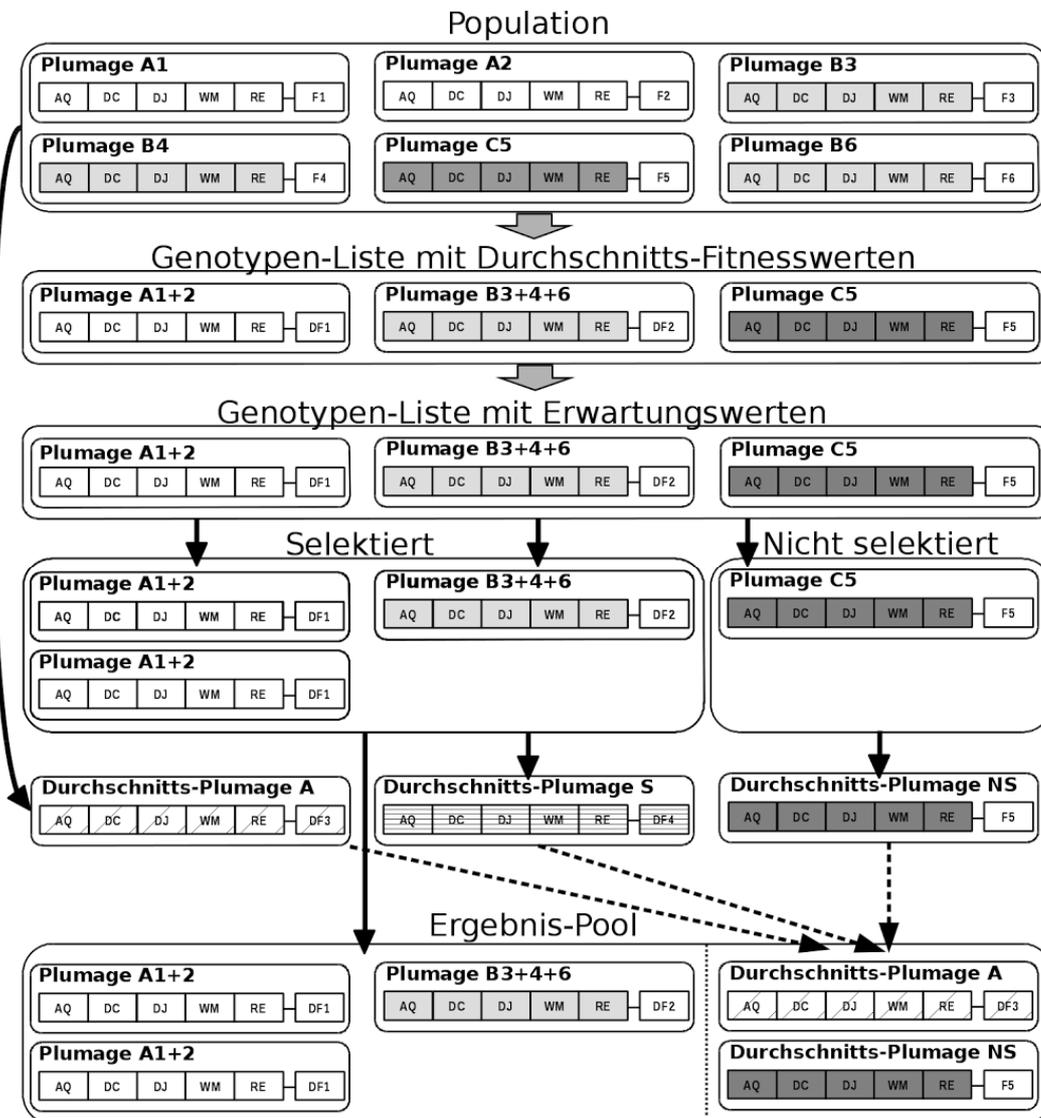


Abbildung 50: Schematischer Ablauf der Deterministic Minority Selection für DMS

Dadurch, dass ein Teil des Ergebnis-Pools der Selektion nicht mit den überdurchschnittlich performanten Genotypen, sondern mit Durchschnitts-Genotypen beziehungsweise Plumages und auch unterdurchschnittlich performanten Genotypen befüllt wird, ist nicht zu erwarten, dass ein Gesamtsystem, dessen Agenten eine DMS-Selektionsvariante einsetzen, eine Spitzenperformanz aufweist. Andererseits besteht der Sinn der verschiedenen DMS-Varianten darin, eine möglichst hohe Stabilität mit überdurchschnittlichen Umsatz- und Gewinnergebnissen zu erreichen. Die Evaluationsergebnisse in Abschnitt 8.4.3 bestätigen, dass dies mit der DMS-Variante erreicht werden konnte.

Die Varianten DMS2 und DMS3 führen hingegen zu etwas unterdurchschnittlichen Systemergebnissen. Es ist zu vermuten, dass dies damit zusammenhängt, dass zur Auffüllung freier Plätze auch originäre Plumages aus der Liste *NichtSelektiert* benutzt werden, die unterdurchschnittlich performant waren. Auffällig ist, dass DMS2 und DMS3 dennoch im Vergleich zur DtA-Selektions-Methode bei TOP- und MID-Szenarien zu einer signifikant besseren Performanz führen. Der Ablauf der gesamten Deterministic Minority Selection ist für die Variante DMS in Abbildung 50 schematisch dargestellt.

5.4.3.3 Rekombination

Die Menge der von der Selektion ausgewählten Genotypen, die im Interims-Pool (siehe Abschnitt 5.4.3.1) gespeichert wurden, werden der Rekombinationsphase des EA als Eltern-Genotypen übergeben. Die Rekombinationsphase kann in MACSIMA zwei grundlegende Methoden, nämlich *Random-Crossover (RaC)* und *n-Punkt-Crossover (nPC)* nutzen, um aus zwei Eltern-Genotypen zwei neue Kind-Genotypen zu erzeugen. In [Nissen 2013, S. 51] wird bemerkt, dass es sich bei der Rekombination um eine Art *Makro-Mutation* handelt, da aus Eltern-Genotypen Kind-Genotypen erzeugt werden, die sich hinsichtlich mehrerer Merkmale von ihren Eltern unterscheiden. Hierdurch können bei der Suche nach einem Lösungsmuster große Teile des Suchraums einbezogen werden.

Die beiden grundlegenden Rekombinations-Verfahren RaC und nPC wurden jeweils in zwei Varianten implementiert. Bei der ersten Variante werden aus der von der Selektion übergebenen Menge die zu rekombinierenden Eltern-Genotypen-Paare nach dem Zufallsprinzip ausgewählt. Bei der zweiten Variante (indiziert durch die Kürzel *Sorted* bzw. *nPCS* und *RaCS*) wird die Menge der Eltern-Genotypen zuerst nach ihren Fitness-Werten sortiert. Darauf folgend wählt das Verfahren die zu „paarenden“ Eltern-Genotypen in der Reihenfolge entsprechend der Sortierung geordnet aus. Die Evaluation der Varianten nPCS und RaCS deutet jedoch darauf hin, dass sich für die Agenten aus ihrer Nutzung im Vergleich zu nPC und RaC keine signifikanten Adaptionsvorteile ergeben.

Random-Crossover (RaC)

Beim *Random-Crossover (RaC)* werden der Menge der bei der Selektion ausgewählten Eltern-Genotypen, die im Interims-Pool gespeichert sind, zufällig zwei entnommen. Aus diesen wird ein Kind-Genotyp erzeugt, dessen Gen-Werte noch nicht instanziiert sind. Anschließend wird für jedes der Gene des Kindes durch eine stochastische Probe gegen den Parameter *preferBetterParent* entschieden, ob dieses Gen vom Vater oder der Mutter geerbt wird.

n-Punkt-Crossover (nPC)

Beim *n-Punkt-Crossover (nPC)* werden zwei Genotypen aus der Menge der selektierten Eltern-Genotypen im Interims-Pool ausgewählt und markiert. Für diese wird anhand einer stochastischen Probe gegen den einstellbaren reellwertigen Parameter *CrossRate*, der die *Kreuzungsrate* angibt und den Wertebereich $[0; 1]$ aufweist, entschieden, ob sie miteinander rekombiniert beziehungsweise gekreuzt werden.

Fällt die stochastische Probe negativ aus, so werden die beiden Genotypen unverändert in die Menge der selektierten Genotypen zurückgelegt. Der Parameter *CrossRate* kann für die Agenten über die MACSIMA-Benutzeroberfläche eingestellt werden. Fällt die stochastische Probe hingegen positiv aus, so werden sie miteinander rekombiniert, indem sie zunächst an n zufällig ausgewählten Punkten zerteilt werden, die sich bei den Genotypen jeweils an der gleichen Position befinden. Der Wert des Parameters n wird ebenfalls über die MACSIMA-Benutzeroberfläche eingestellt.

Die sich daraus ergebenden Teilstücke eines Eltern-Genotyps werden danach mit dem sie jeweils ergänzenden Teilstück des anderen Genotyps wieder zusammengesetzt. Durch eine solche Kreuzung zweier Eltern-Genotypen, die in Abbildung 51 veranschaulicht ist, werden zwei neue Kind-Genotypen erzeugt.

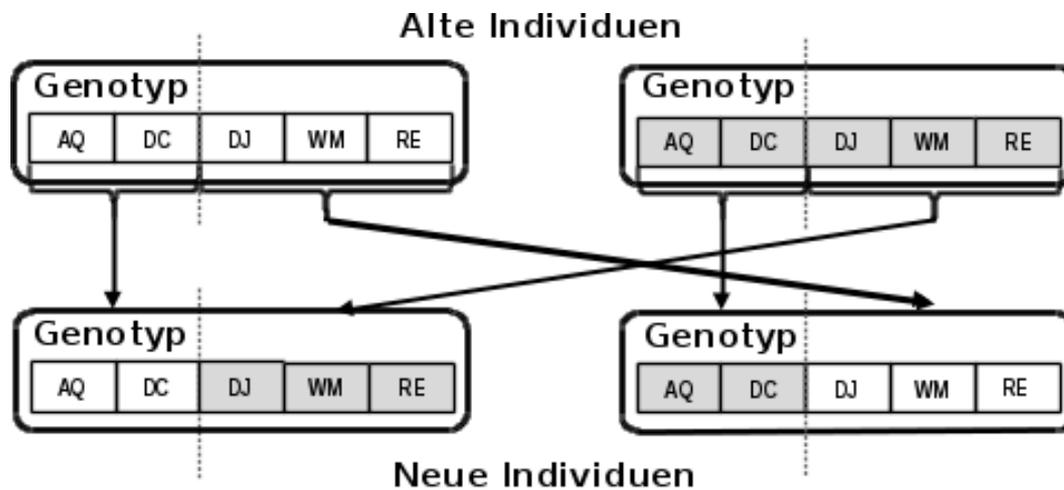


Abbildung 51: Kreuzung zweier Genotypen beziehungsweise Individuen⁵⁴

Die beiden Eltern-Genotypen werden danach im Interims-Pool gelöscht und durch die zwei neu erzeugten Kind-Genotypen ersetzt. Dieser Vorgang wird so lange wiederholt, bis der Interims-Pool keine zwei unmarkierten Eltern-Genotypen mehr enthält. Sollte ein einzelner unmarkierter Eltern-Genotyp übrig bleiben, so bleibt dieser unverändert.

5.4.3.4 Mutation

Die Mutationsphase schließt sich an die Rekombinationsphase an und operiert daher auf einem Interims-Pool, der in der Regel statt selektierter Eltern-Genotypen mittlerweile aus diesen neu erzeugte Kind-Genotypen enthält. Die Mutation kann die Gen-Werte der Kind-Genotypen innerhalb eines einstellbaren Wertebereichs verändern (siehe unten).

Die Mutations-Methoden sorgen so im Gegensatz zu den Rekombinations-Methoden, die die gegebenen Gen-Werte von Eltern-Genotypen neu anordnen, dafür, dass auch neue Gen-Werte erzeugt werden, die der Agent noch nicht in seinem Gen-Pool hatte. Dies führt zu einer höheren Gen-Vielfalt und damit auch zum Ausprobieren neuer Gen- beziehungsweise Strategie-Kombinationen. Insofern stellt die Mutation sozusagen das „kreative Element“ des verwendeten EA dar.

Die aus erprobten Eltern-Genotypen per Rekombination erzeugten Kind-Genotypen weisen eine erhöhte Wahrscheinlichkeit auf, sich in folgenden Preisverhandlungen ebenfalls gut zu behaupten. Da die Mutation deren Gene verändert, besteht jedoch das Risiko, dass erfolgreich adaptierte Gen-Werte verloren gehen beziehungsweise so verändert werden, dass Erfolg des Adaptions- respektive Lernprozesses zunichte gemacht werden. Daher sollte mit dem Grad an Mutation, den man als Designer eines Agenten zulässt, vorsichtig umgehen. [Weicker 2015, S. 10] sagt dazu:

„Große Veränderungen, die in einem Schritt durch Mutation entstanden sind, werden häufig wieder schnell aus der Population verdrängt, da durch die enge Verknüpfung und Wechselwirkung der Gene elementare negative Eigenschaften bei Großmutationen kaum vermeidbar sind.“

⁵⁴ Quelle: In Anlehnung an [Zenner 2004]

Die Evaluationsergebnisse in Kapitel 8 zeigen, dass es für die verschiedenen Mutations-Verfahren in der betrachteten Anwendungsdomäne jeweils einen optimalen Grad gibt, bei dessen Unter- oder Überschreitung die Selbstkoordinationsleistung des Systems abnimmt.

Die Einstellung des Mutations-Grades erfolgt über die Nutzeroberfläche und untergliedert sich bei der Spezifikation eines Szenarios in die Parametrisierung von *Mutations-Rate* und *Mutations-Weite* für die von den Agenten verwendeten Mutations-Methoden.

Die Mutations-Rate ergibt sich aus der Einstellung des reellwertigen Parameters *Mutation-Rate*, dessen zulässiger Wertebereich im Intervall $[0; 1]$ liegt. Durch eine stochastische Probe gegen den jeweils spezifizierten Wert wird für ein zufällig ausgewähltes Gen (siehe die GMU-Methode) oder alle Gene (siehe die SMU-Methode) eines Genotyps entschieden, ob eine Mutations-Operation durchgeführt wird.

Die Mutations-Weite wird über den Parameter *GaussWidth* eingestellt (siehe Abschnitt 6.2.2.2). Dieser gibt die Breite einer Normalverteilung um den ursprünglichen Gen-Wert des Kind-Genotypen an, innerhalb derer die verwendete Mutations-Methode einen neuen Gen-Wert bestimmt.

Wie in Abschnitt 5.3.2 ausgeführt, handelt es sich bei den Werten, die die Strategieparameter beziehungsweise Gene prinzipiell annehmen können, um reelle Werte aus dem Intervall $[0; 1]$. In der aktuellen Implementierung wurde dieser Wertebereich sogar auf das Intervall $[0,05; 0,95]$ eingeschränkt. Dies ist sinnvoll, da der Agent bei seiner Entscheidungsfindung zum Beispiel gegen die Werte seiner Gene AQ und SF stochastische Proben durchführt und so seine Entscheidung durch den Gen-Wert nie vollständig determiniert ist.

Wird ein Gen mutiert, so geschieht dies anhand der folgenden Berechnungsvorschrift:

$$Gen_{neu} = Gen_{alt} + GaussWidth * nextGaussian(), \text{ wobei}$$

GaussWidth die Breite einer Normalverteilung um den Mittelwert Gen_{alt} angibt und

nextGaussian() eine Zufallszahl aus einer Normalverteilung mit dem Mittelwert 0 und der Standardabweichung 1 liefert.

Dadurch liegt der Wert von Gen_{neu} mit hoher Wahrscheinlichkeit im Intervall $[Gen-Wert_{alt} - GaussWidth; Gen-Wert_{alt} + GaussWidth]$. Anschließend wird überprüft, ob der Wert von Gen_{neu} auch im erlaubten Wertebereich, das heißt im Intervall $[0,05; 0,95]$ liegt. Unterschreitet oder überschreitet Gen_{neu} einen der Intervallgrenzwerte, so wird er auf den entsprechenden Grenzwert gesetzt.

Die von den Agenten nutzbaren Mutations-Methoden bestehen in der *General Mutation (GMU)* und der *Single Mutation (SMU)*, die im Folgenden erläutert werden.

Single Mutation (SMU)

Bei der Mutations-Methode *Single Mutation (SMU)* werden alle im Interims-Pool enthaltenen Genotypen der Reihe nach ausgewählt. Anschließend wird für jedes ihrer sechs Gene per stochastischer Probe gegen die vom Experimentator eingestellte Mutations-Rate einzeln entschieden, ob es mutiert wird oder nicht. Bei dieser Methode besteht also eine gewisse Wahrscheinlichkeit dafür, dass der Genotyp komplett verändert wird. Die Veränderung erfolgt im Rahmen der eingestellten Mutations-Weite (siehe oben).

General Mutation (GMU)

Ähnlich wie SMU funktioniert auch die *General Mutation (GMU)*. Allerdings wird maximal ein Gen-Wert mutiert. Auch hier wird gemäß einer stochastischen Probe gegen die eingestellte Mutations-Rate entschieden, ob ein im Interims-Pool enthaltenen Kind-Genotyp überhaupt mutiert wird oder nicht.

Fällt die stochastische Probe positiv aus, so wird genau eines der sechs Gene zufällig ausgewählt und wie bei der SMU-Methode unter Berücksichtigung der eingestellten Mutations-Weite mutiert. Die übrigen 5 Gene werden von GMU nicht verändert. Abbildung 52 zeigt für einen Genotyp die Mutation seines Gens WM beziehungsweise Weight_Memory.

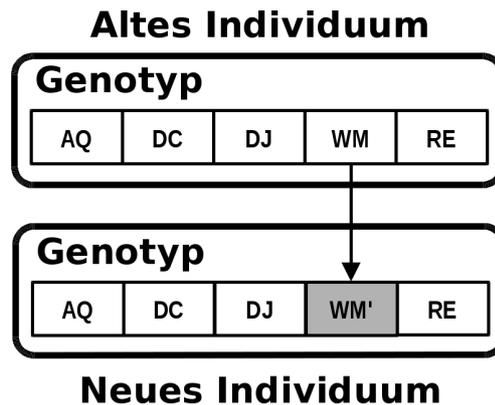


Abbildung 52: Beispiel für die Mutation eines Gens⁵⁵

5.4.4 Ersetzungsschemata

Will man mit der (gegebenenfalls durch Mutation veränderten) Kinder-Generation, die nach der Mutationsphase im Interims-Pool enthalten ist, die alten Genotypen im Gen-Pool des Verhandlungsmoduls ersetzen, so besteht das Risiko, besonders gut angepasste und dadurch erfolgreiche Verhandlungsstrategien zu verlieren, die eventuell nicht gleichwertig durch Kind-Genotypen ersetzt werden können. Für letzteres besteht durch die vom EA vorgenommene Neuordnung und Veränderung der Gene zumindest keine Garantie.

Im Extremfall könnten sogar alle Kind-Genotypen so „entarten“, dass keiner der Genotypen der Kinder-Generation das Niveau der Eltern-Generation erreicht, so dass ein Agent alle Erfolge seines bisherigen Adaptionsprozesses mit einem Schlag verlieren würde.

Um ebendies zu vermeiden, wurden die folgenden beiden Ersetzungsschema-Methoden implementiert. Ihnen ist gemeinsam, dass sie eine vom Agenten-Designer festgesetzte Anzahl von R Plätzen im Interims-Pool vor dem eigentlichen Start des EA besetzen.

Daher sollte die Zahl R in Relation zu der Größe des Pools angemessen gewählt sein. Denn falls durch das Ersetzungsschema ein zu großer Anteil der im Pool vorhandenen Plätze vorab besetzt wird, so wird die Wirksamkeit des evolutionären Adaptionsprozesses limitiert. Die Zahl und der Anteil der vom EA neu erzeugten unterschiedlichen Kind-Genotypen kann dann so gering ausfallen, dass es für sie sehr schwer wird, sich im Gen-Pool des Agenten über mehrere Generationen hinweg durchzusetzen.

⁵⁵ Quelle: In Anlehnung an [Zenner 2004] (betreute Diplomarbeit)

5.4.4.1 Elitismus

Nutzt ein EA als Ersetzungsschema den sogenannten *Elitismus*⁵⁶ (*E*), so sorgt der Elitismus dafür, dass die Genotypen der *R* Plumages, welche die höchste Fitnessbewertung innerhalb der Ausgangspopulation aufweisen, unverändert in den neu aufzufüllenden Strategie- beziehungsweise Genotypen-Pool übernommen werden. Dies führt zu einem gesicherten Überleben starker Gen-Kombinationen von Generation zu Generation. Da besonders erfolgreiche Genotypen aber in der Regel in den Plumages der Population häufiger auftreten, kann diese Methode zu einer ungewollt starken Dominanz von wenigen Genotypen im neuen Pool führen.

5.4.4.2 Schwacher Elitismus

Verwendet ein EA als Ersetzungsschema den sogenannten *schwachen Elitismus* (*WE*⁵⁷), so werden wie beim Elitismus die Genotypen der gemäß ihres Fitness-Werts *R* besten Plumages der Ausgangspopulation ausgewählt und für die Übernahme in den neuen Strategie-Pool vorgemerkt.

Allerdings werden sie nicht direkt beziehungsweise unbedingt unverändert in den Nachfolge-Pool übernommen. Sondern sie werden vorher (wie die durch Rekombination erzeugten Kind-Genotypen des Interims-Pools) der vom Agenten eingesetzten Mutations-Methode übergeben.

Da die Mutation die Gen-Werte eines Genotyps verändert, verringert sich beim schwachen Elitismus im Vergleich zum Elitismus das Risiko, dass die Gen-Vielfalt im neuen Strategie-Pool zu schnell in zu starkem Maße eingeschränkt wird.

Denn beim Elitismus kann es sein, dass ein bestimmter Genotyp mehrmals unverändert in den neuen Pool übernommen wird, falls dieser mehrfach mit überdurchschnittlich hohen Fitness-Werten in den Plumages der Population enthalten ist. Dadurch kann es im neuen Pool sogar zu einer Dominanz eines bestimmten Genotyps der Ausgangspopulation kommen, was sich auf die Flexibilität und Adaptionfähigkeit des Agenten unvorteilhaft auswirkt.

Im Gegensatz dazu sorgt der schwache Elitismus durch die Anwendung einer Mutations-Methode dafür, dass ein solcher Genotyp nicht mehrmals unverändert von der Ausgangspopulation übernommen wird. Und er bietet dadurch einem Agenten die Möglichkeit, seine guten Verhandlungsstrategien durch eine gemäßigte Mutation eventuell noch ein Stück weit besser zu machen, ohne sich für die einzelnen Gene von einem als gut bewerteten Teil des Suchraums zu weit zu entfernen.

5.4.5 Spezialverfahren

In den vorhergehenden Abschnitten wurden vielfältige Einstellungen und Methoden erläutert, mittels derer die Adaptionprozesse der MACSIMA-Agenten konfiguriert werden können. Eine zusammenfassende Übersicht der Einstellmöglichkeiten findet sich in Tabelle 4.

Weitere Details zu den Einstellmöglichkeiten für MACSIMA-Simulationsszenarien finden sich in Abschnitt 5.2.1 (siehe Tabelle 2) und in den Tabellen zu der für MACSIMA-Simulationsszenarien verwendeten Notation in Abschnitt 6.3.

⁵⁶ Im Englischen: *Elitism*

⁵⁷ Abkürzung der englischen Bezeichnung *Weak Elitism*.

Darüber hinaus gibt es noch weitere optional einstellbare Spezialverfahren wie das *Highest-Fitness-First-Verfahren (HFF)* und das *Ticket-Limitierungs-Verfahren*, die im Rahmen dieser Arbeit neu entwickelt wurden und in den folgenden Abschnitten erläutert werden.

Bei diesen Spezialverfahren handelt es sich nicht um notwendige, sondern nur um optionale Komponenten des Adaptionsprozesses eines MACSIMA-Agenten. Sie wurden entwickelt, um bei der Analyse von Simulationsergebnissen entdeckte Verbesserungspotenziale auszuschöpfen und dadurch die Zielgerichtetheit und Geschwindigkeit des Adaptionsprozesses zu erhöhen.

Pool-Einstellungen	Populations-Einstellungen
<ul style="list-style-type: none"> – Pool-Größe (P) – Pool-Anzahl – Sell-Pool – Buy-Pool 	<ul style="list-style-type: none"> – Populations-Größe (Pn) – Populations-Anzahl – Sell-Population – Buy-Population
Lern-Modus	Fest definierte Lernverfahren
<ul style="list-style-type: none"> – NoLearning – 1-Pool-Modus – 2-Pool-Modus 	<ul style="list-style-type: none"> – STDEA-Verfahren
Informationsaustausch-Einstellungen	Informationsnutzungs-Modus
<ul style="list-style-type: none"> – NoExchange (N) – Everybody (E) – InGroup (G) – Informationsaustausch-Quote (xPyF) 	<ul style="list-style-type: none"> – Internes Lernen (INT) – Externes Lernen (EXT) – Gemischtes Lernen (MIX)
Fitness-Berechnungsmethoden	Selektions-Methoden
<ul style="list-style-type: none"> – PmA-Methode – AvP-Methode – BaP-Methode – MdP-Methode – RMF-Methode 	<ul style="list-style-type: none"> – Binary competition (BiC) – Roulette Wheel Selection (RWS) – Deterministic Selection (DtS) – Deterministic Average selection (DtA) – Delete Worst Selection (DwS) – Deterministic Minority Selection (DMS) – Deterministic Minority Selection 2 (DMS 2) – Deterministic Minority Selection 3 (DMS 3)
Rekombinations-Methoden	Mutations- Methoden
<ul style="list-style-type: none"> – n-Punkt-Crossover (nPC) – Random Crossover (RaC) – No Recombination (NoR) 	<ul style="list-style-type: none"> – Single Mutation (SMU) – General Mutation (GMU)
Ersetzungsschema-Einstellungen	Spezialverfahren
<ul style="list-style-type: none"> – Elitismus (E) – Schwacher Elitismus (WE) 	<ul style="list-style-type: none"> – Highest-Fitness-First-Verfahren (HFF) – Ticket-Limitierungs-Verfahren (T)

Tabelle 4: Einstellungen für die Instanziierung des Adaptionsprozesses eines Agenten

5.4.5.1 Highest-Fitness-First-Verfahren

Normalerweise wird aus dem Pool des adaptiven Verhandlungsmoduls, sofern er mehr als ein Element enthält, immer ein Genotyp zufällig ausgewählt und in der Verhandlung eingesetzt. Die Reihenfolge, in der die Genotypen eingesetzt werden, ist dabei beliebig beziehungsweise hängt von der Reihenfolge ab, in der der Pool vom evolutionären Algorithmus mit den Genotypen der Kinder-Generation aufgefüllt wird.

Wird jedoch das *Highest-Fitness-First-Verfahren (HFF)* eingesetzt, so wird vom EA für jeden Kind-Genotyp, der in den Pool aufgenommen wird, ein sogenannter *Potenzialwert*

übergeben, der dem aus den Fitness-Bewertungen seiner Eltern-Genotypen gemittelten Wert entspricht. Die Idee dabei ist, dass ein Kind-Genotyp, dessen Eltern-Genotypen hohe Fitness-Bewertungen erzielt haben, ebenfalls ein hohes Potenzial aufweist, vergleichbare Fitness-Bewertungen zu erreichen.

Ist HFF aktiviert, so wählt das Verhandlungsmodul den nächsten in einer Verhandlung einzusetzenden Genotyp nicht mehr zufällig aus, sondern es sortiert die Genotypen im Pool gemäß ihrer Potenzialwerte und selektiert die Genotypen anhand der sich ergebenden Reihenfolge, wobei als erstes der Genotyp mit dem höchsten Potenzialwert eingesetzt wird.

Dies bietet vor allem beim gemischten Lernen Vorteile, da dann eine vom Agenten verwaltete Population nicht nur durch intern generierte Plumages, sondern auch durch externe Plumages, die der Agent von anderen Agenten empfängt, aufgefüllt wird.

In diesem Fall besteht nämlich die Gefahr, dass die Population durch den Erhalt von externen Plumages so schnell aufgefüllt wird, dass der Schwellenwert für den Start des EA erreicht wird, bevor der Agent in der Lage war, sämtliche in seinem Pool vorhandenen Plumages in Verhandlungen erfolgreich einzusetzen. Die in den nicht eingesetzten Genotypen codierten erlernten Verhandlungsstrategien gehen dem Agenten dann verloren und sind in seinem weiteren Adaptionsprozess nicht mehr nutzbar. Ob dies eintritt, hängt vor allem von den Größen von Pool und Population und deren Verhältnis zueinander ab. Ist die Größe des Pools nur unwesentlich kleiner als die Größe der Population, so steigt die Wahrscheinlichkeit hierfür an.

Das HFF-Verfahren sorgt dafür, dass Genotypen mit einem hohen Potenzial als erste in den vom Agenten geführten Verhandlungen eingesetzt werden und erhöht so die Wahrscheinlichkeit dafür, dass ihr Gen-Material in die Population aufgenommen und an die nachfolgende Generation weitergegeben wird. Dadurch verhindert HFF, dass eigenes erlerntes Verhalten schnell wieder verlernt und durch von außen übernommene Verhaltensweisen ersetzt wird.

Die Evaluationsergebnisse zeigen, dass der Einsatz des HFF-Verfahrens zu einem deutlichen Zuwachs an Performanz führen kann (siehe Abschnitt 8.6). Eine mögliche Erweiterung dieses Verfahrens könnte darin bestehen, beim gemischten Lernen einen Teil der in einer Population vorhandenen Plätze für intern generierte Plumages zu reservieren. Der Lernprozess eines Agenten könnte dann erst starten, nachdem die reservierten Plätze mit eigenem Gen-Material aufgefüllt wurden. Dies würde die Fortpflanzungswahrscheinlichkeit für die im Pool des Agenten enthaltenen Genotypen beziehungsweise die Wahrscheinlichkeit für die Weitergabe des vom Agenten in seinem bisherigen Adaptionsprozess erlernten Verhaltens erhöhen.

5.4.5.2 Ticket-Limitierung

In den Simulationen hat sich gezeigt, dass ein Agent einen Genotyp unter Umständen über einen längeren Zeitraum in vielen aufeinander folgenden Verhandlungen einsetzt, ohne mit ihm einen erfolgreichen Verhandlungsabschluss zu erzielen.

Da der aktuell eingesetzte Genotyp normalerweise nur dann gegen einen anderen Genotyp aus dem Pool ausgetauscht wird, nachdem er erfolgreich eingesetzt wurde, kann ein „erfolgloser“ Genotyp den Einsatz der restlichen Genotypen im Pool über längere Zeit „blockieren“ und damit insbesondere beim gemischten Lernen deren Fortpflanzungswahrscheinlichkeit reduzieren.

Darüber hinaus führt das Festhalten an erfolglosen Genotypen und das damit verbundene Blockieren von alternativen Verhandlungsstrategien aber auch dazu, dass die Population

langsamer gefüllt wird und die Lernfrequenz des Agenten sinkt, so dass sein Adaptionprozess behindert wird und dadurch in der Regel auch seine Performanz sinkt. Dieser Effekt ist insbesondere beim internen Lernen zu beobachten.

Um dies zu vermeiden und zu gewährleisten, dass erfolglos eingesetzte Verhandlungsstrategien regelmäßig gegen neue ausgetauscht werden, wurde das *Ticket-Limitierungs-Verfahren* entwickelt. Bei diesem wird eine Ticket-Variable eingeführt und mit einem Wert initialisiert, der festlegt, nach wie vielen direkt aufeinanderfolgenden und mit einem Abbruch endenden Verhandlungen der aktuell eingesetzte Genotyp gegen einen neuen Genotyp aus dem Pool ausgetauscht wird. Bei jedem Verhandlungsabbruch wird der Wert der Ticket-Variable um eins vermindert. Bei einer erfolgreich abgeschlossenen Verhandlung erhält die Variable hingegen wieder ihren Ausgangswert. Sinkt der Wert der Ticket-Variable auf Null, so wird der eingesetzte Genotyp ausgewechselt.

Abbildung 53 verdeutlicht, wie das Verfahren der Ticket-Limitierung abläuft, wenn der Agent eine Nachricht über den erfolgreichen Abschluss (ACCEPT) oder den Abbruch (REJECT) einer Verhandlung erhält.

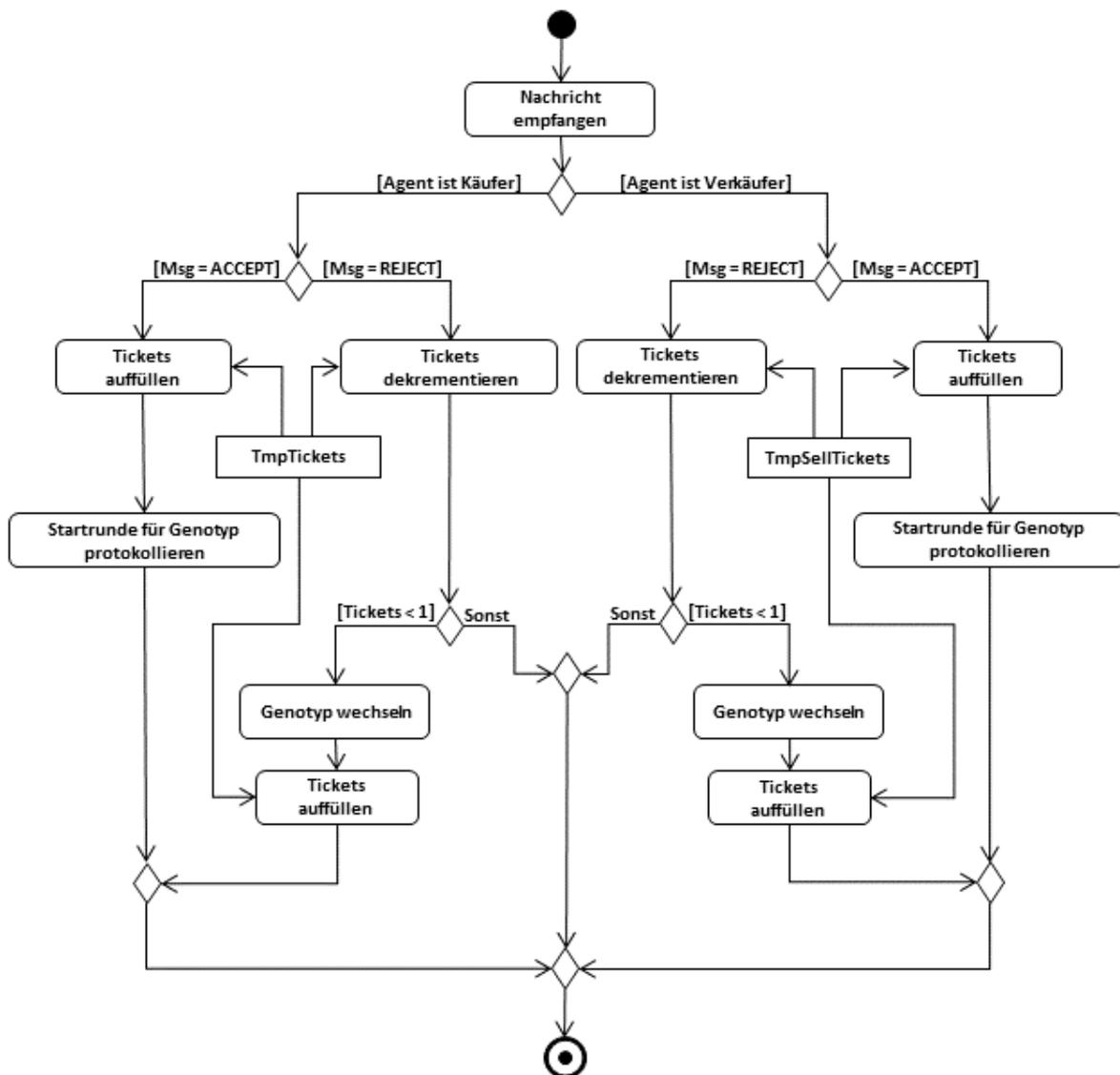


Abbildung 53: UML-Aktivitätsdiagramm für das Ticket-Limitierungs-Verfahren⁵⁸

⁵⁸ Quelle: In Anlehnung an [Stuedter 2010, S. 42] (betreute Diplomarbeit)

Sowohl als Verkäufer als auch als Käufer überprüft er, ob diese Nachricht ein REJECT enthält und dekrementiert dann für den eingesetzten Genotyp die entsprechende Ticket-Variablen beziehungsweise die Anzahl seiner noch vorhandenen „Tickets“. Sind für den aktuell eingesetzten Genotyp keine Tickets mehr vorhanden, so wird er zwangsweise gegen einen anderen aus dem gleichen Pool ausgewechselt. Enthält die empfangene Nachricht hingegen die Information, dass ein Abschluss erzielt wurde, so werden die Anzahl der Tickets und die Startrunde des eingesetzten Genotyps auf den Anfangswert zurückgesetzt.

5.4.6 Das STDEA-Adaptionsverfahren

Die grundsätzliche Idee zur dezentralen Adaption innerhalb von MAS auf der Grundlage eines EA und des Austauschs von Plumages zwischen den Agenten wurde von Smith und Taylor in [Smith & Taylor 1998] beschrieben. Das von ihnen vorgestellte Konzept wurde in [Eymann 2000] als *STDEA* (*Smith and Taylor's distributed evolutionary algorithm*) bezeichnet und in AVALANCHE sowie in MACSIMA implementiert.

STDEA weist eine Menge von festen Einstellungen für das evolutionäre Lernen sowie für Informationsaustausch und -nutzung auf. Bei diesem Verfahren verfügt ein Agent lediglich über eine Pool-Größe von $P = 1$, also nur über einen einzigen Genotyp, den er in seinen Verhandlungen einsetzt. Gelingt dem Agenten ein erfolgreicher Verhandlungsabschluss, so ordnet er diesem Genotyp mit der Fitness-Bewertungsmethode *PmA* (siehe Abschnitt 5.4.2.1) einen Fitness-Wert zu und generiert ein Plumege, das er anschließend an einen innerhalb seiner Agenten-Gruppe zufällig ausgewählten Agenten sendet.

Die Agenten sammeln Plumages, die sie empfangen, in einer Population. Überschreitet die Anzahl der in der Population eines Agenten gesammelten Plumages einen definierten Schwellenwert, so startet der Agent einen *einfachen evolutionären Algorithmus*, der die Phasen Selektion, Rekombination und Mutation aufweist. In [Smith & Taylor 1998] war dieser Schwellenwert auf fünf und in den mit MACSIMA durchgeführten Testreihen auf zwanzig Plumages festgelegt. Bei STDEA sind die Phasen des EA durch die folgenden Methoden realisiert:

1. Selektion: Aus der Population wird das Plumege ausgewählt, das den höchsten Fitnesswert aufweist und der in diesem enthaltene Genotyp selektiert.
2. Rekombination: Der selektierte Genotyp wird mit dem aktuell im Pool des Agenten liegenden Genotyp rekombiniert, indem die RaC-Methode mit dem Parameterwert *preferBetterParent* = 0,5 angewendet wird. Dies bedeutet, dass ein Kind-Genotyp erzeugt wird, bei dem jedes einzelne Gen mit einer Wahrscheinlichkeit von jeweils 50 Prozent von einem der beiden Eltern-Genotypen geerbt wird.
3. Mutation: Sie nutzt die SMU-Methode mit einer Mutations-Rate von 0,05 und einer Mutations-Weite von 0,1.

Anschließend wird die Population geleert und mit dem aus dem EA resultierenden neuen Genotyp der einzige im Pool enthaltene Genotyp ersetzt.

6 Simulationsinfrastruktur

6.1 Einleitung und Übersicht

Im Folgenden wird die Simulationsinfrastruktur von MACSIMA beschrieben, die für die in Kapitel 8 durchgeführten Simulationen verwendet wird. Mit ihrer Hilfe kann über ein mächtiges User Interface eine Vielzahl an Simulations-Szenarien mit einer Fülle an heterogenen Parametereinstellungen effizient, komfortabel und präzise definiert und simuliert werden, um damit in einer Reihe von Simulationsläufen zu experimentieren.

Darüber hinaus bietet die Simulationsinfrastruktur umfangreiche Protokollierungs-Funktionalitäten. Sie ist in der Lage, Protokoll-Dateien zu den Zuständen von Verhandlungen, Agenten, Gen-Pools, Lernverfahren usw. zu generieren und für die protokollierten Simulations-Rohdaten automatisierte Auswertungsprozesse anzustoßen.

Diese Auswertungsprozesse nutzen die Verfahren der in Kapitel 7 vorgestellten Evaluationsmethodik, die Werkzeuge zur Auswertung und Verdichtung der generierten Protokoll-Dateien bereitstellt. Damit werden die Simulations-Rohdaten in gut strukturierte und für Evaluationszwecke (siehe Kapitel 8) effizient nutzbare Simulations-Ergebnisdaten überführt.

Dies ist wichtig, da die Simulationsläufe in MACSIMA nicht-deterministisch verlaufen und somit keiner wie der andere ist – gerade auch, wenn man für das gleiche Szenario wiederholt Simulationen durchführt. Der Zufall hat unter anderem einen großen Einfluss darauf,

- welche Agenten miteinander verhandeln,
- welche in Genotypen codierte Verhandlungsstrategien sie verwenden,
- wie solche Verhandlungen im Detail ablaufen und
- auf welche Weise die Agenten aus den in den Verhandlungen gemachten Erfahrungen mittels ihrer *evolutionären Algorithmen (EA)* lernen.

Es entwickelt sich in jedem Simulationslauf jeweils ein emergentes Systemverhalten, das je nach Parametrisierung des zugrundeliegenden Simulations-Szenarios zu mehr oder weniger stark variierenden Ergebnisvariablen führt.

Um trotz solcher Streuungseffekte die Auswirkungen einer großen Vielfalt an interessanten Szenario-Parameter-Einstellungen auf die bei einer Simulation zu erwartenden Dynamiken und Ergebnisse zielgerichtet untersuchen zu können, bedarf es einer umfangreichen und gut strukturierten Datenbasis. Eine solche Datenbasis wird mittels der Simulations- und Protokollierungs-Funktionalitäten von MACSIMA geschaffen.

Um darauf in der Folge die in Kapitel 7 beschriebene Evaluationsmethodik effizient anwenden zu können, gerade auch im Fall einer Vielzahl an simulierten und evaluierten Szenarien wie in Kapitel 8, nutzt die Simulationsinfrastruktur darüber hinaus auch eine spezielle Notation. Ihre Konventionen zur Benennung der von generierten und verarbeiteten Dateien wird in Abschnitt 6.3 beschrieben.

Die vorgestellte Notation erlaubt sowohl eine effiziente und eindeutige Benennung von heterogen parametrisierten Simulations-Szenarien mit ihren zugehörigen Simulations-Rohdaten und -Ergebnissen, als auch bereits eine präzise Kurzbeschreibung ihrer wesentlichen Parametereinstellungen. Da sowohl die Anzahl der Szenarien-, Protokoll- und Auswertungsdateien als auch deren Gesamtdatenvolumen sehr groß werden kann, hilft die Notation dabei, die generierten Dateien in allen Phasen des Parametrisierungs-, Simulations-, Auswertungs- und Evaluationsprozesses einander zuzuordnen bzw. voneinander unterscheiden zu können.

Die weitreichenden Funktionalitäten der MACSIMA-Simulationsinfrastruktur schaffen so die Grundlage zur Durchführung von großangelegten Evaluationsprozessen mit einer Masse an Szenarien und generierten Daten. Im Evaluationsteil⁵⁹ dieser Arbeit (siehe Kapitel 8) wurden beispielsweise mehrere hundert heterogene Szenarien definiert, simuliert und ausgewertet, um unterschiedliche Fragestellungen zu beleuchten.

Die Beschreibung der Simulationsinfrastruktur in diesem Kapitel gliedert sich in

- die Beschreibung der MACSIMA-Simulationsumgebung in Abschnitt 6.2, gemeinsam mit den Parametrisierungsmöglichkeiten für Güter und Agenten, Strategie- und EA-Einstellungen sowie für den Start von Simulationen, welche ihre Benutzeroberfläche zur Verfügung stellt (s. Abschnitte 6.2.2.1 – 6.2.2.3),
- die Definition und Erklärung einer Notation für zu simulierende Szenarien in Abschnitt 6.3, welche eine hohe Aussagekraft bzgl. der Szenario-Parametrisierung aufweist und daher die Generierung, Weiterverarbeitung und Analyse von umfangreichen Mengen an Simulations- und Auswertungsergebnissen wesentlich vereinfacht, sowie in
- die Erläuterung der Art und Weise, mit der Simulationsläufe protokolliert werden (s. Abschnitt 6.4).

6.2 MACSIMA-Simulationsumgebung

Mit MACSIMA ist es möglich, mit den in 6.2.1.1 dargelegten Spezifikations- und Simulations-Funktionalitäten eine sehr große Anzahl von Szenarien zu definieren und diese als .sce-Dateien in einem Verzeichnis zu speichern.

Jedes zu simulierende Szenario bzw. seine .sce Datei werden dabei gemäß der in Abschnitt 6.3 beschriebenen Notation mit einem Namen versehen, der hinsichtlich der Parameter-Einstellungen des Szenarios eine hohe Aussagekraft besitzt.

Im Anschluss daran kann man MACSIMA starten und in der graphischen Benutzeroberfläche einstellen, mit welcher Anzahl an Simulationsrunden und Wiederholungen jedes der Szenarien simuliert werden soll (s. Abschnitt 6.2.2.3). Dabei lassen sich mittels der Features *Directory Simulation* und *Batch Simulation* alle oder einige wenige der in dem Verzeichnis abgelegten Szenarien im Rahmen eines Batch-oder Directory-Laufes simulieren und darauf folgend automatisiert auswerten (s. u.).

Die Agenten der MACSIMA-Simulationsumgebung protokollieren ihren inneren Zustand sowie ihre Aktivitäten, Verhandlungsschritte und Endresultate in mehreren getrennt und sta-

⁵⁹ Die im Evaluationsteil ausgewertete Datenmenge befindet sich im beinahe zweistelligen Terabyte-Bereich. Die Simulationen wurden auf einem Linux-System mit einer aktuellen Intel i7-Architektur mit 16 GB RAM durchgeführt.

tistisch evaluierbaren Protokoll-Dateien. Diese Rohdaten werden während der Simulationszeit anhand der beim Simulationsstart vorgenommenen Einstellungen (s. Abschnitt 6.2.2.3) in das Verzeichnis geschrieben, in dem die Simulations-Szenarien liegen bzw. gestartet wurden. Details zur Protokollierung der Simulationsläufe finden sich in Abschnitt 6.4.

6.2.1 Grundlegende Einstellungen

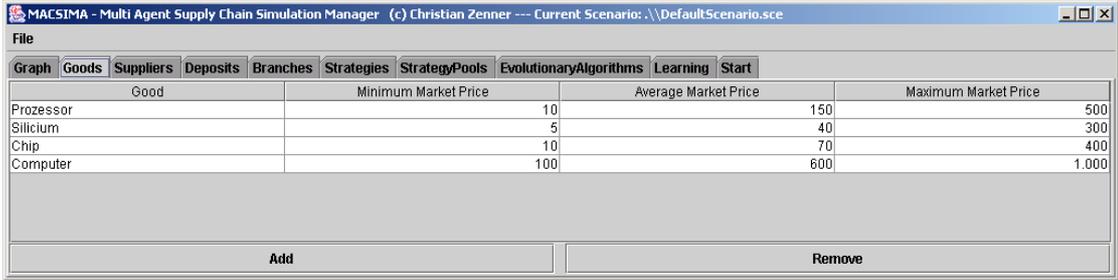
Die durchgeführten Simulationsläufe konzentrieren sich zumeist auf verschiedene Instanziierungen eines fünfstufigen Wertschöpfungsnetzwerks mit linearem Güterdurchfluss für die Computer-Fertigung. Die simulierten Szenarien enthalten in den meisten Fällen jeweils zehn Agenten auf jeder der fünf Stufen dieser Wertschöpfungskette, insgesamt also 50 Agenten, die je nach den Szenario-Einstellungen miteinander kooperieren, Erfahrungen austauschen und daraus lernen können – oder eben nicht bzw. nur eingeschränkt. Die Basis-Topologie, die im Rahmen der Simulationsläufe zumeist verwendet wird, wurde bereits in Abschnitt 5.2.1 (s. Abbildung 34) vorgestellt.

Die im Rahmen der in Kapitel 8 vorgestellten Evaluation durchgeführten Simulationen gingen dabei häufig über jeweils 200.000 Runden, was einerseits groß genug für aussagekräftige Ergebnisse und andererseits praktikabel ist, da die Simulationen dann in akzeptabler Zeit terminieren und das Volumen der generierten Roh- und Ergebnisdaten noch handhabbar bleibt. Sofern im Evaluationsteil der Arbeit Einstellungen verwendet werden, die von diesen Basis-Einstellungen abweichen, wird gesondert darauf hingewiesen.

Im Folgenden wird die graphische Benutzeroberfläche von MACSIMA beschrieben, über welche die Parametereinstellungen bei der Definition von Szenarien vorgenommen werden.

6.2.1.1 Einstellungen für Güter und Agenten

Güter stellen die einfachsten simulationsrelevanten Entitäten dar. Sie werden über die Eingabe ihres Namens sowie weiteren Informationen zu den für sie geltenden marktbezogenen Preisen, nämlich ihrem minimalen (*Minimum Market Price*), durchschnittlichen (*Average Market Price*) und maximalen (*Maximum Market Price*) Marktpreis definiert, wie aus dem folgenden Bildschirmfoto (siehe Abbildung 54) ersichtlich ist.



Good	Minimum Market Price	Average Market Price	Maximum Market Price
Prozessor	10	150	500
Silicium	5	40	300
Chip	10	70	400
Computer	100	600	1.000

Abbildung 54: Einstellungen für Güter und deren Preisgrenzen in MACSIMA

Im Rahmen der durchgeführten Simulationen haben sich bestimmte Preisgrenzen-Einstellungen als gute und praktikable Standard-Einstellungen erwiesen, die aber jederzeit vom Szenario-Designer durch ein selbstdefiniertes Preisgefüge ersetzt werden können. Die im Evaluationsteil der Arbeit (s. Kapitel 8) verwendeten Preisgrenzen-Einstellungen sind im Anhang erläutert.

Im Supply Chain Simulation Manager können für die drei generischen Agenten-Typen *Supplier*, *Deposit* und *Branch* Einstellungen vorgenommen werden. Die Eigenschaften und die Architektur dieser generischen Agenten-Typen sind in Abschnitt 5.2.2 beschrieben. Für sie bietet die graphische Benutzeroberfläche als Parametrisierungsmöglichkeiten

- die Definition der Höhe ihres Startkapitals (*Initial Cash*), mit dem sie in das marktliche Wertschöpfungsnetzwerk eintreten,
- die Angabe eines Betrags an Lebenshaltungskosten (*Cost of Life*), die einem Agenten für jede Simulationsrunde abgezogen werden und
- je nach Agenten-Typ die Zuordnung einer Menge von Input- und/oder Output-Gütern (*InputGoods* und *OutputGoods*) aus der Menge an definierten, im Wertschöpfungsnetzwerk gehandelten Gütern.

Zusätzlich können in Abweichung zu den in der Güter-Definition allgemein festgesetzten Preisschranken für die von einem Agenten gehandelten Güter auch Minimal-, Durchschnitts- und Maximalpreis spezifiziert werden, die nur agentenindividuell gelten.

Sollten keine agentenindividuellen Preisgrenzen-Einstellungen vorgenommen werden, gelten die bei der Güterdefinition spezifizierten Preisinformationen und -grenzen.

Der eingestellte Minimalpreis eines Agenten für ein von ihm verwaltetes Output-Gut repräsentiert einen absoluten Mindestpreis, unterhalb dessen er zur Deckung von Fixkosten im Normalfall nicht bereit ist, ein Stück davon zu verkaufen. Der von ihm in Verhandlungen genutzte Mindestpreis darf (vergleiche Abschnitt 5.3.3) nicht darunter, aber in Abhängigkeit von Beschaffungs- und Produktionskosten sowie der vom Agenten gewünschten Gewinnmarge darüber liegen. Eine Ausnahme hiervon bilden nur Preisadjustierungen beziehungsweise Sonderverkäufe, die vom Preisautomaten initiiert werden (siehe Abschnitt 5.3.3.7).

Bei Supplier-Agenten kann über die Einstellung *Output Frequency* die Anzahl der Simulationsrunden definiert werden, die der Agent benötigt, um sein Output-Gut nach einem erfolgreichen Deal wieder zum Verkauf anbieten zu können.

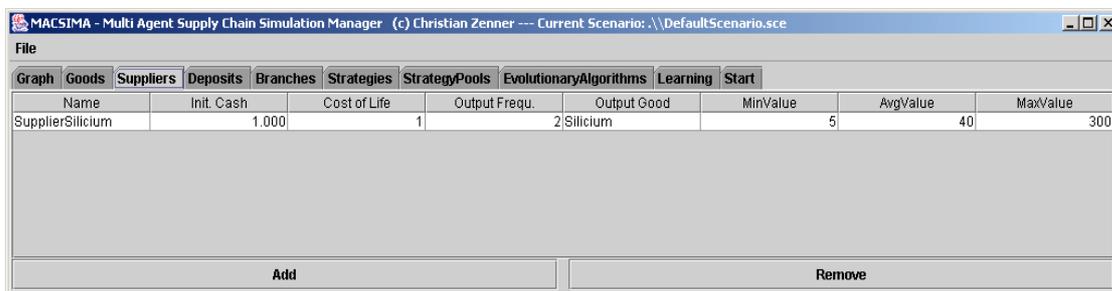


Abbildung 55: Einstellungen für Supplier-Agenten in MACSIMA

Deposit-Agenten besitzen eine Produktionsfunktion. Sie kaufen Input-Güter und produzieren daraus Output-Güter, die sie auf dem Markt anbieten und verkaufen.

Ihre Produktionsfunktion benötigt die Angabe von Produktionskosten (*Production Cost*). Darüber hinaus gibt der Parameter *Production Time* an, wie viele Simulationsrunden sie benötigen, um aus den Input-Gütern ein Output-Gut herzustellen.

Zudem verwenden diese Agenten den Parameter *Profit Wish*, über den die von einem Agenten gewünschte prozentuale Gewinnmarge, d. h. ein prozentualer Aufschlag auf seine Beschaffungs- und Produktionskosten, eingestellt werden kann, die der Agent in seinen Verhandlungen nach Möglichkeit zu erzielen versucht.

Name	Init. Cash	Cost of Life	Producing Time	Output Good	MinValue	AvgValue	MaxValue	InputGoods	Production Cost	ProfitWish
DepositChip	1.000	1	2	Chip	10	70	400	Silicium	5	0,1
DepositProze...	1.000	1	2	Prozessor	10	150	500	Chip	5	0,1
DepositComp...	1.000	1	2	Computer	100	400	600	Prozessor	5	0,1

Abbildung 56: Einstellungen für Deposit-Agenten in MACSIMA

Branch-Agenten dienen in MACSIMA der Modellierung der Nachfrage und repräsentieren konsumierende Entitäten. Sie kaufen daher Input-Güter, für die analog zu den Supplier-Agenten verschiedene Preisgrenzen, z. B. Maximalpreise, definiert werden können.

Name	Init. Cash	Cost of Life	InputGoods
BranchComputer	1.000	1	Computer

Abbildung 57: Einstellungen für Branch-Agenten in MACSIMA

6.2.2 User Interface

Der Supply Chain Simulation Manager MACSIMA erlaubt bei der Definition von Simulationsszenarien vielfältige Einstellmöglichkeiten. Auf diese kann über im Java-Fenster vorhandene Tabs zugegriffen werden, welche die Einstellmöglichkeiten thematisch gliedern und die jeweils über einen tabellarischen Aufbau verfügen. Es existieren Tabs zu den Entitäten

- Güter (*Goods*) und
- Agenten (*Suppliers*, *Deposits* und *Branches*)

sowie zu den Themenkomplexen

- Topologie (*Graph*),
- Gen- beziehungsweise Verhandlungsstrategie-Einstellungen (*Strategies*),
- Lernverfahren (*Evolutionary Algorithms* und *Learning*)

Außerdem bietet das Tab *Start* eine zusammenfassende Übersicht, über welche die in den anderen Tabs vorgenommenen Einstellungen komfortabel kombiniert und rekombiniert sowie die wichtigsten Einstellungen für den Simulationsstart vorgenommen werden können.

Die meisten Tabs sind tabellarisch organisiert und es können Zeilen und textuelle oder numerische Einträge in deren Zellen hinzugefügt oder entfernt werden. Darüber hinaus können viele Eigenschaften über Auswahl-Boxen aktiviert oder deaktiviert werden. Bestimmte Einstellungen werden auch über weitere Fenster vorgenommen, die über Buttons geöffnet werden können.

6.2.2.1 Strategie-Einstellungen

Bei jedem Genotyp im Gen-Pool eines Agenten handelt es sich um eine *Strategie-Parameter-Kombination*, deren Parameter den Genen eines Agenten entsprechen. Die Strategie-Parameter werden in Abschnitt 5.3.2 definiert und erläutert. Gemeinsam spezifizieren sie eine Verhandlungsstrategie des Agenten.

Zu Simulationsbeginn muss der zunächst leere Gen-Pool eines Agenten mit initialen Genotypen beziehungsweise Verhandlungsstrategien gefüllt werden, damit der Agent überhaupt in erste Verhandlungen starten kann. Diese Verhandlungsstrategien behält der Agent bei, bis in Abhängigkeit von seinen Lernfähigkeiten gegebenenfalls evolutionäre Anpassungsprozesse erfolgen.

Je nach seiner definierten Größe wird der Gen-Pool eines Agenten zum Simulationsstart anhand der Strategiekombinationen mit so vielen Genotypen befüllt, wie er maximal enthalten darf. Bei einer Gen-Pool-Größe von n , müssen daher auch n initiale Genotypen erzeugt werden. Die Größe des Gen-Pools wird im Tab *Learning* festgelegt wird (siehe Abbildung 61 in Abschnitt 6.2.2.2).

Der Tab *Strategies* dient dazu, Funktionen zu definieren, anhand derer ein initialer Genotyp im Gen-Pool erzeugt beziehungsweise die Gen-Werte eines Genotypen initialisiert werden. Jede Zeile entspricht einer solchen *Initialisierungsfunktion*, deren Name in der ersten Spalte festgelegt wird. Vollständig spezifizierte Initialisierungsfunktionen werden zur Wiederverwendung gespeichert und in der *Strategies*-Tab-Ansicht in Tabellenform angezeigt. Initialisierungsfunktionen können hinzugefügt, bearbeitet und gelöscht werden.

Nach der ersten Spalte folgen für jedes einzelne Gen eines Genotyps jeweils weitere vier Spalten, in denen definiert wird, wie sein Gen-Wert genau initialisiert wird. In der ersten dieser (pro Gen) vier Spalten mit der Benennung *Setting* kann eingestellt werden, ob der jeweilige Gen-Wert statisch (*Static*) festgelegt sein soll oder anhand einer Normalverteilung innerhalb eines bestimmten Wertebereichs randomisiert (*Random*) bestimmt wird.

Name	AcqSetting	AcqMean	AcqWidth	Acq	DCS...	DC...	DC...	DC	DJS...	DJ...	DJ...	DJ	SFS...	SF...	SF...	SF	WMGe...	WM...	WM...	WM	RepS...	Rep...	Rep...	Rep
Acq 0,3	Static			0,3	Static			0,25	Static			0,15	Static			0,75	Static			0,2	Static			1
Acq 0,4	Static			0,4	Static			0,25	Static			0,15	Static			0,75	Static			0,2	Static			1
Acq 0,5	Static			0,5	Static			0,25	Static			0,15	Static			0,75	Static			0,2	Static			1
Acq 0,6	Static			0,6	Static			0,25	Static			0,15	Static			0,75	Static			0,2	Static			1
Acq 0,7	Static			0,7	Static			0,25	Static			0,15	Static			0,75	Static			0,2	Static			1
Acq 0,2	Static			0,2	Static			0,25	Static			0,15	Static			0,75	Static			0,2	Static			1
Acq 0,8	Static			0,8	Static			0,25	Static			0,15	Static			0,75	Static			0,2	Static			1
Random Acq 0,05-0,95	Random	0,5	0,9	0	Static	0	0	0,25	Static	0	0	0,15	Static	0	0	0,75	Static	0	0	0,2	Static	0	0	1

Abbildung 58: Einstellungen für die Erzeugung von initialen Verhandlungsstrategien in MACSIMA

Bei der Einstellung *Static* wird das jeweilige Gen mit genau dem Wert initialisiert, der in der vierten genbezogenen Spalte angegeben ist. Diese Spalte trägt als Überschrift den Namen des entsprechenden Gens. Zum Beispiel wird in der Abbildung 55 bei der Initialisierungsfunktion „Acq 0,3“ das Acquisitiveness-Gen mit dem Wert 0,3 statisch initialisiert. Dies

bedeutet, dass das Acquisitiveness-Gen aller initialen Genotypen, die mit dieser Initialisierungsfunktion erzeugt werden, immer den Wert 0,3 aufweist. Da die Funktion „Acq 0,3“ auch alle anderen Gen-Werte statisch initialisiert, weisen alle mit ihr erzeugten initialen Genotypen zum Simulationsstart die gleichen Gen-Werte auf.

In vielen Fällen ist jedoch zur Erzeugung einer gewissen Gen-Vielfalt die Einstellung *Random* zu bevorzugen. Wird sie vorgenommen, so werden die Einstellungen in den genbezogenen Spalten *Mean* und *Width* (d. h. in der zweiten und dritten Spalte für jedes Gen) relevant. Die Belegung der entsprechenden Gen-Werte eines erzeugten Genotyps erfolgt dann randomisiert – und zwar innerhalb einer Gauß'schen Normalverteilung, deren Mittelwert und Breite den unter *Mean* und *Width* spezifizierten Parameter-Werten entsprechen.

Ein *Mean*-Parameter-Wert von 0,5 und ein *Width*-Parameter-Wert von 0,9 führen also dazu, dass die Genotypen des Agenten für das mit den Parametern korrespondierende Gen mit zufälligen, normalverteilten Werten innerhalb des Intervalls [0,05; 0,95] belegt werden. In Abbildung 56 weist beispielsweise die Initialisierungsfunktion „Random Acq 0,05-0,95“ entsprechende Einstellungen für das Acquisitiveness-Gen AQ auf.

Im Tab *StrategyPools* (siehe Abbildung 59) wird anschließend definiert, mit welchen der unter dem Tab *Strategies* spezifizierten Initialisierungsfunktionen die initialen Genotypen generiert werden, mit denen der Gen- beziehungsweise Strategie-Pool eines Agenten zum Simulationsstart befüllt wird.

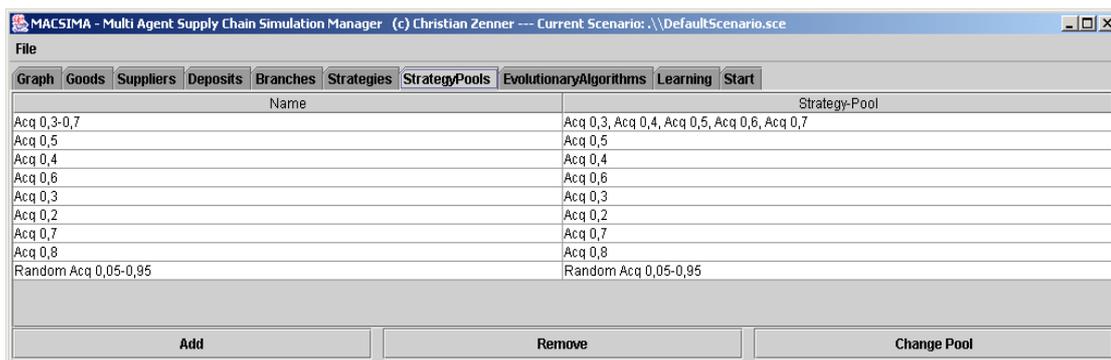


Abbildung 59: Kombination von Initialisierungsfunktionen für die initiale Befüllung des Strategie-Pools von MACSIMA-Agenten

Bei der initialen Befüllung besteht auch die Möglichkeit, die einzelnen Genotypen für den Gen-Pool mit unterschiedlichen Initialisierungsfunktionen zu erzeugen. Zu diesem Zweck können verschiedene Initialisierungsfunktionen in geordneten Listen zusammengefasst beziehungsweise miteinander kombiniert und mit einem eindeutigen Namen versehen werden. Anhand ihres Namens sind sie dann im *Start*-Tab (s. Abschnitt 6.2.2.3) bei der Definition von Szenarien auffind- und auswählbar und können bei der Definition von Szenarien leicht verwendet und gegebenenfalls ausgetauscht werden.

Enthält eine Liste nur eine Initialisierungsfunktion, so werden bei einer Gen-Pool-Größe von n alle n initialen Genotypen eines Agenten gemäß dieser Funktion erzeugt. Sind in solch einer Liste mehr als n Elemente enthalten, so werden die n initialen Genotypen nur mit den ersten n Initialisierungsfunktionen in der Liste initialisiert, indem diese der spezifizierten Reihenfolge nach angewendet werden. Weitere nachfolgende Listenelemente werden ignoriert.

Enthält eine entsprechende Liste mindestens zwei, aber weniger als n Initialisierungsfunktionen, so werden die Initialisierungsfunktionen „rollierend“ angewendet. Sind beispielsweise $n-1$ Funktionen in der Liste definiert, so werden mit diesen die ersten $n-1$ initialen Genotypen

generiert. Der n -te Genotyp wird anschließend wieder unter Verwendung der ersten in der Liste enthaltenen Initialisierungsfunktion erzeugt.

6.2.2.2 Einstellungen für das evolutionäre Lernen

Im Tab *EvolutionaryAlgorithms* (s. Abbildung 60) werden die von den Agenten verwendbaren Lernverfahren definiert. Hierzu werden nach der Benennung des zu definierenden Lernverfahrens in den Spalten *Selection*, *Recombination*, *Mutation* und *Replacement* zunächst die Parameter-Einstellungen für die genetischen Operatoren von MACSIMA (vgl. Abschnitte 5.4.3 – 5.4.4) vorgenommen. Zusätzlich erfolgt in der Spalte *FitnessCalculation* die Auswahl der zu verwendenden Fitness-Bewertungsmethode (siehe Abschnitt 5.4.2).

Gemeinsam spezifizieren die vorgenommenen Einstellungen die für einen Lernalgorithmus grundlegenden Parameter-Einstellungen, die anschließend im Tab *Learning* (s. Abbildung 61) um weitere Einstellungen, zum Beispiel bezüglich der Größe von *Population* und *Gen-Pool* ergänzt beziehungsweise komplettiert werden.

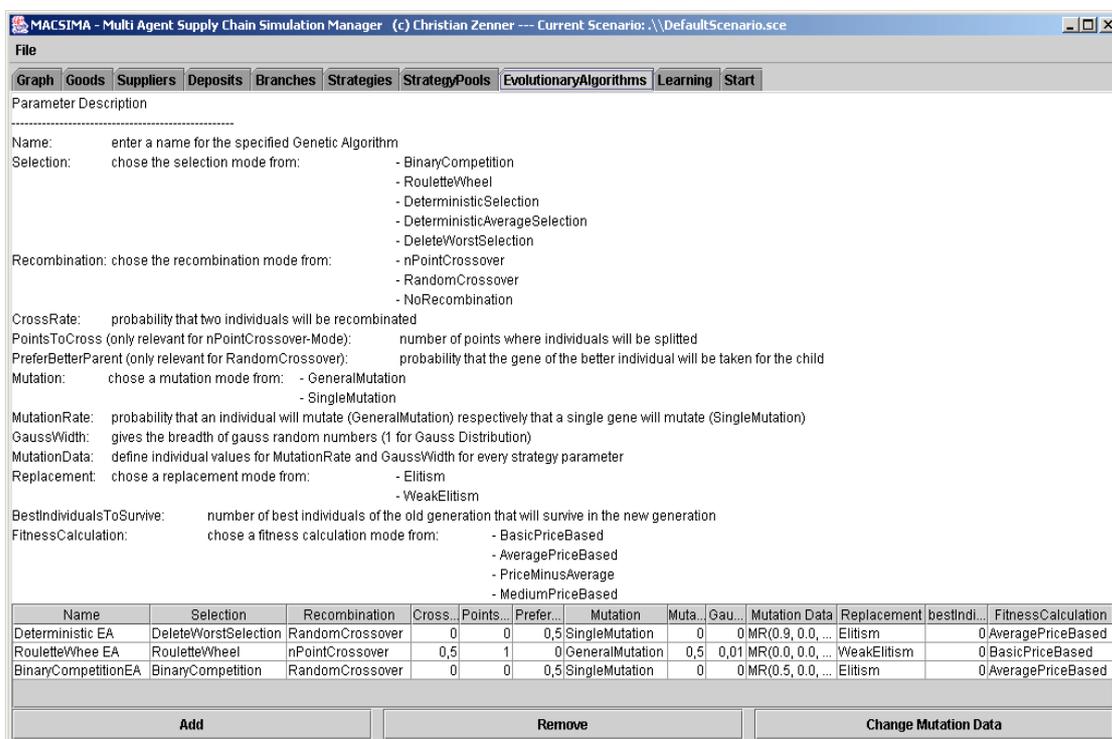


Abbildung 60: Detail-Einstellungen für das Evolutionäre Lernen in MACSIMA im Tab *EvolutionaryAlgorithms*

Die Funktion der genetischen Operatoren und der Fitness-Berechnungen wurde im Kapitel 5.4 bereits ausführlich erläutert. Die in MACSIMA gegebenen Einstellmöglichkeiten werden daher im Folgenden nur kurz beschrieben:

- Die Spalte *Selection* dient der Auswahl des Selektionsverfahrens.
- In der Spalte *Recombination* wird das Rekombinations-Verfahren ausgewählt.
- Bei Auswahl von *nPointCrossover* gibt der Parameter *CrossRate* die Wahrscheinlichkeit dafür an, dass zwei Genotypen miteinander gekreuzt werden. In diesem Fall ist der Parameter *PointsToCross* mit anzugeben, der die Anzahl der Schnittpunkte vorgibt, an denen eine Kreuzung durchgeführt werden soll. Bei 6 Genen ist eine Kreuzungsoperation nur an maximal 5 Positionen möglich.

- Bei Selektion von *RandomCrossover* muss keine *CrossRate*, aber der Parameter *PreferBetterParent* eingestellt werden. Dieser gibt die Wahrscheinlichkeit dafür an, dass pro Gen jeweils der Eltern-Genotyp mit geringerer Fitness den Gen-Wert des Eltern-Genotyps mit höherer Fitness übernimmt.
- In der Spalte *Mutation* kann zwischen den Mutations-Methoden *GeneralMutation* und *SingleMutation* ausgewählt werden.
- Bei Auswahl von *GeneralMutation* erfolgt nur eine Mutationsoperation an einem einzelnen Gen. Hierfür müssen Einstellungen zu *MutationRate* und *GaussWidth* vorgenommen werden.
- *MutationRate* gibt die Wahrscheinlichkeit für die Durchführung der Mutationsoperation an.
- *GaussWidth* bestimmt die Breite der Gauß'schen Normalverteilung, innerhalb derer mutiert werden kann.
- Bei Selektion von *SingleMutation* müssen *MutationRate* und *GaussWidth* pro Gen spezifiziert werden, welches über *Change Mutation Data* in einem separaten Fenster erfolgt. Die vorgenommenen Einstellungen sind anschließend in der Spalte *MutationData* ersichtlich.
- Die Auswahl des Ersetzungsschemas erfolgt in der Spalte *Replacement*.
- Unter *bestIndividualsToSurvive* kann die Anzahl an Genotypen eingetragen werden, die in eine nachfolgende Population übernommen werden sollen.
- In der Spalte *FitnessCalculation* wird die zu verwendende Fitness-Berechnungsmethode angegeben.

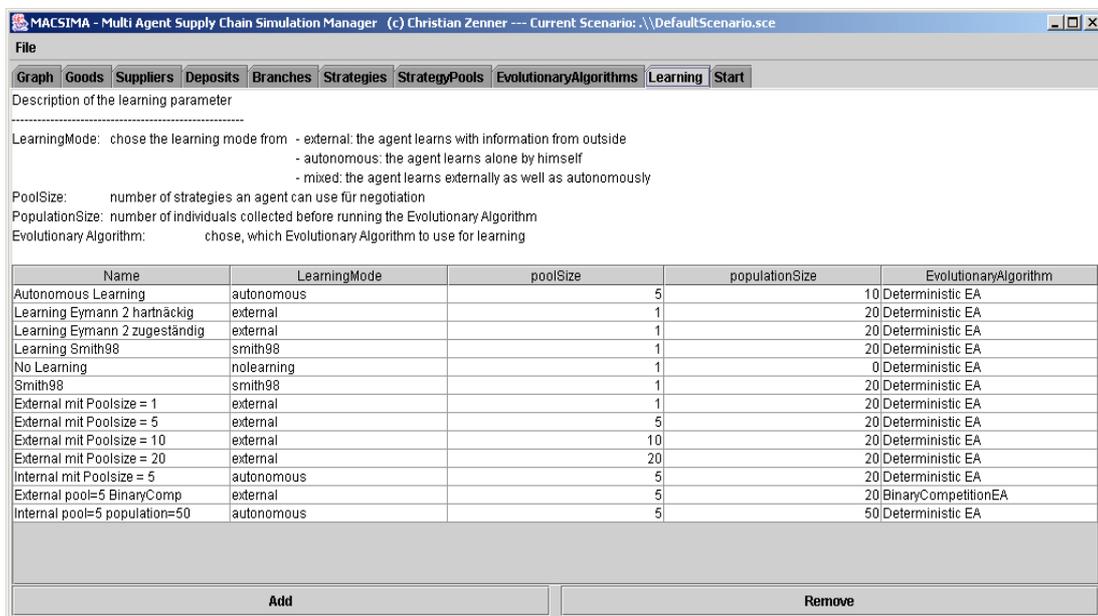


Abbildung 61: Informationsaustausch-, Pool- und Population-Einstellungen für das Evolutionäre Lernen in MACSIMA im Tab *Learning*

Nach der grundlegenden Spezifikation von evolutionären Lernverfahren im Tab *EvolutionaryAlgorithms* erfolgt für jedes der Lernverfahren im Tab *Learning* eine Komplettierung seiner Spezifikation durch die folgenden Einstellungen:

- In der Spalte *Name* wird ein eindeutiger Name für das Lernverfahren vergeben.

- Unter *LearningMode* erfolgt die Auswahl des Lernmodus (*external*, *internal*, *mixed*, *noLearning* und *Smith98*).
- In den Spalten *poolSize* und *populationSize* werden die Größen von Pool und Population definiert.
- Unter *EvolutionaryAlgorithm* stehen die im Tab *EvolutionaryAlgorithms* vordefinierten grundlegenden EA-Spezifikationen zur Auswahl.

6.2.2.3 Einstellungen für den Simulationsstart

Im Tab *Start* werden nun die in den vorhergehenden Tabs vorgenommenen Einstellungen miteinander kombiniert, um zu simulierende Szenario zu definieren und deren Simulation nach Eingabe von weiteren simulationsspezifischen Parametern zu starten. Ein komplettes Simulations-Szenario wird in diesem Tab durch die folgenden Einstellungen definiert:

- Innerhalb der Spalte *Agent* erfolgt die Auswahl der vordefinierten Agenten-Typen. Im in Abbildung 62 dargestellten Beispiel sind dies gemäß der Basis-Topologie von MACSIMA die fünf Agenten-Typen *SupplierSilicium*, *DepositChip*, *DepositProcessor*, *DepositComputer* sowie *BranchComputer*.
- Die unter *Quantity* angegebene Zahl definiert dabei die Anzahl der Agenten, die pro Typ im System gestartet werden.
- In der Spalte *GroupID* werden Agenten zu Gruppen zusammengefasst, um für diese weitere Einstellungen, nämlich z. B. bzgl. des Informationsaustauschs innerhalb der Gruppe vornehmen zu können.
- Letzteres erfolgt in der Spalte *Information Exchange Type*, wo der Modus des Informationsaustauschs auf *NoExchange*, *InGroup* und *Everybody* gesetzt werden kann.
- Unter *StrategyPool* werden anschließend den vorher definierten Agentengruppen die in den Tabs *Strategies* und *StrategyPools* spezifizierten Initialisierungsfunktionen zugeordnet, mit deren Hilfe die initialen Verhandlungsstrategien der Agenten erzeugt werden.
- Innerhalb der Spalte *Learning* wird abschließend definiert, welche der komplett eingestellten Lernverfahren von den verschiedenen definierten Agenten bzw. Agentengruppen zur Adaption an sich verändernde Gegebenheiten im Netzwerk verwendet werden.

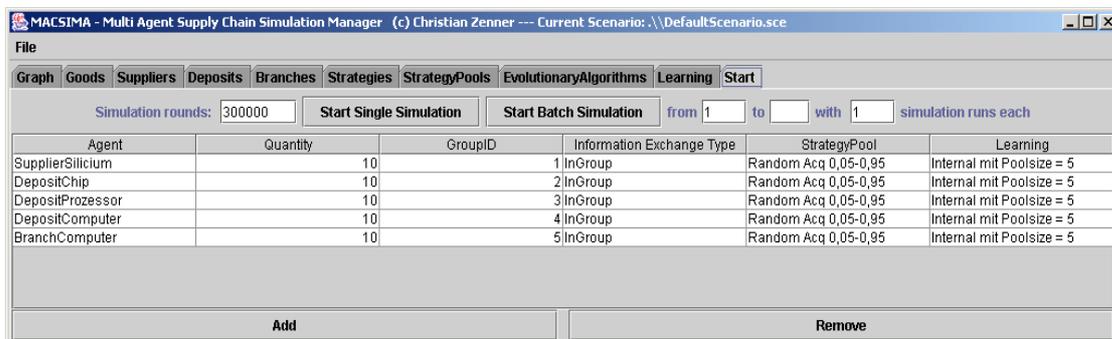


Abbildung 62: Einstellungen für den Simulationsstart in MACSIMA

Definierte Szenarien können über das File-Menü als Datei (mit der Endung *.sce*) in ein Verzeichnis gespeichert und geladen werden. Die Simulation eines aktuell geladenen Szenarios kann mittels *Start Single Simulation* gestartet werden, nachdem Eintragungen zur Anzahl

der Simulationsrunden (*Simulation Rounds*) und zur Anzahl der gewünschten Simulationslauf-Wiederholungen (im Eingabefeld *with #⁶⁰ simulation runs each*) gemacht wurden.

Weiterhin können mittels *Start Batch Simulation* Stapelverarbeitungs- bzw. Simulationsprozesse durchgeführt werden, bei denen im MACSIMA Start-Verzeichnis befindliche Szenarien mit Dateinamen der Form *BatchScenario#.sce⁶¹* nacheinander simuliert werden. Dies erfolgt in aufsteigender Reihenfolge gemäß der Eingabe einer Start- und Endnummer für die zu simulierenden Szenarien in den Eingabefeldern *from* und *to*. Erfolgt bei *to* keine Eingabe, so werden ausgehend von der Startnummer alle in direkt aufeinander folgender Nummerierung benannten Batch-Szenarien nacheinander simuliert.

Darüber hinaus bietet MACSIMA die komfortable Funktion *Directory Simulation*, bei der alle in einem Verzeichnis befindlichen Szenarien nacheinander simuliert werden. Diese Funktion bietet auch die Möglichkeit, nach Abschluss aller Simulationsläufe eine automatische Auswertung der Simulationsergebnisse (s. Abschnitt 7.3) anzustoßen.

6.3 Notation für die Simulationsszenarien

Die vielen möglichen einstellbaren Parameter in MACSIMA erfordern es, die unten beschriebene Notation einzuführen, um die Parametrisierung der simulierten Szenarien präzise und einfach mit einem eindeutigen Namen beschreiben zu können.

Die Komponenten des Szenario-Namens spezifizieren die wichtigsten für ein Szenario eingestellten Parameter. Diese entsprechen den Eingaben in verschiedenen Bereichen der im vorhergehenden Abschnitt 6.2 beschriebenen MACSIMA-Benutzeroberfläche.

Die angegebene Notation basiert auf den in [Walz 2007] und [Ruß 2009c] vorgestellten Grundlagen, berücksichtigt aber darüber hinausgehend zwischenzeitlich neu hinzugekommene Parametrisierungsmöglichkeiten. Die Namensbestandteile sind zumeist durch einen Bindestrich voneinander separiert.

Der Namensaufbau orientiert sich prinzipiell an der folgenden Reihenfolge, wobei einige Namensbestandteile zwingend und andere optional sind:

- Stufennummer (gegebenenfalls inklusive der Agenten-Anzahl)
- Gen-Werte
- Lernmodus mit Informationsaustausch-Einstellungen
- Pool- und Populationsgröße
- Selektionsverfahren
- Rekombinations-Verfahren (mit Parameter-Einstellungen)
- Mutation (mit Parameter-Einstellungen)
- Ersetzungsschema (mit Individuen-Anzahl)
- Fitness-Berechnungsmethode
- Pool-Anzahl

⁶⁰ Das Zeichen # muss hierbei einer natürlichen Zahl entsprechen.

⁶¹ Die Nummerierung # muss hier einer natürlichen Zahl entsprechen und fortlaufend inkrementell vergeben werden.

- Parameter für den Genotypen-Wechsel bei Verhandlungen
- Weitere optionale Einstellungen (zum Beispiel zum Startkapital der Agenten)

Optional ist in diesem Zusammenhang so zu verstehen, dass entsprechende Angaben zu Einstellungen nur dann erforderlich sind, wenn diese bei der Definition des Szenarios auch tatsächlich vorgenommen wurden. Die Abkürzungen für die Parameter sind mit ihrer jeweiligen Semantik in den Tabellen 5, 6 und 7 aufgeführt.

Abkürzung	Name	Parameter / Erläuterung
Optional: Angaben zu Agentenzahlen und Stufen		
A	Agenten-Anzahl	natürliche Zahl, die die Anzahl gleich parametrisierter Agenten auf einer Stufe angibt
S	Stufen-Nummer	natürliche Zahl, die die Stufe angibt; mit „+“ werden mehrere Stufen angegeben (z. B. S1+3) und mit „-“ zwei Zahlen, die von Stufe bis Stufe angeben (z. B. S1-3)
Optional: Angabe der Gen-Werte		
Gen()	AQ, DC, DJ, SF, WM, R	siehe Abschnitt 5.3.2
fixAll()		Alle Gene werden zum Simulationsstart auf den gleichen Wert gesetzt
randGen		Die Werte aller Gene werden zum Simulationsstart randomisiert generiert.
Nicht-optional: Verwendeter Lernmodus		
NoLearning		Lernen ist ausgeschaltet.
INT	Internes Lernen	Informationsaustausch-Modus: (G. E oder N)
EXT	Externes Lernen	Informationsaustausch-Modus: (G. E oder N)
MIX	Gemischtes Lernen	Informationsaustausch-Modus: (G. E oder N)
Smith98	STDEA-Adaptionsverfahren von Smith und Taylor	Für Benchmarking-Zwecke verwendetes Lernverfahren (siehe Abschnitt 5.4.6 beziehungsweise [Smith & Taylor1998])
Nicht-optional: Art des Informationsaustauschs zwischen den Agenten		
N	No Exchange	kein Informationsaustausch
G	InGroup	Austausch von Plumages ⁶² innerhalb der definierten Agentengruppen
E	Everybody	Austausch von Plumages mit allen Agenten des Szenarios
Nicht-optional: Pool- und Populations-Größe		
P	PoolSize	natürliche Zahl, welche die Pool-Größe angibt
Pn	Populationsgröße	natürliche Zahl (\geq Pool-Größe), welche die Größe der Population angibt

Tabelle 5: Teil 1 der Notation für die MACSIMA-Simulationsszenarien (zu Basis-Einstellungen bzw. grundlegenden Namenskomponenten)

⁶² Bei einem Plumage handelt es sich um eine Datenstruktur, die einen Genotypen und einen für diesen berechneten Fitness-Wert enthält (siehe Abschnitt 5.4.1).

In Tabelle 6 sind dabei alle nicht-optionalen Namensbestandteile aufgeführt, welche die Parametrisierung der Lernverfahren betreffen (s. Abschnitt 6.3.1).

Abkürzung	Name	Parameter / Erläuterung
Nicht-optional: Selektions-Methoden		
BiC	Binary Competition	siehe Abschnitt 5.4.3.1
RWS	Roulette Wheel Selection	siehe Abschnitt 5.4.3.1
DMS	Deterministic Minority Selection	siehe Abschnitt 5.4.3.2
DMS2	Deterministic Minority Selection 2	siehe Abschnitt 5.4.3.2
DMS3	Deterministic Minority Selection 3	siehe Abschnitt 5.4.3.2
DtS	Deterministic Selection	siehe Abschnitt 5.4.3.1
DtA	Deterministic Average	siehe Abschnitt 5.4.3.1
DwS	Delete Worst Selection	siehe Abschnitt 5.4.3.1
Nicht-optional: Rekombinations-Methoden		
nPC	n-Punkt-Crossover	(Rekombinationswahrscheinlichkeit., Anzahl der Kreuzungspunkte, x) siehe Abschnitt 5.4.3.3
RaC	Random Crossover	(Rekombinationswahrscheinlichkeit, x, PreferBetterParent) siehe Abschnitt 5.4.3.3
NoR	No Recombination	Die Rekombinationsphase ist innerhalb des Lernverfahrens ausgeschaltet.
Nicht-optional: Mutations-Methoden		
SMU	Single Mutation	(Mutationswahrscheinlichkeit, Mutations- beziehungsweise Gauß-Weite) siehe Abschnitt 5.4.3.4
GMU	General Mutation	(Mutationswahrscheinlichkeit, Mutations- beziehungsweise Gauß-Weite) siehe Abschnitt 5.4.3.4
Nicht-optional: Ersetzungsschema		
E	Elitismus (engl. Elitism)	(natürliche Zahl für die Anzahl der BestIndividualsToSurvive) siehe Abschnitt 5.4.4
WE	Schwacher Elitismus (engl. Weak Elitism)	(natürliche Zahl für die Anzahl der BestIndividualsToSurvive) siehe Abschnitt 5.4.4
Nicht-optional: Fitness-Bewertungsmethode		
AvP	Average Price Method	siehe Abschnitt 5.4.2.2
BaP	Basic Price Method	siehe Abschnitt 5.4.2.3
MdP	Medium Price Method	siehe Abschnitt 5.4.2.4
PmA	Price minus Average Method	siehe Abschnitt 5.4.2.1
RMF	Relative Medium Price Fraction Method	siehe Abschnitt 5.4.2.5

Tabelle 6: Teil 2 der Notation für die MACSIMA-Simulationsszenarien (zu den Detail-Einstellungen für die evolutionären Lern-Algorithmen)

In Tabelle 7 sind optionale Namensbestandteile aufgeführt, die in Abschnitt 6.3.2 im Detail erläutert werden.

Abkürzung	Name	Parameter / Erläuterung
Optional: Anzahl der verwendeten Gen-Pools (und Populationen)		
2Pool	2 getrennte Gen-Pools	Für Details zur Verwendung von 2 getrennten Gen-Pools für den Einkauf und Verkauf siehe Abschnitt 5.3.3.2. Diese Komponente ist optional; wird nur 1 Pool verwendet, so entfällt sie.
Optional: Feinjustierung des Informationsaustausches		
xPyF	Informationsaustausch-Quote	Ist als Informationsaustausch InGroup eingestellt, so kann durch die Spezifikation der Parameter x und y der Informationsaustausch feinjustiert werden. Das Kürzel xPyF bedeutet dabei: An x Prozent der Agenten einer Gruppe G in y Prozent der der Abschluss-Fälle ein Plumage senden. Ist das Kürzel nicht spezifiziert, so gilt die Standardeinstellung 100P100F.
Optional: Sonstige Verfahren		
HFF	Highest Fitness First	siehe Abschnitt 5.4.5.1
T	Ticket-Limitierung	siehe Abschnitt 5.4.5.2
Weitere optionale Komponenten		
IC	Initial Cash	Anfangsausstattung eines Agenten mit Geldeinheiten
G	gegen	Abgrenzung von 2 oder mehr heterogenen Agenten-Parametrisierungen

Tabelle 7: Teil 3 der Notation für die MACSIMA-Simulationsszenarien (zu weiteren optionalen Einstellungen bzw. Namenskomponenten)

6.3.1 Nicht-optionale Notationselemente

Im Folgenden wird die Semantik der vorgestellten Notation anhand des Szenario-Namens

MIX(G)-P3-Pn40-RWS-nPC(0,5-1-x)-GMU(0,5-0,01)-WE(1)-MdP-2Pool

im Detail erläutert. Diese Beispiel-Benennung enthält dabei zunächst nur nicht-optionale Namensbestandteile.

Spezifikation von Lernmodus und Informationsaustausch: MIX(G)

Die erste zwingend erforderliche Namenskomponente einer Szenario-Benennung spezifiziert den Lernmodus, den die Agenten verwenden. Ist *Int* angegeben, so bedeutet dies, dass autonom und „intern“ gelernt wird – weitere einstellbare Lern-Modi sind *externes Lernen (EXT)* sowie *gemischtes Lernen (MIX)*.

Auf welche Form des Informationsaustauschs zwischen den Agenten sich diese Angabe bezieht, spezifiziert die nächste Namenskomponente, die sich in einer runden Klammer an die Angabe des Lernmodus anschließt.

Hier kann der Informationsaustausch ganz ausgeschaltet (das heißt auf *NoExchange (N)* eingestellt) werden oder mit allen Agenten, also *Everybody (E)*, erfolgen. Darüber hinaus kann

der Austausch von Plumages (siehe Abschnitt 5.4.1) aber auch auf den Austausch innerhalb einer *Gruppe (G)* beschränkt sein.

Im Beispiel bedeutet $MIX(G)$, dass alle Agenten innerhalb der Gruppe G *gemischtes Lernen* nutzen. Dies bedeutet, dass ein Agent nicht nur alle durch seine eigenen Verhandlungen intern erzeugten Plumages in seine Population aufnimmt, sondern auch die extern von anderen Agenten erzeugten und an ihn gesendeten Plumages.

Letztere stellen codierte Erfahrungswerte aus Verhandlungen dar, an welchen der Agent selbst nicht teilgenommen hat. Externe Plumages (siehe Abschnitt 5.4.1.3) werden ihm von „befreundeten“ Agenten zugeschickt, die sich mit ihm in der gleichen Agenten-Gruppe befinden.

Die Zuordnung von Agenten zu Agentengruppen wird in der MACSIMA-Benutzeroberfläche definiert. Eine Agenten-Gruppe kann l bis n Agenten einer Stufe und zusätzlich ggf. auch Agenten benachbarter Stufen enthalten. Hat ein Gruppenmitglied ein internes Plumage generiert, so sendet er dieses an mehrere Mitglieder seiner Gruppe, wobei der Sendeprozess an die einzelnen Gruppenmitglieder jeweils mit einer in MACSIMA definierbaren Wahrscheinlichkeit erfolgt, die standardmäßig auf 50% eingestellt ist, aber auch verändert werden kann.

Spezifikation der Pool- und Populations-Größe: $P3-Pn40$

Nach der Spezifikation des Lernmodus und der Art des Informationsaustauschs erfolgen in der Notation Angaben zu den Größen des vom Lernverfahren verwendeten Gen-Pools⁶³ P sowie der Genotypen-Population Pn , jeweils definiert durch eine natürliche Zahl.

Der Gen-Pool enthält in der oben genannten Szenario-Beispielbenennung drei Genotypen. Und die Population kann im Beispiel vierzig Plumages aufnehmen, bis der Schwellenwert für den Start des Lernverfahrens (siehe Abschnitt 5.4.1) erreicht ist.

Spezifikation der Selektions-Methode: RWS

Die darauffolgende Namenskomponente spezifiziert das beim Lernen verwendete Selektionsverfahren. MACSIMA stellt dem Szenario-Designer dabei eine Vielzahl an Verfahren (vergleiche die Abschnitte 5.4.3.1 und 5.4.3.2) zur Verfügung, nämlich die

- *Binary Competition (BiC)*,
- *Roulette Wheel Selection (RWS)*,
- *Deterministic Selection (DtS)*,
- *Deterministic Average Selection (DtA)*,
- *Delete Worst Selection (DwS)* sowie
- verschiedene Verfahren der *Deterministic Minority Selection (DMS, DMS2, DMS3)*

Spezifikation der Rekombinations-Methode: $nPC(0,5-1-x)$

Als nächstes erfolgt die Angabe, welche Form der Rekombination der Evolutionäre Lernalgorithmus verwendet. Dem Szenario-Designer steht hierbei die Wahlmöglichkeit zwischen

- *n-Punkt-Crossover (nPC)*,
- *Random-Crossover (RaC)* und

⁶³ Ein Gen-Pool besteht aus einer Menge von Genotypen, die aus Genen aufgebaut sind (vgl. Abschnitt 5.4.1).

- dem Ausschalten der Rekombination (*NoRecombination*)

offen. Für die Rekombinations-Methoden *nPC* und *RaC* erfolgt dabei in runden Klammern die detaillierte Spezifikation ihrer Parametereinstellungen, separiert durch einen Bindestrich.

In den *nPC*-Parametereinstellungen erfolgt an der ersten Position die Angabe der Wahrscheinlichkeit dafür, dass zwei Genotypen miteinander gekreuzt werden, gefolgt von der Anzahl der Schnitte, die zur Kreuzung in den Genotypen durchgeführt werden an der zweiten Position. An der dritten Position steht das Zeichen „x“, welches als eine Art „syntaktischer Zucker“ Verwendung findet, um die Notation für die Parametereinstellungen der Rekombinations-Methoden *nPC* und *RaC* einerseits zu vereinheitlichen und andererseits die Unterschiede zwischen den Methoden zu verdeutlichen. Im speziellen *nPC*-Fall verdeutlicht das „x“, dass *nPC* keine Einstellung eines dritten Parameters erfordert.

Bei der Random-Crossover-Methode ist z. B. im Unterschied zu den *nPC*-Einstellungen das Zeichen „x“ standardmäßig an der zweiten Position der Parametereinstellungen zu finden, da bei dieser Methode im Gegensatz zur *nPC*-Methode keine Schnittanzahl angegeben werden muss.

Zusammenfassend gibt in unserem Beispiel der Namensabschnitt *-RWS-nPC(0,5-1-x)-* konkret an, dass die Parameter für das Lernverfahren der Agenten so gesetzt sind, dass eine Selektion von Genotypen mittels des *Roulette-Wheel-Selection*-Verfahrens erfolgt, an dass sich eine Rekombinationsphase mittels *n-Point-Crossover* anschließt. Dabei wird mit einer Rekombinationswahrscheinlichkeit von 0,5 bzw. 50% an genau einer Schnittstelle der Genotypen ein Crossover durchgeführt.

Wäre im Beispiel anstatt eines *n-Point-Crossovers* mit den Einstellungen *-nPC(0,5-1-x)-* ein *Random Crossover* mit der Parametrisierung *-RaC(0,5-x-0,6)-* eingestellt worden, so würde das Crossover folgendermaßen ablaufen:

Für jeden der 6 Gen-Werte der beiden für ein Crossover ausgewählten Genotypen würde unabhängig voneinander anhand des ersten Parameters – der Rekombinationswahrscheinlichkeit 0,5 bzw. 50% – jeweils entschieden werden, ob eine Überschreibung des aktuellen Gen-Werts mit dem entsprechenden Wert des jeweils anderen Genotypen überhaupt erfolgt.

Der dritte Parameter – hier 0,6 bzw. 60% – würde dann die Wahrscheinlichkeit dafür spezifizieren, dass der Eltern-Genotyp mit der schlechteren Fitnessbewertung den Gen-Wert vom Eltern-Genotyp mit der besseren Fitnessbewertung übernimmt.

Bei dieser Vorgehensweise erfolgt keine klassische Kreuzung der Eltern-Genotypen, so dass an der zweiten Stelle des Tupels keine Angabe zu der Anzahl der Kreuzungspunkte erfolgen muss.

Spezifikation der Mutations-Methode: GMU(0,5-0,01)

Der dritte genetische Operator eines evolutionären Algorithmus ist die Mutation. In MACSIMA sind die Verfahren

- *Single Mutation (SMU)* und
- *General Mutation (GMU)*

implementiert. In runden Klammern erfolgt für *SMU* und *GMU* die Angabe der beiden Parameter des jeweiligen Verfahrens. In beiden Fällen definiert dabei der erste Parameter, mit welcher Wahrscheinlichkeit mutiert wird. Und der zweite Parameter gibt die Gauß-Weite an, d. h. ein Maß dafür, wie weit sich der Wert eines mutierten Parameters vom Ausgangswert unterscheiden darf.

Im Beispiel-Szenario wird die Mutation mittels der Methode *GMU* durchgeführt. Dabei erfolgt eine Mutation mit einer Wahrscheinlichkeit von 50%, wobei die Gaußweite auf 0,01, d. h. 1% eingestellt ist.

Spezifikation des Ersetzungsschemas: WE(1)

Anhand der nächsten Namenskomponente lässt sich erkennen, welches Ersetzungsschema (vergleiche Abschnitt 5.4.4) am Ende des evolutionären Algorithmus Verwendung findet. MACSIMA stellt die Varianten

- *Weak Elitism (WE)* und
- *Elitism (E)*

zur Verfügung. Eine in runden Klammern angegebene natürliche Zahl legt die Anzahl der besten Individuen fest, die in die neue Generation übernommen werden sollen. Die Ersetzung erfolgt im Beispiel-Szenario mittels der Methode *Weak Elitism*, wobei die 1 bei *WE(1)*- angibt, dass genau ein Genotyp in die nachfolgende Generation übernommen wird.

Spezifikation der Fitness-Berechnungsmethode: WE(1)

Der letzte nicht-optionale Namensbestandteil besteht in der Angabe der benutzten Fitness-Berechnungsmethode. Dabei stehen die Methoden

- *Average Price (AvP)*,
- *Basic Price (BaP)*,
- *Medium Price (MdP)*,
- *Price Minus Average (PmA)* und
- *Relative Medium Price Fraction (RMF)*

zur Auswahl. Die Fitness eines Plumages wird im Beispiel anhand der Methode *MediumPriceBased* berechnet, die im Abschnitt 5.4.2.4 im Detail beschrieben wird.

6.3.2 Optionale Notationselemente

In diesem Abschnitt werden die optionalen Elemente der oben vorgestellten Notation aufgelistet. Entsprechende Angaben werden nur bei Simulationen verwendet, die dies erfordern, da entsprechende Einstellungen vorgenommen wurden.

So könnte der oben beschriebene Szenario-Beispielname um ein Präfix mit Angaben zu Stufen und der Anzahl gleich parametrisierter Agenten auf diesen Stufen ergänzt werden.

Ein Präfix der Form *A5S2+S3-* würde bedeuten, dass 5 Agenten auf der Stufe 2 sowie alle Agenten auf der Stufe 3 des Szenarios die in den nachfolgenden Namenskomponenten angegebenen Parametereinstellungen aufweisen, nämlich *MIX(G)-P3-Pn40-RWS-nPC(0,5-1-x)-GMU(0,5-0,01)-WE(1)-MdP-2Pool*.

Dieses Präfix kann durch die Angabe der initialen Gen-Werte, mit denen diese Agenten auf den verschiedenen Stufen am Simulationsbeginn gestartet werden, ergänzt werden. Eine Komponente *-(0,2-0,5-0,5-0,5-0,5-0,5)* würde bedeuten, dass der Initialwert des Gens *Acquisitiveness (AQ)* 0,5 beträgt und alle anderen Gene mit dem Startwert 0,5 initialisiert werden.

Neben diesen Präfixen gibt es für die Erweiterung einer Szenario-Definition bzw. ihrer Benennung auch optionale Suffixe, welche die Anzahl der verwendeten Gen-Pools (bzw. Populationen) und die Verwendung spezieller Mechanismen zur Performanzsteigerung bzgl. des Strategie- bzw. Genotypen-Wechsels bei Verhandlungen spezifizieren.

Was die Anzahl der verwendeten Gen-Pools (getrennt nach dem Einsatzzweck, das heißt Einkauf und Verkauf) angeht, so ist standardmäßig immer mindestens ein Pool aktiviert, so dass im 1-Pool-Fall die Hinzunahme einer entsprechenden Namenskomponente nicht erforderlich ist. Sind jedoch 2 Gen-Pools aktiviert, so wird dies durch das Suffix *-2Pool* deutlich gemacht.

Was die Verwendung von Mechanismen zur Performanzsteigerung angeht, so wird, falls das *Highest-Fitness-First-Verfahren* (vergleiche Abschnitt 5.4.5.1) eingestellt sein sollte, die Szenario-Bezeichnung um das Suffix *-HFF* ergänzt. Ebenso wird, falls die Verwendung von Tickets eingestellt beziehungsweise das Verfahren *Ticket-Limitierung (T)* (vergleiche Abschnitt 5.4.5.2) aktiviert sein sollte, die Szenario-Benennung durch ein nachfolgendes Kürzel *-Tn* komplettiert, wobei das *n* in diesem Fall einer natürlichen Zahl entspricht, welche die Ticketanzahl spezifiziert.

Abschließend können, falls erforderlich, weitere optionale Parameter als Suffix ergänzt werden, z. B. Angaben über die initiale Ausstattung der Agenten mit Geld, d. h. ihren *Initial Cash (IC)*. Das Suffix *-IC(500T)* würde dabei bedeuten, dass die Agenten in einer Simulation mit einem Geldbestand von 500.000 *Geldeinheiten (GE)* starten.

Sollten die Agenten des Szenarios nicht homogen, sondern heterogen parametrisiert sein, d. h. z. B. auf Stufe 2 ein bestimmtes Lernverfahren verwenden und auf Stufe 3 ein anderes, so würden die Beschreibungen dieser beiden unterschiedlichen Lern-Parametrisierungen durch die Bezeichnung *-gegen-* voneinander abgegrenzt werden.

6.4 Protokollierung von Simulationsläufen

6.4.1 Generierung der Protokoll-Dateien

Die Agenten in MACSIMA protokollieren ihren inneren Zustand sowie ihre Aktivitäten, Verhandlungsschritte und Endresultate in mehreren getrennten, statistisch auswertbaren Protokoll-Dateien. Diese Simulations-Rohdaten werden während der Simulationszeit in das Verzeichnis geschrieben, in dem die Simulations-Szenarien liegen bzw. gestartet wurden und umfassen u. a. Informationen über die Zustände von Agenten, die Evolution ihrer individuellen Verhandlungsstrategien sowie den Verlauf und die Ergebnisse ihrer Verhandlungen.

Die erzeugten Simulations-Rohdaten lassen sich anschließend mit einer ganzen Reihe von komplexen Perl- und Shell-Skripten quantitativ auswerten. Dabei werden sie zu Kennzahlen für die einzelnen Agenten als auch für Agentengruppen und alle Stufen des MACSIMA-Wertschöpfungsnetzwerks verdichtet, wie in Kapitel 7 im Detail dargelegt wird.

In Abhängigkeit von den Einstellungen in der Textdatei *stat.properties* lassen sich für MACSIMA bis zu sechs verschiedene Protokoll-Dateien aktivieren bzw. deaktivieren, die verschiedene Aspekte eines Simulationslaufs dokumentieren.

Dabei ist die Datei *stat.properties* gemeinsam mit der Szenario-Datei, welche die Bezeichnung *Szenario-Name.sce* trägt, im gleichen Verzeichnis abzulegen, in dem eine Simulation gestartet wird. Der Szenario-Name wird gemäß der oben in Abschnitt 6.3 eingeführten Notation gebildet.

Je nach den in `stat.properties` gesetzten Flags generiert MACSIMA bis zu sechs verschiedene Protokoll-Dateien mit der Namensform:

Szenario-Name_proposalLineLaufnummer.csv
 Szenario-Name_salesLaufnummer.csv
 Szenario-Name_populationLaufnummer.csv
 Szenario-Name_genotypesLaufnummer.csv
 Szenario-Name_priceDistributionLaufnummer.csv
 Szenario-Name_obituaryLaufnummer.csv

Der Namens-Präfix einer Protokoll-Datei besteht dabei aus dem Szenario-Namen, gefolgt von einer Bezeichnung für Art und Inhalt der Protokoll-Datei sowie danach aus einer Laufnummer, die die Nummer des Simulationslaufs angibt, bei dem die Datei erzeugt wurde – bei 4 Simulationsläufen, die für ein Szenario wiederholt durchgeführt werden, kann sie die Werte 1 bis 4 annehmen.

In Abhängigkeit von den aktivierten Protokoll-Dateien fallen für die vier pro Szenario durchgeführten Simulationsläufe Protokoll-Daten in unterschiedlichem Umfang an. In der Regel betragen diese bei den Standardeinstellungen, bei denen die Generierung der `proposalLine`-, `sales`-, `genotypes`- und `obituary`-Protokoll-Dateien aktiviert ist, pro Szenario bereits mehrere Gigabyte.

6.4.2 Format und Verarbeitbarkeit der Protokoll-Dateien

Die Protokoll-Dateien weisen alle das `.csv`-Dateiformat⁶⁴ auf, bei dem die protokollierten, einfach strukturierten Simulationsdaten – durch Kommata (oder ein anderes definiertes Zeichen) voneinander separiert – gespeichert werden. Dieses Dateiformat eignet sich einerseits sehr gut für die Generierung von Tabellen und Listen unterschiedlicher Länge. Andererseits ist dieses Format so einfach gehalten, dass entsprechende `csv`-Datensätze recht universell weiterverarbeitet und/oder importiert werden können, z. B. durch

- Perl-Skripte (für statistische Auswertungen und die Verdichtung der Daten) und
- Plot-Programme wie GNU-Plot (für die graphische Analyse von Simulationsläufen als Grundlage einer qualitativen Evaluation) oder
- unterschiedlichste Anwendungsprogramme, wie z.B. aus dem Bereich Tabellenkalkulation (zum Beispiel Microsoft Excel und Libre Office Calc).

6.4.3 Inhalte der Protokoll-Dateien

Die `proposalLine`-Dateien enthalten die Verläufe sämtlicher während eines Simulationslaufs stattgefundenen Verhandlungen zwischen 2 Agenten – d.h. alle abgegebenen Gebote und Gegengebote sowie die Information, ob diese angenommen oder abgelehnt wurden. Hieraus lässt sich beispielweise ermitteln, weshalb Verhandlungen gescheitert sind, wie das Verhältnis von begonnenen zu erfolgreich abgeschlossenen oder abgebrochenen Verhandlungen aussieht usw. Sie stellen die größten Protokoll-Dateien dar.

In den `sales`-Dateien werden die erfolgreichen Verhandlungsabschlüsse (sogenannte *Deals*) protokolliert. Jeder Deal wird während einer laufenden Simulation genau einmal dokumentiert – nämlich von dem Verkäufer des Gutes. In einer `sales`-Datei sind neben den Deals mit den erzielten Preisen auch weitere Informationen gespeichert, unter anderem auch über die

⁶⁴ `csv` ist die Abkürzung für „comma-separated values“.

Genotypen, die die Agenten bei der Verhandlung verwendet haben und die Anzahl der Verhandlungsrunden, bei denen die Genotypen bzw. die in ihnen codierten Verhandlungsstrategien eingesetzt wurden. Daher lassen sich aus *sales*-Dateien mittels verschiedener Auswertungsskripte viele unterschiedliche Kennzahlen berechnen.

In den *population*-Dateien werden zum einen alle Plumages⁶⁵ der Population eines Agenten vor der Selektionsphase seines evolutionären Algorithmus erfasst und zum anderen die in der Folge selektierten Plumages nach der Selektion (bzw. vor der Rekombinationsphase). Zusammen mit dem Agentennamen und den Gen-Werten sowie der Fitness des Plumages werden auch die aktuelle Runde, Preisinformationen sowie die Anzahl der Runden, in denen ein Genotyp von dem Agenten verwendet wurde, protokolliert.

In den *genotypes*-Protokoll-Dateien werden alle Genotypen jedes Agenten erfasst, die sich zu einem Zeitpunkt bzw. einer bestimmten Simulationsrunde in seinem Gen-Pool befinden – ergänzt um weitere Angaben, unter anderem zum Cash-Bestand des Agenten. Dabei wird protokolliert, mit welchem Gen-Pool an Genotypen ein Agent startet und welche der Genotypen darin im Verlauf des Simulationslaufs durch seinen EA und die damit induzierte Adaption durch andere Genotypen ersetzt werden.

Anhand der *priceDistribution*-Dateien lassen sich die Preisentwicklungen für die im System gehandelten Güter nachvollziehen – sowohl für den einzelnen Agenten als auch für die Agenten-Gesellschaft. Hierzu werden für jeden Agenten grundlegende Daten wie der Agentenname, die aktuelle Runde, das gehandelte Gut und der erzielte Deal-Preis protokolliert, aber auch darüber hinausgehende Daten wie seine individuellen Preisschranken und statistische Daten wie z. B. der bei der letzten vorhergehenden Verhandlung erzielte Preis oder der vom Agenten über eine Reihe von Deals hinweg erzielte gewichtete Durchschnittspreis.

Sollte ein Agent insolvent werden, so wird in der *obituary*-Datei eines Simulationslaufs protokolliert, in welcher Runde dieser Agenten-Zustand eingetreten ist. Insolvente Agenten werden vom Marktplatz entfernt.

⁶⁵ Vergleiche Abschnitt 5.4.1.

7 Evaluationsmethodik

7.1 Einleitung und Übersicht

Im Folgenden wird die Evaluationsmethodik von MACSIMA vorgestellt, mit deren Hilfe die quantitative und qualitative Evaluation der in Abschnitt 6.4 beschriebenen Simulations-Rohdaten von MACSIMA durchgeführt und die in Kapitel 8 vorgestellten Evaluationsergebnisse erzielt werden. Mit ihrer Hilfe lassen sich vielfältige Hypothesen und Fragestellungen auf verschiedenen Systemebenen formulieren und untersuchen.

Hierzu werden die Simulations-Rohdaten

- mit leistungsfähigen Skripten weiterverarbeitet und zu aussagekräftigen quantitativen Kennzahlen verdichtet,
- in Ergebnistabellen überführt, die für quantitative Analysen auf drei Evaluationsebenen effizient nutzbar sind, sowie
- zur Schaffung zusätzlicher qualitativer Analysemöglichkeiten auf geeignete Weise visualisiert, da bestimmte Evaluationsfragestellungen mit einem rein quantitativen Analyseansatz nur schwer beleuchtet werden können.

Bei der Betrachtung der drei Evaluationsebenen sind unterschiedliche Aspekte von Relevanz. Um diese quantitativ beleuchten zu können, bedarf es der Erzeugung geeigneter Kennzahlen sowie ihrer ebenenbezogenen Zuordnung.

Die quantitative Auswertung des Rohdaten-Bestandes erfolgt dabei mit Perl- und Shell-Skripten, welche die Rohdaten zu Kennzahlen verdichten. Deren Berechnung erfolgt dabei entweder aus

- den über einen einzelnen Simulationslauf anfallenden Daten oder
- den über alle Simulationsläufe hinweg anfallenden Daten bzw. aggregierten laufbezogenen Kennzahlen für die einzelnen Läufe.

Einen Überblick über die verwendeten betriebswirtschaftlichen Kennzahlen und Maße der deskriptiven Statistik gibt Abschnitt 7.3.

Manche Kennzahlen bzw. Ergebnisvariablen, wie beispielsweise Umsatz und Gewinn sowie die Anzahl erfolgreicher oder abgebrochener Verhandlungen, sind sowohl für die einzelnen Agenten, als auch für Agentengruppen, die Stufen des MACSIMA-Wertschöpfungsnetzwerks sowie das Gesamtsystem interessant und aussagekräftig.

Andere, wie beispielsweise Kennzahlen bzgl. der Gewinnverteilung, spielen nur auf bestimmten, z. B. auf Gruppen fokussierenden Evaluationsebenen eine Rolle.

Die Evaluationsebenen bzw. -perspektiven gliedern sich daher in die Betrachtung von

- einzelnen Agenten und deren Interaktionsmöglichkeiten (*Mikro-Ebene*) sowie
- von Agentengruppen (*Meso-Ebene*) und
- von allen Agenten eines Wertschöpfungsnetzwerks (*Makro-Ebene*).

Die Herleitung, Definition und Semantik dieser verschiedenen Ebenen der Systemanalyse fußen auf Forschungsergebnissen aus dem Gebiet der *Sozionik* und werden in Abschnitt 7.2 erläutert.

Die Zuordnung der mit den Skripten erzeugten Ergebniskennzahlen zu Ebenen mündet in die Definition von sieben komplexen, aber dennoch sehr übersichtlich strukturierten Ergebnistabellen.

Diese sind quantitativ miteinander vergleichbar und eignen sich gut, um das sich aus den Szenario-Parametrisierungen ergebende Systemverhalten und damit verbundene Phänomene auf der Makro-, Meso- und Mikro-Ebene der Agenten-Gesellschaft auf effiziente Weise quantitativ zu untersuchen. Dies ist insbesondere bei der Analyse eines großen Bestands an Simulationsdaten hilfreich, wie er beispielsweise im Evaluationsteil dieser Arbeit in Kapitel 8 auftritt.

Die Struktur der Ergebnistabellen, ihre Zuordnung zu den drei definierten Analyse- bzw. Evaluationsebenen sowie die Generierung der in ihnen enthaltenen Kennzahlen und statistischen Lage- und Streuungsmaße werden in Abschnitt 7.3 dargestellt und detailliert erläutert.

Darüber hinausgehend muss jedoch auch berücksichtigt werden, dass sich aufgrund des inhärenten Nicht-Determinismus in jedem Simulationslauf jeweils ein emergentes Systemverhalten entwickelt.

Je nach Parametrisierung des zugrundeliegenden Simulations-Szenarios führt dies innerhalb des Simulationszeitraums zu unterschiedlichen Entwicklungen, z. B. was die Entwicklung der Güterpreise angeht, und damit zu mehr oder weniger stark variierenden Ergebnisvariablen. Zur detaillierten Analyse der auftretenden Streuungseffekte bedarf es zusätzlicher Evaluationsinstrumente.

In Kapitel 7.4 wird daher die für die quantitative Evaluation zur Verfügung stehende Methodik um Werkzeuge für die qualitative Evaluation von Simulationsergebnissen ergänzt. Es werden Möglichkeiten zur Visualisierung von innerhalb von Simulationsläufen auftretenden Dynamiken beschrieben, die sich quantitativ nur schwer fassen bzw. beschreiben lassen.

Die für die quantitative Evaluation zur Verfügung stehende Methodik mit ihren Werkzeugen und Strukturen eignet sich aufgrund ihrer kompakten, verdichteten Darstellung von Kennzahlen und Maßen ideal für ein schnelles Benchmarking bzw. die Einordnung und Gruppierung von großen Mengen an Szenarien und ihrer Simulationsergebnisse anhand des Vergleichs ihrer wesentlichen quantitativen Charakteristika.

Die für eine qualitative Evaluation zur Verfügung stehenden Werkzeuge sind dazu eine wertvolle Ergänzung, da sie weitergehende Fragestellungen untersuchbar machen, die einer rein quantitativ-basierten Evaluation nicht zugänglich sind. Dies betrifft insbesondere auch die Adaptions- und Lernprozesse und die davon abhängigen emergierenden dynamischen Marktentwicklungen.

Sowohl die quantitativ-basierte Herangehensweise als auch die qualitativ-basierte besitzen Vorzüge, aber auch Einschränkungen, so dass umfassende Analysen interessanter Phänomene auf Makro-, Meso- und Mikro-Ebene der Agenten-Gesellschaft häufig nur durch die wechselseitige Nutzung beider erfolgreich durchführbar sind.

Häufig befruchten sich die beiden Sichten gegenseitig, indem die Betrachtung aus der einen Sicht zu Hypothesen führt, die durch die Nutzung der Möglichkeiten der anderen erhärtet oder verworfen werden kann.

7.2 Evaluationsebenen

7.2.1 Untersuchungsperspektiven aus Sicht der Sozionik

Um ein Benchmarking unterschiedlicher Szenario-Parametrisierungen durchführen zu können, bedarf es einer vergleichenden Analyse der von ihnen induzierten ökonomischen Effekte auf verschiedenen Ebenen des in MACSIMA realisierten Wertschöpfungsnetzwerks.

[Schillo 2004] hat sich am DFKI in Saarbrücken im Rahmen des Schwerpunktprogrammes *Sozionik - Erforschung und Modellierung künstlicher Sozialität* der Deutschen Forschungsgemeinschaft (DFG) mit einem Wissenstransfer zwischen den Forschungsdisziplinen der *Verteilten Künstlichen Intelligenz (VKI)* und der *Soziologie* beschäftigt.

Sozionik ist, wie [Malsch 1999] schreibt, ein „*Forschungsfeld zwischen Soziologie und Künstlicher Intelligenz (KI) ...*“, in dem es um die Frage geht, „... *wie es möglich ist, gesellschaftliche Vorbilder und soziologische Konzepte aufzugreifen, um daraus Modelle künstlicher Sozialität und intelligente Computertechnologien zu entwickeln.*“ Bei den hierbei behandelten Forschungsfragen geht es u. a. um

- das „*Potential der ‚Multiagentensysteme‘ (MAS) der VKI für die Simulation gesellschaftlicher Zusammenhänge*“,
- die „*Frage nach der Exploitierbarkeit soziologischer Theorien für den Bau von dynamischen und skalierbaren Multiagentensystemen.*“ sowie
- „...*die Chancen und Risiken von ‚Artificial Societies‘⁶⁶ im Internet.*“

Weitere Erläuterungen zum Forschungsgebiet der Sozionik finden sich u. a. in [Fischer et al. 2005; Malsch 2001; Malsch 2002].

Vor diesem Hintergrund lag der Fokus von [Schillo 2004] insbesondere auf der Anwendung von Konzepten der Soziologie zur Erhöhung der Robustheit von Multiagenten-Systemen und der Analyse ihrer Koordinationsstrukturen.

In diesem Zusammenhang wurde auch das zentrale Mikro-Makro-Problem der Verteilten Künstlichen Intelligenz [Weiss 1999] beleuchtet, bei dem es kurzgefasst um das zielgerichtete Design von Multiagenten-Systemen unter Berücksichtigung der Mikro-Ebene (Agenten-Perspektive) und der Makro-Ebene (System-Perspektive) geht.

Ein von der Mikro-Ebene ausgehender Entwurfsprozess erfolgt „bottom-up“ und legt seinen Fokus auf die Eigenschaften bzw. die Architektur der Agenten sowie ihre Interaktionsfähigkeiten. Der Nachteil solch eines agentenzentrierten Design-Ansatzes besteht darin, dass das resultierende globale Koordinationsergebnis des Gesamtsystems bei einer großen Anzahl von Agenten in den meisten Fällen kaum mehr vorhersehbar ist.

Im Gegensatz dazu legt ein von der Makro-Ebene ausgehender Entwurfsprozess seinen Fokus auf die Eigenschaften und die Leistung des Gesamtsystems und versucht, dessen Agentenpopulation so mit Agenten auszustatten bzw. zu „befüllen“, dass ein gewünschtes Systemverhalten resultiert (s. a. Kapitel 3.3).

Die *Soziologie* [Esser 1993] beschäftigt sich als Wissenschaft mit der empirischen und theoretischen Erforschung des sozialen Verhaltens innerhalb von aus Menschen bestehenden

⁶⁶ „...kommenden Agentengesellschaften aus künstlichen ‚Akteuren‘ und menschlichen ‚Agenten‘...“ [Malsch 1999]

Gesellschaften⁶⁷. Gemäß ihrer Untersuchungsgegenstände wird sie klassischerweise in die Forschungsgebiete

- *Makro-Soziologie* (Gesellschaftsanalyse: Gesellschaft, Kollektiv, Struktur, System, etc.) und
- *Mikro-Soziologie* (Akteuranalyse: Akteur, Individuum, soziales Handeln, etc.)

unterteilt. Hinzu kam die

- *Meso-Soziologie* (Analyse intermediärer Ebenen und Zusammenhänge: Gruppe, Organisation, Subsysteme, etc.),

um eine Verbindung zwischen der Mikro- und Makro-Soziologie zu schaffen und Unzulänglichkeiten einer einseitigen Betrachtung von Fragestellungen aus Mikro- oder Makro-Perspektive zu überwinden (sogenannter *Mikro-Makro-Link*, s. dazu [Alexander 1987]) und sowohl Rückkopplungseffekte zwischen diesen Ebenen als auch dazwischenliegende intermediäre Ebenen und Zusammenhänge zu untersuchen.

Nach Coleman's *Makro-Mikro-Makro-Modell* [Coleman 1991; Greve et al. 2009] wirken sich gesellschaftliche Strukturen und Phänomene auf der Makro-Ebene auf das Verhalten der Akteure innerhalb der Gesellschaft (Individual- bzw. Mikro-Ebene) aus, die davon ausgehend anhand ihrer Ressourcen, Fähigkeiten und Handlungsoptionen Entscheidungen treffen und Handlungen vornehmen. In ihrer Summe wirken diese individuellen Handlungen dann wieder zurück auf die Gesellschaft bzw. aggregieren sich zu neuen Makro-Phänomenen. Die Analyse von Systemübergängen soll daher ausgehend von einer Makro-Ebene zu einer Mikro-Ebene und über die Meso-Ebene zurück zu einer Makro-Sicht erfolgen.

Anhand des Vergleichs des Mikro-Makro-Problems der VKI mit dem konzeptionell ähnlichen Mikro-Makro-Link-Problems der Soziologie erweitert [Schillo 2004] unter Anwendung von Erkenntnissen der Meso-Soziologie die von der VKI beim Systementwurf zu berücksichtigenden Perspektiven um eine „intermediäre“ Meso-Ebene, die sich mit der Bildung bzw. Entstehung von Gruppen und Organisationen beschäftigt und als Bindeglied zwischen Mikro- und Makro-Ebene fungiert. Als solche beeinflusst sie das Systemverhalten in wesentlichem Maße.

Ausgehend von diesen Erkenntnissen wurden von [Schillo 2004] in der MAS-Forschungsgruppe am DFKI für sieben verschiedene agentenbasierte Organisationsformen - darunter auch die Organisationsform „Markt“ - simulationsbasiert empirische Untersuchungen zur Bildung organisationaler Netzwerke in der Wertschöpfungsnetzwerk-Domäne durchgeführt. Die untersuchte Organisationsform „Markt“ war dabei (soziologisch adäquat) im Wesentlichen dadurch gekennzeichnet, dass die Interaktionen zwischen den autonomen Agenten kurzfristiger Art sowie auf den ökonomischen Austausch fokussiert waren und sich ad hoc auf Basis der individuellen Präferenzstrukturen der Marktteilnehmer ergeben haben. Ebenso war es den Agenten möglich, in den Markt einzutreten oder ihn zu verlassen.

Aufgrund der in weiten Bereichen gegebenen Analogien zu MACSIMA (vgl. hierzu auch die Kapitel 3.1.2 sowie 3.3.3) erscheint es sinnvoll, bei der Analyse der Simulationsergebnisse von MACSIMA neben der Mikro- und Makro-Ebene auch eine Analyse auf der Meso-Ebene mit einzubeziehen⁶⁸.

⁶⁷ Vgl. auch <http://de.wikipedia.org/wiki/Soziologie>

⁶⁸ In diesem Zusammenhang sind auch andere Veröffentlichungen aus der Sozionik zum Thema des kollektiven organisationalen Lernens [Florian 2004] und der Selbstorganisation und Evolution in Multiagentensystemen [Klügl 2004] von Relevanz.

Dabei beschränken sich die in dieser Arbeit für die Meso-Ebene durchgeführten Untersuchungen (s. Kapitel 7) häufig auf Gruppen autonomer, einander wohlgesonnener (benevolenter) Agenten, die auf dem Markt miteinander kooperieren, indem sie (in Form von Plumages) Informationen bzgl. verwendeter Verhandlungsstrategien und damit erzielter Erfolge untereinander austauschen – was die typischen Gegebenheiten auf den meisten Märkten widerspiegelt.

Weitergehende komplexe, ggf. sogar rekursive Verschmelzungs-, Transformations- und Zerfallsprozesse zwischen autonomen und teilautonomen Agenten, die auf einem Markt unter Aufgabe eines Teiles ihrer Autonomie holonische agentenbasierte Organisationsformen⁶⁹ bilden, sind hingegen nicht Untersuchungsgegenstand dieser Arbeit. Beiträge und Verweise zu holonischen Multiagenten-Systemen in der Supply-Web-Domäne finden sich jedoch in den Kapiteln 2.7 und 4.2 sowie in [Schillo2004], [Bürckert et al.1998], [Fischer et al. 2003] und [Ruß 2001c].

Zusammenfassend erscheint es sinnvoll, eine adäquate simulationsbasierte empirische Untersuchung des Verhaltens von marktbasierenden Agenten-Gesellschaften in einer Wertschöpfungsnetzwerk-Domäne auf drei Untersuchungsebenen, nämlich einer Mikro-, einer Meso- und einer Makro-Ebene durchzuführen, die im Folgenden genauer charakterisiert werden.

7.2.2 Makro-, Meso- und Mikro-Ebene

Die Auswertung und Analyse von Simulationsergebnissen erfolgt im Rahmen dieser Arbeit grundsätzlich auf drei Untersuchungsebenen. Diese beleuchten gemäß der Ausführungen im vorhergehenden Abschnitt 7.2.1 aus unterschiedlichen Betrachtungsperspektiven die verschiedenen sozialen Ebenen der Agenten-Gesellschaft sowie ihre Zusammenhänge und weisen daher einen unterschiedlichen Fokus und Aggregationsgrad auf. Im Folgenden wird ihre Abgrenzung voneinander ausführlich erläutert:

1. **Makro-Ebene:** Auf der Makro-Ebene werden die das Gesamtsystem betreffenden Phänomene untersucht. Das Benchmarking unterschiedlicher Szenario-Parametrisierungen erfolgt hier auf der Basis von ökonomischen Kennzahlen, die für die Gesamtperformanz des marktlich organisierten Agenten-Netzwerks von Relevanz sind und daher ausgehend von den erzeugten Simulations-Rohdaten den höchsten Aggregationsgrad aufweisen. Dieser Ebene werden daher Kennzahlen wie beispielsweise
 - der Gesamt-Umsatz und der Gesamt-Gewinn,
 - die Anzahl an insgesamt begonnenen und abgebrochenen Verhandlungen sowie
 - Kennzahlen zu den durchschnittlichen Gewinnen und Umsätzen pro Transaktion und
 - Kennzahlen zu den Güterpreisen
 zugeordnet. Die genaue Zuordnung ist anhand zweier simulationslaufübergreifender Auswertungstabellen ersichtlich, nämlich einer Makro-Umsatz-Tabelle sowie einer Makro-Preis-Tabelle, die für Analysen auf der Makro-Ebene (s. Abschnitt 7.3.2.1) – aus den Rohdaten aller (in der Regel vier) Simulationsläufe – generiert werden.
2. **Meso-Ebene:** Auf der Meso-Ebene werden die einzelnen Wertschöpfungsstufen und die innerhalb der Stufen ggf. gebildeten Agentengruppen untersucht. Neben übergreifenden Kennzahlen für die Wertschöpfungsstufen (wie z. B. Stufen-Gewinn und

⁶⁹ wie z. B. Virtuelle Unternehmen oder Strategische Netzwerke

–Umsatz) finden sich hier auch für Agentengruppen innerhalb einer Stufe des Wertschöpfungsnetzwerks Angaben zum Umsatz- und Gewinnanteil der Gruppe am Stufenumsatz- und –gewinn, so dass die Performanz von mehreren Agentengruppen miteinander verglichen werden kann. Für Analysen auf der Meso-Ebene werden die zur Verfügung stehenden Simulationsdaten über alle Simulationsläufe hinweg verdichtet und in Kennzahlen und Lage- sowie Streuungsmaße überführt, die einer einzelnen übergreifenden Meso-Tabelle zugeordnet werden. Eine Darstellung die Struktur der Meso-Tabelle findet sich in Abschnitt 7.3.2.2 der Arbeit.

3. **Mikro-Ebene:** Auf der Mikro-Ebene erfolgt die gruppenbezogene Betrachtung der Performanz einzelner Agenten. Für jeden Agenten ist ersichtlich, welcher Agentengruppe er angehört, welche individuellen Kennzahlen er aufweist (z. B. Agenten-Umsatz und –Gewinn, Anzahl seiner Transaktionen usw.) und wie diese zu den aggregierten Kennzahlen seiner Gruppe (z. B. hinsichtlich seines Anteils am Gruppen-Gewinn oder –Umsatz) in Relation stehen. Auf der Basis der Informationen auf der Mikro-Ebene können daher u. a. Fragen zu gruppenbezogenen Phänomenen wie der Umsatz- und Gewinnverteilung innerhalb von Agentengruppen beantwortet werden. Entsprechende Mikro-Tabellen werden in der gegenwärtigen Realisierung nicht über alle Simulationsläufe hinweg, sondern pro Lauf erzeugt, so dass insgesamt vier Mikro-Tabellen für Detailanalysen zur Verfügung stehen. Ein Beispiel für den genauen Aufbau der Mikro-Tabellen findet sich in Abschnitt 7.3.2.3.

Für jede simulierte Szenario-Parametrisierung stehen so nach der Auswertung der Rohdaten insgesamt sieben Ergebnistabellen zur Verfügung, mit denen Fragestellungen auf der Makro-, Meso- und Mikro-Ebene der Agenten-Gesellschaft auf einer quantitativen Basis effizient untersucht werden können. Wie die darin enthaltenen Kennzahlen generiert und den verschiedenen Tabellen zugeordnet werden, wird im folgenden Abschnitt erläutert.

7.3 Quantitative Evaluation

7.3.1 Generierung von Kennzahlen

Alle quantitativen Kennzahlen und statistischen Maße werden durch Weiterverarbeitung der von der MACSIMA-Simulationsinfrastruktur erzeugten Simulations-Rohdaten (vgl. Kapitel 6) generiert. Die Weiterverarbeitung erfolgt anhand einer ganzen Reihe von komplexen Perl- und Shell-Skripten, die speziell für die Auswertung von MACSIMA-Simulations-Rohdaten entwickelt wurden.

Die Verdichtung der Rohdaten zu Kennzahlen und Maßen erfolgt dabei je nach Verwendungszweck einer Kennzahl entweder unmittelbar aus den über einen einzelnen Simulationslauf hinweg anfallenden Daten oder aus den über alle Simulationsläufe hinweg anfallenden Daten bzw. aggregierten Kennzahlen für die einzelnen Läufe. Dies erfolgt aufgrund der großen Masse an Rohdaten automatisch.

Die berechneten statistischen Kennzahlen und Lage- sowie Streuungsmaße werden anschließend speziellen Ergebnistabellen für die Makro-, Meso- und Mikro-Analyse zugeordnet, die von den verwendeten Skripten ebenfalls automatisch erzeugt werden. Deren genaue Struktur wird in Abschnitt 7.3.2 näher erläutert und im Anhang in Abschnitt 11.3 weiter veranschaulicht.

Diese Ergebnistabellen sind zu dem Zweck gestaltet worden, die Kennzahlen kompakt und auf den Untersuchungsgegenstand bezogen zusammenzustellen, so dass Analysen sowie ein Benchmarking von verschiedenen Szenarien auf Makro-, Meso- und Mikro-Ebene effizient

durchgeführt werden können. Die für diese Tabellen relevanten Themen, von denen ihre Struktur bestimmt wird, beinhalten u. a.

- den Vergleich wirtschaftsbezogener Kennzahlen (wie z. B. Preise, Umsatz, Gewinn, deren Verteilungen),
- die Betrachtung von Wertschöpfungsstufen sowie von Agentengruppen des simulierten Szenarios und deren Gegenüberstellung, sowie
- die Gegenüberstellung mehrerer Läufe eines Simulationsszenarios unter denselben Bedingungen.

7.3.2 Ergebnistabellen

Bei der Auswertung der Simulations-Rohdaten für die normalerweise vier für ein Szenario durchgeführten Simulationsläufe werden automatisiert sieben standardisierte Ergebnistabellen generiert. Diese Ergebnistabellen finden im Rahmen der Evaluation von Simulationsszenarien in Kapitel 8 Verwendung.

Wie bereits erwähnt werden für jedes Szenario standardmäßig vier Simulationsläufe vorgenommen, um vor dem Hintergrund des inhärenten Nichtdeterminismus und den daraus resultierenden Streuungseffekten für die Ergebnisvariablen einerseits eine gewisse statistische Glättung der Ergebnisse vornehmen zu können und andererseits bestimmte statistische Maße auch sinnvoll berechnen zu können. Zur Identifizierung von sogenannten Ausreißern wären mindestens drei Läufe erforderlich.

Die sieben Ergebnistabellen sind nach dem vorgestellten Makro-, Meso- und Mikro-Ebenen-Konzept gegliedert bzw. benannt. Dabei handelt es sich um

- zwei verschieden strukturierte Tabellen für die Makro-Ebene, die sich aus der Aggregation der Daten aller Simulationsläufe ergeben - mit den Benennungen
 - Szenarioname.tabelle.makro.preis.csv und
 - Szenarioname.tabelle.makro.umsatz.csv,
- einer Medio-Ergebnistabelle, die sich ebenfalls durch Aggregation über alle Simulationsläufe hinweg ergibt - mit der Benennung
 - Szenarioname.tabelle.medio.csv, sowie
- vier gleich strukturierten Ergebnistabellen für die Mikro-Ebene, die jeweils pro Simulationslauf des Szenarios erzeugt werden - mit der Benennung
 - Szenarioname.tabelle.mikro_LaufNr.csv.

Die einzelnen Tabellen enthalten jeweils betriebswirtschaftliche Kennzahlen zu den einzelnen Läufen sowie aggregierte statistische Maße, wie z. B. für viele Ergebnisvariablen den Mittelwert, den Median, die Varianz, den Variationskoeffizienten sowie die Standardabweichung oder den Herfindahl-Index (s. Abschnitt 7.3.3). Entsprechenden Kennzahlen- und Maß-Werte werden von der Auswertungsinfrastruktur automatisiert aus den Rohdaten berechnet.

Die Tabellen sind so standardisiert, dass sie für den Großteil der relevanten Untersuchungen verwendet werden können. Um die Ergebnistabellen kompakter und übersichtlicher darstellen zu können, werden häufig vorkommende Bezeichnungen abgekürzt. Da manche Abkürzungen tabellenübergreifend Verwendung finden, werden einige von ihnen in der folgenden Übersicht kurz erläutert.

Abk.	Bedeutung	Abk.	Bedeutung	Abk.	Bedeutung
#	Anzahl	Ges	Gesamt	St	Stufe(n)
AbQ	Abschlussquote	Gr	Gruppe(n)	Su	Supplier
abs	absoluter	MW	Mittelwert	T	Tausend
Ag	Agenten	P	Proposal(s)	U	Umsatz
An	Anteil	pM	Prozentualer erwarteter Mittelwert	VaK	Variationskoeffizient
Br	Branch	Pr	Processor	V-D	Verkaufs-Deal(s)
Ch	Chip	R	Rejects		
D	Deal(s)	Ru	Runden		
De	Deposit	SA	Standardabweichung		
E-D	Einkaufs-Deal(s)	Sf	Sterbefälle		
G	Gewinn	Si	Silicium		

Tabelle 8: In den Ergebnistabellen verwendete Abkürzungen

Im Folgenden wird die Struktur der Ergebnistabellen für die Makro-, Meso- und Mikro-Ebene zusammen mit den zugehörigen Kennzahlen anhand von Beispielen erläutert. Dabei werden die beiden Ergebnistabellen für die Makro-Ebene, nämlich die Makro-Umsatz- sowie die Makro-Preis-Tabelle, als erstes vorgestellt.

7.3.2.1 Makro-Ergebnistabellen

Für Analysen auf Makro-Ebene werden pro simuliertem Szenario jeweils zwei Makro-Ergebnistabellen erzeugt, nämlich

- eine *Makro-Umsatz-Tabelle* sowie
- eine *Makro-Preis-Tabelle*.

Diese weisen eine unterschiedliche Struktur auf, da sich umsatz- und gewinnbezogene Kennzahlen auf Agenten sowie Agentengruppen beziehen, wohingegen sich Kennzahlen für die Entwicklung und Streuung von Güterpreisen auf die im System gehandelten Güter beziehen.

Natürlich stehen die Kennzahlen beider Tabellen in einem Zusammenhang, wodurch eine detaillierte Analyse der in den Simulationen auftretenden Makrophänomene gewährleistet ist. Neben eher einfach berechenbaren Kennzahlen enthalten die beiden Tabellen auch Statistik-Maße, die die ökonomischen Entwicklungen, die innerhalb eines oder mehrerer Simulationsläufe emergieren, bis zum einem gewissen Grad fassbar und nachvollziehbar machen.

7.3.2.1.1 Makro-Umsatz-Ergebnistabelle

Die Makro-Umsatz-Ergebnistabelle stellt eine übergreifende Gesamtsicht auf die durchgeführten Simulationsläufe bereit. Sie enthält verschiedene betriebswirtschaftliche Kennzahlen, mit denen Fragestellungen zu den Themen Umsatz und Gewinn sowie Kommunikations- und Markteffizienz bearbeitet werden können.

Die Struktur der Makro-Umsatz-Tabelle ist in den beiden folgenden Tabellen dargestellt. Sie gliedert sich in zwei Substrukturen. Die erste (s. Tabelle 9) stellt Kennzahlen-Werte dar, die über die *gesamte* Agenten-Gesellschaft erhoben werden.

Szenario-Name	GesU	GesG	U/V-D	G/V-D	#V-D	#V-R	#P	#V-D/#Ru
Lauf 1 (L1)								
Lauf 2 (L2)								
Lauf 3 (L3)								
Lauf 4 (L3)								
Mittelwert (MW)								
Standardabweichung (SA)								
Variationskoeffizient (VK)								
Varianz (σ)								
Startwert								
Endwert								
Minimum								
Maximum								
Median								
Oberes Quartil								
Unteres Quartil								
Herfindahl-Index (HHI)								

Legende:

GesU:	Gesamt-Umsatz	#V-D:	Anzahl der Verkaufs-Deals
GesG:	Gesamt-Gewinn	#V-R:	Anzahl der Verkaufs-Rejects
U/V-D:	Umsatz pro Verkaufs-Deal	#P:	Anzahl der Proposals
G/V-D:	Gewinn pro Verkaufs-Deal	#Ru:	Anzahl der Runden

Tabelle 9: Teil 1 der Struktur der Makro-Umsatz-Ergebnistabelle

Tabelle 9 enthält Spalten für die folgenden Kennzahlen bzw. Ergebnisvariablen:

1. Gesamt-Umsatz des Szenarios (GesU)
2. Gesamt-Gewinn des Szenario (GesG)
3. Umsatz pro Verkaufs-Deal (U/VD): gibt an, wieviel Agenten bei dem Verkauf eines Gutes durchschnittlich umsetzen bzw. welchen Durchschnittspreis sie dabei erzielen.
4. Gewinn pro Verkaufs-Deal (G/VD): gibt an, wieviel Agenten bei dem Verkauf eines Gutes durchschnittlich verdienen bzw. welchen absoluten Durchschnittsgewinn sie dabei erzielen.
5. Anzahl der Verkaufs-Deals (#VD): entspricht der Anzahl der erfolgreich abgeschlossenen Verhandlungen bzw. erfolgreich getätigten Transaktionen.
6. Anzahl von Rejects bei Verhandlungen (#R): entspricht der Anzahl der abgebrochenen Verhandlungen.
7. Erfolgsquote ($\#VD / \#Ru$): macht eine Aussage über die relative Häufigkeit erfolgreicher Verhandlungen.

Daran schließt sich die zweite Substruktur an (s. Tabelle 10). Sie enthält Kennzahlen-Werte, die nicht für die Agenten-Gesellschaft, sondern für die in ihr enthaltenen Agentengruppen berechnet werden, nämlich die

8. Umsatzanteile der Agentengruppen, die die Stufen des Wertschöpfungsnetzwerks bilden, die
9. Gewinnanteile der Agentengruppen, die die Stufen des Wertschöpfungsnetzwerks bilden sowie die
10. Anzahl der Sterbefälle (#Sf).

Die ersten vier Ergebnis-Zeilen der Tabelle sind laufbezogen. Sie werden mit Kennzahlen-Werten befüllt, die für jeden der normalerweise vier Simulationsläufe getrennt voneinander berechnet werden.

Szenario-Name	Umsatzanteil der Stufe in %				Gewinnanteil der Stufe in %			
	Gr SuSi	Gr DeCh	Gr DePr	Gr DeCo	Gr SuSi	Gr DeCh	Gr DePr	Gr DeCo
Lauf 1 (L1)								
Lauf 2 (L2)								
Lauf 3 (L3)								
Lauf 4 (L3)								
Mittelwert (MW)								
Standardabweichung (SA)								
Variationskoeffizient (VK)								
Varianz (σ)								
Startwert								
Endwert								
Minimum								
Maximum								
Median								
Oberes Quartil								
Unteres Quartil								
Herfindahl-Index (HHI)								

Legende:

Gr SuSi: Gruppe aller Silicium-Supplier-Agenten Gr DePr: Gruppe aller Prozessor-Deposit-Agenten

Gr DeCh: Gruppe aller Chip-Deposit-Agenten Gr DeCo: Gruppe aller Computer-Deposit-Agenten

Tabelle 10: Teil 2 der Struktur der Makro-Umsatz-Ergebnistabelle

Die darauffolgenden Zeilen enthalten über diese 4 Kennzahlen-Werte hinweg ermittelte, das heißt laufübergreifende statistische Maße in der Reihenfolge

- Mittelwert, Standardabweichung, Variationskoeffizient und Varianz,
- Startwert, Endwert, Minimum und Maximum sowie
- Median, oberes und unteres Quartil und
- Herfindahl-Index.

Zugegebenermaßen besitzen diese Werte für eine Testreihe mit nur vier Simulationsläufen eine eher geringe Aussagekraft. Werden mehr Simulationsläufe durchgeführt, passen sich aber Auswertungslogik und Tabellenstruktur daran an, so dass die entsprechenden Tabelleneinträge in anderen Auswertungs- und Analyse--Settings durchaus sehr nützlich sind.

Die Makro-Umsatz-Tabelle bietet auf diese Weise eine gute Gesamtsicht auf den Markt bzw. das Wertschöpfungsnetzwerk mit seinen Stufen. Die kompakte Darstellung wesentlicher ökonomischer Kennzahlen ermöglicht eine effiziente Analyse und Bewertung der Gesamtleistung des Systems. Bei den untenstehenden Tabellen handelt es sich um eine exemplarische Darstellung der Makro-Umsatz-Tabelle. Aufgrund der Breite ihrer im Original zusammenhängenden Struktur erfolgt die Darstellung hier in zwei Teilen.

7.3.2.1.2 Makro-Preis-Ergebnistabelle

Die Makro-Preis-Ergebnistabelle bietet einen kompakten und ausführlichen Überblick zu den Preisniveaus aller im System gehandelten Güter, d. h. der laufbezogenen Entwicklung und Streuung der Güterpreise sowie zu laufübergreifenden diesbezüglichen Abweichungen. Dies beinhaltet Informationen zu der Anzahl an erfolgreich abgeschlossenen Verhandlungen (sogenannten *Verkaufs-Deals*) sowie statistische Preisinformationen. Sie besteht, wie in Tabelle 11 dargestellt, aus einer laufbezogenen sowie einer laufübergreifenden Substruktur. Die obere, laufbezogene Hälfte der Tabelle ist nach Gütern sowie den durchgeführten Simulationsläufen gegliedert und beinhaltet daher Kennzahlen bzgl. der Güter, die pro Lauf getrennt berechnet werden. Die untere Tabellenhälfte beinhaltet pro gehandeltem Gut jeweils laufübergreifend berechnete Kennzahlen und Maße.

Szenario-Name																
Laufbezogen																
Gut	Silicium				Chip				Prozessor				Computer			
Lauf i	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Verkaufs-Deals (#VD)																
Mittelwert																
Standardabweichung (SA)																
Variationskoeffizient (VaK)																
Varianz (σ)																
Startpreis																
Endpreis																
Maximalpreis																
Minimalpreis																
Median																
Oberes Quartil																
Unteres Quartil																
Laufübergreifend																
	Silicium				Chip				Prozessor				Computer			
Verkaufs-Deals (#VD)																
Mittelwert																
Standardabweichung (SA)																
Variationskoeffizient (VaK)																
Varianz (σ)																
Mittlerer Startpreis																
Mittlerer Endpreis																
Maximalpreis																
Minimalpreis																
Median																
Oberes Quartil																
Unteres Quartil																

Legende:

#VD: Anzahl der Verkaufs-Deals VaK: Variationskoeffizient
SA: Standardabweichung

Tabelle 11: Struktur der Makro-Preis-Ergebnistabelle

Die Werte der statistischen Maße beziehen sich in beiden Substrukturen jeweils nur auf die Daten der bei erfolgreich abgeschlossenen Verhandlungen erzielten Verkaufspreise. Diese pro Gut erzielten Preise werden bei den Berechnungen gemäß ihrer Reihenfolge als Zeitreihe behandelt. Die verwendeten Maße sind

1. Mittelwert: der mittlere laufbezogen oder laufübergreifend erzielte Preis für ein Gut
2. Standardabweichung (SA)
3. Variationskoeffizient (VaK)
4. Varianz (σ^2)
5. Startpreis: der zu Simulationsbeginn erzielte Preis für ein Gut
6. Endpreis: der zu Simulationsende erzielte Preis für ein Gut
7. Maximalpreis: der laufbezogen oder laufübergreifend erzielte höchste Preis für ein Gut
8. Minimalpreis. der laufbezogen oder laufübergreifend erzielte geringste Preis für ein Gut
9. Median
10. Oberes Quartil
11. Unteres Quartil

Die Definitionen der verwendeten statistischen Maße finden sich in Abschnitt 7.3.3.

7.3.2.2 Meso-Ergebnistabelle

Die Meso-Ergebnistabelle (siehe Tabelle 12) enthält Kennzahlen zu Agentengruppen und Stufen (s. Zeilenbeschriftungen) und setzt diese in einigen ihrer Spalten auch zueinander in Relation.

Die erste obere Tabellenhälfte bietet eine Sicht auf eine ganze Reihe von Performanz-Kennzahlen für die Agentengruppen in jedem einzelnen der vier durchgeführten Simulationsläufe.

Jeder der Agentengruppen (SupplierSilicium, DepositChip, DepositProcessor1, DepositProcessor2, DepositComputer und BranchComputer) werden in den Spalten der Tabelle daher nicht einzelne Kennzahlen-Werte, sondern Werte-Quadrupel zugeordnet.

Diese Quadrupel enthalten die Kennzahlen-Werte der einzelnen vier Läufe für die jeweilige Gruppe. In der Regel teilen sich daher jeweils 2 benachbarte Zeilen und Spalten die gleiche übergeordnete Beschriftung.

In der Beispiel-Tabelle 12 entsprechen vier Agentengruppen genau den Stufen, d. h. sie beinhalten alle Agenten der jeweiligen Stufe. Die Stufe der DepositProcessor-Agenten besteht jedoch im Beispiel (gemäß der Definition in der medio.gruppen-Datei) aus zwei Agentengruppen, nämlich den Gruppen DepositProcessor1 und DepositProcessor2.

Die erste und dritte Spalte geben Stufenumsatz (St-U) und Stufengewinn (St-G) noch unabhängig von den Agentengruppen für die jeweiligen Stufen, d. h. alle diesen Stufen zugeordneten Agenten, aufsummiert an.

In den Spalten zu Stufenumsatz (St-U) und Stufengewinn (St-G) enthalten die den Gruppen DepositProcessor1 und DepositProcessor2 zugeordneten Quadrupel also die gleichen Werte, in den weiteren Spalten (z. B. Gruppenanteil am Stufenumsatz) werden den beiden Agentengruppen jedoch ihre individuellen Werte zugeordnet.

Die in Tabelle 12 dargestellte Struktur der Meso-Tabelle ist wegen des Umfangs der Tabelle bzw. aus Gründen der Übersichtlichkeit nicht ganz vollständig, sondern zeigt einen wesentlichen Ausschnitt bzgl. der Kennzahlen

- Stufen-Umsatz (St-U)
- Gruppen-Umsatz-Anteil (Gr-UAn) am Stufen-Umsatz in Prozent (%)
- Stufen-Gewinn (St-G)
- Gruppen-Gewinn-Anteil (Gr-GAn) am Stufen-Gewinn in Prozent
- Gruppen-Gewinn-Anteil am Gesamt-Gewinn (GesG) in Prozent
- Anzahl (#) der Deals (bzw. erfolgreichen Verhandlungsabschlüsse) pro Gruppe
- Anzahl der Rejects (bzw. Verhandlungsabbrüche) pro Gruppe
- Abschlussquote (AbQ) pro Gruppe
- Sterbefälle (Sf) pro Gruppe (d. h. Anzahl der während des Simulationslaufs insolvent werdenden Agenten)

Sicht auf die einzelnen Simulationsläufe																		
Lauf	Stufen-Umsatz (St-U)		Gr-UAn am St-U in %		Stufen-Gewinn (St-G)		Gr-GAn am St-G in %		Gr-GAn am GesG in %		#Deals pro Gruppe		#Rejects pro Gruppe		AbQ pro Gruppe		Sf	
	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
	3	4	3	4	3	4	3	4	3	4	3	4	3	4	3	4	3	4
Gr SuSi			%	%			%	%	%	%								
Gr DeCh			%	%			%	%	%	%								
Gr DePr1			%	%			%	%	%	%								
Gr DePr2			%	%			%	%	%	%								
Gr DeCo			%	%			%	%	%	%								
Gr BrCo	0	0	-	-	0	0	-	-	0%	0%								
	0	0	-	-	0	0	-	-	0%	0%								
Laufübergreifende Sicht auf die Simulationsläufe 1 - 4																		
	MW	SA	MW	SA	MW	SA	MW	SA	MW	SA	MW	SA	MW	SA	MW	SA	MW	SA
	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK
Gr SuSi																		
Gr DeCh																		
Gr DePr1																		
Gr DePr2																		
Gr DeCo																		
Gr BrCo	0	0	-	-	0	0	-	-	0	0								
	-	-	-	-	-	-	-	-	-	-								

Legende:

Gr SuSi: Gruppe aller Silicium-Supplier-Agenten
 Gr DeCh: Gruppe aller Chip-Deposit-Agenten
 Gr DePr: Gruppe aller Prozessor-Deposit-Agenten
 Gr DeCo: Gruppe aller Computer-Deposit-Agenten
 Gr BrCo: Gruppe aller Computer-Branch-Agenten
 St-U: Stufen-Umsatz
 St-G: Stufen-Gewinn
 Gr-UAn: Gruppen-Umsatzanteil

Gr-GAn: Gruppen-Gewinnanteil
 GesG: Gesamt-Gewinn
 #Deals: Anzahl der Deals
 MW: Mittelwert
 Rel. SW: Relative Spannweite
 SA: Standardabweichung
 VaK: Variationskoeffizient

Tabelle 12: Wesentliche Spalten der Meso-Ergebnistabelle

Nicht in Tabelle 12 dargestellte Spalten der Meso-Ergebnistabelle enthalten zusätzlich Werte zu den betriebswirtschaftlichen Kennzahlen

- Anzahl der Einkaufs-Deals (#E-D) pro Gruppe
- Einkaufs-Abschlussquote (E-AbQ) pro Gruppe

- Anzahl der Verkaufs-Deals (#V-D) pro Gruppe
- Anzahl der Verkaufs-Rejects (#V-R) pro Gruppe
- Verkaufs-Abschlussquote (V-AbQ) pro Gruppe
- Gewinn pro Verkaufs-Deal (G/V-D) pro Gruppe
- Umsatz pro Verkaufs-Deal (U/V-D) pro Gruppe
- Absoluter Gruppen-Umsatz (Abs GrU)
- Absoluter Gruppen-Gewinn (Abs GrG)

Während in der oberen Hälfte der Meso-Tabelle die Quadrupel die Werte der jeweiligen vier Simulationsläufe abbilden, entsprechen sie in der unteren Hälfte den folgenden vier Statistik-Maßzahlen, welche nicht für einzelne Simulationsläufe, sondern laufübergreifend pro Gruppe berechnet werden:

- Arithmetischer Mittelwert
- Standardabweichung
- Relative Spannweite
- Variationskoeffizient

7.3.2.3 Mikro-Ergebnistabelle

Für Analysen auf der Mikro-Ebene wird pro Szenario für jeden durchgeführten Simulationslauf eine laufbezogene Mikro-Ergebnistabelle (siehe Tabelle 13) generiert. In ihr sind die laufbezogenen Ergebnisse aller in dem Simulationsszenario definierten und anhand der medio.gruppen-Datei zur Auswertung vorgesehenen Gruppen enthalten.

Die übergeordnete Struktur jeder dieser Mikro-Ergebnistabellen folgt dabei der Einteilung nach Agentengruppen. Im Beispiel sind fünf Gruppen definiert, denen jeweils alle Agenten einer Stufe des Wertschöpfungsnetzwerks angehören.

Daher gliedert sich die Beispiel-Mikro-Tabelle 13 in fünf gruppenbezogene Sub-Strukturen. Die in jeder von ihnen abgebildeten Informationen und Kennzahlen sind:

1. Gruppe (Gr): Angabe des Namens der Agentengruppe
2. Anzahl der Agenten (#Ag), die zur Gruppe gehören.
3. Zu erwartender prozentualer Mittelwert (pM), der für jeden Agenten bei einer statistischen Gleichverteilung von Umsatz, Gewinn oder der Anzahl der Transaktionen innerhalb der Gruppe zu erwarten wäre. Ein Vergleich der tatsächlichen Werte mit diesem Idealwert ist im Rahmen von Gruppen- und Stufentests interessant.
4. Gewinn (G)
5. Umsatz (U)
6. Anzahl der Deals (D)
7. Anzahl der Einkaufs-Deals (E-D)
8. Anzahl der Verkaufs-Deals (V-D)
9. Gewinn pro Verkaufs-Deal (G/V-D)
10. Umsatz pro Verkaufs-Deal (U/V-D)

In der ersten Zeile jeder gruppenbezogenen Substruktur werden für alle Kennzahlen die gruppenbezogenen Gesamtwerte angegeben.

	Gruppe (Gr)	#Ag	pM	Gewinn (G)	Umsatz (U)	#Deals (D)	E-D	V-D	G/V-D	U/V-D
	SuSi	10	10,00%				0			
Maximum							0			
Minimum							0			
Mittelwert							0			
Median							0			
Relative Spannweite							-			
Standardabweichung							0			
Herfindahl-Index							-			
	DeCh	10	10,00%							
Maximum										
Minimum										
Mittelwert										
Median										
relative Spannweite										
Standardabweichung										
Herfindahl-Index										
	DePr	10	10,00%							
Maximum										
Minimum										
Mittelwert										
Median										
relative Spannweite										
Standardabweichung										
Herfindahl-Index										
	DeCo	10	10,00%							
Maximum										
Minimum										
Mittelwert										
Median										
Relative Spannweite										
Standardabweichung										
Herfindahl-Index										
	BrCo	10	10,00%	0	0			0	-	-
Maximum				0	0			0	-	-
Minimum				0	0			0	-	-
Mittelwert				0	0			0	-	-
Median				0	0			0	-	-
relative Spannweite				-	-			-	-	-
Standardabweichung				0	0			0	-	-
Herfindahl-Index				-	-			-	-	-

Legende:

#Ag: Anzahl der Agenten
 pM: prozentualer Mittelwert
 V-D: Anzahl der Verkaufs-Deals
 U/V-D: Umsatz pro Verkaufs-Deal
 G/V-D: Gewinn pro Verkaufs-Deal

SuSi: Gruppe aller Silicium-Supplier-Agenten
 DeCh: Gruppe aller Chip-Deposit-Agenten
 DePr: Gruppe aller Prozessor-Deposit-Agenten
 DeCo: Gruppe aller Computer-Deposit-Agenten
 BrCo: Gruppe aller Computer-Branch-Agenten

Tabelle 13: Struktur der Mikro-Ergebnistabelle

In den weiteren Zeilen finden sich statistische Maßzahlen, die aus den individuellen Einzelwerten aller Agenten der Gruppe zu den Kennzahlen 4 bis 10 errechnet werden. Die berechneten Kennzahlen und Maße der deskriptiven Statistik sind

- Maximum,
- Minimum,
- Mittelwert,
- Median,
- Relative Spannweite,
- Standardabweichung und
- Herfindahl-Index.

7.3.3 Verwendete Maße und Kennzahlen der Statistik

Bei der Auswertung der Rohdaten der Simulationsläufe werden mit Hilfe von verschiedenen Perl- und Shell-Skripten statistische Maße und Kennzahlen automatisiert berechnet. Maße und Kennzahlen überführen Merkmale der Simulationsläufe in analysier- und interpretierbare Zahlenwerte bzw. *Key Performance Indicators (KPI)*. Sie machen die Eigenschaften bzw. Effekte bestimmter Szenario-Parametrisierungen quantifizierbar, so dass die Eigenschaften

- wiederholter Simulationsläufe des gleichen Szenarios,
- unterschiedlicher Szenarien sowie auch
- die Auswirkungen der Verwendung unterschiedlicher Koordinationsmechanismen (wie bilaterale Verhandlungen im Vergleich zu unterschiedlichen Auktionsmechanismen)

im Rahmen eines Benchmarkings quantitativ miteinander verglichen werden können. Die in den Ergebnistabellen verwendeten Maße und Kennzahlen werden im Folgenden vorgestellt bzw. definiert.

7.3.3.1 Herfindahl-Index

Beim *Herfindahl-Index (HHI bzw. H)* [Bleymüller et al. 1996] handelt es sich um eine Kennzahl, die der Messung der Konzentration einer Verteilung auf Individuen (beispielsweise in Märkten) dient. Er wird in der Makro-Umsatz- sowie in der Mikro-Ergebnistabelle verwendet.

Der Index gibt Auskunft darüber, wie sich eine aufzuteilende Menge von Objekten (z. B. eine Geld-, Produktions- oder Absatzmenge) innerhalb einer Gruppe von Individuen bzw. Agenten (z. B. Produzenten) oder zwischen verschiedenen Gruppen (z. B. Wertschöpfungsstufen oder Produzentengruppen) verteilt. Die Verwendung dieser Kennzahl macht Sinn, da sich der in einem Szenario erzielte Gesamt-Umsatz und –Gewinn in der Regel nicht gleichmäßig verteilt – weder zwischen den Gruppen noch innerhalb von ihnen.

Für N Individuen a_i wird der *normalisierte* Herfindahl-Index H wie folgt berechnet:

$$H := \frac{\sum_{i=1}^N a_i^2}{\left(\sum_{i=1}^N a_i\right)^2}, \text{ mit } i \in \{1; N\}$$

Dabei entspricht a_i dem auf ein Individuum entfallenden Anteil. Der Index entspricht der Summe aller quadrierten Markt-, Umsatz- oder Gewinnanteile aller Marktteilnehmer und liegt mathematisch betrachtet im Wertebereich $[1/N; 1]$. Der Maximalwert 1 wird erreicht, wenn eine maximale Konzentration, d. h. eine Monopolsituation, vorliegt. Der Minimalwert $1/N$ wird im Falle einer Gleichverteilung der zu verteilenden Objekte unter den Marktteilnehmern (also bei Gleichheit aller a_i) erreicht. Bei Werten von $H < 0,1$ spricht man von geringer, bei Werten von $H > 0,18$ in der Regel bereits von hoher Konzentration.

7.3.3.2 Weitere Lage- und Streuungsmaße

Die weiterhin verwendeten statistischen Methoden sind den Grundlagen im Bereich der klassischen Statistik zuzurechnen, weshalb sie im Folgenden nur kurz behandelt werden. Sie helfen dabei, die umfangreichen protokollierten Bestände an quantitativen Simulationsergebnisdaten zu interpretierbaren Kennzahlen und Maßen zu verdichten, wie es für die Evaluation von Simulationen (siehe Kapitel 8) erforderlich ist.

Dadurch lassen sich signifikante Charakteristika von sowie Unterschiede zwischen Ergebnissen unterschiedlicher Szenario-Parametrisierungen erkennen, die aufgrund ihres Umfangs weder aus den Rohdaten noch aus einer Visualisierung in Form von Diagrammen (s. Abschnitt 7.4) zu ersehen wären.

Je nach Verwendungszweck kommen der arithmetische Mittelwert, der Median oder weitere Lage- und Streuungsmaße zum Einsatz. Diese werden im Folgenden in Anlehnung an [Graeber 2006] und [Ferschl 1985] kurz erläutert.

7.3.3.2.1 Lagemaße

In Abhängigkeit vom Verwendungszweck bzw. Untersuchungsfokus werden unterschiedliche Lagemaße wie z. B. der arithmetische Mittelwert oder der Median verwendet.

Der *arithmetische Mittelwert* \bar{x} ist die durch die Anzahl N der Messwerte dividierte Summe aller Messwerte x_i (mit $i \in \{1; N\}$):

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$$

Der arithmetische Mittelwert leistet eine einfache Durchschnittsberechnung, die für viele Fälle ausreicht, in denen die Verteilung von Messwerten relativ symmetrisch ist und nicht mehrere Maxima auftreten. Sofern Häufungen von Messwerten innerhalb bestimmter Wertebereiche vorkommen und insbesondere mehrere Maxima auftreten, hat der Median eine darüber hinausgehende Aussagekraft.

Der *Median* oder Zentralwert x_z ist der Wert, der in der Mitte der Variationsreihe liegt:

$$x_z = \frac{x_{N+1}}{2} \quad \text{falls } N \text{ ungerade}$$

$$x_z = \frac{1}{2} \cdot \left(x_{\frac{N}{2}} + x_{\frac{N}{2}+1} \right) \quad \text{falls } N \text{ gerade}$$

7.3.3.2.2 Streuungsmaße

Die *Varianz* ist wohl das am häufigsten verwendete Streuungsmaß der deskriptiven Statistik. Sie misst die Streuung der Messwerte relativ zu ihrem arithmetischen Mittelwert \bar{x} bzw. entspricht dem arithmetischen Mittel der Abweichungsquadrate der Messwerte:

$$\text{Var}(x) = \sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2$$

Die *Standardabweichung* σ ist die Quadratwurzel der Varianz:

$$\sigma = \sqrt{\text{Var}(x)}$$

Da sie in dem gleichen Wertebereich wie die Messwerte x_i liegen, ist sie oftmals leichter interpretierbar.

Dort, wo die Standardabweichung vieler Einzelergebnisse betrachtet wird, ist es wichtig, diesen Wert zu der Größenordnung der Werte in Bezug zu setzen, damit ein Vergleich verschiedener Simulationsläufe möglich wird.

Hierfür findet der *Variationskoeffizient* v_σ [Kohn 2004] Verwendung, ein auf das arithmetische Mittel bezogenes relatives Streuungsmaß, welches das Verhältnis der Standardabweichung zum arithmetischen Mittelwert angibt:

$$v_\sigma = \frac{\sigma}{\bar{x}}$$

Die *Relative Spannweite* \bar{R} ist ebenfalls ein relatives Streuungsmaß, das sich auf das arithmetische Mittel bezieht:

$$\bar{R} = (x_{max} - x_{min})/\bar{x}$$

Im Unterschied zum Variationskoeffizienten finden bei ihrer Berechnung ausschließlich das Maximum sowie das Minimum und der arithmetische Mittelwert aller Messwerte Berücksichtigung. Daher weist sie gegenüber Ausreißern keine Robustheit auf.

7.4 Qualitative Evaluation

Die Simulationsläufe von MACSIMA erzeugen Datenbestände, die ohne eine Weiterverarbeitung kaum vernünftig analysier- und interpretierbar sind. Eine geeignete Darstellung kann im Wesentlichen quantitativ oder graphisch erfolgen.

Da die Aussagekraft der im vorhergehenden Abschnitt 7.3 vorgestellten Kennzahlen und statistischen Maße bzgl. bestimmter Evaluationsfragestellungen begrenzt ist, wurden über den quantitativen Analyseansatz hinausgehende Visualisierungsmöglichkeiten implementiert, die der qualitativen Analyse dienen. Diese machen charakteristische Eigenschaften eines Simulationslaufs auf einen Blick interpretierbar und weisen ggf. auf Aspekte hin, die anhand der quantitativen Analysemethoden weiter untersucht werden können.

7.4.1 Visualisierungsoptionen für Simulationsergebnisse

Aus den protokollierten Simulations-Rohdaten können zur Visualisierung von Genen und Kennzahlen mit Hilfe von Perl- und Shell-Skripten sowie GnuPlot⁷⁰ automatisiert Diagramme in verschiedenen Formaten generiert werden, die beispielsweise

- den Evolutionsprozesses der durch Gene codierten Strategie-Parameter der Agenten und Agentengruppen sowie
- die auftretenden Preisschwankungen der im Szenario gehandelten Güter

⁷⁰ <http://www.gnuplot.info>

im Zeit- bzw. Simulationsverlauf veranschaulichen. Hieraus können leicht qualitative Erkenntnisse gewonnen werden, beispielsweise bzgl. des Adaptionsverhalten der Agenten – sowohl individuell als auch innerhalb von Gruppen, Wertschöpfungsstufen oder auch innerhalb des Gesamtsystems.

Die folgenden ausgewählten Skripte leisten folgende Visualisierungen:

1. Visualisierung von Preisentwicklungen:
 - a. *preisPlotSales.sh* (u. a. mit Optionen `-g` für grau und `-c` für Color): generiert aus den sales-Dateien der Läufe jeweils einen Preis-Plot.
2. Visualisierung von Genen und Genotypen:
 - a. *genSales.sh*: plottet Gene aus allen 4 Sales-Dateien und zeichnet die Gene für Buyer und Seller in *Szenarioname_SingleSales.png* und gemischt für Käufer und Verkäufer in *Szenarioname_MixedSales.png* und alle Gene zusammen in *Szenarioname_AllSales.png*.

Die Ausgabe dieses Skripts beinhaltet Grafiken der folgenden Art:

- *_MixedSales.png*: stellt für alle Gene jeweils ein Gen sowohl für den Buy- als auch für den Sell-Pool eines Agenten in einem Diagramm dar.
 - *_AllSales.png*: enthält alle Gene eines Pools eines Agenten in einem Diagramm.
 - *_SingleSales.png*: stellt die Entwicklung eines Gens eines Agenten-Pools in einem Diagramm dar.
- b. *genAvgTest.sh*: plottet Gen-Durchschnitte für alle Gruppen und Stufen aus allen vier Sales-Dateien.
Seine Ausgabe besteht aus den Grafiken *MixedAvg.png* und *AllAvg.png*, wobei erstere Grafik die Gen-Durchschnitte von Buyer- und Seller-Agent bzgl. eines Gens plottet und letztere alle Gene eines Pools.
 - c. *genGeno.sh*: leistet das Plotten der Gene sowohl für den Buy- als auch für den Sell-Pool der Agenten auf Basis der Protokoll-Datei *genotypes.csv* - inklusive des Plottens der Liquidität der einzelnen Agenten (*Cash* bzw. *CA*).
 - d. *genPopu.sh*: arbeitet auf den *{name}_population\${i}.csv*-Dateien und erzeugt PNG-Grafiken mit dem Plot der in Abschnitt 5.4.2 erläuterten Preisinformationen (*currentPrice*, *weighted_Average*, *maxPrice*, *basicPrice*, *mediumPrice*, *minPrice*) mit der Benennung *Szenarioname_AllPricesPopu.png*.

Das Skript zeichnet außerdem alle Gene, die die Genotypen in der Population besitzen, für Buyer- und Seller-Agenten sowie deren Fitness und die Runden, in denen der Genotyp im Einsatz war in Grafik-Dateien mit der Endung *_RoundsFor-CurrentGenotypeAvg_SinglePopu.png*.

Außerdem stellt dieses Skript die einzelnen Gene und Werte auch gemischt für Buyer und Seller dar und speichert diese in Grafik-Dateien mit der Endung *_MixedPopu.png*.

3. Visualisierung von Profiten
 - a. *profitTest.pl*: leistet die Berechnung von *dealNettoProfit* und *dealBruttoProfit* sowie deren Summe über die Laufzeit für jeden einzelnen Agenten für einen Lauf.

Im Folgenden finden sich Beispiele für einige der generierten Grafiken.

7.4.2 Visualisierung von Kennzahlen

In Abbildung 63 und Abbildung 64 wurden mit Hilfe des Skripts *preisPlotSales.sh* die Preisentwicklungen für unterschiedliche Szenario-Parametrisierungen visualisiert.

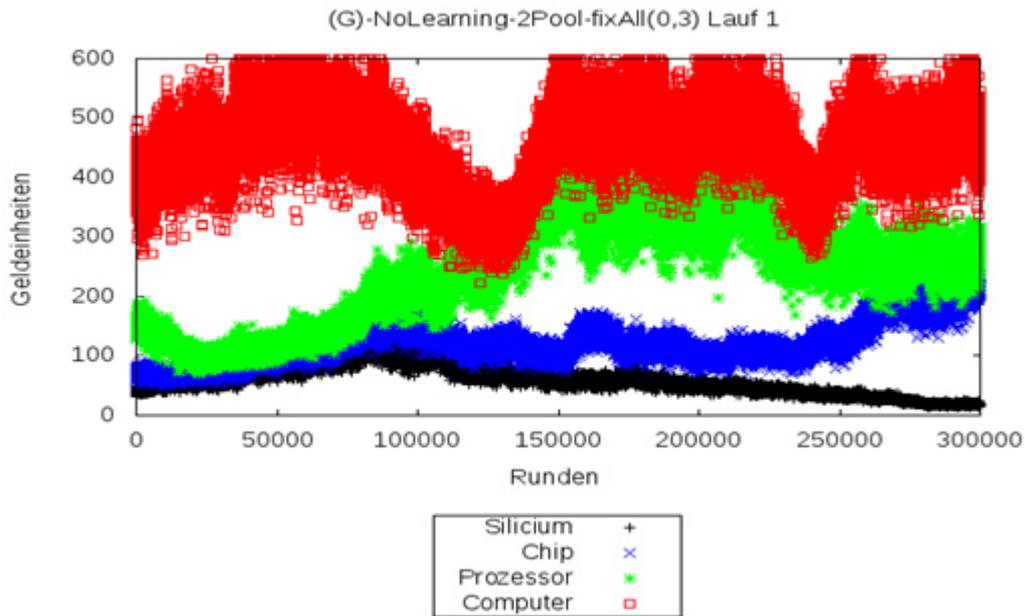


Abbildung 63: Visualisierung der Entwicklung der Güterpreise während eines Simulationslaufs mit dem Skript *preisPlotSales.sh*

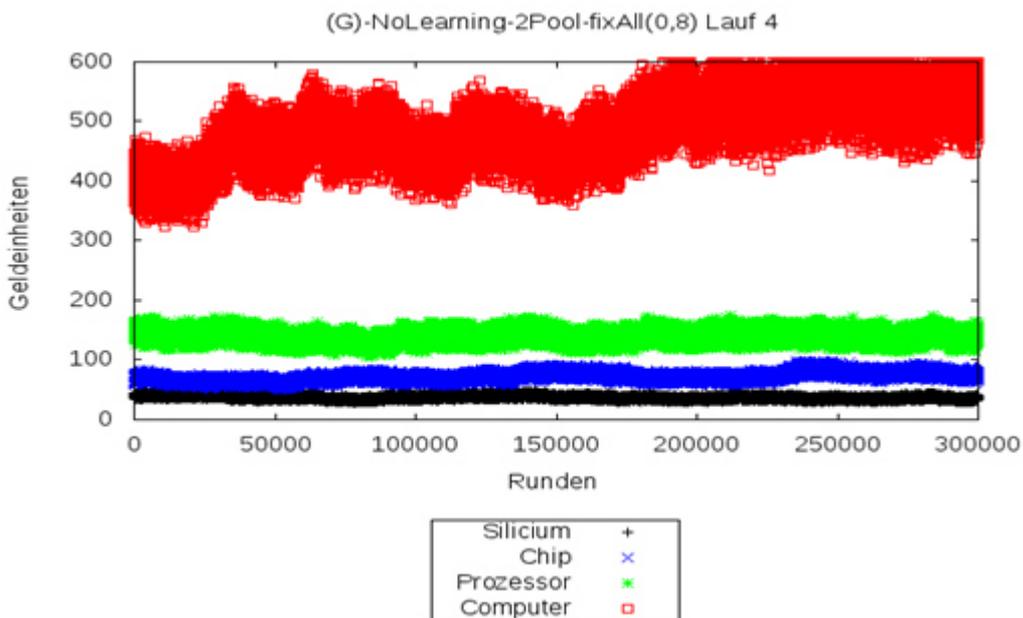


Abbildung 64: Preisentwicklung für eine andere Szenario-Parametrisierung – erzeugt mit dem Skript *preisPlotSales.sh*

Die beiden folgenden Abbildungen zeigen Visualisierungsmöglichkeiten für die Entwicklung der Kennzahlen *Maximalpreis* und *Verhandlungsdauer*.

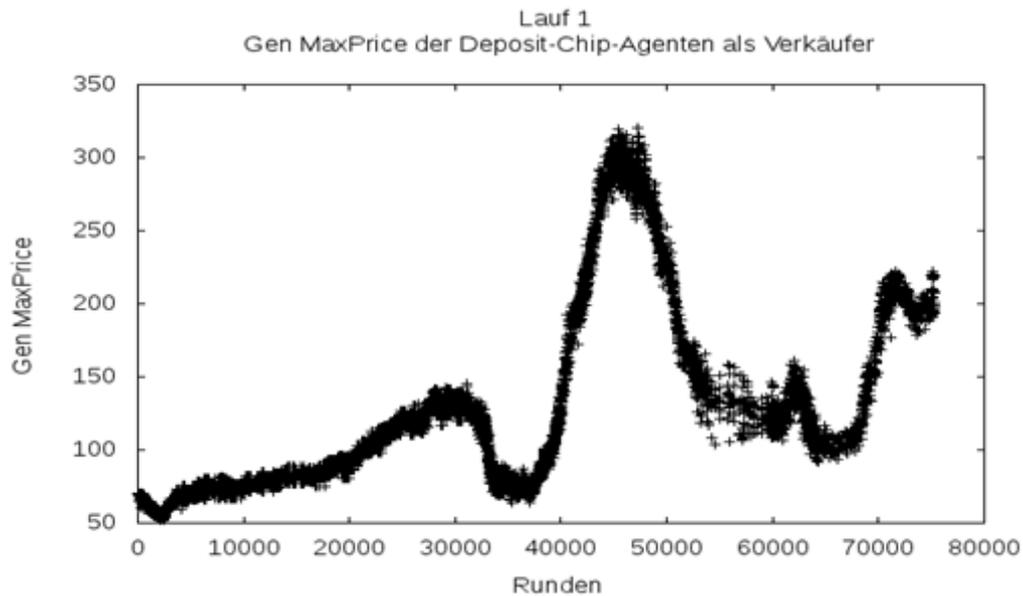


Abbildung 65: Entwicklung der internen Maximalpreis-Schranken (MaxPrice) von Chip-Produzenten – erzeugt mit dem Skript *genSales.sh*

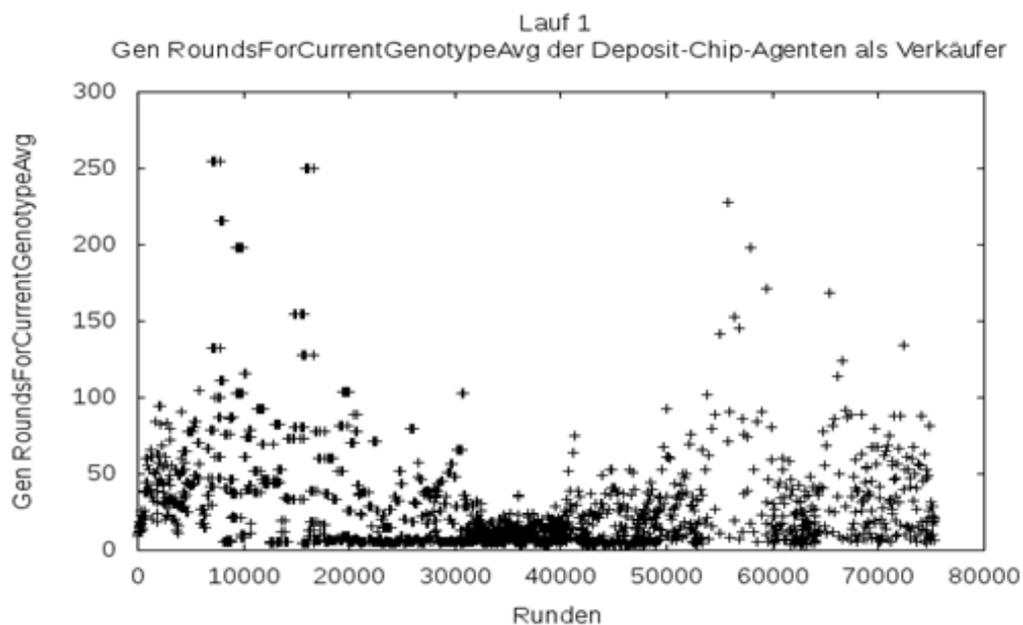


Abbildung 66: Entwicklung der Verhandlungsdauer bzw. der Rundenzahl, die eine Verhandlungsstrategie bis zu einem erfolgreichen Abschluss benötigt

Der Vergleich der Grafiken macht die Vorzüge einer Visualisierung für die Analyse von Simulationsläufen und ihren Ergebnissen deutlich. Durch sie entstehen qualitative Analyse-möglichkeiten, welche die quantitative Analyse sinnvoll ergänzen. Beispielsweise erleichtert es die Visualisierung, bei der Evaluation die Untersuchungshypothese zu formulieren, dass eine unterschiedliche Parametrisierung der Agenten-Gene zum Simulationsstart zu einer unterschiedlichen Dynamik der Güterpreis-Entwicklung und -Streuung führen kann.

Die folgende Abbildung wurde mit dem Skript *genSales.sh* erzeugt.

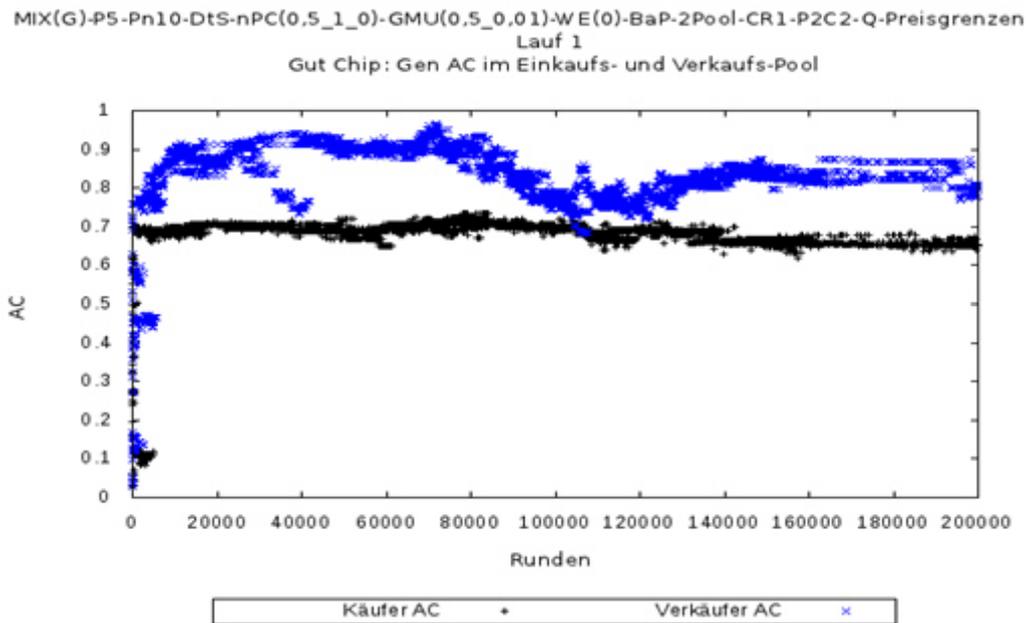


Abbildung 67: Entwicklung des Gens Acquisitiveness von Käufern und Verkäufern des Gutes Chip

Das Diagramm zeigt, dass die Gen-Pools beider Agentengruppen mit einer randomisiert erzeugten, breiten Verteilung des Gens Acquisitiveness in die ersten Verhandlungen starten, welches ihre „Beharrlichkeit“ bzgl. ihrer Verhandlungsposition codiert (zur genaueren Erklärung der Wirkungsweise des Gens siehe Abschnitt 5.3.2).

Dadurch, dass *gemischtes Lernen* innerhalb der Gruppen (MIX(G), siehe Abschnitt 5.4.1.4) aktiviert ist, tauschen die Agenten innerhalb ihrer Gruppen Informationen über erfolgreich abgeschlossene Verhandlungen aus bzw. übermitteln sich gegenseitig Gen-Wert-Kombinationen, die zu sogenannten *Deals* geführt haben – und die diese Agenten dann auch häufiger in darauffolgenden Verhandlungen benutzen.

In Abbildung 67 ist daher schon kurz nach Simulationsstart eine Häufung bestimmter Acquisitiveness-Gen-Werte innerhalb des Gen-Pools der Agentengruppen zu beobachten. Da in der Szenario-Parametrisierung *Deterministic Selection (DtS)* (siehe Abschnitt 5.4.3.1) als Selektions-Methode eingestellt wurde, kommt es bezüglich der Genotypen bzw. Strategieparameter-Kombinationen mit ihren Genen zu einem raschen Selektionsprozess. Dieser sorgt dafür, dass sich die Acquisitiveness-Gen-Werte der beiden miteinander verhandelnden Agenten sehr schnell auf einem relativ stabilen Niveau einpendeln. Dabei scheint es für die Verkäufer des knappen Gutes *Chip* individuell rational zu sein, relativ „hart“ zu verhandeln, d. h. nur in sehr geringerem Maße Preisnachlässe zu gewähren. Für die Käufer des Gutes scheint es hingegen vorteilhaft zu sein, den Preisvorstellungen der Verkäufer in stärkerem Maße entgegenzukommen – vermutlich führt dies zu einem Vorteil im Konkurrenzkampf mit den anderen Agenten auf der gleichen Stufe des Wertschöpfungsnetzwerks.

Die folgende Abbildung visualisiert für den gleichen Simulationslauf den Adaptionsprozess des Gens *Delta_Jump* (siehe Abschnitt 5.3.2) innerhalb des Gen-Pools der Agentengruppen Chip-Produzenten und den das Gut *Chip* nachfragenden Prozessor-Produzenten.

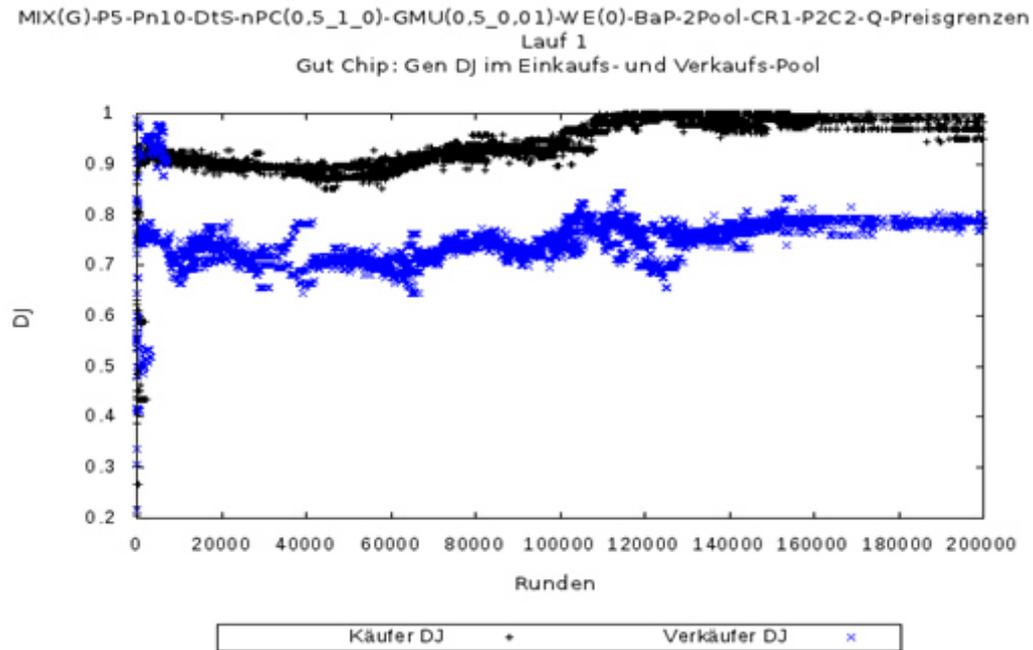


Abbildung 68: Entwicklung des Gens *Delta_Jump* von Verkäufern bzw. Produzenten des Guts Chip sowie von Käufern (Prozessor-Produzenten)

Das *Delta_Jump*-Gen codiert grob gesagt die von den Agenten gewünschte Marge, das heißt die Spanne zwischen dem individuell gefühlten durchschnittlichen Marktpreis für ein Gut, den ein Agent auf Basis seiner letzten Transaktionen anhand einer individuellen Gewichtungsfunktion berechnet, und der ersten Preisvorstellung, mit welcher der Agent in die Verhandlung startet.

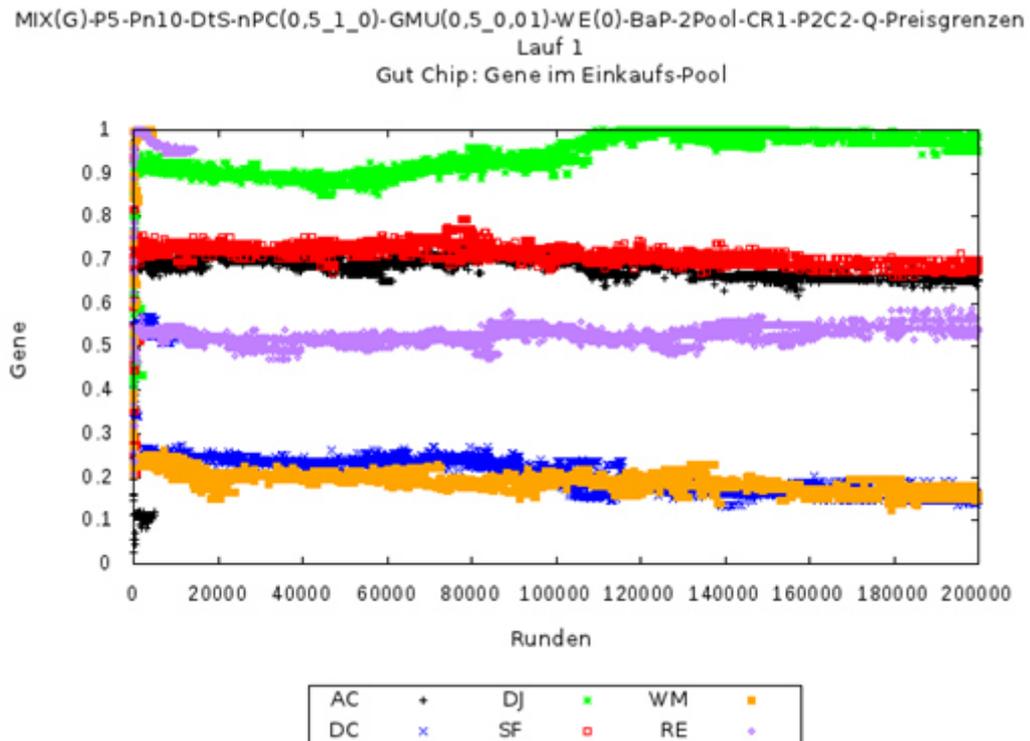


Abbildung 69: Entwicklung aller Gene im Einkaufspool der Chip-Produzenten

Der Adaptionsprozess von *Delta_Jump* (siehe Abbildung 68) läuft aufgrund der gleichen Parametrisierung des Lernverfahrens sehr ähnlich zu dem Acquisitiveness-Adaptionsprozess in Abbildung 67 ab, was nicht überrascht.

In ihrer Kombination lassen die beiden Visualisierungen in Abbildung 67 und Abbildung 68 den Schluss zu, dass es für Verkäufer des Gutes *Chip* eine dominante Strategie ist, mit im Vergleich zu den Nachfragern noch moderaten Margenwünschen in eine Verhandlung zu starten, dann aber in stärkerem Maße auf ihren Preisvorstellungen zu beharren, als dies ihre Verhandlungssopponenten tun. Dies macht Sinn, da so für die Anbieter die Chance steigt, dass sich die Nachfrager überhaupt auf Preisverhandlungen für das von ihnen angebotene Output-Gut einlassen (im Sinne von Lock-Angeboten) und nicht aufgrund einer Wucher-Vermutung Verhandlungen vorzeitig abbrechen.

Die als Käufer handelnden Agenten adaptieren sich an diese Verkäufer-Strategie, indem sie mit im Vergleich höheren Margenvorstellungen starten, dann aber in den folgenden Verhandlungsschritten die Verhandlung mit einer geringeren Wahrscheinlichkeit abbrechen und den Verkäufern eher entgegen kommen. Die Abbildung 69 und Abbildung 70 zeigen, dass auch alle Gene des Gen-Pools der Agentengruppen gleichzeitig visualisiert werden können.

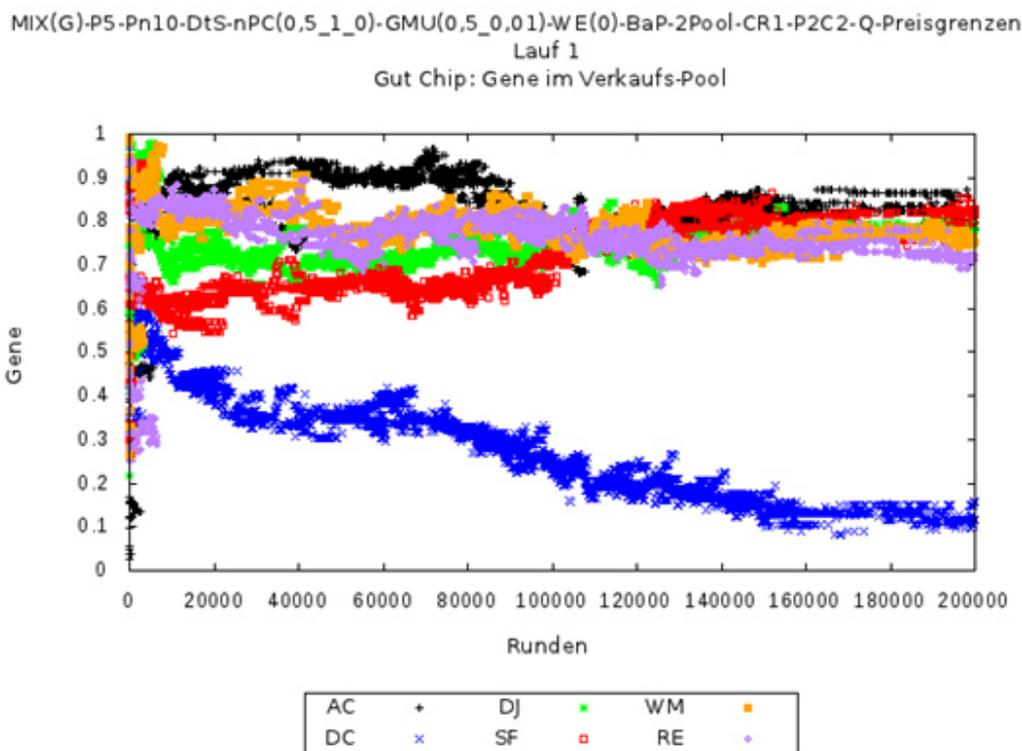


Abbildung 70: Entwicklung aller Gene im Verkaufspool der Chip-Produzenten

Bei den in diesem Abschnitt gezeigten Beispielen an Visualisierungsmöglichkeiten für Kennzahlen und Gene handelt es sich nur um einen kleinen Ausschnitt dessen, was anhand der in Abschnitt 7.4.1 kurz aufgezählten Visualisierungsskripte möglich ist. Weitere Beispiele finden sich im Evaluationsteil der Arbeit in Kapitel 8 sowie im Anhang.

Die skizzierten qualitativen Analysemethoden ergänzen die in Abschnitt 7.3 aufgezeigte quantitative Analysemethodik ausgezeichnet, da die dort notwendige und sinnvolle Verdichtung von Simulationsdaten zwar einerseits oft erst zu effizient interpretierbaren Ergebnissen führt, aber natürlich andererseits einen Verlust an Information zur Folge hat, so dass sich insbesondere Fragestellungen, die dynamische Entwicklungen betreffen, rein quantitativ nur eingeschränkt oder gar nicht beantworten lassen.

8 Evaluation

8.1 Einleitung und Übersicht

MACSIMA baut auf den konzeptionellen Ideen von Prof. Dr. Torsten Eymann [Eymann 2000; Eymann 2003; Eymann 2013] und dem von ihm implementierten AVALANCHE-System auf. MACSIMA bietet jedoch im Vergleich mit AVALANCHE eine Reihe von Erweiterungen, die im Evaluationsfokus dieses Kapitels liegen. Diese bestehen in

- der Fähigkeit der Agenten, aus gemachten Erfahrungen mit Hilfe komplexer Evolutionärer Algorithmen (EA) zu lernen, inklusive
 - der Realisierung des Konzepts des *2-Pool-separierten Lernens* (s. Kapitel 5.3), wodurch eine spezifisch auf Kauf- oder Verkaufsprozesse gerichtete Adaptierung der Agenten realisiert wird,
 - unterschiedlichen Varianten der neuartigen Selektions-Methode *Deterministic Minority Selection (DMS)* und
 - dem neu entwickelten *Highest-Fitness-First-Verfahren* sowie
- erweiterten sozialen Fähigkeiten zum gezielten Informationsaustausch bezüglich auf dem Markt gemachter Erfahrungen innerhalb von definierbaren Agentengruppen.

Die für das Evolutionäre Lernen vorhandenen und flexibel einstellbaren Parametrisierungsmöglichkeiten gehen über die Fähigkeiten von vorhergehenden Ansätzen (beispielsweise [De Wilde et al. 2003; Smith & Taylor 1998; Smith et al. 1999]) hinaus und werden hinsichtlich ihrer Effekte auf das Verhalten der Agenten und des Gesamtsystems detailliert untersucht.

Wesentliche Forschungsfrage war dabei, wie sich die in MACSIMA realisierten Erweiterungen der Ansätze von AVALANCHE auf die Selbstkoordinationsfähigkeiten des gesamten Wertschöpfungsnetzwerks und seine Performanz auswirken - sowohl isoliert betrachtet, als auch in ihrer Kombination.

Insbesondere wurde untersucht, ob eine Parametrisierung des Lernprozesses und des Informationsaustauschs existiert, von der – wenn sie von allen Agenten im Wertschöpfungsnetzwerk homogen genutzt wird – ein den Gesamtnutzen des Multiagentensystems maximierender Effekt erwartet werden kann.

Im Folgenden wird zunächst das Verhalten der Simulationssysteme AVALANCHE und MACSIMA qualitativ miteinander verglichen – mit dem Ergebnis, dass sich die Agenten in beiden Systemen relativ identisch verhalten, wenn die Fähigkeiten der MACSIMA-Geschäftsagenten auf die bereits in AVALANCHE verfügbaren limitiert sind.

Darauffolgend werden bei ausgeschalteter Lernfähigkeit der Agenten die Effekte der Variation der Startwerte von Genen betrachtet – sowohl isoliert voneinander, als auch in Kombination.

Anschließend wird der Untersuchungsfokus auf die Parametrisierung der Lern-Komponente der MACSIMA-Agenten gelegt. Die zur Verfügung stehenden genetischen Operatoren und

Einstellmöglichkeiten werden in umfangreichen Testreihen hinsichtlich ihrer Effekte auf die Entwicklung wesentlicher Performanz-Kennzahlen der Simulations-Szenarien evaluiert.

Danach wird analysiert, wie sich verschiedene Informationsaustausch-Modi auf diese Performanz-Kennzahlen des Gesamtsystems auswirken. Insbesondere wird dabei das Zusammenspiel der Informationsaustausch-Modi mit dem genetischen Operator der Mutation sowie verschiedenen Fitness-Bewertungsmethoden beleuchtet.

Hierbei wird auch eine Testreihe durchgeführt, bei der nur die Agenten auf der mittleren Stufe des Wertschöpfungsnetzwerks mit Lern- und Informationsaustauschfähigkeiten ausgestattet sind.

Das Kapitel schließt mit einer Testreihe, die die Gesamtperformanz des agentenbasierten Wertschöpfungsnetzwerks bei verschiedenen ausgewählten Parameter-Einstellungen miteinander vergleicht. Aus der Evaluation der Testreihen, die in Abschnitt 11.1 beschrieben sind, lassen sich unter anderem folgende Erkenntnisse ableiten:

- Internes Lernen mit evolutionären Algorithmen ohne die Nutzung von Informationen anderer Agenten erhöht die Selbstkoordinations-Leistung der Agenten und die Gesamtperformanz des Systems signifikant.
- Werden weitere MACSIMA-Erweiterungen verwendet, so ergeben sich weitere Leistungsverbesserungen.
- Erlaubt man den Agenten einen gruppenbezogenen Informationsaustausch innerhalb ihrer Wertschöpfungsstufe, so werden im Vergleich mit den in [Eymann 2000] untersuchten Parametrisierungen Leistungssteigerungen um das bis zu Zehnfache erreicht.
- Der Austausch von Informationen über Verkaufsabschlüsse zwischen den Agenten erhöht ihre Performanz, aber das richtige Maß ist entscheidend, damit es beim evolutionären Lernen nicht eventuell zu einem Overfitting und damit für die Agenten zu einem Verlust an Adaptionfähigkeit kommt.

Neben diesen Erkenntnissen wird anhand der Ergebnisse der Testreihen auch deutlich, dass Einstellungen bzgl. der verwendeten Fitness-Bewertungsfunktion und die Feinparametrisierung der Mutations-Methoden auch entscheidende Einflüsse auf die Selbstkoordinationsfähigkeit des Multiagenten-Systems haben.

Die Evaluationsergebnisse geben insgesamt wertvolle Hinweise darauf, was bei der Parametrisierung des Lernverfahrens, des Informationsaustauschs zwischen den Geschäftsagenten und sonstiger MACSIMA-Erweiterungen zu beachten ist, um innerhalb der Anwendungsdomäne zu einer wünschenswerten Systemperformanz zu gelangen.

8.1.1 Veröffentlichungen

Zu der im Rahmen dieser Dissertation geleisteten Forschungsarbeit sind auch verschiedene Forschungsergebnisse zu zählen, die mit einer Basis-Version des hier verwendeten MACSIMA- Systems erzielt wurden. Diese wurden in den Publikationen [Ruß & Walz 2009a; Ruß & Walz 2009b; Ruß & Walz 2009c; Ruß & Walz 2009d; Ruß & Walz 2012] veröffentlicht.

Die nachfolgend beschriebenen Ergebnisse wurden mit einer erweiterten Version des MACSIMA-Systems generiert und werden in dieser Arbeit erstmals vorgestellt.

8.2 Systemverhalten von AVALANCHE und MACSIMA

In [Eymann 2000] wurde mit dem AVALANCHE-System eine Reihe von Experimenten durchgeführt, um das Koordinationsverhalten von Agenten auf elektronischen Marktplätzen zu untersuchen. Ein detaillierter Überblick über den Inhalt und die Parameter-Einstellungen dieser Experimente wird in Abschnitt 11.1.1 gegeben. In den beschriebenen Untersuchungen wurden insbesondere die Startwerte der Gene AQ, DC und SF variiert – sowohl einzeln als auch parallel beziehungsweise in Kombination miteinander. Zudem wurde in [Eymann 2000] die Adaptionfähigkeit der Geschäftsagenten bei Verwendung des STDEA-Verfahrens beobachtet.

Im Unterschied zu Experimenten mit MACSIMA wurden die vorgenommenen AVALANCHE-Experimente in der Regel zeitbasiert durchgeführt, indem ausgehend vom Startzeitpunkt einer Simulation deren Laufzeit gemessen wurde.

Im Gegensatz dazu werden Testreihen mit MACSIMA in der Regel rundenbasiert durchgeführt, so dass die Simulationsdauer von MACSIMA-Szenarien nicht in Zeiteinheiten, sondern in einer Runden-Anzahl gemessen wird. Dies hat den Vorteil, dass die mit MACSIMA generierten Simulationsergebnisse weder durch die Leistungsfähigkeit des verwendeten Simulations-Servers noch durch die Teilung von dessen System-Ressourcen mit anderen Rechenprozessen signifikant beeinflusst werden.

Ein weiterer wichtiger Unterschied zwischen den Testergebnissen von AVALANCHE und MACSIMA besteht darin, dass die in [Eymann 2000] durchgeführten Evaluationen qualitativer Art waren. Durch die Ausstattung von MACSIMA mit der in Kapitel 6 beschriebenen Simulationsinfrastruktur und der in Kapitel 7 erläuterten Evaluationsmethodik werden mit MACSIMA vorgenommene Experimente in der Regel sowohl quantitativ als auch qualitativ analysiert.

Im Folgenden werden Diagramme, die in [Eymann 2000] der qualitativen Analyse von Simulationsergebnissen dienten, mit Diagrammen verglichen, die auf der Basis von MACSIMA-Simulationsergebnissen automatisiert generiert wurden.

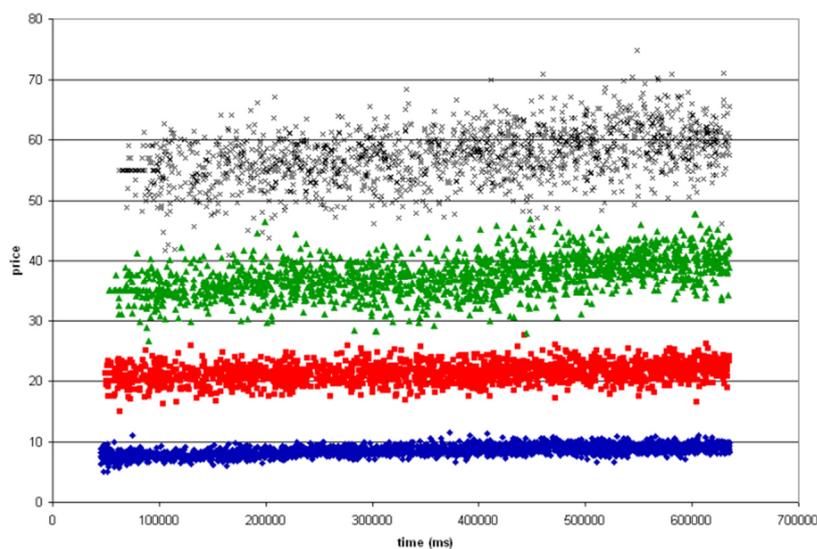


Abbildung 71: Verlauf des AVALANCHE-Experiments Eymann_1 mit festgelegten Startpreisen und homogener Strategieverteilung (ohne Lernen)⁷¹

⁷¹ Quelle: [Eymann 2000, S. 178], Abbildung 26

Dieser Vergleich hat den Zweck, anhand von einigen Parameter-Einstellungen aus [Eymann 2000] stichprobenhaft aufzuzeigen, dass sich die Geschäftsagenten in MACSIMA und AVALANCHE sehr ähnlich verhalten, wenn die von MACSIMA gebotenen Erweiterungen ihrer Fähigkeiten gemäß der Einstellungen in den Simulationsszenarien von den Agenten nicht genutzt werden.

Abbildung 71 ist [Eymann 2000, S. 179] entnommen und gehört zum Experiment Eymann_1 (siehe Tabelle 15 in Abschnitt 11.1.1). Das Diagramm zeigt, wie sich ausgehend von festgelegten Startpreisen für die Güter durch marktliche Selbstkoordination ein stabiles Preissystem bildet. Bei Verwendung der gleichen Einstellungen lassen sich in MACSIMA-Simulationen vergleichbare Entwicklungen beobachten, wie Abbildung 72 veranschaulicht.

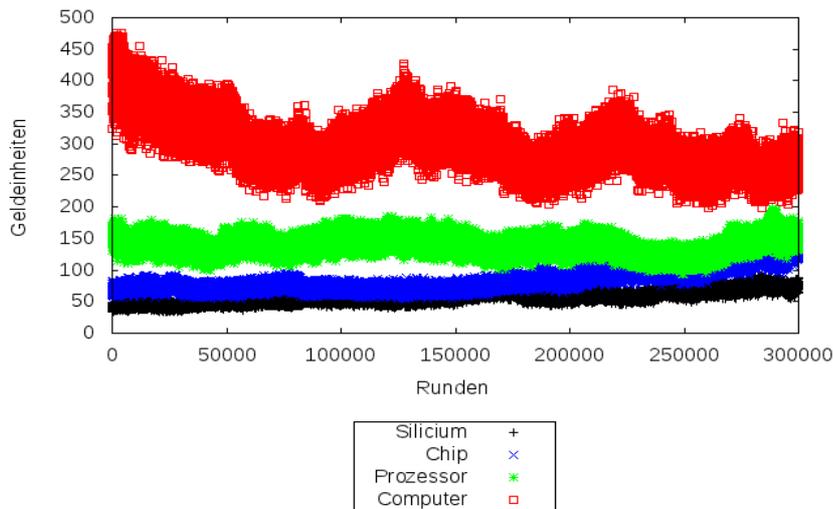


Abbildung 72: Verlauf des MACSIMA-Experiments mit den Einstellungen von Eymann_1 (ohne die Verwendung von MACSIMA-Erweiterungen)

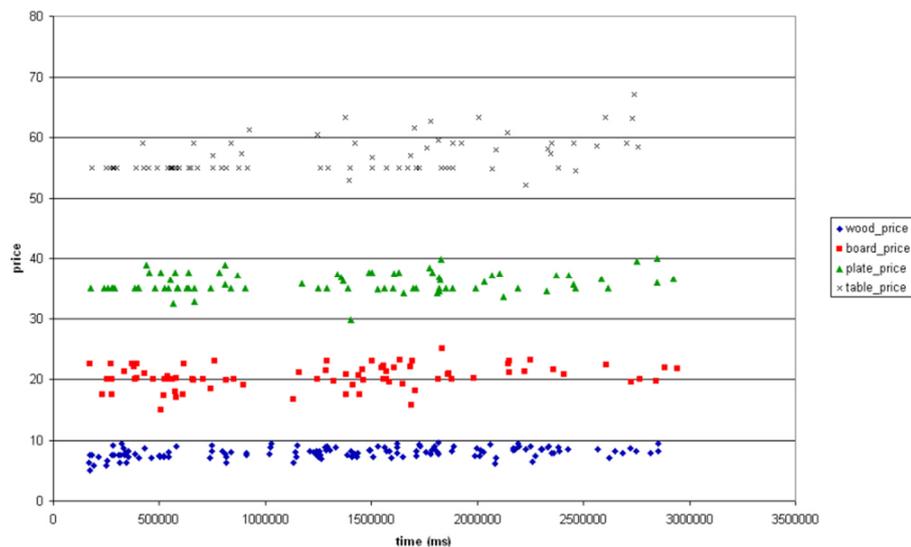


Abbildung 73: Preisentwicklung in AVALANCHE bei einem AQ-Wert von $0,9^{72}$

Abbildung 73 stammt aus [Eymann 2000, S. 182] und gehört zum Experiment Eymann_2 (siehe Tabelle 15 in Abschnitt 11.1.1). Sie verdeutlicht den Einfluss der Initialisierung des Werts für das Gen Acquisitiveness (AQ). Ist der AQ-Gen-Wert homogen für alle Agenten

⁷² Quelle: [Eymann 2000, S. 182], Abbildung 28

relativ hoch eingestellt (im dargestellten Simulationsverlauf war $AQ = 0,9$), so werden im Verlauf der Verhandlungen nur sehr wenige Konzessionen eingegangen.

Dies wirkt sich auf die Anzahl der Verkaufsabschlüsse beziehungsweise Transaktionen auf dem Gesamtmarkt negativ aus; es finden bei dem AQ-Wert von 0,9 nur 366 Transaktionen statt, wohingegen es bei einem AQ-Wert von 0,5 noch 7115 waren.

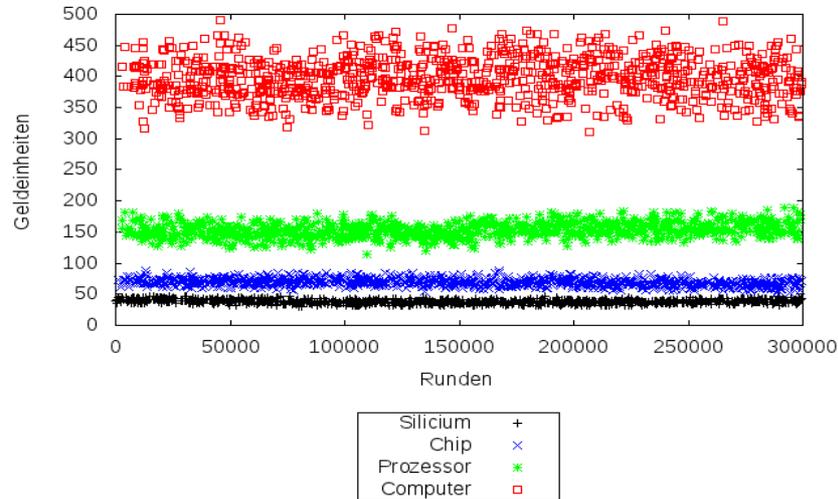


Abbildung 74: Verlauf des MACSIMA-Experiments mit den Einstellungen von Eymann_2 (ohne MACSIMA-Erweiterungen)

Eine von der Tendenz her gleiche Entwicklung zeigt sich bei Verwendung der Eymann_2-Einstellungen auch in MACSIMA-Simulationen, wie Abbildung 74 verdeutlicht, in der die Anzahl von Transaktionen (bei der Einstellungen $AQ = 0,9$) im Vergleich zu Abbildung 72 deutlich nachlässt.

Dieser Effekt setzt sich fort, wenn der AQ-Initialwert weiter erhöht wird, wie Abbildung 75 deutlich macht.

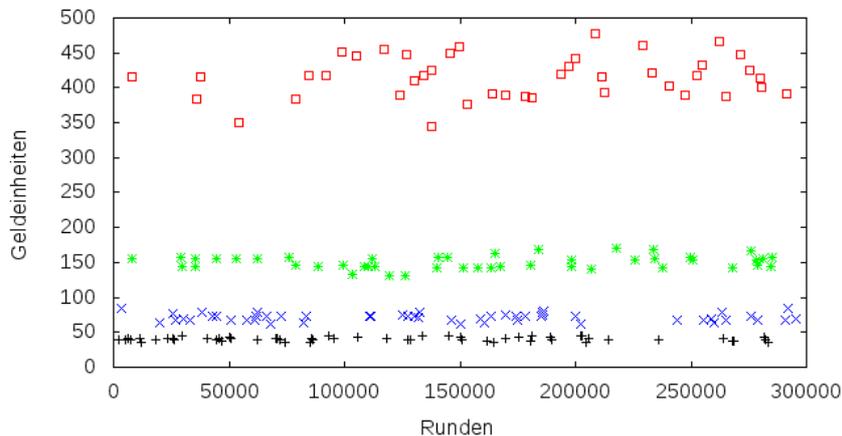


Abbildung 75: Verlauf des MACSIMA-Experiments mit $AQ = 0,95$ und den sonstigen Einstellungen von Eymann_2 (ohne MACSIMA-Erweiterungen)

Abbildung 76 zeigt Preisentwicklungen in AVALANCHE auf Basis der Parameter-Einstellung Eymann_3 (siehe Tabelle 15 in Abschnitt 11.1.1) und stammt aus [Eymann 2000, S. 183]. Die gezeigten Preisentwicklungen sind darauf zurückzuführen, dass die Zimmermann-Agenten, die auf derselben Wertschöpfungsstufe wie die Prozessor-Produzenten in MACSIMA stehen, mit einem höheren AQ-Startwert ($AQ = 0,6$) als die anderen Agenten-Typen (AQ homogen bei 0,4) initialisiert werden. Dadurch, dass die Agenten auf

den benachbarten Stufen mehr Konzessionen eingehen, schaffen es die Zimmermann-Agenten, sich eine relativ hohe Gewinnspanne zu sichern.

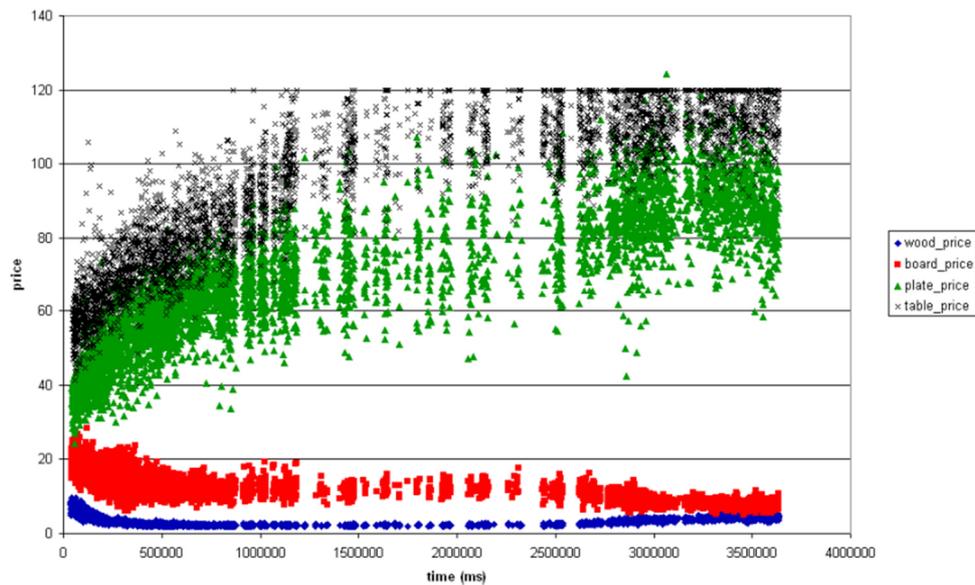


Abbildung 76: AVALANCHE-Preisentwicklung in einem Szenario mit hartnäckigen Zimmermann-Agenten⁷³ (mit einem höheren AQ-Wert als die übrigen Agenten)

Bei den gleichen Einstellungen zeigt sich in MACSIMA ein von der Tendenz her sehr ähnliches Agenten- und Systemverhalten, wie Abbildung 77 veranschaulicht. Den Agenten auf der dritten Stufe des Wertschöpfungsnetzwerks gelingt es genau wie in dem AVALANCHE-Experiment, den Verkaufspreis des von ihnen produzierten Guts zu steigern und gleichzeitig gegenüber den Verkäufern des von ihnen nachgefragten Guts sinkende Einkaufspreise durchzusetzen.

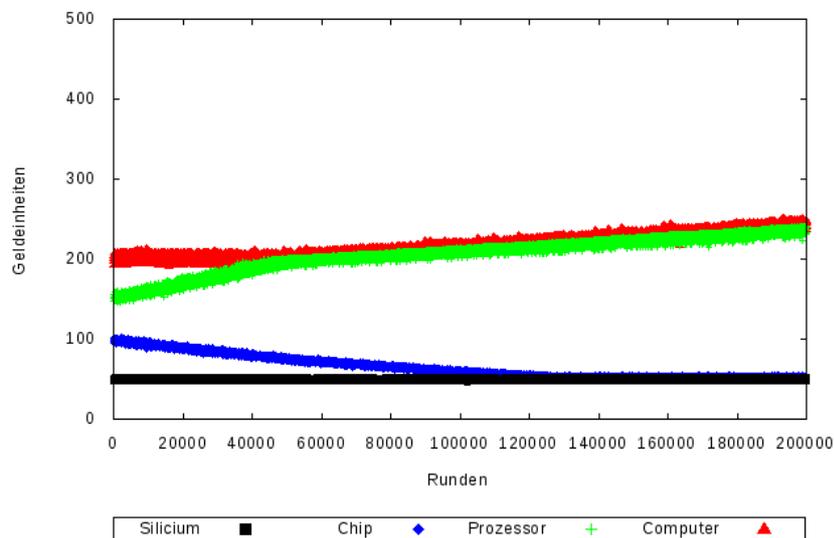


Abbildung 77: MACSIMA-Preisentwicklung mit hartnäckigen Prozessor-Produzenten

Bei einer genauen Beobachtung verschiedener Diagramme in [Eymann 2000], so zum Beispiel auch von Abbildung 76, fällt auf, dass Handlungspausen auftreten. Obwohl bei den mit AVALANCHE durchgeführten Simulationen in vielen Fällen der Zugriff auf Server-Ressourcen (wie CPU-Zeit und Speicher) dafür verantwortlich ist (siehe [Eymann 2000, S. 183]), kam

⁷³ Quelle: [Eymann 200, S.183], Abbildung 29

es bei MACSIMA-Simulationsläufen oftmals auch ohne Ressourcen-Mangel zu ähnlichen Effekten.

So konnten bei der Simulation von MACSIMA-Szenarien häufig Handelspausen beobachtet werden, die mehrere Tausend Simulationsrunden andauerten. Oft führte dies bei den betroffenen Szenarien, die häufig hohe AQ-Werte aufwiesen, dazu, dass die Geschäftsagenten zwar bis zur letzten im Experiment eingestellten Simulationsrunde versuchten, Transaktionen abzuschließen, dazu aber lediglich bis zur Hälfte der Dauer des Experiments in der Lage waren.

Eine Analyse der Gründe hierfür offenbarte, dass ein Zustand eintrat, in denen die Geschäftsagenten begonnene Verhandlungen sehr schnell abbrachen oder sie erst gar nicht mehr aufnahmen, weil sie mit ihren Preisvorstellungen relativ weit auseinander lagen. Tritt eine „Durststrecke“ auf, in der ein Agent wenige Verkaufsabschlüsse tätigen kann, so kann dies dazu führen, dass sich sein interner bzw. vermuteter Marktpreis für ein Gut von dessen tatsächlichem Marktpreis abkoppelt. Dann ist es sinnvoll, dass ein Agent über einen korrigierenden Mechanismus verfügt, der beispielsweise den von ihm für ein produziertes Gut geforderten Verkaufspreis nach einer bestimmten Anzahl erfolgloser Verhandlungen sukzessive senkt, bis der Agent wieder einen Verkaufserfolg erzielen kann. Ein entsprechender Mechanismus wurde mit dem sogenannten Preis-Automaten realisiert, der in Kapitel 5 beschrieben ist. Durch die Verwendung des Preis-Automaten kann die Anzahl auftretender Handelspausen sehr stark, das heißt auf ein in typischen MACSIMA-Diagrammen nicht mehr sichtbares Maß reduziert werden.

Insgesamt lässt sich anhand des qualitativen Vergleichs der Simulationslauf-Diagramme von AVALANCHE und MACSIMA erkennen, dass sich die Agenten in beiden Systemen sehr ähnlich bzw. von der Tendenz her auf die gleiche Art verhalten, wenn die Fähigkeiten der MACSIMA-Geschäftsagenten auf die bereits in AVALANCHE verfügbaren limitiert sind.

Da die Grundlagen des MACSIMA-Systems auf den von Prof. Dr. Torsten Eymann im AVALANCHE-System implementierten Ansätzen beruhen, war dies zwar so zu erwarten. Dennoch wurde der qualitative Vergleich beider Systeme mit großer Genauigkeit durchgeführt, um diese Erwartung zu bestätigen.

8.3 Variation der Gen-Startwerte

Die quantitative Evaluation des MACSIMA Systems beginnt mit Simulationen, bei denen die Lern-Komponente der Agenten ausgeschaltet ist. In diesem Fall startet jeder der Agenten mit Strategie-Einstellungen, die sich zur Laufzeit nicht verändern. Dies bedeutet, dass der jeweilige Agent nicht in der Lage ist, seine Verhandlungsstrategie auf der Basis seiner Erfahrungswerte zu adaptieren und zu optimieren und mit seiner gegebenen Verhandlungsstrategie einfach weiter macht.

Vor diesem Hintergrund ist die Auswahl von zueinander passenden Start-Einstellungen für die Gene beziehungsweise Strategie-Parameter für die Performanz des Gesamtsystems von entscheidender Bedeutung. Sind alle Agenten einer Wertschöpfungsstufe mit Gen-Einstellungen parametrisiert, die zu zahlreichen Verhandlungsabbrüchen führen, so verringert sich die Anzahl der Transaktionen und der Güterfluss innerhalb des Gesamtsystems wird gebremst, wodurch es innerhalb des Wertschöpfungsnetzwerks zu Umsatz- und Gewinneinbrüchen kommt.

Die entsprechenden Start-Einstellungen für die Gene beeinflussen die Leistungsfähigkeit des Wertschöpfungsnetzwerks aber auch dann, wenn die Agenten mit Lern- bzw. Adaptionsfähigkeiten ausgestattet sind. Dies rührt daher, dass die Agenten aus ihren Erfahrungen lernen,

welche insbesondere am Anfang der Simulation von den Start-Einstellungen der Gene bzw. Verhandlungsstrategien abhängen. Sind diese schlecht bzw. von einer optimalen oder zumindest guten Strategiekombination sehr verschiedenen, so werden die Agenten anfangs nur wenige positive Erfahrungen sammeln können, auf deren Basis ein effizientes Lernen und eine schnelle Adaption möglich ist.

Nachdem im Folgenden zunächst die Effekte der Variation der Startwerte einzelner Gene betrachtet werden, werden in Abschnitt 8.3.3 die Effekte von Startwert-Kombinationen untersucht.

8.3.1 Isolierte Variation des Gens Acquisitiveness (AQ)

Bereits in [Eymann 2000] wurden mehrere Tests zur Variation des Gens Acquisitiveness (AQ) durchgeführt. Es wurde gezeigt, dass das AQ-Gen ein wichtiges Gen ist und großen Einfluss auf das Verhandlungsergebnis hat.

Das folgende Experiment mit MACSIMA bestätigt dies. Seine Basis-Parametrisierung orientiert sich an den Einstellungen des AVALANCHE-Experiments Eymann_1 ($\langle AC = 0,5; DC = 0,25; DJ = 0,15; SF = 0,75; WM = 0,2; RP = 1,0 \rangle$), siehe Tabelle 15 in Abschnitt 11.1.1). Bis auf das Gen AQ werden alle Agenten homogen mit diesen statischen Gen-Startwert-Einstellungen parametrisiert.

In Abweichung zum Eymann_1-Test wird jedoch der Startwert des Gens AQ systematisch um jeweils 0,1 verändert. Dieses Vorgehen führt bei ausgeschalteter Lern-Funktionalität zu Simulationsergebnissen, die in Abbildung 78 dargestellt sind.

Die Grafik verdeutlicht, wie wichtig das Acquisitiveness-Gen ist. Ist sein Wert zu hoch, so sinkt die Anzahl der Verkaufsabschlüsse drastisch – bei gleichzeitigem starkem Anstieg der Zahl nicht erfolgreich abgeschlossener Verhandlungen. Bei einem Wert ab ca. 0,7 kommt es ohne die Verwendung des Preisautomaten (siehe Kapitel 5) immer häufiger zu Stockungen beziehungsweise Handlungspausen, die die Selbstkoordination des Wertschöpfungsnetzwerks so beeinträchtigen können, dass keine Transaktionen mehr stattfinden und der Güterfluss teilweise oder vollständig zum Erliegen kommt.

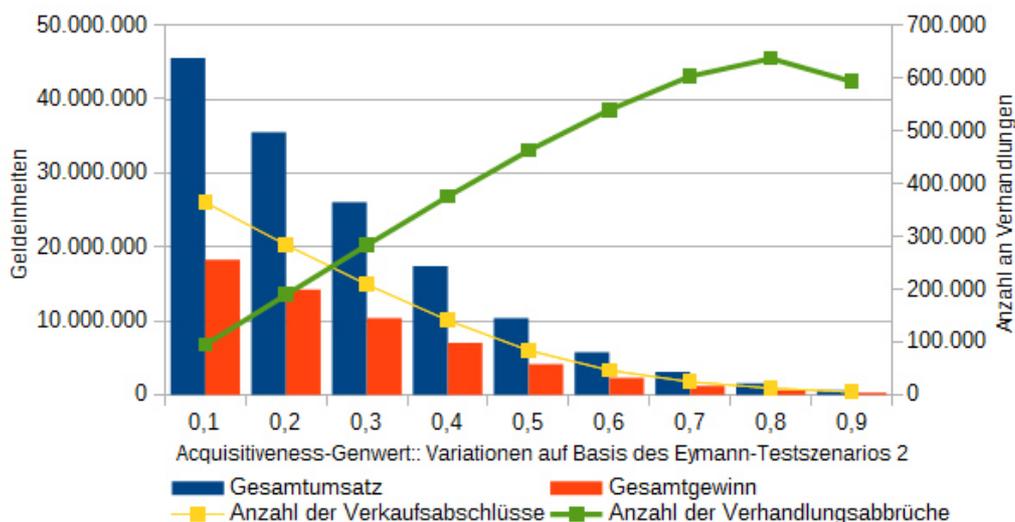


Abbildung 78: Effekte bei Variation des Startwerts des Gens Acquisitiveness

8.3.2 Isolierte Variation des Gens Satisfaction (SF)

Die folgende Abbildung basiert auf einem Test-Aufbau, der an dem Eymann_1-Test angelehnt ist. Statt wie Abbildung 78 den Startwert des Gens AQ zu verändern, wird dieser statisch mit 0,5 initialisiert und stattdessen der Startwert des Gens SF um jeweils 0,1 verändert. Die Lern-Komponente der Agenten war wie bei der isolierten Betrachtung des AQ-Gens im vorhergehenden Abschnitt ausgeschaltet.

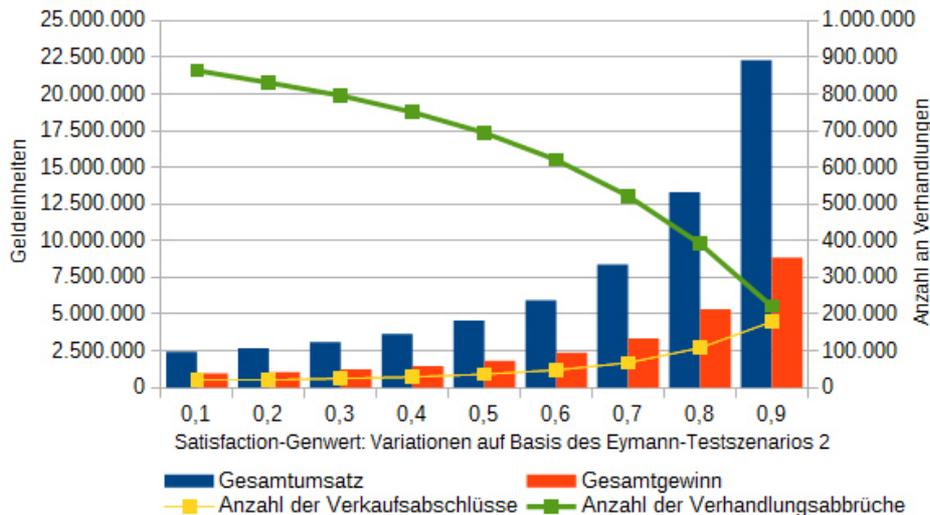


Abbildung 79: Effekte bei Variation des Startwerts des Gens Satisfaction

Bei einem niedrigen SF-Startwert sind alle Agenten unzufrieden, so dass sie häufiger Verhandlungen abbrechen. So kommen beispielsweise bei einem SF-Startwert von 0,1 50 Agenten in 200.000 Runden lediglich zu 20.000 Verkaufsabschlüssen. Ursache hierfür ist die hohe Zahl an Verhandlungsabbrüchen, die mit steigendem SF-Wert immer stärker zurückgeht. Umgekehrt steigt mit zunehmendem SF-Wert die Anzahl der Verkaufsabschlüsse.

Die beiden obigen Untersuchungen beziehen sich nur auf die isolierte Variation der Startwerte einzelner Gene, basierend auf den Parameter-Einstellungen des Eymann_1-Tests.

Insofern müssen entsprechende Variationen einzelner Gen-Werte auf Basis einer anderen Basis-Startwert-Parametrisierung nicht unbedingt die gleichen Ergebnisse erzielen. Insbesondere kann vermutet werden, dass die von den Genen induzierten Effekte von ihrer Kombination mit anderen Gen-Werten abhängen und daher nicht ausschließlich isoliert voneinander betrachtet werden sollten.

8.3.3 Variation von Startwert-Kombinationen

Um die Bandbreite der Wirkung variierender Gen-Startwert-Kombinationen auf Performanz-Kennzahlen wie Umsatz und Gewinn besser abschätzen zu können und ein Gefühl dafür zu entwickeln, welche Kombinationen an Gen-Startwerten zu bestimmten wünschenswerten Effekten führen, wird in diesem Abschnitt eine relativ große Stichprobe aus der Menge an möglichen Gen-Startwert-Einstellungen untersucht.

Für diese Stichprobe wurden unterschiedliche Szenarien definiert und simuliert, bei denen die Lern-Komponente der Agenten ausgeschaltet ist. Die untersuchte Testreihe wird in Abschnitt 11.1.2 erläutert. Die Wahl einer Stichprobe war notwendig, da es 59049 (9 hoch 5) Möglichkeiten an Startwert-Kombinationen gibt, wenn die Gene in 0,1-Schritten variiert werden.

Wie Abbildung 80 zeigt, konnte auf Basis der Stichprobe eine breite Abdeckung des Ergebnisraums für die Zielvariablen durchschnittlicher Gesamtumsatz, Anzahl an Verkaufsabschlüssen sowie Anzahl an Verhandlungsabbrüchen erreicht werden.

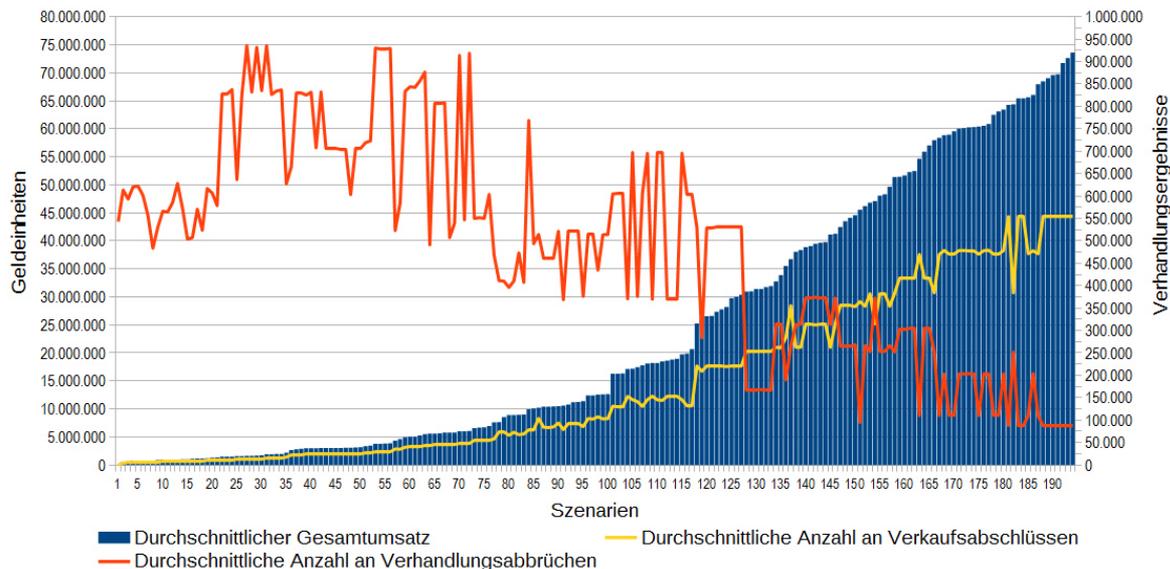


Abbildung 80: Verteilung von Gesamtumsatz, Verkaufsabschlüssen und Verhandlungsabbrüchen bei den durchgeführten Gen-Startwert-Tests

Die Analyse der Ergebnisse dieser umfangreichen Testreihe legt nahe, dass die in den AVALANCHE-Experimenten gewählten Gen-Parametrisierungen, die im Wertebereich

$$\langle AC = [0,05; 0,95]; DC = [0,05; 0,65]; DJ = 0,15; SF = 0,75; WM = 0,2; RP = 1,0 \rangle$$

liegen, im Hinblick auf eine hohe erwartbare Systemperformanz (bezüglich der in Abbildung 80 betrachteten Zielvariablen, aber auch weiterhin wie beispielsweise dem Gesamtgewinn oder dem Gewinnanteil am Umsatz) nicht ideal sind.

Stattdessen lassen die Ergebnisse darauf schließen, dass von Szenarien mit einer Parametrisierung der Gen-Startwerte, die im Bereich der in Abschnitt 11.1.5.2 erläuterten TOP-Randomisierung liegt, nämlich mit gleichverteilten Gen-Startwerten im Wertebereich

$$\langle AC = [0,1; 0,6]; DC = [0,4; 0,9]; DJ = [0,4; 0,9]; SF = [0,4; 0,9]; WM = [0,1; 0,6] \rangle,$$

eine besonders hohe Gesamtperformanz zu erwarten ist.

Hierbei ist zu bemerken, dass bei der TOP-Randomisierung in Abschnitt 11.1.5.2 für das Gen RP eigentlich eine Gleichverteilung innerhalb des Wertebereichs $[0,05; 0,95]$ angegeben wird. Dieses Gen wird aber bei der gegenwärtigen Implementierung nicht genutzt, sondern dient als Platzhalter für zukünftige konzeptionelle Erweiterungen, beispielsweise zur Berücksichtigung von Wissen über die Vertrauenswürdigkeit von Verhandlungspartnern oder von bestimmten bei Auktionen relevanten Aspekten. Daher ist die genaue Art der Initialisierung des Gens RP für die folgenden Testreihen nicht von Relevanz.

8.4 Parametrisierung der Lern-Komponente

In diesem Abschnitt werden verschiedene, die Lern-Komponente der MACSIMA-Agenten betreffende Einstellmöglichkeiten miteinander verglichen, um empfehlenswerte und weniger empfehlenswerte Einstellungen voneinander zu differenzieren.

Da die Menge der möglichen Kombinationen an Parameter-Einstellungen auch für die Lern-Komponente kombinatorisch explodiert, kann das Ziel der Untersuchung nicht darin bestehen, eine optimale Parametrisierung des evolutionären Lernens zu finden. Neben der kombinatorischen Explosion kommt noch erschwerend hinzu, dass sich die genetischen Operatoren mit ihren Einstellungen gegenseitig beeinflussen und die von anderen Operatoren bewirkten Effekte sowohl hemmen als auch vergrößern können (vgl. [Weicker 2015, S. 116 f.]).

Dennoch verfolgen die durchgeführten Experimente und Analysen den Zweck, zu untersuchen, welche Einstellungen innerhalb der Anwendungsdomäne zu einem mehr oder weniger wünschenswerten Agenten- und Systemverhalten führen. Anhand der gewonnenen Erkenntnisse werden grundsätzliche Empfehlungen hinsichtlich der Lern-Parametrisierung ausgesprochen und Hinweise gegeben, was bei den Lern-Einstellungen besonders zu beachten ist.

Die durchgeführten Testreihen legen ein besonderes Gewicht auf die Parametrisierung des *2-Pool-separierten Lernens* (siehe Abschnitt 5.4.1.2).

8.4.1 Effekte der Relation von Pool und Population

Im Folgenden werden für die Pool-Größen P3, P5, P7 und P10 die sich aus ihrem Zusammenspiel mit unterschiedlichen Populations-Größen ergebenden Effekte auf die Gesamtporformanz eines agentenbasierten Wertschöpfungsnetzwerks untersucht. Die Details zur durchgeführten Testreihe werden in Abschnitt 11.1.3 beschrieben.

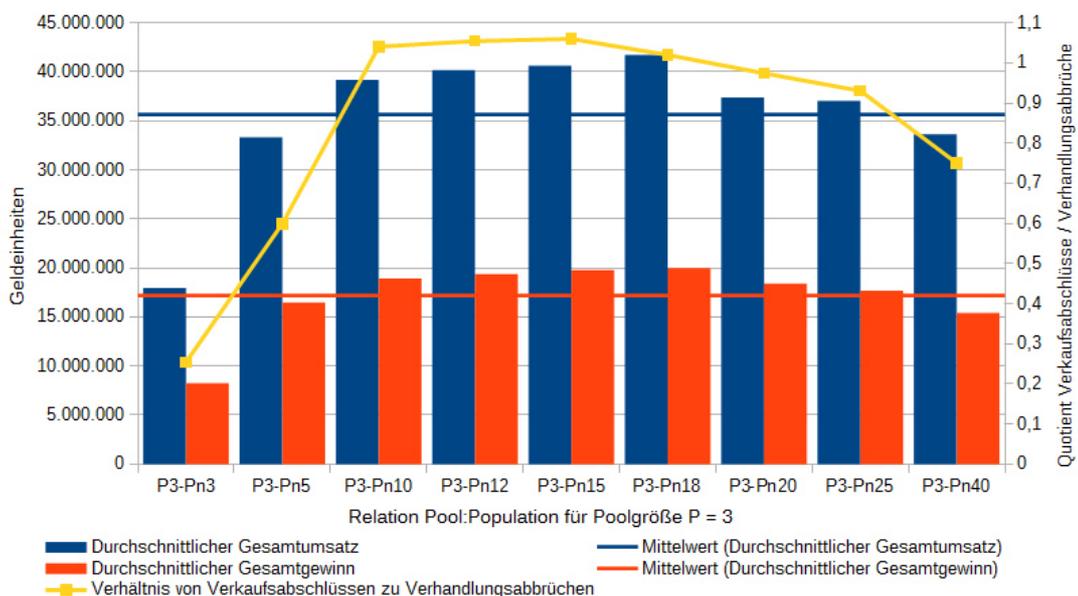


Abbildung 81: P:Pn-Relationstests für P = 3

In Abbildung 81 erkennt man, dass bei einer festen Poolgröße von 3 eine anwachsende Populationsgröße bis zu einem Grenzwert von 18 zu besseren Umsätzen und Gewinnen führt. Danach fällt die Performanz des Gesamtsystems in Bezug auf den durchschnittlich erreichbaren Gesamtumsatz und -gewinn wieder ab.

Während das diesbezüglich beste Ergebnis bei P3-Pn18, also einer P:Pn-Relation von 1:6 erreicht wird, wird das beste Verhältnis zwischen Verkaufsabschlüssen und Verhandlungsabbrüchen bereits bei der Einstellung P3Pn15 erzielt, wobei dieses Verhältnis auch bei der Einstellung P3Pn18 noch sehr gut ist.

Dieses Ergebnis lässt sich dadurch erklären, dass mit einer anwachsenden Populations-Größe immer mehr Plumages gespeichert werden, so dass das sich anschließende Evolutionäre Lernverfahren aus einer größeren Menge an Gen-Material selektieren kann. Durch dieses Mehr an Eingabe steigt die Leistungsfähigkeit des evolutionären Lernens beziehungsweise die Qualität seiner Ausgabe bis zu einer bestimmten Grenze an.

Ab dieser bringt noch mehr an Eingabe-Information in Bezug auf die Ausgabe weniger an Qualitäts-Zugewinn, als durch den selteneren Start des Lernverfahrens gleichzeitig verloren geht. Denn das Lernverfahren wird nur gestartet, wenn die Population vollständig gefüllt ist, also bei einer Populations-Größe von Pn40 normalerweise deutlich seltener als beispielsweise bei einer Größe von Pn10.

Bei einer Pool-Größe von P3 befinden sich im Verhandlungs-Pool immer drei Genotypen beziehungsweise Verhandlungsstrategien, die in den bilateralen Verhandlungen eingesetzt werden. Erhöht man die Poolgröße auf P5, so werden immer fünf Genotypen in den Verhandlungen verwendet.

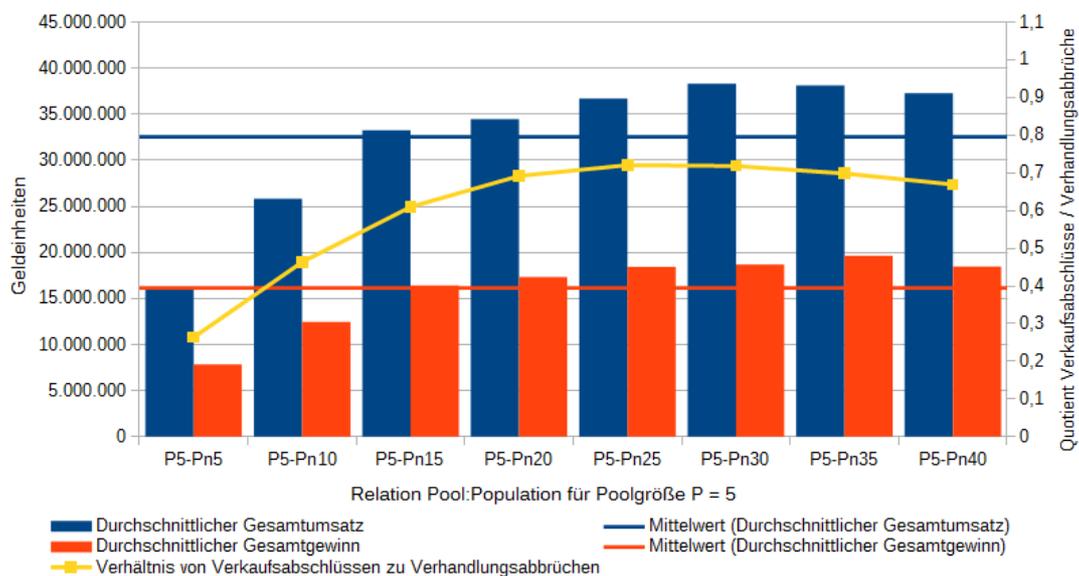


Abbildung 82: P:Pn-Relationstests für P = 5

Gemäß Abbildung 82 liegt für die Pool-Größe P5 das Optimum, was den erzielbaren durchschnittlichen Gesamtumsatz und das Verhältnis von Verkaufsabschlüssen zu Verhandlungsabbrüchen angeht, bei der Einstellung P5Pn30. In Bezug auf den durchschnittlichen Gesamtgewinn liegt das Optimum hingegen bei P5-Pn35.

Die Abbildungen zeigen, dass bereits eine kleine Vergrößerung des Pools von P3 auf P5 fast eine Verdopplung der optimalen Populations-Größe bewirkt. Wie bei P3 liegt auch im P5-Fall ein optimales P:Pn-Verhältnis im Bereich von 1:6 oder 1:7.

Bei einer Pool-Größe von P7 liegt das optimale P:Pn-Verhältnis auch im Bereich von 1:6 oder 1:7, wie Abbildung 83 zeigt. Das gleiche optimale Verhältnis ist auch für die Pool-Größe von P10 zu beobachten (siehe Abbildung 84).

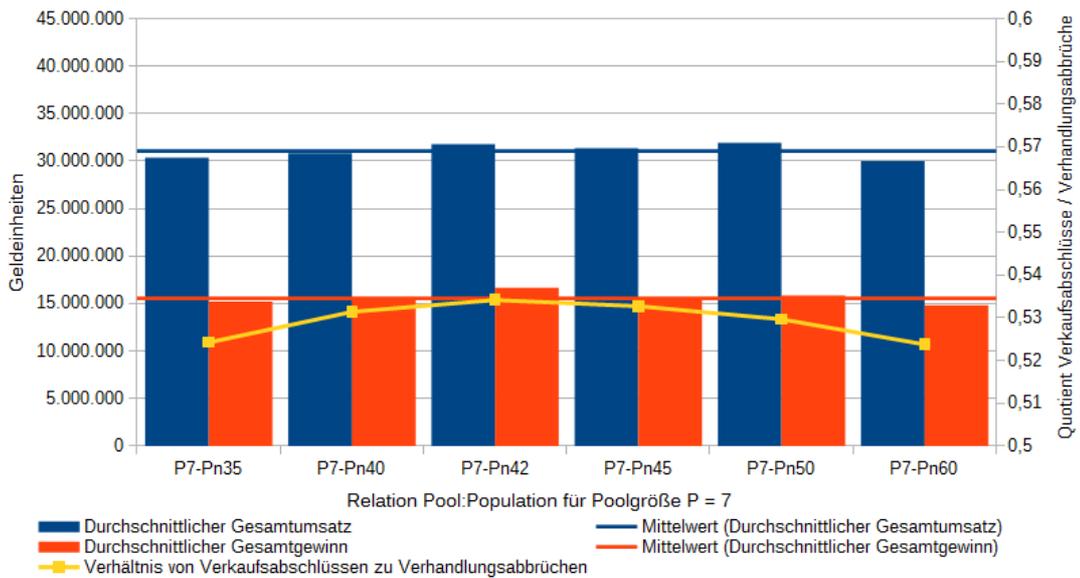


Abbildung 83: P:Pn-Relationstests für P = 7

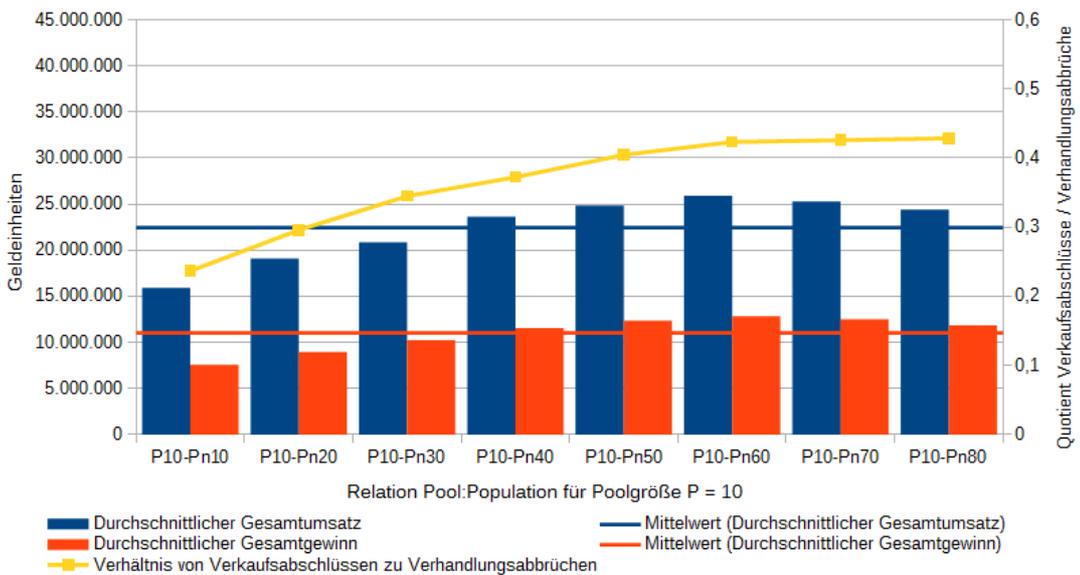


Abbildung 84: P:Pn-Relationstests für P = 10

Insgesamt zeigt die Testreihe, dass eine P:Pn-Relation von 1:6 oder 1:7 über alle betrachteten Pool-Größen hinweg hinsichtlich der beobachteten Ergebnisvariablen vorteilhaft ist. Daher werden nachfolgende Testreihen jeweils mit einer dieser P:Pn-Einstellungen durchgeführt.

Mit einem durchschnittlichen Gesamtumsatz von circa 32,5 Millionen Geldeinheiten konnten die P5-Szenarien aus Abbildung 82 die P7-Szenarien aus Abbildung 83 und die P10-Szenarien aus Abbildung 84 deutlich übertreffen. Mit einem durchschnittlichen Gesamtumsatz von knapp über 35 Millionen Geldeinheiten konnten die P3-Szenarien aus Abbildung 81 noch etwas besser abschneiden.

In den folgenden Testreihen werden die Effekte von genetischen Operatoren untersucht. Da deren Arbeit von der Größe von Pool P und Population Pn beeinflusst wird, erscheint es sinnvoll, in den folgenden Testreihen eher mittlere Größen für P und Pn zu betrachten.

Aus diesem Grund werden in den weiteren Testreihen die P:Pn-Einstellungen P5Pn30 und P5Pn35 verwendet.

8.4.2 Effekte der Fitness-Methoden

Im Folgenden werden die Effekte der Fitness-Methoden PmA, RMF, BaP, MdP und AvP auf Ergebnisvariablen wie den durchschnittlichen Gesamtumsatz und –gewinn sowie das Verhältnis von Verkaufsabschlüssen zu Verhandlungsabbrüchen analysiert. Die Details zur durchgeführten Testreihe werden in Abschnitt 11.1.4 beschrieben. Alle Simulationen der Testreihe wurden sowohl mit den P:Pn-Einstellungen P5Pn30 als auch P5Pn35 durchgeführt.

In Bezug auf die Effekte auf die Ergebnisvariablen fallen die Ergebnisse sowohl bei P5Pn30 (siehe Abbildung 85) als auch P5Pn35 (siehe Abbildung 86) fast identisch aus. In beiden Fällen haben die Fitness-Methoden MdP und AvP den größten positiven Effekt. Bei der Einstellung P5Pn35 führt die Methode AvP zum besten Systemergebnis, bei der Einstellung P5Pn30 ist AvP der Methode MdP leicht unterlegen.

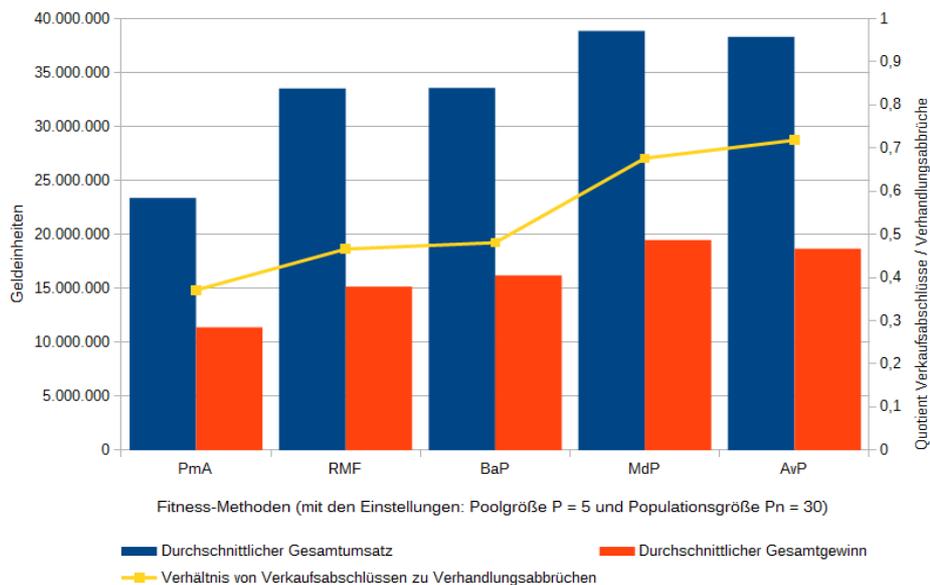


Abbildung 85: Fitness-Methoden-Ergebnisse für Pool- und Populations-Größe P5Pn30

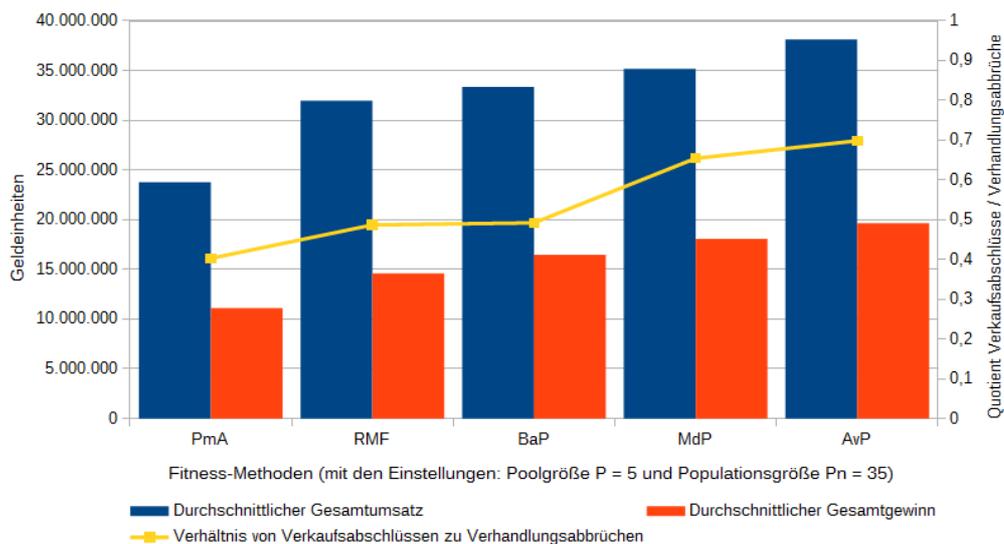


Abbildung 86: Fitness-Methoden-Ergebnisse für Pool- und Populations-Größe P5Pn35

Deutlich am schlechtesten wirkt sich die Verwendung der Fitness-Methode PmA aus. Die Methoden RMF und BaP liegen in beiden Abbildungen fast gleichauf, wobei BaP in beiden Fällen zu einer leicht besseren Performanz des Gesamtsystems führt.

Innerhalb der Testreihe waren die erzielten durchschnittlichen Gesamtumsätze aller Fitness-Methoden bei der Einstellung P5Pn30 höher als bei der P5Pn35-Parametrisierung. Bei beiden P:Pn-Einstellungen wurde das jeweils beste Verhältnis von Verkaufsabschlüssen zu Verhandlungsabbrüchen bei der Verwendung der AvP-Methode erreicht.

Insgesamt lässt sich festhalten, dass für die Anwendungsdomäne die Verwendung der Fitness-Methoden MdP und AvP empfohlen werden kann, insbesondere in Kombination mit der P:Pn-Einstellung P5Pn30.

Folgende Untersuchungen werden daher bevorzugt anhand von Testreihen durchgeführt, deren Szenarien-Definitionen diese Parameter-Einstellungen beinhalten.

8.4.3 Effekte der Selektions-Methoden

Alle Szenarien der durchgeführten Testreihe beinhalten 2-Pool-separiertes Lernen, die Initialisierung von 50 Agenten und bauen auf dem Szenario-Grundmuster

MIX(G)-P5-Pn30-[Selektion]-nPC(0,5-1-0)-GMU(0,5-0,01)-[Elitismus]-AvP-[Startwerte]

auf. Die Testreihe ist in Abschnitt 11.1.5 ausführlich beschrieben.

Im Rahmen der Testreihe wurden die Effekte der acht Selektions-Methoden BiC, DMS, DMS2, DMS3, DtA, DtS, DWS und RWS auf den durchschnittlichen Gesamtumsatz des Wertschöpfungsnetzwerks untersucht.

Die innerhalb der Testreihe verwendeten Randomisierungs-Einstellungen sind in Abschnitt 11.1.5.2 beschrieben und ergaben sich durch die Überprüfung von Gemeinsamkeiten zwischen Simulationsergebnissen aus der Testreihe zu Startwert-Effekten (siehe 11.1.2), die bzgl. des erwirtschafteten Umsatzes in TOP-, MID- und LOW-Klassen eingeordnet werden konnten. Dass die Randomisierung passend vorgenommen wurde, ist aus Abbildung 88 und Abbildung 89 ersichtlich, da die Szenarien mit TOP-Randomisierung bei jeder Selektions-Methode den größten Umsatzanteil erzielen konnten.

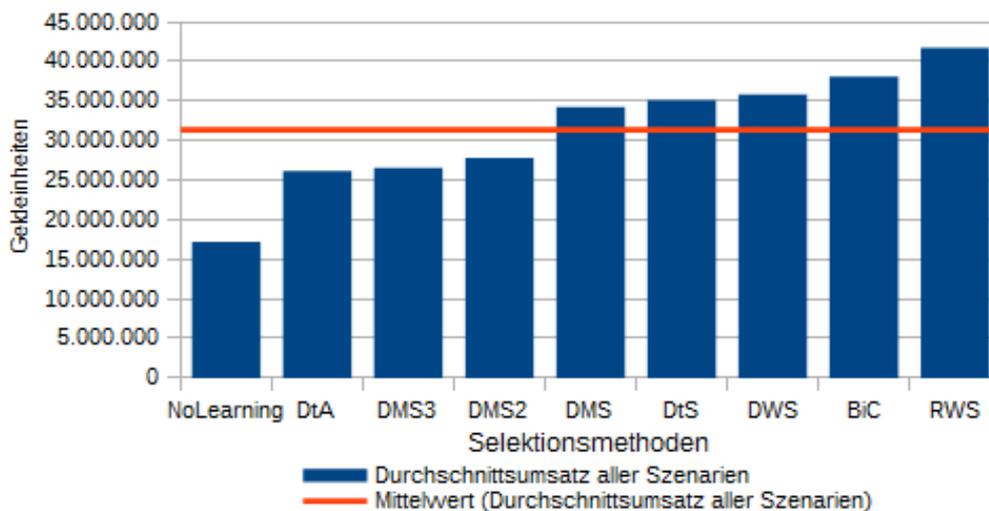


Abbildung 87: Selektions-Methoden-Testergebnisse 1

Um abzusichern, dass das Evolutionäre Lernen vorteilhaft ist, wurden in der Testreihe auch Simulationen durchgeführt, bei denen das Lernen ausgeschaltet war.

Die entsprechenden Szenarien mit NoLearning-Einstellung schneiden signifikant schlechter ab als alle anderen Szenarien mit eingeschalteter Lern-Komponente und Selektion, wenn der Durchschnitt aus allen Szenarien (das heißt den LOW-, MID und TOP-Performance-Szenarien) gebildet wird (vergleiche Abbildung 87).

Lediglich wenn sie mit TOP-Randomisierungs-Einstellungen starten, können Szenarien mit NoLearning-Einstellung einen relativ guten Durchschnittsumsatz erzielen. Dies macht die Bedeutung der Gen-Initialisierungen zum Systemstart noch einmal deutlich. Bei einer über die gesamte Agenten-Population hinweg geeigneten Parametrisierung der Gen-Startwerte kann das Gesamtsystem auch ohne eine weitere Adaption gute Ergebnisse erzielen.

Auf einem realen elektronischen Markt (siehe Abschnitt 3.3.3) sowie weiteren behandelten Einsatzszenarien (siehe Abschnitt 4.2) ist aber davon auszugehen, dass es immer eine Anzahl an Teilnehmern beziehungsweise Agenten geben wird, die sich eigennützig verhalten und versuchen, strategische Vorteile zu erzielen. Dieser Umstand und weitere möglicherweise störende Umweltcharakteristika (vergleiche 2.3.3) und -einflüsse führen dazu, dass NoLearning-Szenarien unter realen Bedingungen wahrscheinlich noch deutlich schlechter abschneiden würden, als in dieser Testreihe.

Neben der NoLearning-Einstellung schneiden auch die Selektions-Methoden DtA, DMS2 und DMS3 insgesamt unterdurchschnittlich ab. Insbesondere zeigt sich, dass eine Generierung eines Durchschnitts-Genotypen, wie ihn das Verfahren DtA erstellt, nicht zu besseren Ergebnissen führt. Dies liegt wahrscheinlich daran, dass durch die Durchschnittsbildung Abhängigkeiten zwischen den Genen ignoriert werden.

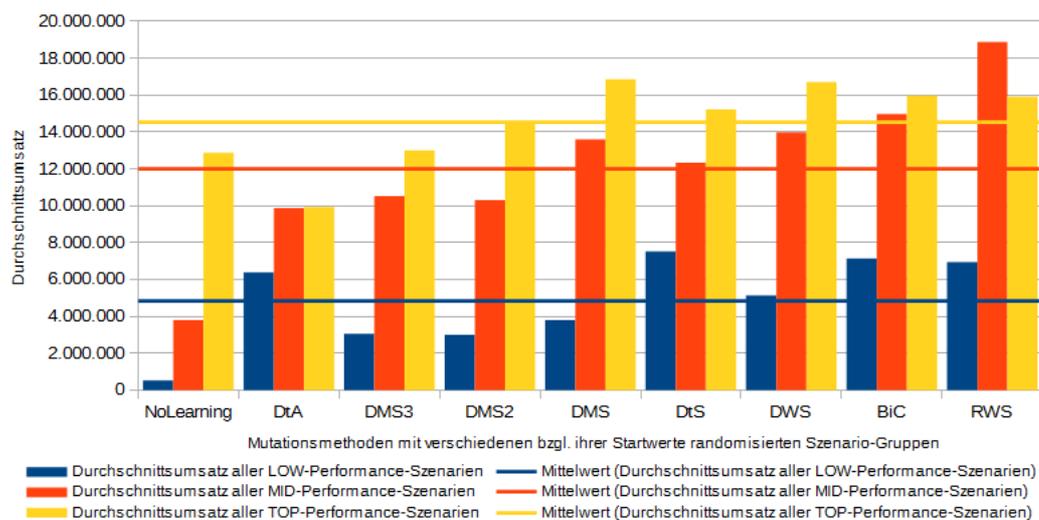


Abbildung 88: Selektions-Methoden-Testergebnisse 2

Die neu entwickelte DMS-Methode (siehe Abschnitt 5.4.3.2) kann sich hinsichtlich des unter ihrer Verwendung durchschnittlich erzielten System-Umsatzes genau in der Mitte der getesteten Einstellungen platzieren. Auffällig ist jedoch, dass es den Agenten bei ihrer Verwendung in Szenarien mit TOP-Parametrisierung gelingt, den insgesamt höchsten Durchschnittsumsatz zu erzielen. Dies weist darauf hin, dass DMS von allen Selektions-Methoden am besten dazu geeignet ist, eine für den Adaptionsprozess vorteilhafte genetische Vielfalt im Gen-Material zu erhalten.

Abbildung 88 und Abbildung 89 zeigen, dass gerade Systeme, die mit einer LOW- oder MID-Gen-Parametrisierung starten, vom Lernen und einer geschickten Wahl der Selektions-Methode profitieren.

Durch ihre Lernfähigkeit können die Agenten ihre Gen-Werte beziehungsweise Verhandlungsstrategien adaptieren und so auch aus einer schlechten Ausgangslage heraus Umsätze erzielen, die deutlich über das hinausgehen, was ohne Lernen oder beim Lernen unter Verwendung einer ungünstigen Selektions-Methode möglich ist.

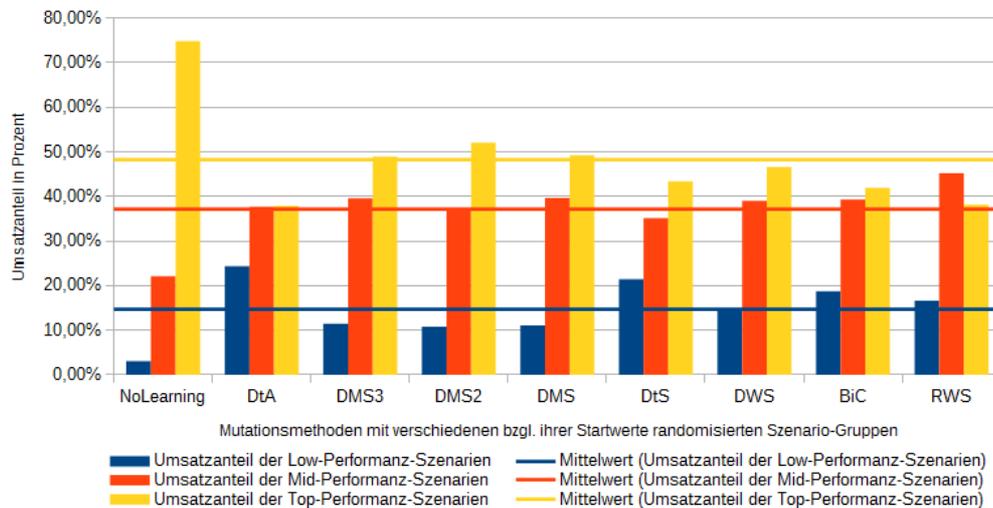


Abbildung 89: Selektions-Methoden-Testergebnisse 3

Zusammenfassend zeigen die Ergebnisse der Testreihe, dass die Selektions-Methoden DMS, DtS, DWS, BiC und RWS empfehlenswert sind. Die das RWS-Verfahren verwendenden Szenarien generieren dabei den höchsten Durchschnittsumsatz.

Aus diesem Grund werden beispielsweise die Untersuchungen zu den Effekten von Ersetzungsschema und Mutation anhand von Testreihen durchgeführt, deren Szenarien die Selektions-Methode RWS verwenden (siehe Abschnitte 11.1.8 und 11.1.9).

8.4.4 Effekte der Rekombinations-Einstellungen

Alle Szenarien der durchgeführten Testreihe beinhalten 2-Pool-separiertes Lernen und die Initialisierung von 50 Agenten.

Der Test der Rekombinations-Methoden nPC und RaC wurde in Kombination mit der Fitness-Methode AvP und den fünf Selektions-Methoden BiC, DMS, DtS, DWS und RWS vorgenommen, die in den vorhergehenden Abschnitten empfohlen wurde.

Die Kombinationen mit mehreren Selektions-Methoden wurden getestet, da sich die genetischen Operatoren mit ihren Einstellungen gegenseitig beeinflussen und die von anderen Operatoren bewirkten Effekte sowohl hemmen als auch vergrößern können (vgl. [Weicker 2015, S. 116 f.]). Die genaue Testreihe ist in Abschnitt 11.1.6 beschrieben.

Abbildung 90 zeigt die Durchschnitte der untersuchten Ergebnisvariablen, die aus allen Simulationsergebnissen gebildet werden, bei denen die jeweilige Rekombinations-Einstellung verwendet wird. In der Abbildung ist die Reihenfolge der verschiedenen Einstellungen nach ihrem Umsatz geordnet.

Es ist ersichtlich, dass die RaC-Einstellungen zu den beiden schlechtesten Ergebnissen führen. Die Ergebnisse der verschiedenen nPC-Einstellungen liegen, was die Werte ihrer Umsatz- und Gewinn-Ergebnisvariablen angeht, relativ nahe beieinander.

Bezieht man in die Analyse die Anzahl der von allen Agenten generierten Verkaufsabschlüsse in die Betrachtung mit ein, so erscheinen die Einstellungen nPC(1-2-X) und nPC(1-1-X) be-

sonders empfehlenswert, da sie nicht nur zu einer überdurchschnittlich hohen Anzahl an Verkaufsabschlüssen, sondern auch zu überdurchschnittlichen Gesamtumsätzen und –gewinnen führen.

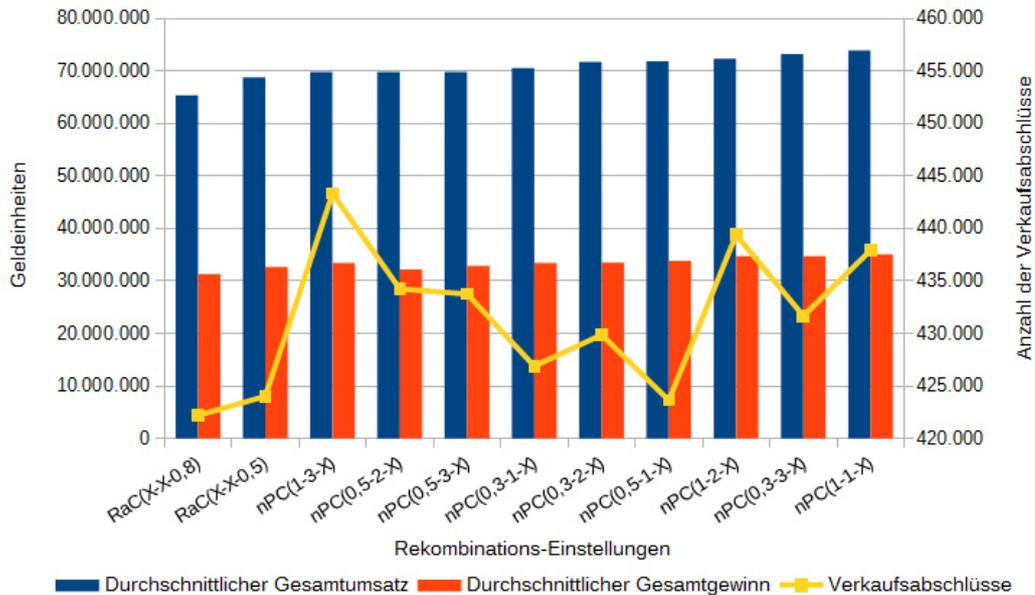


Abbildung 90: Effekte der Rekombination auf Gesamtumsatz und -gewinn sowie die Anzahl an Verkaufsabschlüssen

Abbildung 91 veranschaulicht, wie sich die getesteten Rekombinations-Einstellungen auf die Anzahl von Verkaufsabschlüssen, Verhandlungsabbrüchen und deren Verhältnis auswirken.

Die Ergebnisse der verschiedenen Einstellungen sind dabei gemäß ihres Effekts auf das Verhältnis von Verkaufsabschlüssen und Verhandlungsabbrüchen ansteigend angeordnet. Auch aus dieser Perspektive betrachtet, gehören die Rekombinations-Einstellungen nPC(1-2-X) und nPC(1-1-X) zu den drei besten getesteten Parametrisierungen.

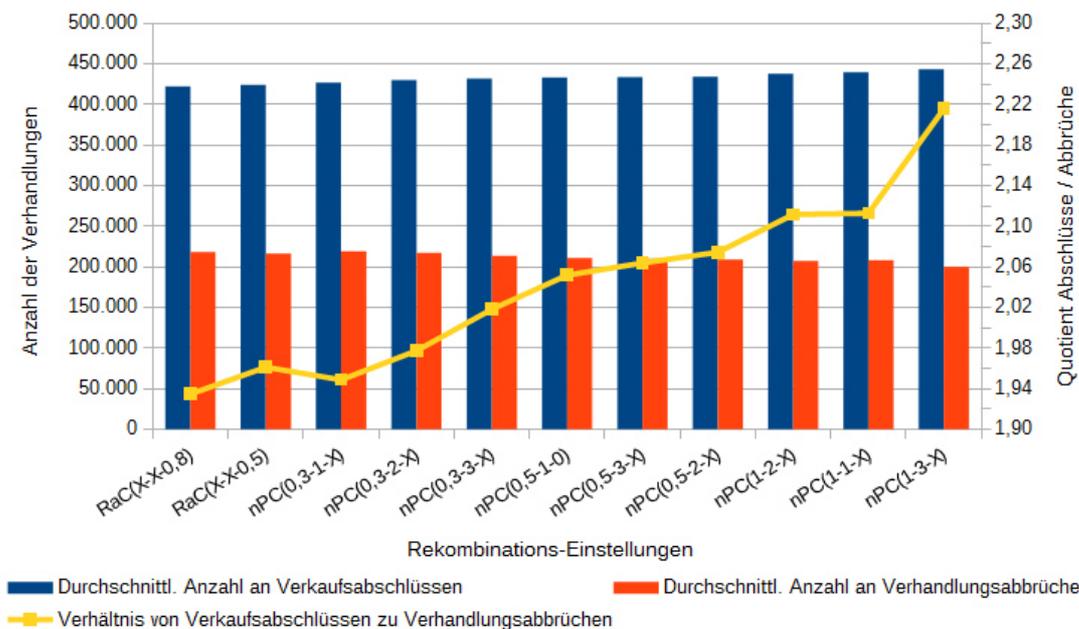


Abbildung 91: Effekte der Rekombination auf die Verhandlungsergebnis-Kennzahlen

In Abbildung 92 und Abbildung 93 werden die beiden schlechtesten Rekombinations-Einstellungen (RaC(X-X-0,8) und RaC(X-X-0,5)) den beiden empfehlenswertesten nPC-Einstellungen (nPC(1-2-X) und nPC(1-1-X)) in Bezug auf verschiedene Performanz-Kennzahlen direkt gegenübergestellt.

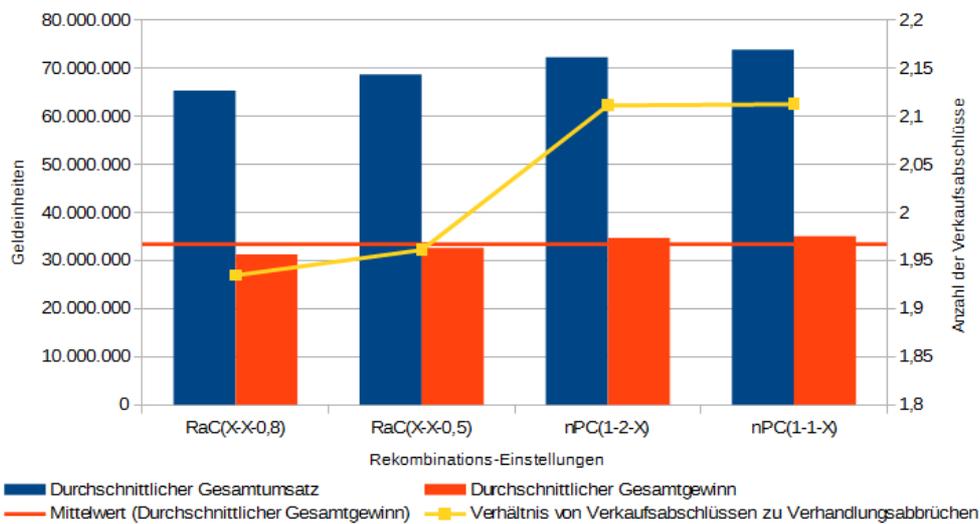


Abbildung 92: Effekte der besten und schlechtesten Rekombination auf Gesamtumsatz und -gewinn u. das Verhältnis der Verhandlungsergebnis-Kennzahlen

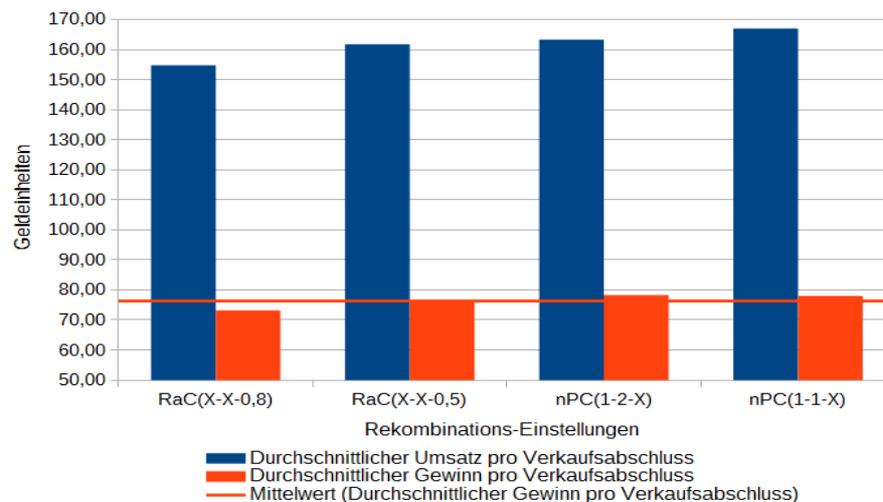


Abbildung 93: Effekte der besten und schlechtesten Rekombination auf den durchschnittlichen Umsatz und Gewinn pro Transaktion

Abbildung 94 stellt dar, wie sich die schlechteste und die empfehlenswerteste der untersuchten Rekombinations-Einstellungen, nämlich RaC(X-X-0,8) und nPC(1-1-X) in Kombination mit den in Abschnitt 8.4.3 empfohlenen fünf Selektions-Methoden auf die Verhandlungsergebnis-Kennzahlen auswirken.

Obwohl die Verwendung von RaC(X-X-0,8) in Kombination mit den meisten Selektionsmethoden zu eher schlechten Ergebnissen führt, ist diese Einstellung in Kombination mit RWS dazu in der Lage, das Ergebnis mit dem höchsten Umsatz und dem besten Verhältnis von Abschlüssen zu Abbrüchen zu bewirken.

Dieses unerwartete Ergebnis bestätigt die oben genannte Erkenntnis aus [Weicker 2015, S. 116 f.], dass sich die genetischen Operatoren mit ihren Einstellungen gegenseitig beeinflussen. Dieses Ergebnis bestätigt auch die Sinnhaftigkeit des aufwendigen Aufbaus der in 11.1.6 beschriebenen Testreihe, auf der dieser Abschnitt basiert.

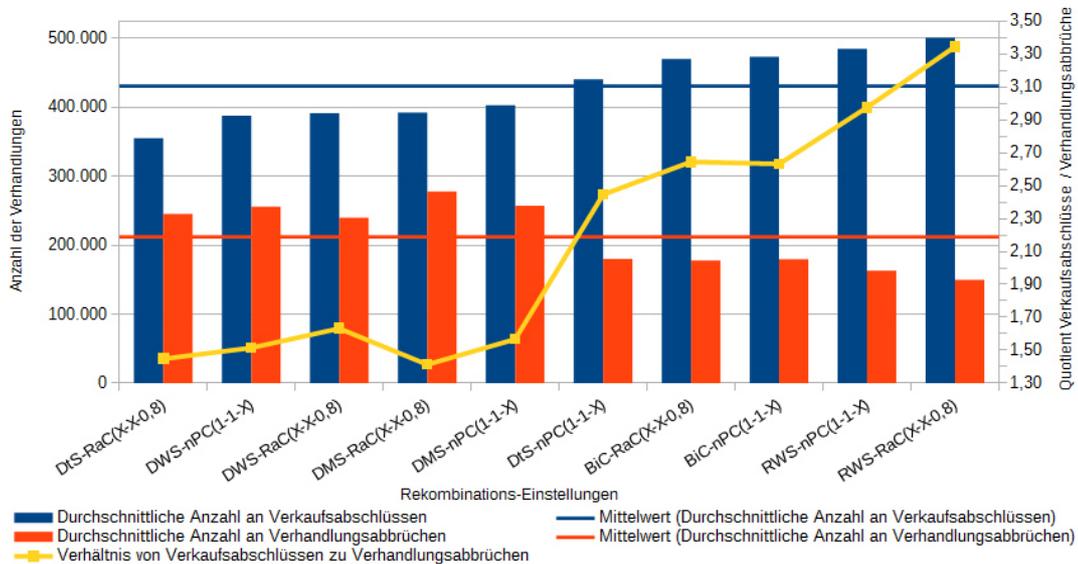


Abbildung 94: Effekte der besten Einstellungen für nPC(1-1-X) und RaC(X-X-0,8) in Kombination mit fünf Selektions-Methoden

Aus den Simulationsergebnissen der Testreihe wurden für die Simulationsszenarien auch weitere Kennzahlen wie zum Beispiel der durchschnittliche Gesamtumsatz und -gewinn sowie die dazugehörigen Variationskoeffizienten berechnet. Die entsprechenden Ergebnisse sind in den folgenden beiden Abbildungen dargestellt.

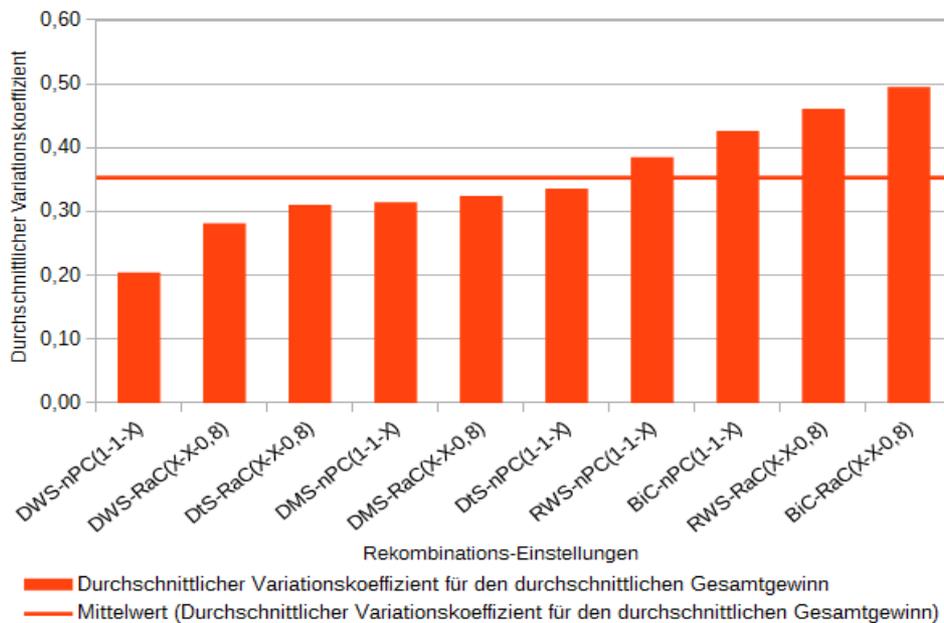


Abbildung 95: Effekte der Rekombination auf die Stabilität des Gesamtgewinns

Es zeigt sich (siehe Abbildung 95 und Abbildung 96), dass von den vier besten Rekombinations-Einstellungen aus Abbildung 94 die Kombination RWS-nPC(1-1-X) hinsichtlich der Ergebnisvariablen Gesamtgewinn und Gesamtumsatz die stabilsten Ergebnisse produziert.

Abbildung 97 zeigt, dass die gleiche Aussage auch in Bezug auf den durchschnittlichen Umsatz und Gewinn pro Transaktion zutrifft.

Zusammengefasst ist für das Anwendungsszenario ein n-Punkt-Crossover (nPC) mit nur einem oder wenigen Schnittpunkten empfehlenswert. So führt die nPC-Einstellung nPC(1-1-X) in Kombination mit der Selektions-Methode RWS in Bezug auf alle untersuchten Ergebnisvariablen zu hervorragenden Ergebnissen, die aus jeder Untersuchungsperspektive heraus zu den besten drei Ergebnissen zu zählen sind.

Insbesondere zeichnet sich diese Kombination durch eine hohe Ergebnis-Stabilität aus, so dass sie sich zur weiteren Analyse im folgenden Abschnitt empfiehlt. Neben der RWS-Methode kann auch die BiC-Methode empfohlen werden.

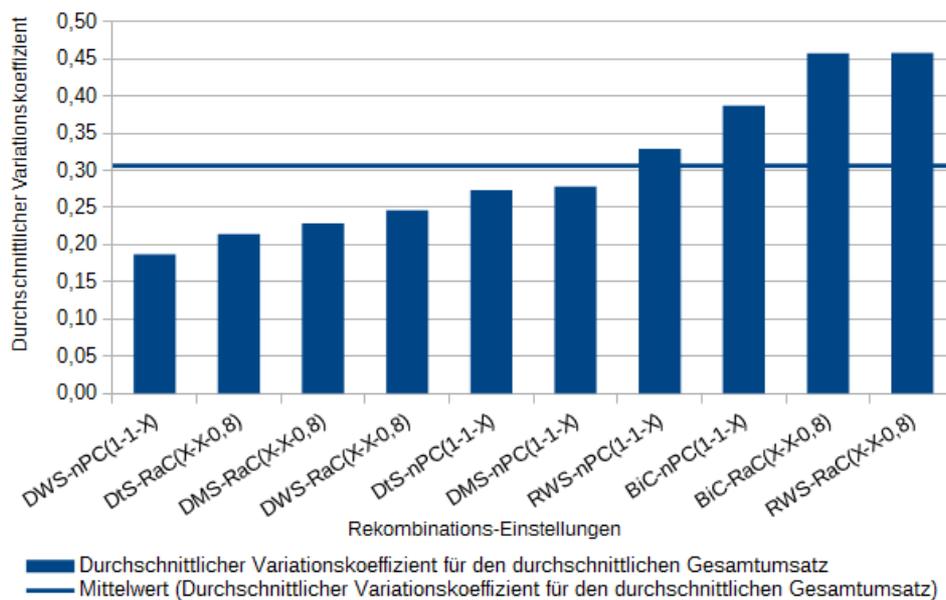


Abbildung 96: Effekte der Rekombination auf die Stabilität des Gesamtumsatzes

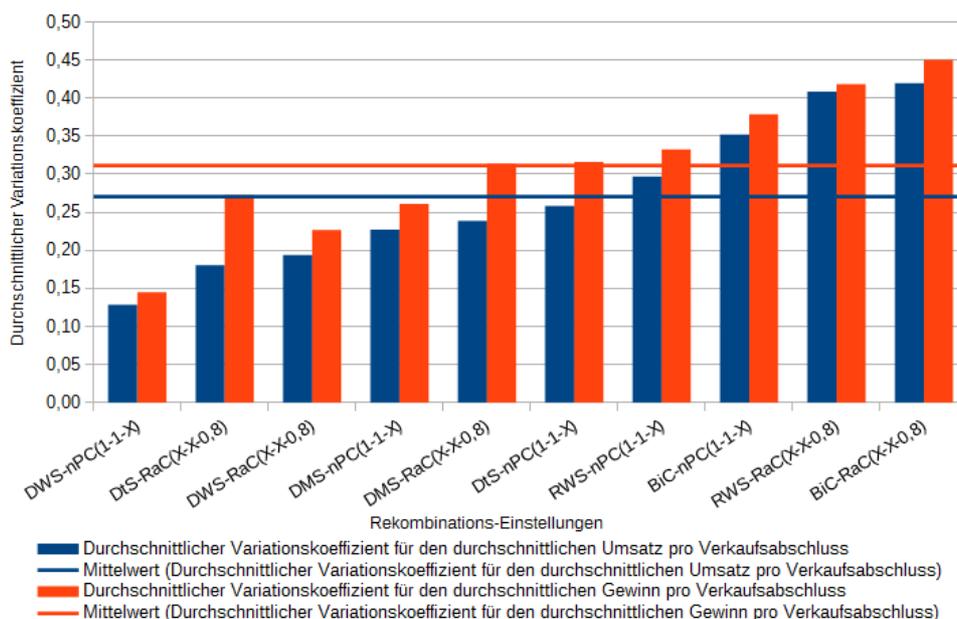


Abbildung 97: Effekte von Rekombinations-Einstellungen auf die Stabilität von Umsatz und Gewinn pro Transaktion

Die Empfehlung von $nPC(1-1-X)$ als nPC -Einstellung mit nur einem Schnittpunkt erscheint auch unabhängig von den oben erläuterten Ergebnissen sinnvoll, da bei der Rekombination aus zwei Genotypen ein neuer entsteht, ohne dass domänenabhängige Interdependenzen zwischen den Genen beachtet werden.

Daher besteht in der Rekombinationsphase die Gefahr, dass den Agenten erfolgreiche Strategie- beziehungsweise Gen-Wert-Kombinationen verloren gehen oder diese „verwässert“ werden, wenn ein für die Anwendungsdomäne sinnvolles Maß überschritten wird.

8.4.5 Effekte der Pool-Separierung

Im Folgenden wird die Performanz des Wertschöpfungsnetzwerks bei 1-Pool-basiertem und 2-Pool-separiertem Lernen miteinander verglichen. Hierzu werden die nPC -Einstellungen $nPC(1-1-X)$ und $nPC(0,5-1-X)$ mit den entsprechenden 1-Pool- und 2-Pool-Einstellungen und – analog zur Testreihe des vorhergehenden Abschnitts – mit der Fitness-Methode AvP und fünf Selektions-Methoden kombiniert. Eine Beschreibung der Testreihe findet sich in Abschnitt 11.1.7.

Der in Abbildung 98 dargestellte, aggregierte Überblick zeigt, dass beide nPC -Einstellungen in Verbindung mit dem 2-Pool-separierten Lernen bezüglich der untersuchten Kennzahlen besser abschneiden als in Verbindung mit dem 1-Pool-basierten Lernen.

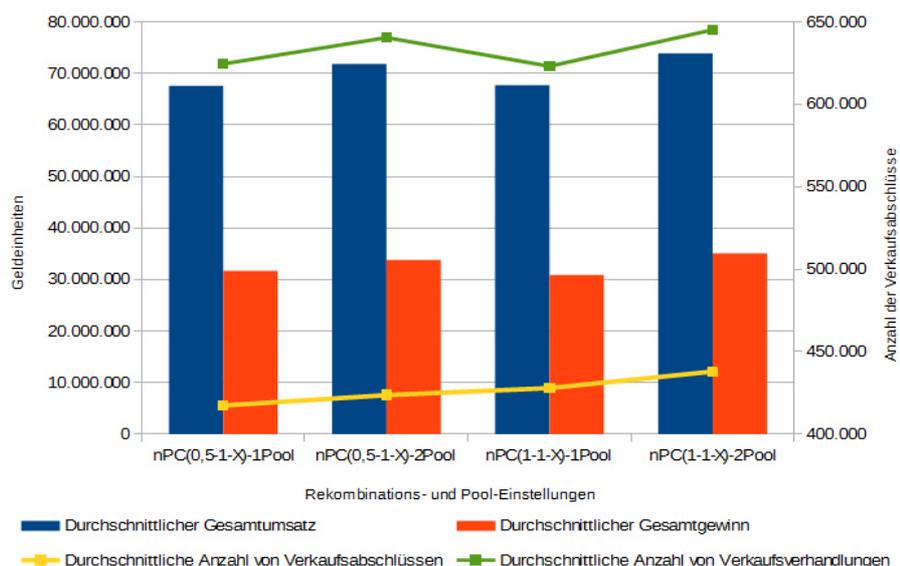


Abbildung 98: Effekte der Pool-Separierung auf Gesamtumsatz, Gesamtgewinn und Verhandlungsergebnis-Kennzahlen

Abbildung 99 zeigt, dass bei Simulationen, in denen alle Agenten homogen das 2-Pool-separierte Lernen nutzen, auch der Umsatz und Gewinn pro Transaktion etwas höher ausfallen.

Betrachtet man die Gesamtmenge aller Szenarien, so fällt auf, dass die Simulationsergebnisse beim 2-Pool-separierten Lernen, was die Anzahl an erzielten Verkaufsabschlüssen und Verhandlungsabbrüchen angeht, weniger stark voneinander abweichen als beim 1-Pool-basierten Lernen (siehe Abbildung 100).

Abbildung 101 veranschaulicht, wie sich die getesteten Einstellungen $nPC(1-1-X)$ und $nPC(0,5-1-X)$ in Kombination mit den fünf Selektions-Varianten auf den in den Simulationen erzielten Gesamtumsatz und –gewinn auswirken.

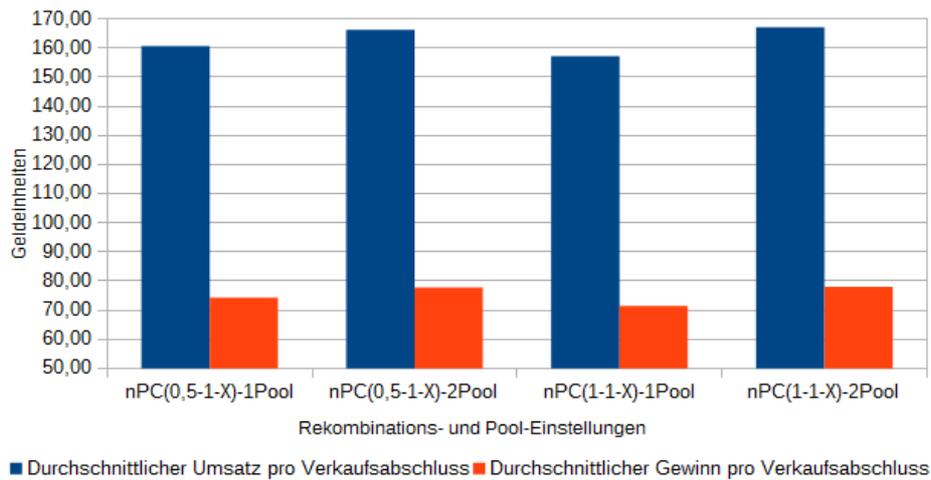


Abbildung 99: Effekte der Pool-Separierung auf Umsatz und Gewinn pro Abschluss

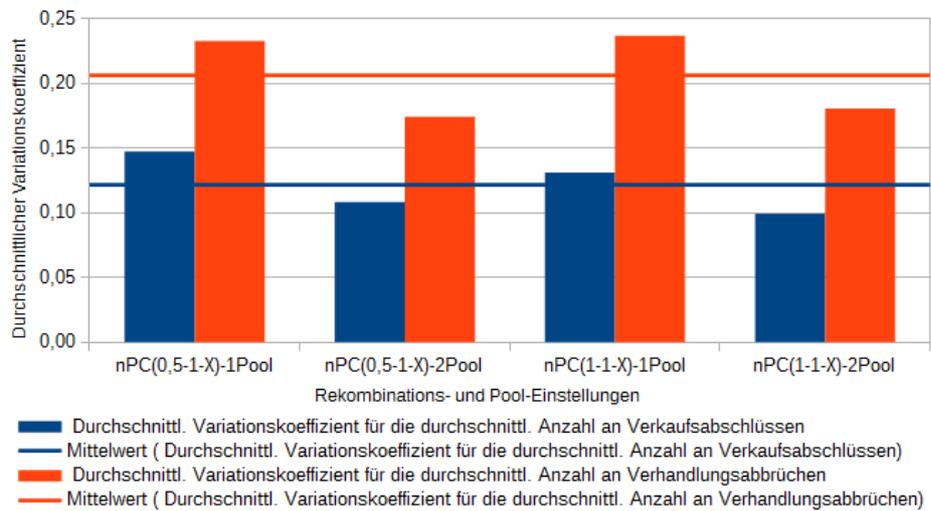


Abbildung 100: Effekte der Pool-Separierung auf die Stabilität der Anzahl an Verkaufsabschlüssen und Verhandlungsabbrüchen

Zur besseren Vergleichbarkeit sind dabei die korrespondierenden 1-Pool- und 2-Pool-Ergebnisse jeweils nebeneinander angeordnet. Es zeigt sich, dass Umsatz und Gewinn beim 2-Pool-separierten Lernen durchgehend besser ausfallen.

Am geringsten fällt der Unterschied bei Verwendung der neu entwickelten DMS-Methode (siehe Abschnitt 5.4.3.2) aus, wahrscheinlich da sie eine für den Adaptionprozess vorteilhafte genetische Vielfalt im Gen-Material erhält.

Der größte Performanz-Sprung bei der Umstellung vom 1-Pool- auf den 2-Pool-Lern-Modus zeigt sich in Kombination mit der *Roulette Wheel Selection (RWS)* (siehe Abbildung 101). Die RWS-Methode führt zum dritthöchsten durchschnittlichen Gesamtumsatz sowie zum zweithöchsten durchschnittlichen Gesamtgewinn.

Darüber hinaus profitieren die Verhandlungsergebnisse vom Einsatz der RWS-Methode. Wie Abbildung 102 veranschaulicht, gelingen den Agenten bei RWS-Nutzung überdurchschnittlich viele Verkaufsabschlüsse, während von ihnen gleichzeitig nur unterdurchschnittlich viele Verhandlungen abgebrochen werden.

Ähnlich gut schneidet die BiC-Methode ab. Die DMS-Methode fällt auf, da sie im 2-Pool-Modus zu den meisten Verhandlungen führt.

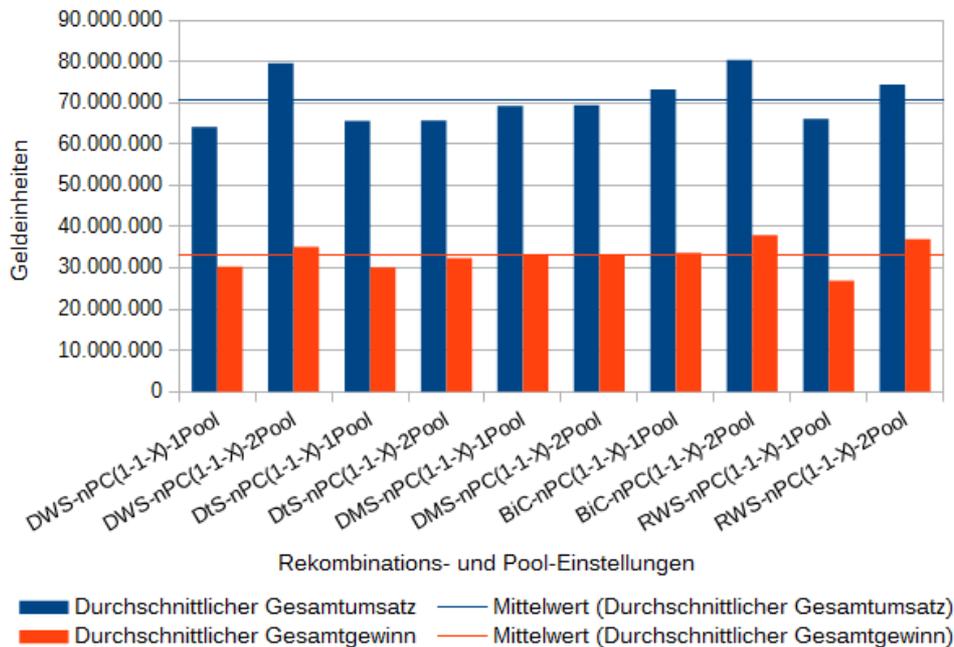


Abbildung 101: Effekte der Pool-Separierung auf den durchschnittlichen Gesamtumsatz und -gewinn

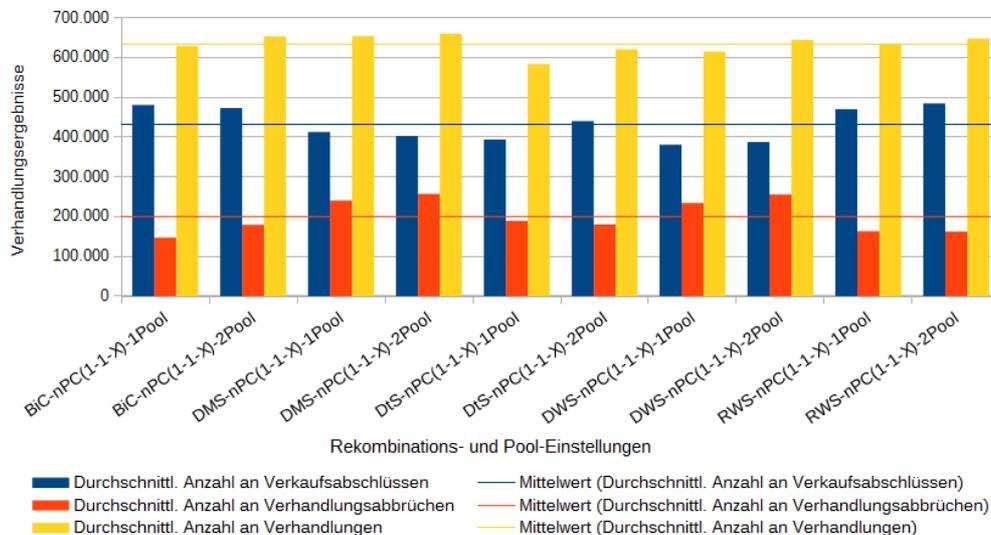


Abbildung 102: Effekte der Pool-Separierung auf die Kennzahlen für Verhandlungsergebnisse

Die Szenarien, die RWS und BiC als Selektions-Methoden verwenden, weisen bei ihren Performanz-Kennzahlen jedoch auch überdurchschnittliche Schwankungen auf, wie aus den folgenden drei Abbildungen ersichtlich ist.

Zusammenfassend zeigen die Ergebnisse der Testreihe, dass die Performanz eines agentenbasierten Wertschöpfungsnetzwerks davon profitiert, wenn alle an ihm partizipierenden Agenten den 2-Pool-separierten Lern-Modus verwenden.

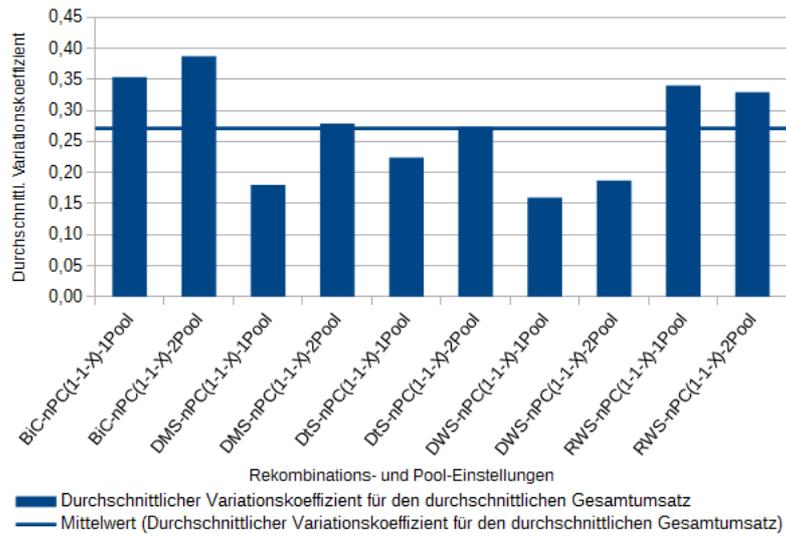


Abbildung 103: Effekte der Pool-Separierung auf die Stabilität des Gesamtumsatzes

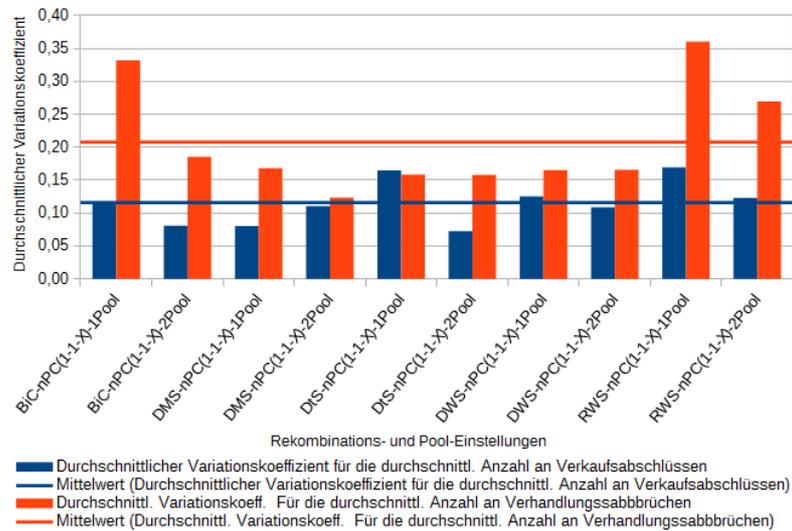


Abbildung 104: Effekte der Pool-Separierung auf die Stabilität der Kennzahlen für Verhandlungsergebnisse

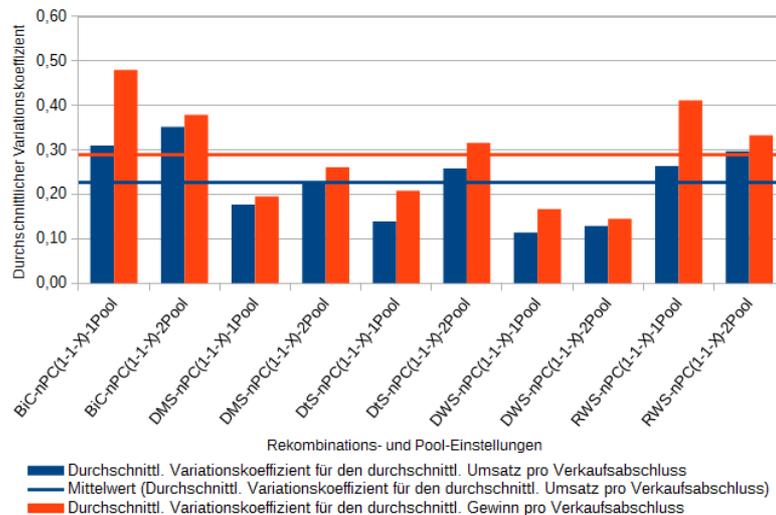


Abbildung 105: Effekte der Pool-Separierung auf die Stabilität des Umsatzes und Gewinns pro Verkaufsabschluss

8.4.6 Effekte der Ersetzungsschema-Einstellungen

Im Folgenden werden die Effekte der Ersetzungsschemata *Elitismus* (*E*) und *Schwacher Elitismus* (*WE*) mit den Parameter-Einstellungen $E(0)$, $E(1)$, $E(2)$, $E(3)$, $E(4)$ sowie $WE(0)$, $WE(1)$, $WE(2)$, $WE(3)$, $WE(4)$ untersucht.

Alle Szenarien der dazu durchgeführten Testreihe beinhalten die Initialisierung von 50 Agenten, die alle den 2-Pool-separierten Lern-Modus nutzen. Die Testreihe nutzt das Szenario-Grundmuster

MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-[Elitismus]-AvP-[Rnd]-50A-2Pool

und ist in Abschnitt 11.1.8 beschrieben.

Abbildung 106 zeigt, dass die Wahl der Ersetzungsschema-Parametereinstellung eine gravierende Auswirkung auf den durchschnittlichen Gesamtumsatz und Gesamtgewinn des Wertschöpfungsnetzwerks hat.

Mit der Ersetzungsschema-Parametereinstellung $WE(3)$ erzielt das Wertschöpfungsnetzwerk den höchsten durchschnittlichen Gesamtumsatz. Dieser ist über fünfmal höher als der Gesamtumsatz, den das System ohne Ersetzungsschema (das heißt mit der Parametereinstellung $E(0)$ oder $WE(0)$) zu erzielen in der Lage ist.

Insgesamt erzielen die Szenarien, welche die *WE*-Methode verwenden, durchgehend einen deutlich höheren Gesamtumsatz und Gesamtgewinn als die Szenarien, welche die *E*-Methode einsetzen.

Während das Multiagentensystem mit der Einstellung $WE(3)$ den höchsten Gesamtumsatz erzielt, liegt der mit $WE(1)$ generierte Gesamtgewinn circa 2 Prozent über dem bei $WE(3)$ erwirtschafteten. Der Gesamtumsatz fällt bei $WE(1)$ etwas geringer als bei $WE(3)$ aus, ist aber immerhin am zweithöchsten.

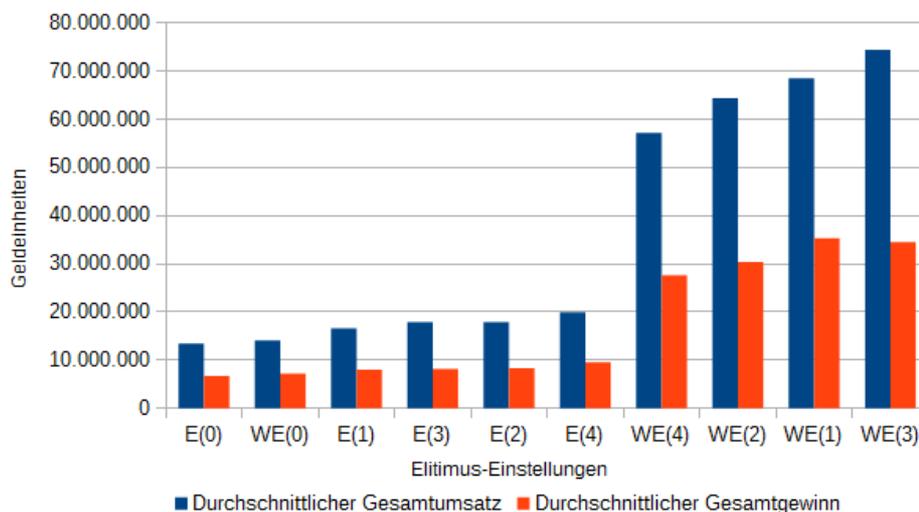


Abbildung 106: Gesamtumsatz und Gesamtgewinn in Abhängigkeit von den Ersetzungsschema-Einstellungen

Wie Abbildung 107 zeigt, ist dies darauf zurückzuführen, dass beim Einsatz der *WE*- statt der *E*-Methode die Anzahl an Verhandlungsabbrüchen sprunghaft sinkt, während gleichzeitig die Anzahl an Verkaufsabschlüssen genauso sprunghaft ansteigt. Während sich die Ergebnisse der *WE*-Einstellungen insgesamt auf einem signifikant höheren Performanz-Niveau als die

der E-Einstellungen bewegen, fallen auf das jeweilige Ersetzungsschema bezogen die Unterschiede zwischen den Fein-Parametrisierungen nur relativ gering aus.

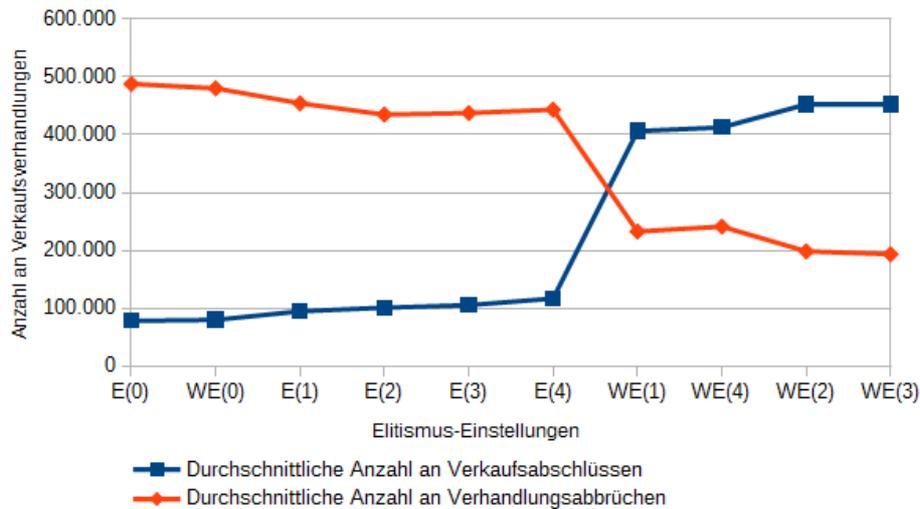


Abbildung 107: Verkaufsabschlüsse und Verhandlungsabbrüche in Abhängigkeit von den Ersetzungsschema-Einstellungen

Aus der vorhergehenden Darstellung ergibt sich, dass der prozentuale Anteil der erfolgreich abgeschlossenen Verhandlungen an der Gesamtzahl der geführten Verhandlungen steigen muss. Dies lässt sich anhand von Abbildung 108 nachvollziehen.

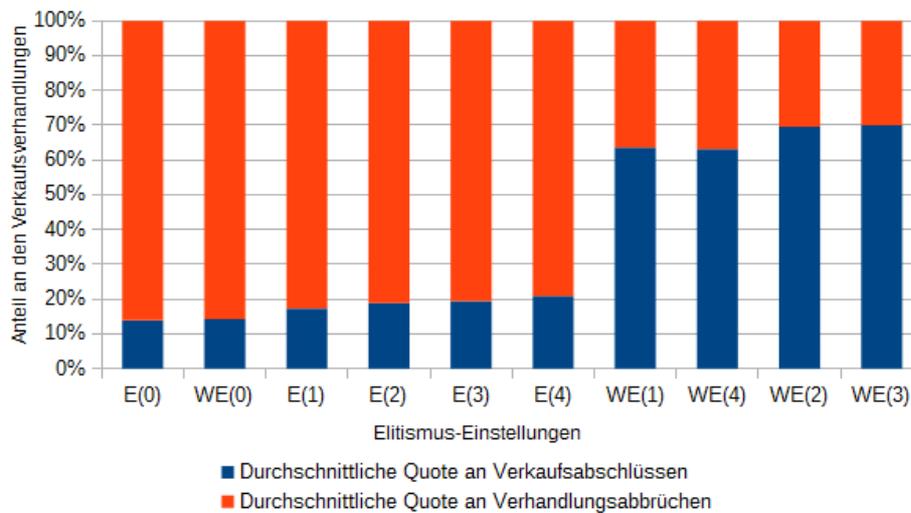


Abbildung 108: Prozentualer Anteil von Verhandlungsabbrüchen und Verkaufsabschlüssen an den geführten Verhandlungen

Zusammenfassend kann für ein aus Agenten bestehendes Wertschöpfungsnetzwerk, bei dem die Agenten eine homogen parametrisierte Lern-Komponente aufweisen, die Verwendung des Ersetzungsschemas Schwacher Elitismus empfohlen werden. Dieses Ersetzungsschema führt zu einer signifikant erhöhten Quote an Verkaufsabschlüssen und damit durchweg dazu, dass vom Gesamtsystem die Erwirtschaftung höherer Umsätze und Gewinne als bei der Verwendung des Ersetzungsschemas Elitismus erwartet werden kann.

Beim detaillierten Vergleich der WE-Einstellungen zeigt sich, dass WE(2) und WE(3) bzgl. der untersuchten Performanz-Kennzahlen führend sind und in etwa auf dem gleichen guten Niveau liegen.

In den folgenden Testreihen wird die Einstellung WE(3) verwendet, da die Agenten-Gemeinschaft innerhalb der Testreihe bei ihrer Nutzung den höchsten durchschnittlichen Gesamtumsatz erzielen konnte.

8.4.7 Effekte der Mutations-Methoden

In diesem Abschnitt werden die Auswirkungen der Mutations-Methoden *General Mutation (GMU)* und *Single Mutation (SMU)* (siehe 5.4.3.4) auf die System-Performanz untersucht. Die Testreihe nutzt das Szenario-Grundmuster

MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-[Mutation]-WE(3)-AvP-[Rnd]-50A-2Pool

und verwendet durchgängig Szenarien, welche die Initialisierung von 50 Agenten beinhalten, und alle den 2-Pool-separierten Lern-Modus nutzen. Die Testreihe ist in Abschnitt 11.1.9 beschrieben.

Da die Mutations-Methoden die Werte einzelner oder mehrerer Gene verändern, besteht wie in der Rekombinations-Phase auch in der Mutations-Phase die Gefahr, dass den Agenten erfolgreiche Strategie- beziehungsweise Gen-Wert-Kombinationen verloren gehen oder diese „verwässert“ werden, wenn ein für die Anwendungsdomäne sinnvolles Maß überschritten wird.

Bei der GMU-Methode (siehe 5.4.3.4) wird, wenn überhaupt, nur ein einziges Gen mutiert. Der erste Parameter von GMU gibt an, wie häufig mutiert wird und der zweite die Gauß-Weite, die das Maß der erlaubten Änderung des Gen-Werts angibt⁷⁴.

Bei der Methode SMU (siehe 5.4.3.4) kann im Extremfall jedes Gen eines Genotypen mutiert werden, da für jedes Gen durch das Ziehen einer Zufallszahl entschieden wird, ob es mutiert wird oder nicht.

Der analysierten Testreihe liegt die Hypothese zugrunde, dass ein „Zuviel“ an Mutation gute Genotypen bzw. Gen-Wert-Kombinationen „zerstören“ und ein „Zuwenig“ negativ für die Genvielfalt und dadurch für das Lernen sein kann. Weiterhin sind die Häufigkeit und die Größe der Mutationssprünge zu beachten, bei denen es auch kein Zuviel und Zuwenig geben darf. Dies alles muss ausbalanciert werden.

Denn große Veränderungen, die durch das Wirken der Mutations-Methoden an Genotypen in einem Schritt entstehen, werden häufig wieder schnell aus der Population verdrängt, da durch die enge Verknüpfung und Wechselwirkungen der Gene elementare negative Eigenschaften bei sogenannten „Groß-Mutationen“ kaum vermeidbar sind (siehe [Weicker 2015]).

Hinsichtlich der Feinparametrisierung von Mutations-Methoden wird daher in der Literatur (siehe [Weicker 2015]) in der Regel eine zufällige Veränderung der Gen-Werte anhand der Normalverteilung vorgeschlagen, da der Grad der Mutation (insbesondere bei Fließkommazahlen) eher gering sein sollte.

Da bei GMU (wenn überhaupt) ein einziges Gen mutiert wird, ist es einfacher als bei SMU, die an die Anwendungsdomäne gut angepasste Mutationshäufigkeit zu ermitteln, die zu einer guten Selbstkoordination und performanten System-Leistung führt.

Abbildung 109 zeigt, dass der erste Parameter-Wert der GMU-Methode nicht auf 1,0, sondern eher auf 0,5 eingestellt werden sollte, da mit der Einstellung 0,5 alle betrachteten Performanz-

⁷⁴ Beispiel: Bei GMU(1,0-0,3) wird immer eines der Gene mutiert (1 = 100%). Bei einer Gauß-Weite von 0,3 und einem Gen-Wert von 0,55 vor der Mutation liegt der mutierte Gen-Wert mit einer hohen Wahrscheinlichkeit im Intervall [0,25; 0,85].

Kennzahlen, insbesondere auch Gesamtumsatz und –gewinn, signifikant besser ausfallen. Es ist also bei der GMU-Methode Szenarien-unabhängig empfehlenswert, einen Genotypen nicht in 100% der Fälle, sondern besser nur in 50% der Fälle zu mutieren. In Bezug auf die Maximierung von Gesamtumsatz und –gewinn des Systems kann für GMU eine Gauß-Weite von circa 0,1 empfohlen werden.

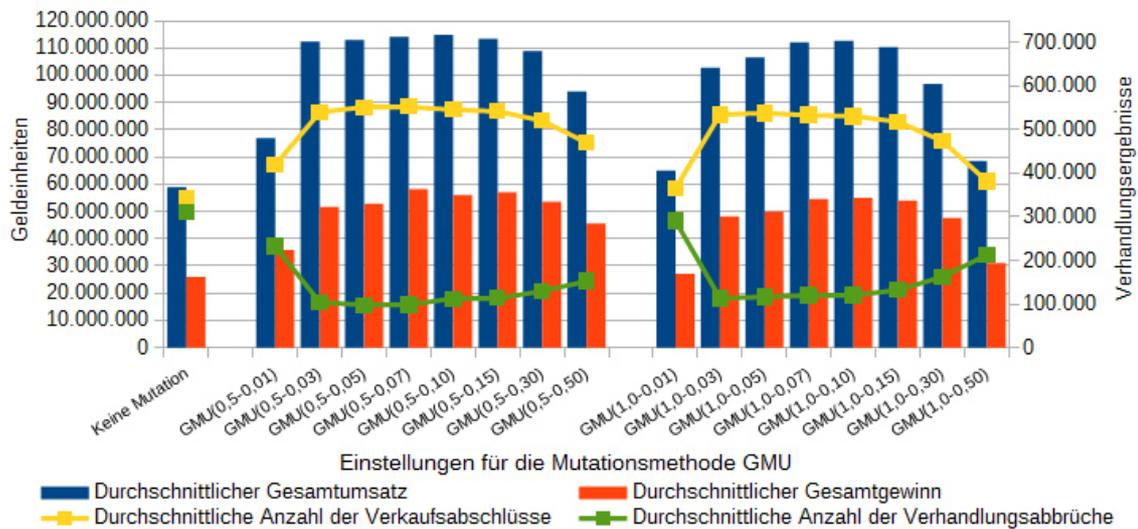


Abbildung 109: Durchschnittlicher Gesamtumsatz und –gewinn in Abhängigkeit von den Einstellungen für die Mutations-Methode GMU

Abbildung 110 macht die Unterschiede der Ergebnisse aus Abbildung 109 deutlich, indem die Differenzen aus den beiden Einstellungen GMU(0,5-X) und GMU(1,0-X) erzielten Kennzahlen-Werten gebildet und dargestellt werden. GMU(0,5-X) führt in allen Fällen zu Ergebnissen, die bezüglich aller vier Kennzahlen besser sind.

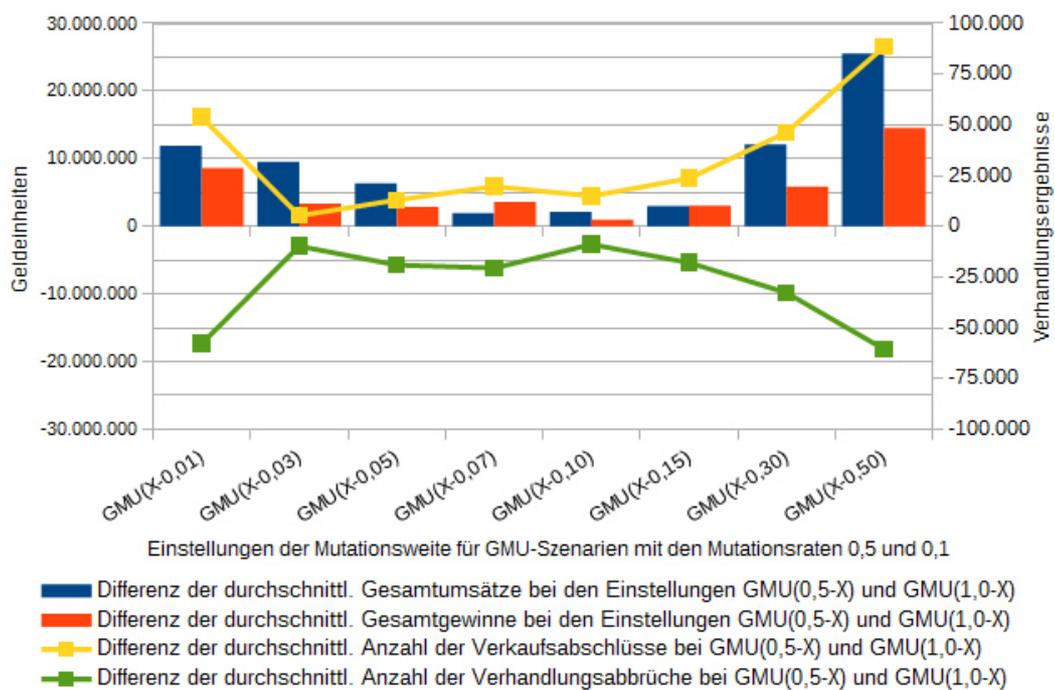


Abbildung 110: Entwicklung von Gesamtumsatz, –gewinn und Verhandlungsergebnissen bei Variation der Gauß-Weite für GMU

Die in den obigen Abbildungen erkannten Tendenzen bestätigen sich auch bei der Betrachtung der Entwicklung von Umsatz und Gewinn pro Transaktion in Abbildung 111.

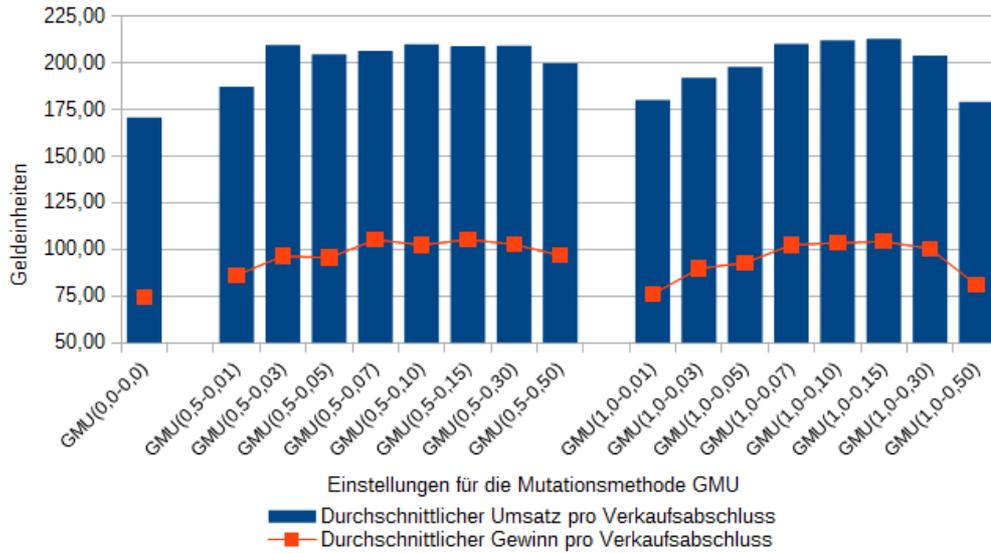


Abbildung 111: Vergleich der Entwicklung des Gesamtumsatzes für GMU mit den Mutationsraten 0,5 und 1,0 bei Variation der Mutations-Weite

Wie Abbildung 112 zu entnehmen ist, führt die Einstellung GMU(0,5-X) sowohl bezüglich der erzielbaren Anzahl an Verkaufsabschlüssen als auch bezüglich der Verhandlungsdauer in allen Fällen beziehungsweise bei allen Einstellungen der Mutations-Weite zu wünschenswerteren Ergebnissen als GMU(1-X).

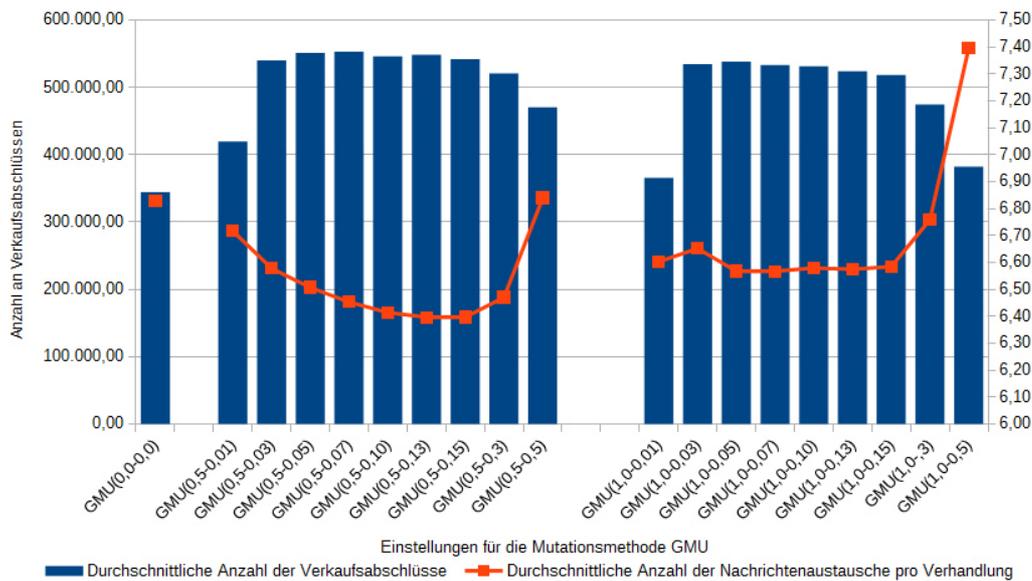


Abbildung 112: Entwicklung der Abschlüsse und Nachrichtenaustausche pro Verhandlung für die Mutationsraten 0,5 und 1,0 bei Variation der Mutations-Weite

Abbildung 113 zeigt, dass es auch bei Verwendung der SMU-Methode fast immer besser ist, seltener, also in 50% der Fälle (SMU(0,5-X)) zu mutieren, anstatt immer zu mutieren und damit jedes einzelne Gen zu verändern. Das Diagramm bestätigt die Vermutung, dass eine nicht zu hohe Mutations-Rate in Verbindung mit einer maßvollen Mutations-Weite für die SMU-Einstellungen sinnvoll ist, weil dadurch das Risiko sinkt, dass Genotypen sich zu stark verändern und damit in nachfolgenden Verhandlungen eine schlechtere Fitness-Bewertung erhalten. Bei SMU(0,5-0,5) übersteigen die Abbrüche bereits die Anzahl erfolgreicher Verhandlungen, was sich noch weiter verstärkt, wenn jedes Gen immer mutiert wird.

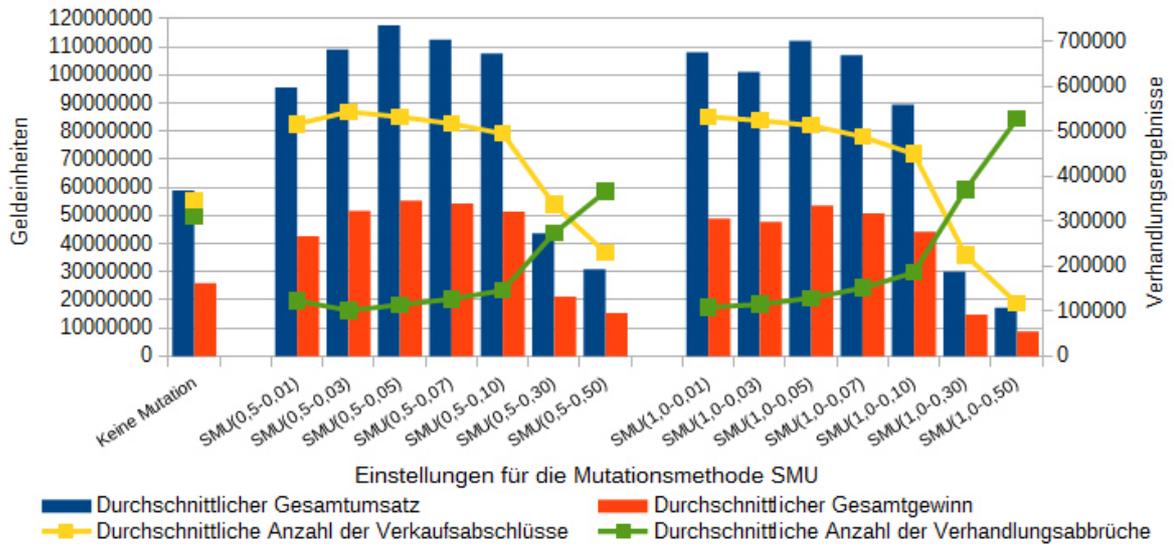


Abbildung 113: Gesamtumsatz, Gesamtgewinn und Verhandlungsergebnisse in Abhängigkeit von den Einstellungen für die Mutations-Methode SMU

Es fällt auf, dass die Ergebnisse bei Einsatz der SMU-Methode bei den meisten Parameter-Einstellungen besser sind, als wenn die Mutation ausgeschaltet ist (in der Abbildung als „keine Mutation“ gekennzeichnet). Werden die Parameter jedoch ungünstig gewählt, kann die System-Performanz auf ein deutlich darunter liegendes Niveau abfallen. Dies konnte bei der GMU-Methode (siehe Abbildung 109) nicht beobachtet werden.

Analog zu Abbildung 110 stellt auch Abbildung 114 die Differenzen der Kennzahlen-Ergebnisse für die Einstellungen SMU(0,5-X) und SMU(1,0-X) kompakt dar. Für eine Gauß-Weite ab 0,03 fallen die SMU(0,5-X)-Ergebnisse dabei immer besser aus.

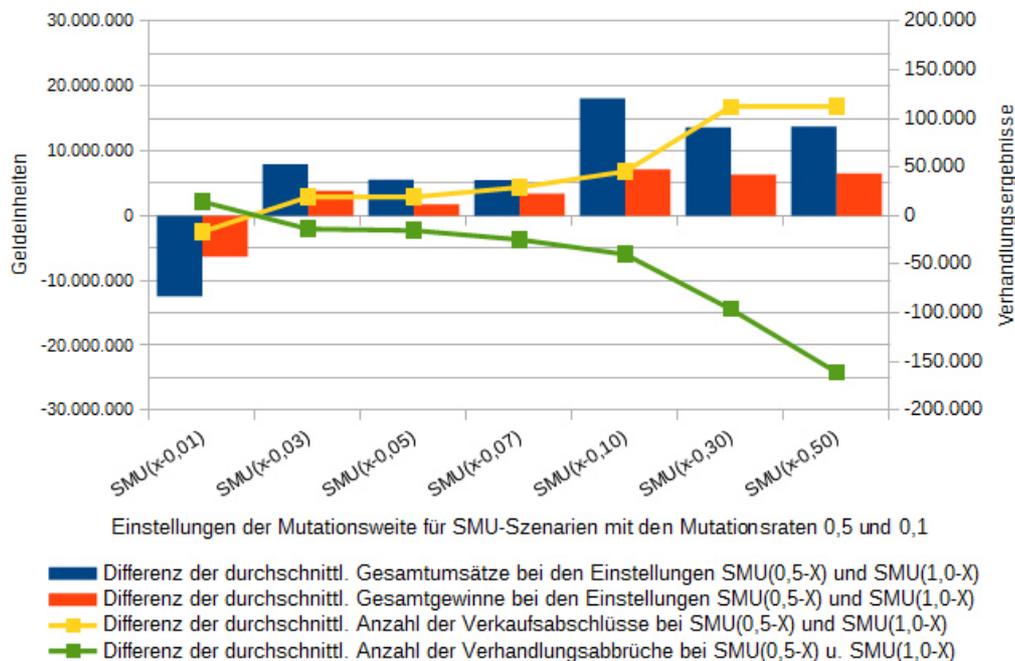


Abbildung 114: Unterschiede bei der Entwicklung von Umsatz, Gewinn und Verhandlungsergebnissen für SMU bei Variation der Mutationsrate

Einen direkten Vergleich der Performanz-Kennzahlen von GMU und SMU für ausgewählte Parameter-Einstellungen bietet Abbildung 115. Bei gleichen Einstellungen führt GMU in vier

von sechs Fällen bezüglich aller Kennzahlen zu besseren Ergebnissen. Dies dürfte daran liegen, dass durch die GMU-Methode ein Genotyp sanfter verändert wird.

In zwei Fällen, nämlich bei GMU(0,5-0,05) und GMU(1,0-0,05) beziehungsweise bei einer Mutations-Weite von 0,05 liefert GMU bessere Verhandlungsergebnisse, aber einen etwas geringeren Gesamtumsatz und –gewinn.

Es ist aber auffällig, dass sich die SMU-Ergebnisse bei größer werdender Mutations-Rate und –Weite viel rasanter verschlechtern, als dies bei der GMU-Methode der Fall ist (vergleiche dazu auch Abbildung 116).

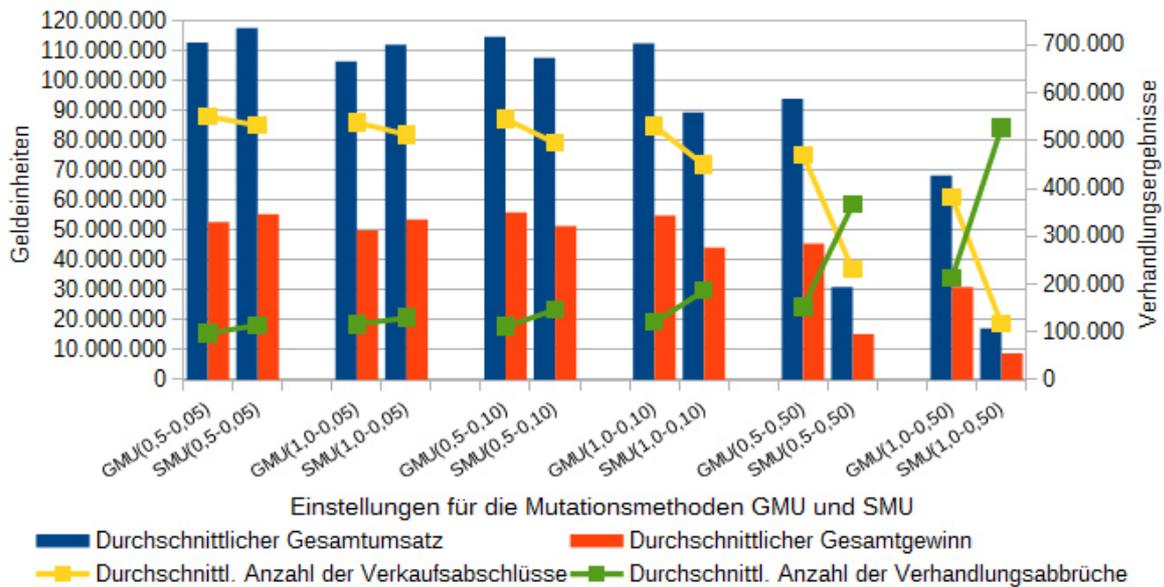


Abbildung 115: Gesamtumsatz, Gesamtgewinn und Verhandlungsergebnissen für die Methoden GMU und SMU bei Variation von Mutations-Rate u. –Weite

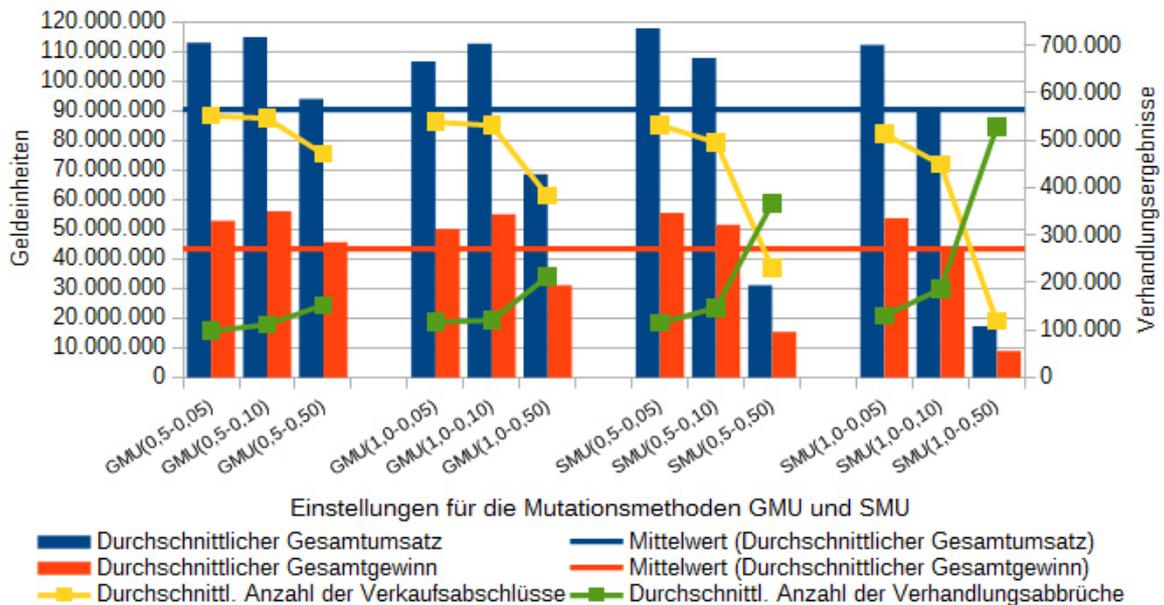


Abbildung 116: Entwicklung von Gesamtumsatz, Gesamtgewinn und Verhandlungsergebnissen für die Methoden GMU und SMU bei Variation der Mutations-Rate und –Weite

Die Ergebnisse zeigen, dass eine Mutations- beziehungsweise Gauß-Weite von 0,05 bis 0,1 bei der GMU-Methode eine gute Wahl ist. Bei der SMU-Methode wäre sie mit 0,03 bis 0,05

deutlich geringer anzusetzen, um gute Ergebnisse zu erzielen. Weiterhin kann als GMU-Parameter-Einstellung eine Mutations-Rate von 0,5 empfohlen werden, mit der signifikant bessere Ergebnisse erzielbar sind als mit 0,1.

Da die GMU-Methode bei den meisten Einstellungs-Kombinationen besser abschneidet als die SMU-Methode und bei ihr die Gefahr, einen Genotyp in einem einzigen Schritt so stark zu verändern, dass seine Fitness-Bewertung „einbricht“, geringer ausfällt, sollte sie beim Einsatz in der Anwendungsdomäne präferiert werden.

Im Durchschnitt hoch-performante Mutations-Einstellungen auszuwählen, die hinsichtlich ihrer Einzelergebnisse aber eine große Schwankungsbreite beziehungsweise einen hohen Variationskoeffizienten aufweisen, ist wenig sinnvoll, da die Gefahr besteht, dass dem System im Einzelfall hohe Umsatz- und Gewinnbeträge verloren gehen. Ein hoher Variationskoeffizient bedeutet in dieser Hinsicht ein hohes Risiko.

Dennoch ist es nicht sinnvoll, entsprechende Einstellungen allein auf Basis der aus ihren Simulations-Ergebnissen berechneten Variationskoeffizienten auszuwählen. Denn ein niedriger Variationskoeffizient macht keine Aussage über die Performanz der Einstellungen hinsichtlich der untersuchten Ergebnisvariablen. Er sagt lediglich aus, wie sehr die Werte der Ergebnisvariablen schwanken. Damit stellen moderate Variationskoeffizienten für die Kennzahlen von Szenarien für die Auswahl geeigneter Mutations-Parameter eine notwendige, aber keine hinreichende Bedingung dar.

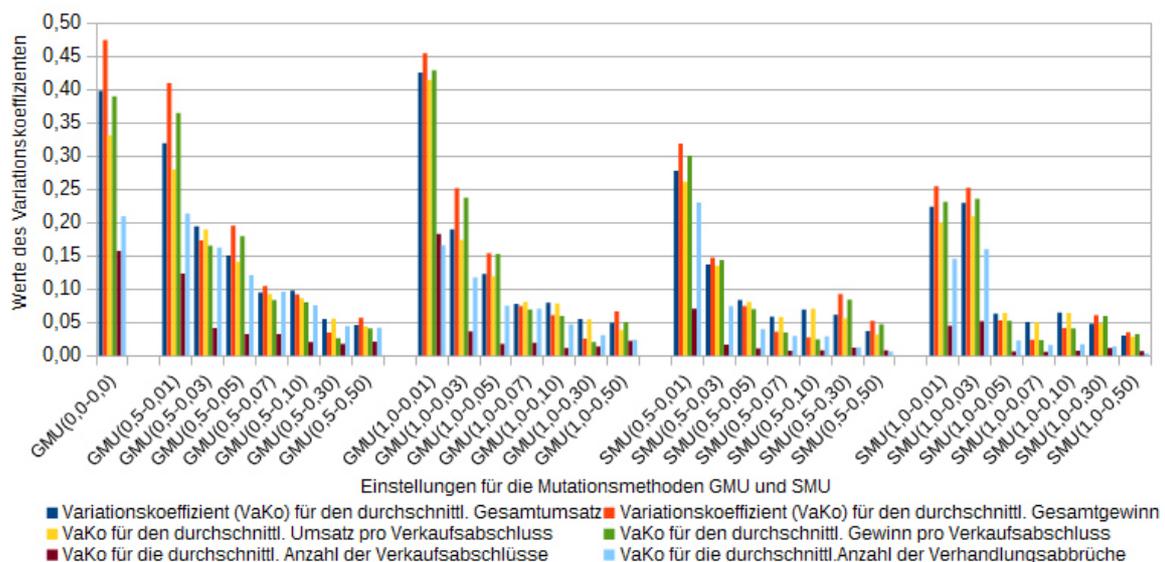


Abbildung 117: Vergleich der Entwicklung verschiedener Variationskoeffizienten für die Methoden GMU und SMU bei Variation von Mutations-Rate u. –weite

Gleichbleibend schlechte Ergebnisse, wie sie beispielsweise bei SMU mit hoher Mutations-Rate und –Weite erzielt werden, weisen daher für die Ergebnisvariablen sehr geringe Variationskoeffizienten auf (vergleiche Abbildung 117). Dennoch können einem Agenten- oder System-Designer entsprechende Einstellungen nicht empfohlen werden.

Vielmehr muss primär eine performante Parameter-Einstellung für eine Mutations-Methode gefunden werden, wie sie oben beispielsweise mit GMU(0,5-0,1) identifiziert wurde. Daran anschließend muss überprüft werden, wie hoch die Schwankungen bei den Ergebnisvariablen beziehungsweise Kennzahlen ausfallen. Aus Abbildung 118 ist ersichtlich, dass auch vor diesem Hintergrund die Einstellung GMU(0,5-0,1) empfohlen werden kann.

Dabei ist hervorzuheben, dass der Einsatz sämtlicher getesteter Mutations-Varianten beziehungsweise der Lern-Komponente mit eingeschalteter Mutations-Phase dazu geführt hat,

dass die Werte der Variationskoeffizienten sämtlicher Kennzahlen im Vergleich zur ausgeschalteten Mutation (bezeichnet als „GMU(0,0-0,0)“) signifikant gesenkt wurden, was anhand von Abbildung 117 und Abbildung 118 nachvollzogen werden kann.

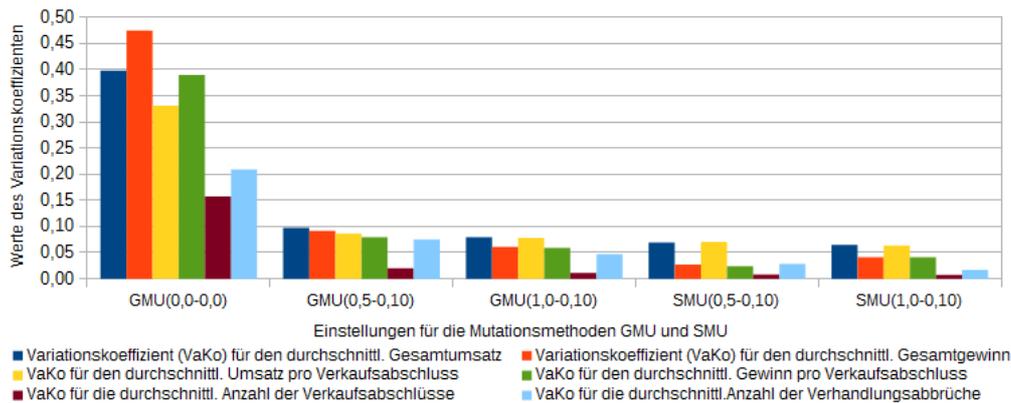


Abbildung 118: Variationskoeffizienten für die Mutations-Methoden GMU und SMU bei Variation von Mutations-Rate u. –Weite

8.5 Informationsaustausch und –nutzung

MACSIMA-Agenten können unmittelbar nach einem erfolgreichen Verhandlungsabschluss Informationen miteinander austauschen, die die abgeschlossene Verhandlung und die von ihnen beobachtete Marktpreissituation für das gehandelte Gut betreffen. Entsprechende Informationen werden von den Agenten in Abhängigkeit von ihren Informationsaustausch-Einstellungen (siehe Abschnitt 5.4.1.3) und ihrer Zugehörigkeit zu einer Agenten-Gruppe in der Form eines Plumages (siehe Abschnitt 5.4.1) an andere Agenten gesendet.

Bezüglich der Informationsaustausch-Einstellungen stehen verschiedene Modi zur Verfügung, die für einen Agenten festlegen, ob er keine Informationen beziehungsweise Plumages sendet, das Senden auf die Mitglieder seiner Gruppe beschränkt ist oder er seine Plumages an alle Agenten eines Simulationsszenarios adressieren kann.

Darüber hinaus kann für den Informationsaustausch eines Agenten eine Quote eingestellt werden, die festlegt, an wieviel Prozent (P) der Agenten, die er gemäß seines Informationsaustausch-Modus adressieren kann, er im Falle (F) eines erfolgreichen Verhandlungsabschlusses ein Plumage sendet. Bei der Standardeinstellung für die Quote, nämlich 100P100F, sendet ein Agent nach jedem erfolgreichen Verhandlungsabschluss das von ihm generierte Plumage an alle beziehungsweise 100 Prozent der von ihm adressierbaren Agenten.

Sendet ein Agent ein Plumage, so enthält dieses Informationen über das gehandelte Gut, den Genotyp, der die Verhandlungsstrategie des Agenten in der Verhandlung bestimmt hat, die für diesen ermittelte Fitness-Bewertung sowie verschiedene Preisinformationen, die die Verhandlungssituation charakterisieren.

Zu den Preisinformationen gehören die Startpreise der Verhandlungspartner, der Einigungsbeziehungsweise Abschlusspreis sowie der (gewichtete) durchschnittliche Marktpreis für das gehandelte Gut, den der Agent, der das Plumage generiert hat, subjektiv beobachtet hat. Dieser sogenannte *weighted Average* wird von einem MACSIMA-Agenten nach jedem erfolgreichen Verhandlungsabschluss auf Basis des Abschlusspreises und seines *Weight_Memory* aktualisiert (siehe Abschnitt 5.3.2).

Weiterhin enthält solch ein Plumage Informationen darüber, ob der in ihm enthaltene Genotyp beim Verkauf oder dem Kauf des gehandelten Gutes eingesetzt wurde.

Empfängt ein MACSIMA-Agent ein von einem anderen Agenten extern generiertes Plumage, so entscheidet seine Einstellung zur Informationsaustausch-Nutzung (siehe Abschnitt 5.4.1.4) darüber, ob er dieses in eine seiner Populationen aufnimmt und es daher in seinem Lernprozess nutzt. Im Modus des Internen Lernens verwendet der Agent keine externen Informationen und nutzt nur interne, das heißt von ihm selbst erzeugte Plumages. Im Modus des Externen Lernens stützt er sich hingegen ausschließlich auf externe Plumages. Dagegen sammelt er im Modus des Gemischten Lernens in den von ihm verwalteten Populationen sowohl interne als auch externe Plumages, deren Gen-Material im späteren EA-basierten Lernprozess durchmischt wird.

8.5.1 Informationsaustausch und Mutation

Bei dieser Testreihe wurde untersucht, wie unterschiedliche, systemweit geltende Informationsaustausch-Einstellungen die Auswirkungen der Mutations-Einstellungen $GMU(0,5-0,01)$, $GMU(0,5-0,1)$, $GMU(1,0-0,1)$, $SMU(0,5-0,1)$ und $SMU(1,0-0,1)$ auf die Performanz des Gesamtsystems beeinflussen.

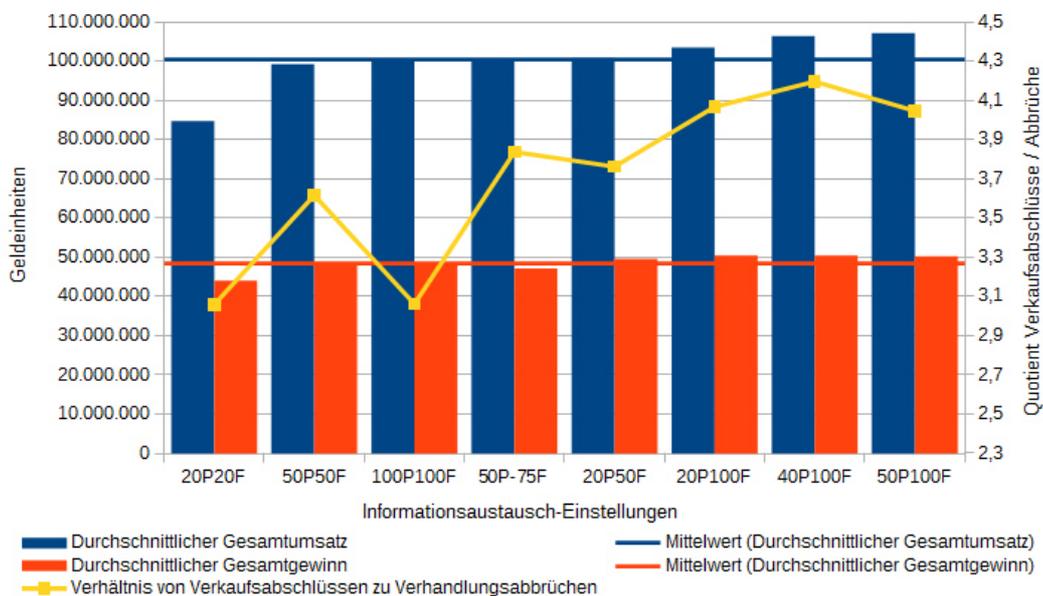


Abbildung 119: Informationsaustausch-Effekte 1

Alle Szenarien der Testreihe beinhalten die Initialisierung von 50 Agenten, die alle den 2-Pool-separierten Lern-Modus nutzen. Es wurden die Informationsaustausch-Einstellungen 20P20F, 20P50F, 20P100F, 40P100F, 50P50F, 50P75F, 50P100F und 100P100P untersucht. Die Testreihe ist in Abschnitt 11.1.10 beschrieben.

Abbildung 119 zeigt eine Aggregation der Simulationsergebnisse, bei denen über die mit allen Mutations-Einstellungen erzielten Ergebnisse der Durchschnitt gebildet wurde. Das Diagramm zeigt, dass es sinnvoll ist, den Informationsaustausch zu begrenzen beziehungsweise auszubalancieren. Die Informationsaustausch-Einstellungen sind gemäß der mit ihnen erzielten Umsatzergebnisse in aufsteigender Reihenfolge angeordnet.

Senden die Agenten einer Gruppe ihre Strategie-Plumages immer an alle anderen Gruppenmitglieder (Einstellung 100P100F), so kann nur das drittschlechteste von acht Umsatzergebnissen erzielt werden. Auch in Bezug auf das Verhältnis von Verkaufsabschlüssen zu Verhandlungsabbrüchen handelt es sich um das zweitschlechteste Ergebnis.

Um ein performantes Ergebnis zu erzielen reicht es jedoch nicht aus, den Informationsaustausch einfach zu beschränken, da das in allen betrachteten Aspekten schlechteste Ergebnis durch eine drastische Reduktion des Informationsaustauschs auf 20P20F erzielt wird.

Die beiden besten Ergebnisse werden durch eine eher moderate Beschränkung des Informationsaustauschs auf 40P100F und 50P100F erzielt. Besonders wichtig scheint dabei zu sein, dass die Zahl der Gruppenmitglieder, an die Informationen weitergegeben werden, nicht zu groß wird. Der Grund dafür könnte darin liegen, dass die Weitergabe von Strategie-Plumages mit besonders hoher Fitness an viele Agenten der gleichen Gruppe relativ schnell dazu führt, dass sich das Gen-Material dieser Plumages innerhalb der Gruppe durchsetzt und anderes Gen-Material verdrängt. Hierbei könnte es auch zu einer Art Overfitting-Effekt kommen.

Dies führt im Verlauf der Simulation zu einer Verringerung oder gar zu einer Zerstörung der genetischen Vielfalt der innerhalb der Gruppe genutzten Verhandlungsstrategien. Dies wiederum hemmt die Adaptionfähigkeit der Agenten im weiteren Verlauf der Simulation, die dann nur noch durch eine höhere Mutations-Rate und –Weite verbessert werden kann. Dies beinhaltet aber, wie oben erläutert, auch Risiken hinsichtlich des Verlusts von Gen-Material, das sich im Simulationsverlauf als für die Agenten vorteilhaft erwiesen hat.

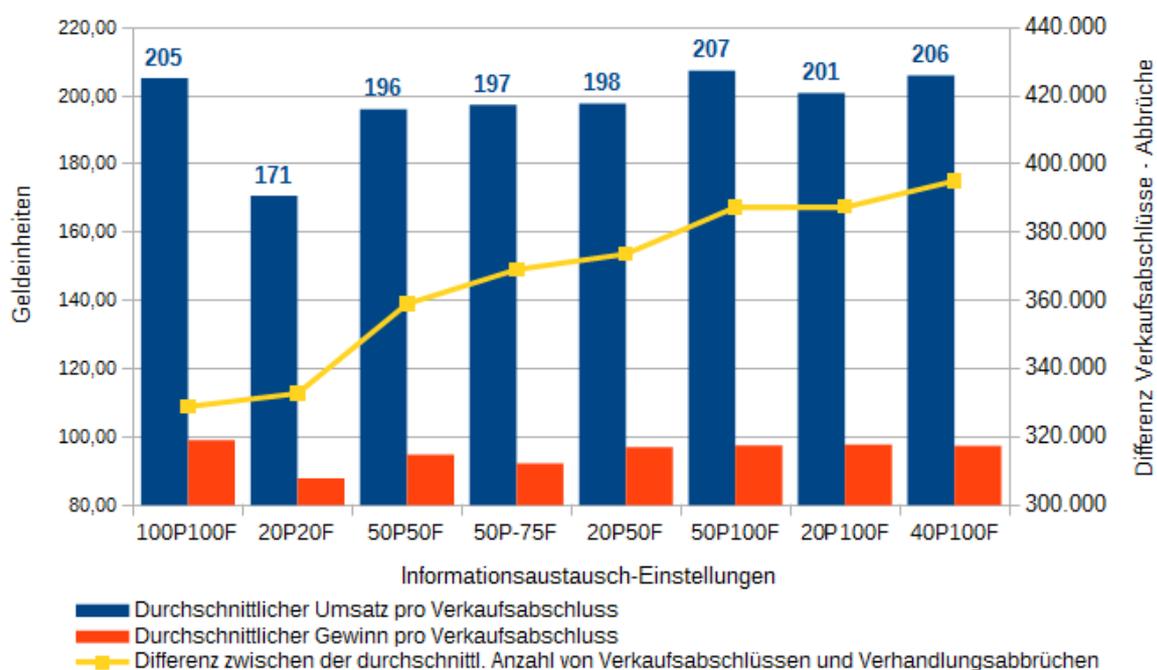


Abbildung 120: Informationsaustausch-Effekte 2

Anhand von Abbildung 120 wird aber auch deutlich, dass ein hoher Informationsaustausch nicht nur negative Effekte hat. So konnten bei den Szenarien mit den Einstellungen 100P100F, 50P100F und 40P100F, die von den getesteten Einstellungen diejenigen mit den meisten geteilten Informationen waren, pro Transaktion auch die höchsten Umsätze und Gewinne realisiert werden. Es kann davon ausgegangen werden, dass der hohe Informationsaustausch zu einer hohen Markttransparenz führt, die letztlich preistreibend wirkt.

In einem Oligopol könnte sich diese Transparenz auch preismindernd auswirken. Da es sich bei dem allen Simulationen zugrundeliegenden Basis-Szenario (siehe Abschnitt 5.2.1) von der Topologie her jedoch um eine Art bilaterales Oligopol⁷⁵ handelt und die Konsumenten auf der letzten Stufe der Wertschöpfungsstruktur gemäß ihrer Nachfragefunktion auch bei einem relativ hohen Preisniveau weiter Güter nachfragen, kommt es bei den Simulationen nicht dazu.

⁷⁵ Eine Marktform, bei der eine relativ geringe Zahl an Anbietern einer ähnlich geringen Zahl an Nachfragern gegenüber steht.

Der Verlauf der Kurve, die die Entwicklung der Differenz zwischen der Anzahl an Verkaufsabschlüssen und Verhandlungsabbrüchen beschreibt, zeigt ebenfalls, dass weder eine extrem hohe (100P100F) als auch eine relativ schlechte (20P20F) Markttransparenz für die System-Performanz optimal sind, da im Vergleich eine mittlere Markttransparenz (40P100F) zu einem Anstieg des Differenzbetrages um beinahe zwanzig Prozent führt.

Abbildung 121 bietet eine Detail-Übersicht der Simulationsergebnisse, die bei der Kombination von GMU-Parametrisierungen mit den verschiedenen getesteten Informationsaustausch-Einstellungen erzielt wurden. Dabei fällt auf, dass mit den Einstellungen 40P100F und 50P100F jeweils besonders wünschenswerte Ergebnisse erzielt werden, mit 20P20F und 100P100F in den meisten Fällen eher schlechte.

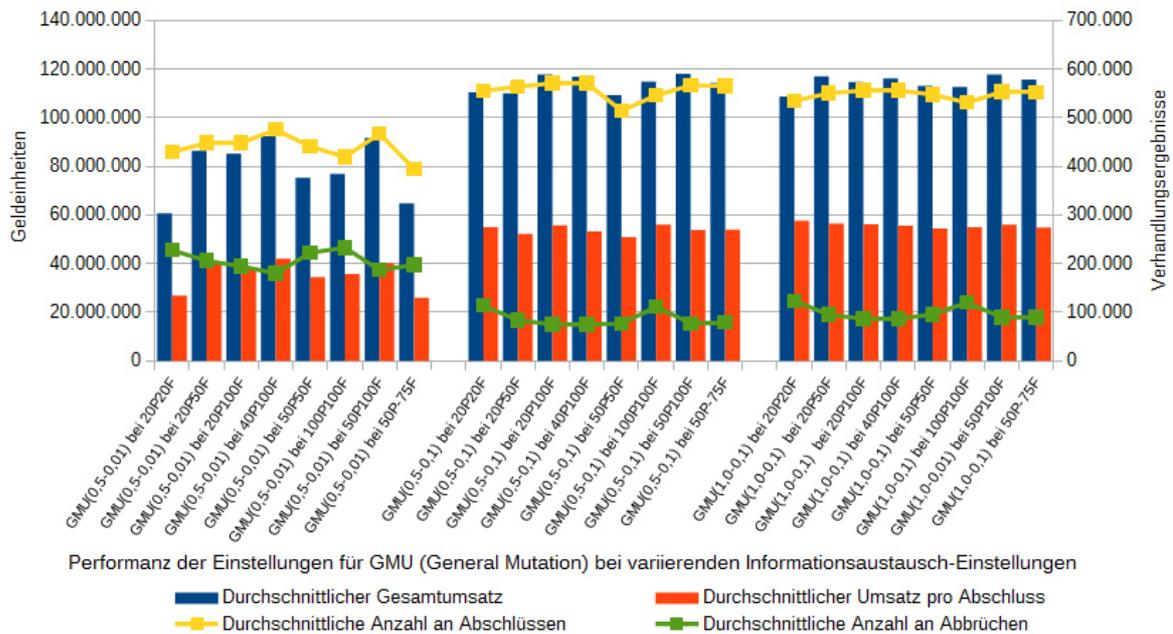


Abbildung 121: Informationsaustausch und General Mutation (GMU)

Analog hierzu bietet auch Abbildung 122 eine entsprechende Detail-Übersicht der SMU-Simulationsergebnisse bei Variation der Informationsaustausch-Einstellungen. Auch beim Einsatz von SMU sind die Einstellungen 40P100F und 50P100F jeweils empfehlenswert. Von den Einstellungen 20P20F, 20P50F und 100P100F ist dagegen eher abzuraten.

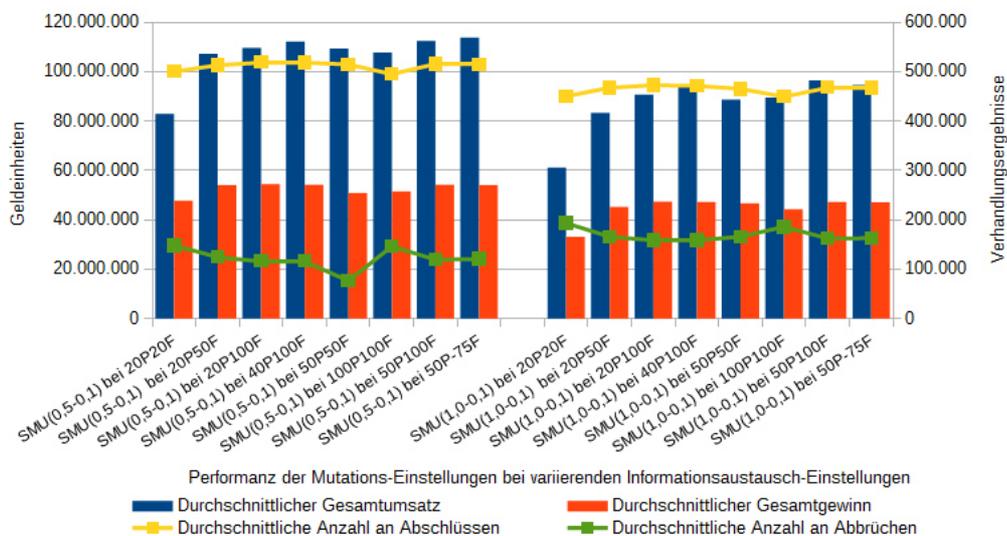


Abbildung 122: Informationsaustausch und Single Mutation (SMU)

Abbildung 123 gibt einen Überblick über alle Simulationsergebnisse der Testreihe. Diese Abbildung findet sich auch in vergrößerter Form in Abschnitt 11.2 im Anhang.

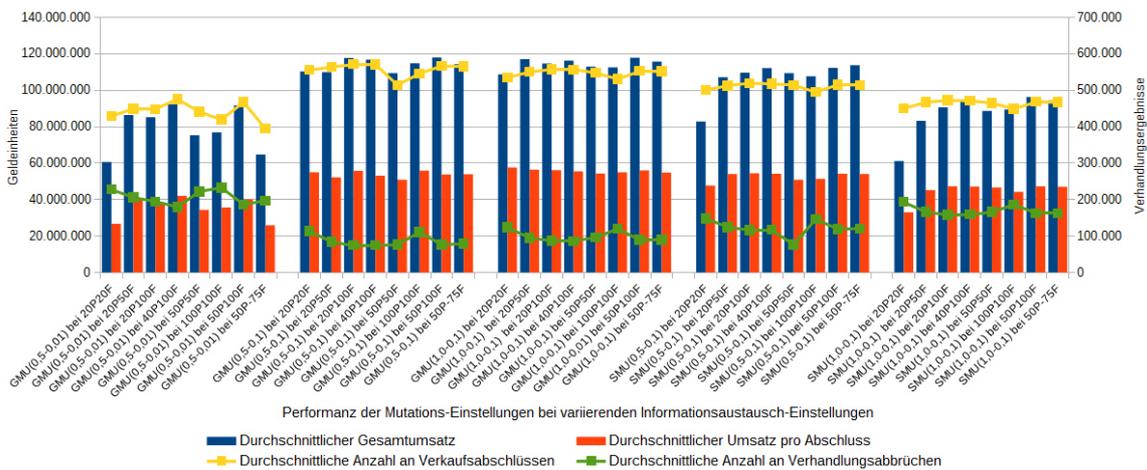


Abbildung 123: Gesamtübersicht kombinierter Effekte von variierenden Informationsaustausch- und Mutations-Einstellungen im Detail

Über alle Informationsaustausch-Einstellungen hinweg ist die Parametrisierung der Mutation mit der Einstellung GMU(0,5-0,1) eine gute Wahl, wie aus der auf Aggregationen beruhenden Abbildung 124 ersichtlich ist.

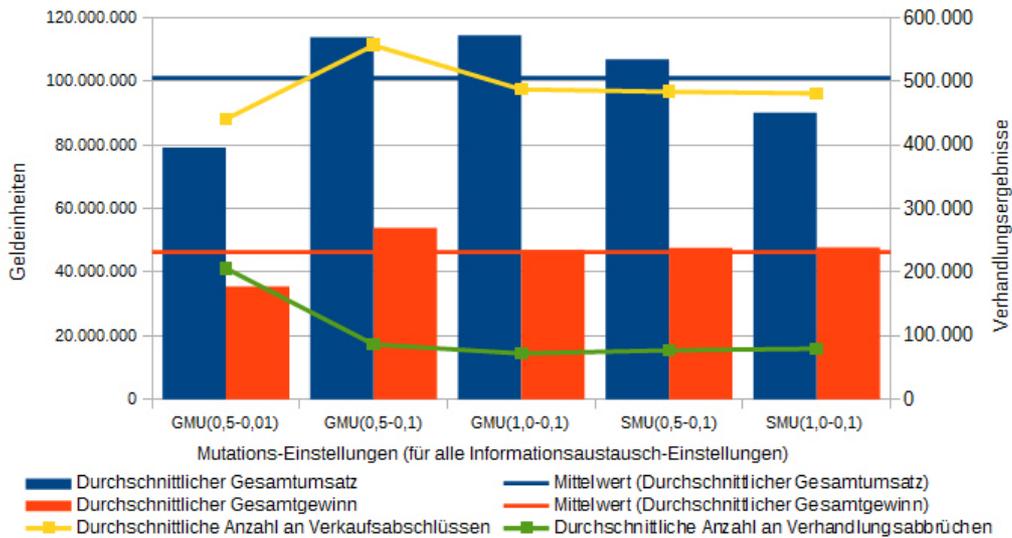


Abbildung 124: Ermittlung der über alle Informationsaustausch-Einstellungen hinweg besten Mutations-Einstellung

8.5.2 Informationsaustausch bei heterogenem Lernen

Es wurde untersucht, wie sich eine in Bezug auf die Informationsaustausch- und Lern-Einstellungen heterogene, das heißt sich für die verschiedenen Agenten-Typen uneinheitliche Parametrisierung, auf die Performanz der Agenten-Typen auswirkt.

Hierzu wurden die Prozessor-Produzenten-Agenten auf der dritten beziehungsweise mittleren Stufe des Wertschöpfungsnetzwerks mit Lernfähigkeiten ausgestattet, während bei allen übrigen Agenten-Typen die Lern-Komponente abgeschaltet wurde. Darüber hinaus wurden in den Szenarien Fitness-Methoden- und ProfitWish-Einstellungen variiert. Die resultierenden Szenarien wurden mit variierenden Informationsaustausch-Einstellungen simuliert. Weitere Details zur Testreihe finden sich in Abschnitt 11.1.11.

Entsprechende Tests werden im MACSIMA-Kontext *Stufentests* genannt. Über alle simulierten Szenarien hinweg wurde der Durchschnittsumsatz und –gewinn für die mittlere, dritte Stufe des Wertschöpfungsnetzwerks ermittelt (siehe Abbildung 125 und Abbildung 126).

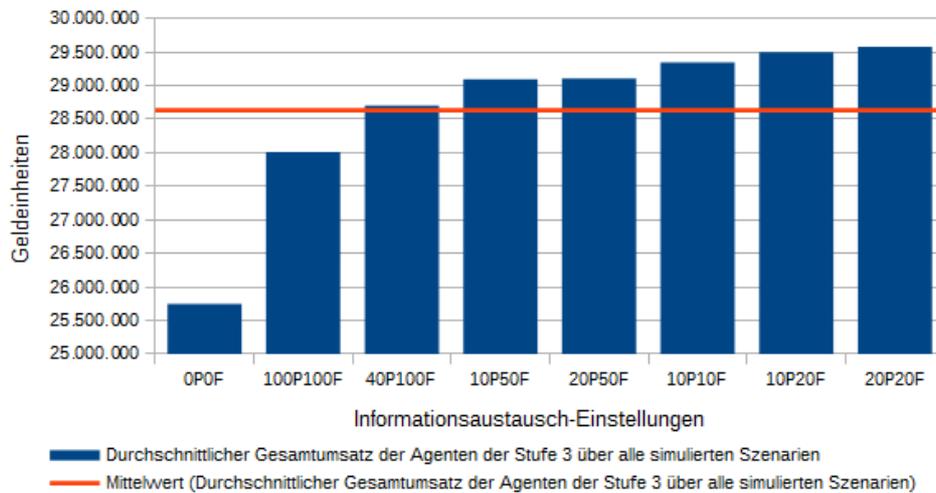


Abbildung 125: Gesamtumsatz der Agenten auf Stufe 3 in Abhängigkeit von den Informationsaustausch-Einstellungen

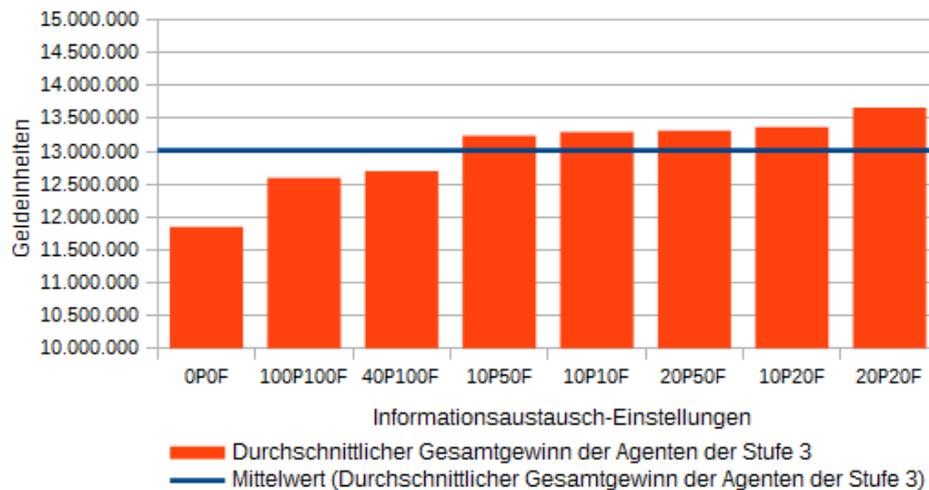


Abbildung 126: Gesamtgewinn der Agenten auf Stufe 3 in Abhängigkeit von den Informationsaustausch-Einstellungen

Die bei den verschiedenen Informationsaustausch-Einstellungen erzielten Gesamtumsätze und –gewinne sind ihrer Höhe nach geordnet.

Wie im vorhergehenden Abschnitt zeigt sich, dass ein hoher Informationsaustausch zu schlechten Performanz-Kennzahlen führt. Der maximal mögliche Informationsaustausch 100P100F erzielt das zweitschlechteste Umsatz- und Gewinn-Ergebnis (vergleiche Abbildung 126).

Jedoch führt ein echter Informationsaustausch in beliebigem Umfang stets zu besseren Umsatz- und Gewinn-Kennzahlen als ein ganz ausgeschalteter Informationsaustausch (OP0F), der zum jeweils schlechtesten Ergebnis führt. Ein Informationsaustausch scheint also immer vorteilhaft zu sein, aber das richtige Maß ist entscheidend, damit es weder beim evolutionären Lernen zu einem Overfitting-Effekt noch zu einer zu hohen Markttransparenz kommt.

20P20F schneidet am besten ab und kann bzgl. des durchschnittlich erzielten Gesamtumsatzes OP0F um 14,8% übertreffen und 100P100F um ca. 5,6%.

Die Durchschnittsumsätze von 20P50F und 10P10F differieren beim Gesamtumsatz um 0,8% und tauschen beim Gesamtgewinn die Plätze (siehe Abbildung 126). 10P10F liegt beim Umsatz vorn, wird aber beim Gewinn von 20P50F um 0,1% minimal übertroffen. 20P20F kann bzgl. des Durchschnittsgewinns 0P0F um 15,4% übertreffen und 100P100F um ca. 8,5%.

Bei den heterogenen Szenario-Einstellungen der Testreihe schneidet der moderate Informationsaustausch 40P100F besser als die Einstellung 100P100F ab. Jedoch wird mit 40P100F kein gutes Ergebnis mehr erzielt, sondern stattdessen nur noch ein deutlich unterdurchschnittliches. Die besten Werte der Performanz-Kennzahlen werden hingegen von der Einstellung 20P20F erzielt, die in der Testreihe des vorhergehenden Abschnitts für die schlechtesten Ergebnisse verantwortlich war.

Hieran zeigt sich, dass eine gute Parametrisierung von Agenten nicht isoliert betrachtet werden darf, sondern von Interdependenzen zwischen den Parameter-Einstellungen und vor allem von den Umweltbedingungen, in denen die Agenten handeln, abhängig ist.

Ein möglicher Erklärungsansatz besteht darin, dass die Produzenten-Agenten der dritten Stufe in der aktuellen Testreihe nicht mehr von anderen lernenden Agenten umgeben sind, die versuchen, sich zu adaptieren und dadurch auch unterschiedliche Verhandlungsstrategien einsetzen. Sondern stattdessen sind sie von nicht-adaptiven Agenten umgeben, die stur genau eine Verhandlungsstrategie verfolgen, mit der sie jeweils initialisiert wurden. Auch wenn diese Verhandlungsstrategien durch eine Randomisierung der zugrundeliegenden Gene voneinander unterschiedlich sind, so führt dies doch zu einer deutlich verminderten Variabilität beziehungsweise Flexibilität der Verhandlungspartner auf der vor- und nachgelagerten Wertschöpfungsstufe.

Unter solchen Umfeld-Bedingungen scheint der Effekt eines Verlusts an Vielfalt im zur Verfügung stehenden Gen-Material bereits auf einem niedrigeren Informationsaustausch-Niveau einzusetzen als bei der Testreihe des vorhergehenden Abschnitts. Dennoch ist auch unter den genannten Bedingungen ein noch niedrigerer 10P10F-Informationsaustausch der Einstellung 20P20F unterlegen.

Abbildung 127 schlüsselt die aggregierten Umsatz-Ergebnisse aus Abbildung 124 weiter nach den verwendeten Fitness-Einstellungen auf. Auch hier sind die Ergebnisse der Informations-Einstellungen gemäß der Höhe ihrer aggregierten Kennzahlen-Ergebnisse geordnet.

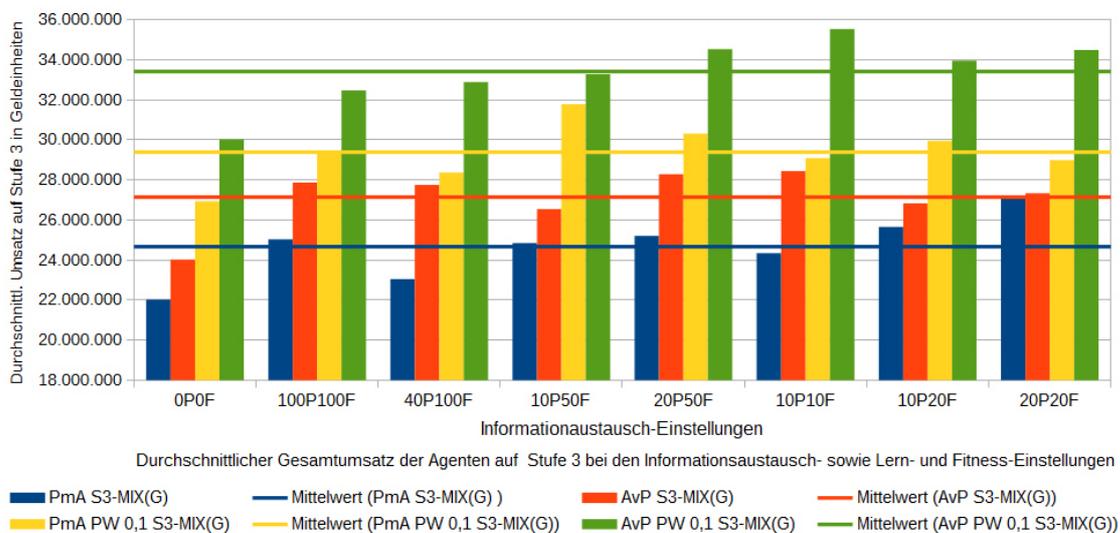


Abbildung 127: Umsätze der Agenten auf Stufe 3 in Abhängigkeit von den Informationsaustausch- und Fitness-Einstellungen

Die Agenten, die die Fitness-Methode PmA verwenden, schneiden sowohl mit als auch ohne einen ProfitWish von 10% (Einstellung PW 0,1) in Bezug auf den von ihnen durchschnittlich erwirtschafteten Umsatz bei allen getesteten Informationsaustausch-Einstellungen schlechter ab als die Agenten, die AvP verwendenden (und ansonsten gleich parametrisiert sind).

Durch eine Kombination der AvP-Einstellung mit einem ProfitWish von 10% werden unter allen Informationsaustausch-Einstellungen jeweils die besten Umsatz- und Gewinn-Ergebnisse erzielt (vergleiche auch Abbildung 128).

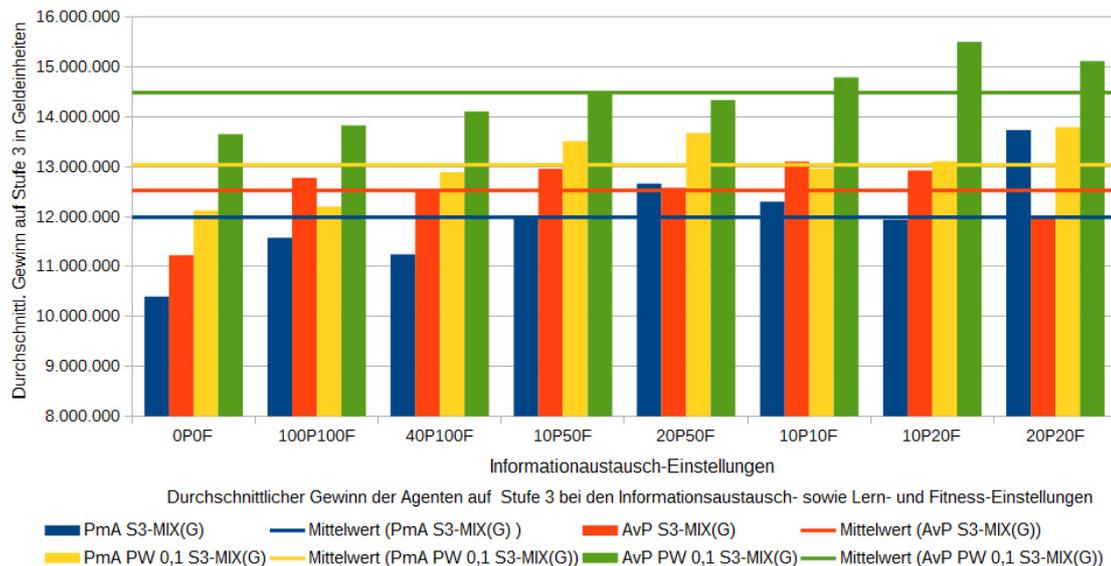


Abbildung 128: Gewinne der Agenten auf Stufe 3 in Abhängigkeit von den Informationsaustausch- und Fitness-Einstellungen

Zusammengefasst hat sich gezeigt, dass die heterogenen Stufentest-Einstellungen zwar tendenziell zu ähnlichen Ergebnissen wie die homogenen Einstellungen bei vorhergehenden Testreihen führen.

Bei genauerer Analyse offenbart sich jedoch, dass bei sich ändernden Umweltbedingungen beziehungsweise sich änderndem Verhalten der Agenten im Umfeld die Parametrisierung davon betroffener einzelner Agenten und Agenten-Gruppen darauf überprüft werden muss, ob aus ihnen weiterhin ein performantes Verhalten emergiert. Gegebenenfalls muss dann erneut eine ausbalancierte Anpassung der Parameter-Einstellungen vorgenommen werden.

8.6 Selbstkoordination mit MACSIMA-Erweiterungen

In den vorangehenden Abschnitten dieses Kapitels wurden die Auswirkungen verschiedener variierender Parametereinstellungen zunächst isoliert voneinander untersucht. Nach Untersuchungen zur Bedeutung der einzelnen Gen-Startwerte und ihrer Kombinationen konzentrierten sich die daran anschließenden Testreihen der Evaluation thematisch auf die verschiedenen Phasen des von den Agenten genutzten evolutionären Lernens.

Aus den Erkenntnissen dieser Untersuchungen wurden Empfehlungen für die Parametrisierung und Randomisierung der Gen-Werte sowie für die Feinjustierung der Lern-Komponente abgeleitet und in den jeweils nachfolgenden Testreihen berücksichtigt.

Danach folgten Testreihen zu den Effekten variierender Informationsaustausch-Einstellungen, die in starkem Maße Interdependenzen mit der Parametrisierung anderer für die System-

Performanz wesentlicher Methoden untersuchten. Hierbei standen vor allem die Einflüsse von Mutations- und Fitness-Methoden im Fokus.

In diesem Abschnitt wird nun eine Testreihe durchgeführt, die den Zweck hat, zu untersuchen, welche Steigerung der System-Performanz beziehungsweise der Selbstkoordinationsleistung im Vergleich zu den mit AVALANCHE durchgeführten Experimenten erreicht werden kann, wenn von den Geschäftsagenten verschiedene in MACSIMA implementierte Erweiterungen der AVALANCHE-Konzepte sowie wesentliche Lern-Einstellungen genutzt werden. Die dazugehörigen Parameter-Einstellungen orientieren sich dabei an den bislang im Rahmen der Evaluation gewonnenen Erkenntnissen. Untersuchungsgegenstand ist also nicht die Performanz von einzelnen Agenten oder Agenten-Gruppen, sondern des agentenbasierten Wertschöpfungsnetzwerks als Ganzem.

Hierzu wurden folgende Arten von Szenarien entworfen und simuliert:

- Szenarien, bei denen die Agenten die Erweiterungen des MACSIMA-Systems (Lernen, HFF und Informationsaustausch) nicht nutzen können.

In diesen Szenarien sind die Geschäftsagenten daher nur auf die Nutzung von Fähigkeiten beschränkt, die mit denen der Agenten im AVALANCHE-System (siehe [Eymann 2000]) vergleichbar sind.

Die entsprechenden Szenarien nennen sich Eymann_1 und Eymann_2 (siehe Abschnitt 11.1.1).

- Szenarien, bei denen die Agenten die Erweiterungen des MACSIMA-Systems (Lernen, HFF und Informationsaustausch) nutzen können. Innerhalb dieser Gruppe an Szenarien lassen sich weiterhin Szenarien unterscheiden,
 - in denen die Agenten nur intern lernen (angezeigt durch das Kürzel INT(G)) und Szenarien,
 - in denen die Agenten gemischt lernen (angezeigt durch das Kürzel MIX(G)).

Alle Szenarien der Testreihe beinhalten die Initialisierung von 50 Agenten, die alle den 2-Pool-separierten Lern-Modus nutzen. Es wurde die Informationsaustausch-Einstellung 50P100F für den Austausch von transaktionsbezogenen Informationen wie dem Verkaufspreis und den verwendeten Verhandlungsstrategie-Parametern zwischen den Agenten einer Wertschöpfungsstufe verwendet. Eine detaillierte Beschreibung der Testreihe mit den Spezifikationen der simulierten Szenarien wird in Abschnitt 11.1.12 gegeben.

Die in Abbildung 129 dargestellten Ergebnisse der Testreihe vergleichen die Selbstkoordinationsleistung der Geschäftsagenten in den Szenarien Eymann_1 und Eymann_2 mit der Selbstkoordinationsleistung in den übrigen Szenarien der Testreihe, die die oben angegebenen und in Abschnitt 11.1.12 im Detail beschriebenen Eigenschaften und Einstellungen aufweisen.

In den Szenarien Eymann_1 und Eymann_2 werden die Agenten dabei zum Simulationsstart homogen mit Gen-Startwerten initialisiert, die den Erläuterungen in Abschnitt 11.1.1 zu entnehmen sind und die den Gen-Startwert-Parametrisierungen von in [Eymann 2000] vorgenommenen Experimenten entsprechen.

Es handelt sich dabei um die beiden aus [Eymann 2000] entnommenen Gen-Startwert-Parametrisierungen, die bei mit MACSIMA durchgeführten Simulationen, bei denen die MACSIMA-Erweiterungen ausgeschaltet und die Agenten daher auf die in AVALANCHE verfügbaren Fähigkeiten beschränkt waren, das jeweils das hinsichtlich der Performanz-Kennzahlen quantitativ beste und schlechteste Simulationsergebnis erzeugt haben.

Abbildung 129 zeigt, dass alle MACSIMA-Erweiterungen, beginnend beim Internen Lernen, zu einer Steigerung der Selbstkoordinationsleistung führen. Der durchschnittliche Gesamtumsatz und der durchschnittliche Gesamtgewinn können bereits durch das *interne Lernen*, aber noch stärker durch das *gemischte Lernen* in Kombination mit der HFF-Methode gesteigert und das Verhältnis von Verkaufsabschlüssen zu Verhandlungsabbrüchen günstiger gestaltet werden.

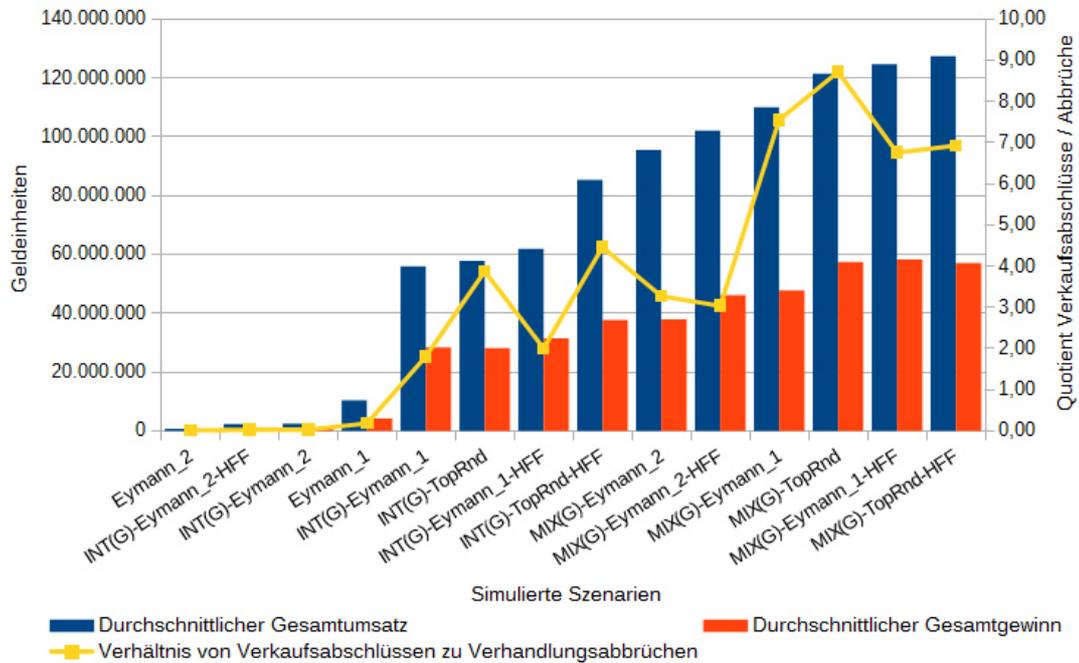


Abbildung 129: Vergleich der Selbstkoordinationsleistung von an [Eymann 2000] orientierten Szenarien mit Szenarien mit MACSIMA-Erweiterungen, Teil 1

Das Szenario mit der höchsten Performanz, MIX(G)-TopRnd-HFF verwendet dabei eine Kombination aus gemischtem Lernen, einer TOP-Randomisierung der Gen-Startwerte gemäß Abschnitt 11.1.5.2, dem Highest-Fitness-First-Verfahren (siehe 5.4.5.1) und dem Informationsaustausch 50P100F zwischen den Agenten einer Wertschöpfungsstufe.

Weiterhin ist zu bemerken, dass MIX(G)-Szenarien immer besser als die vergleichbaren INT(G)-Szenarien abschneiden. Dies unterstreicht die Vorteile des realisierten Konzepts des Gemischten Lernens. Zudem schneiden Szenarien mit eingeschalteter HFF-Methode auch immer besser ab, als die korrespondierenden Szenarien, bei denen HFF ausgeschaltet ist.

Die gleichen Aussagen gelten auch, wenn die in Abbildung 130 dargestellten Ausprägungen weiterer Performanz-Kennzahlen betrachtet werden. Die Umsätze und Gewinne pro Transaktion werden signifikant gesteigert.

Ebenso steigt der Anteil der mit einem Verkaufsabschluss endenden Verhandlungen an der Gesamtzahl der Verhandlungen bzw. die Differenz zwischen Verkaufsabschlüssen und Verhandlungsabbrüchen (vergleiche auch Abbildung 131).

Auch bezüglich dieser Performanzkennzahlen erzielen die Szenarien, die *gemischtes Lernen* mit der HFF-Methode und/oder einer Top-Randomisierung der Gen-Startwerte kombinieren, die beste Selbstkoordinationsleistung.

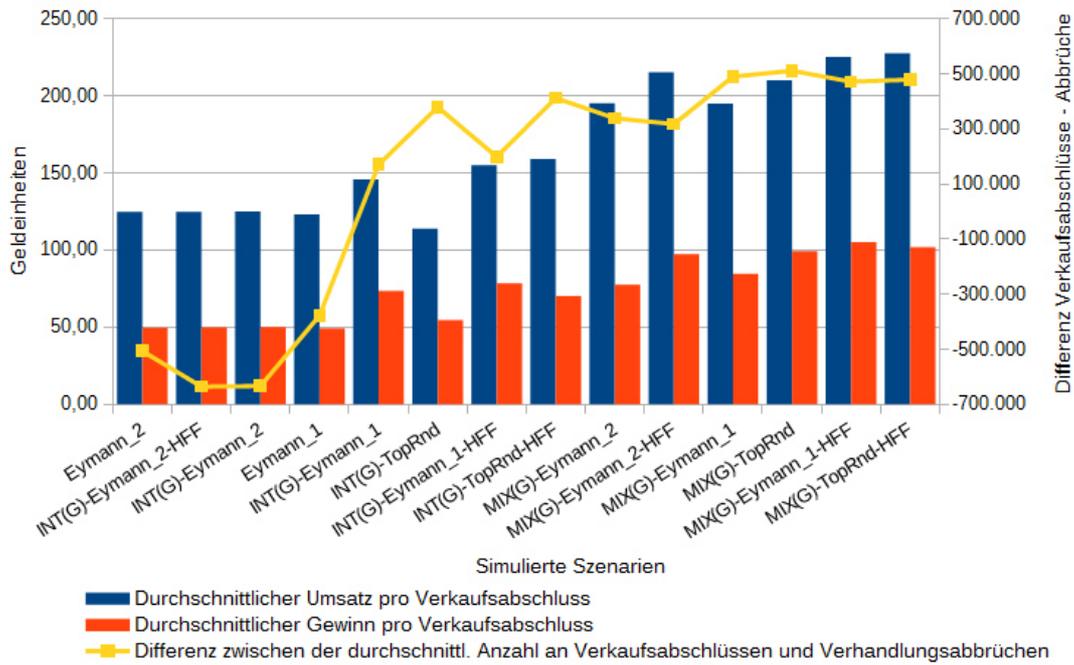


Abbildung 130: Vergleich der Selbstkoordinationsleistung von an [Eymann 2000] orientierten Szenarien mit Szenarien mit MACSIMA-Erweiterungen, Teil 2

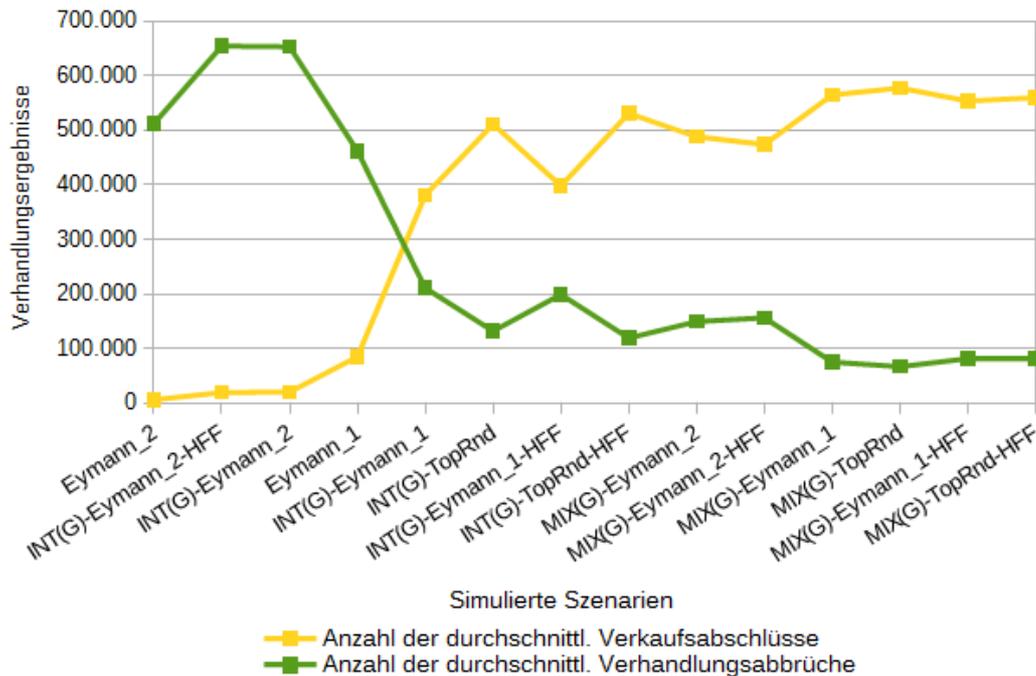


Abbildung 131: Verkaufsabschlüsse und Verhandlungsabbrüche in an [Eymann 2000] orientierten Szenarien und Szenarien mit MACSIMA-Erweiterungen

Zusammenfassend lassen sich aus der Analyse dieser Testreihe die folgenden Erkenntnisse ableiten:

- Internes EA-Lernen (INT(G)) erhöht die Gesamtleistung des Systems im Vergleich zu den an [Eymann 2000] orientierten Szenarien Eymann_1 und Eymann_2 schon signifikant.

- Wird die HFF-Option angeschaltet, so ergibt sich nochmals ein Leistungssprung.
- Erlaubt man den Agenten einen gruppenbezogenen Informationsaustausch (mit der Einstellung 50P100F) innerhalb ihrer Wertschöpfungsstufe und die Nutzung von Gemischtem Lernen (MIX(G)-Szenarien), so werden Leistungssteigerungen von circa 1.000 % im Vergleich zum mit AVALANCHE-Einstellungen simulierten Eymann_1-Szenario erreicht (s. Abbildung 129).
- Der Austausch von transaktionsbezogenen Informationen wie dem Verkaufspreis und den verwendeten Verhandlungsstrategie-Parametern zwischen den Agenten erhöht ihre Performanz signifikant.

8.7 Zusammenfassung der Evaluationsergebnisse

Die durchgeführten Analysen begannen in Abschnitt 8.2 mit einem qualitativen Vergleich des Verhaltens der Simulationssysteme AVALANCHE und MACSIMA. Es wurde gezeigt, dass bei gleichen Simulationseinstellungen in beiden Systemen die Emergenz eines von der Tendenz her sehr ähnlichen Agenten- und Systemverhaltens beobachtet werden kann, wenn die Fähigkeiten der MACSIMA-Geschäftsagenten auf die bereits in AVALANCHE verfügbaren limitiert sind.

Darauffolgend wurden in Abschnitt 8.3 bei ausgeschalteter Lernkomponente der Agenten die Effekte der Variation der Startwerte von Genen betrachtet – sowohl isoliert voneinander, als auch in Kombination. Die isolierte Variation des Gens Acquisitiveness (AQ) in Abschnitt 8.3.1 und des Gens Satisfaction (SF) in Abschnitt 8.3.2 zeigte, dass wie in AVALANCHE auch in MACSIMA bereits die Variation einzelner Gen-Werte große Auswirkungen auf das Systemergebnis hat.

Die Analyse einer großen Menge beziehungsweise Stichprobe an verschiedenen Kombinationen von Gen-Startwerten in Abschnitt 8.3.3 führte zu dem Ergebnis, dass die von den Genen induzierten Effekte von ihrer Kombination mit anderen Gen-Werten abhängen und von Szenarien mit einer Parametrisierung der Gen-Startwerte, die gleichverteilt im Wertebereich

$$\langle AC = [0,1; 0,6]; DC = [0,4; 0,9]; DJ = [0,4; 0,9]; SF = [0,4; 0,9]; WM = [0,1; 0,6] \rangle,$$

liegen, eine besonders hohe Gesamtperformanz in Bezug auf die Ergebniskennzahlen Gesamtumsatz und Gesamtgewinn zu erwarten ist.

Darauffolgend wurde in umfangreichen Testreihen untersucht, wie sich die Einstellung der wesentlichen Parameter der Lern-Komponente der MACSIMA-Agenten auf die wichtigsten Performanz-Kennzahlen eines Wertschöpfungsnetzwerks auswirken. In den meisten durchgeführten Testreihen nutzten die Agenten dabei das 2-Pool-separierte Lernen (siehe Abschnitt 5.4.1.2).

Dabei stellte sich in Abschnitt 8.4.1 heraus, dass es sich hinsichtlich der beobachteten Ergebnisvariablen über alle betrachteten Pool-Größen hinweg vorteilhaft auswirkt, wenn Pool-Größe und Populations-Größe eine Relation von 1:6 oder 1:7 aufweisen. Besonders gute Ergebnisse konnten mit den Pool-Größen 3 und 5 erzielt werden.

Bei der Untersuchung der Effekte verschiedener Fitness-Methoden in Abschnitt 8.4.2 zeigte sich, dass für die Anwendungsdomäne die Verwendung der Fitness-Methoden MdP und AvP besonders empfohlen werden kann. Besonders gute Ergebnisse konnten in Kombination mit der P:Pn-Einstellung P5Pn30 erzielt werden. Vom Einsatz der PmA-Methode ist hingegen eher abzuraten.

In Abschnitt 8.4.3 wurde untersucht, wie sich das evolutionäre Lernen bei Verwendung der acht Selektions-Methoden BiC, DMS, DMS2, DMS3, DtA, DtS, DWS und RWS auf den durchschnittlichen Gesamtumsatz des Wertschöpfungsnetzwerks auswirkt. Die erzielten Ergebnisse wurden mit den Ergebnissen verglichen, die bei ausgeschalteter Lernkomponente generiert wurden.

Daraus konnte die Erkenntnis abgeleitet werden, dass das evolutionäre Lernen vorteilhaft ist, da Szenarien, in denen die Agenten nicht lernen können, im Durchschnitt signifikant schlechter abschneiden als Szenarien, in denen die Agenten lernen können und die getesteten Selektions-Methoden verwenden. Auf der Grundlage der Testergebnisse können die Selektions-Methoden DMS, DtS, DWS, BiC und RWS empfohlen werden. Die das RWS-Verfahren verwendenden Szenarien generieren dabei den höchsten Durchschnittsumsatz.

Die neu entwickelte DMS-Methode (siehe Abschnitt 5.4.3.2) konnte insgesamt nur durchschnittlich gute Ergebnisse erzielen. Es fiel jedoch auf, dass ihre Verwendung in Szenarien mit einer empfehlenswerten Startwert-Parametrisierung der Gene dazu führte, dass ein sehr hoher Durchschnittsumsatz erzielt werden konnte. Dies weist darauf hin, dass die Selektions-Methode DMS sehr gut dazu geeignet ist, eine für den Adaptionsprozess vorteilhafte genetische Vielfalt im Gen-Material zu erhalten.

Die Analyse der Effekte der Rekombinations-Methoden nPC und RaC in Kombination mit mehreren Selektions-Methoden in Abschnitt 8.4.4 ergab, dass die RaC-Methode im Vergleich zur nPC-Methode zu schlechteren Ergebnissen führt. Für das Anwendungsszenario konnte ein n-Punkt-Crossover (nPC) mit nur einem oder wenigen Schnittpunkten, also zum Beispiel mit den Einstellungen nPC(1-2-X) oder nPC(1-1-X), als besonders empfehlenswert eingestuft werden. Dabei konnte die Einstellung nPC(1-1-X) in Kombination mit der Selektions-Methode RWS hinsichtlich der Ergebnisvariablen Gesamtgewinn und Gesamtumsatz die stabilsten Ergebnisse generieren. Neben der RWS-Methode bietet sich auch die BiC-Methode für die Verwendung in Kombination mit dem n-Punkt-Crossover an.

Im Anschluss an diese Untersuchung wurde in Abschnitt 8.4.5 die Performanz des Wertschöpfungsnetzwerks bei 1-Pool-basiertem und 2-Pool-separiertem Lernen miteinander verglichen. Dabei wichen die Simulationsergebnisse beim 2-Pool-separierten Lernen, was die Anzahl an erzielten Verkaufsabschlüssen und Verhandlungsabbrüchen angeht, weniger stark voneinander ab als beim 1-Pool-basierten Lernen. Es zeigte sich, dass Umsatz und Gewinn beim 2-Pool-separierten Lernen durchgehend besser als beim 1-Pool-basierten Lernen ausfallen und die Performanz eines agentenbasierten Wertschöpfungsnetzwerks davon profitiert, wenn alle an ihm partizipierenden Agenten den 2-Pool-separierten Lern-Modus verwenden. Dabei führte die DMS-Methode im 2-Pool-Modus zu den meisten Verhandlungen.

Die Variation der Ersetzungsschema-Einstellungen in Abschnitt 8.4.6 zeigte, dass die Wahl der Ersetzungsschema-Parametereinstellung eine gravierende Auswirkung auf den durchschnittlichen Gesamtumsatz und Gesamtgewinn des agentenbasierten Wertschöpfungsnetzwerks hat. Die Verwendung des Ersetzungsschemas Schwacher Elitismus (WE) kann besonders empfohlen werden, da mit ihm innerhalb der Testreihe bei der Einstellung WE(3) ein durchschnittlicher Gesamtumsatz erzielt werden konnte, der über fünfmal höher war als der Gesamtumsatz, den die Agentengesellschaft ohne die Verwendung eines Ersetzungsschemas generieren konnte.

In Abschnitt 8.4.7 wurden die Auswirkungen der Mutations-Methoden *General Mutation (GMU)* und *Single Mutation (SMU)* auf die System-Performanz analysiert. Die Analyse der Testreihe ergab, dass die GMU-Methode in den meisten Fällen besser abschneidet als die SMU-Methode. Mit der Einstellung GMU(0,5-0,1) konnte eine besonders performante Parameter-Einstellung für die Mutation identifiziert werden. Es konnte beobachtet werden,

dass die Mutations-Einstellungen besonders große Auswirkungen auf die Systemperformanz haben und eine gute Ausbalancierung von Mutations-Rate und –Weite wesentlich ist, um eine performante Systemleistung und eine gute Ergebnisstabilität zu erzielen.

Danach wurde untersucht, wie sich verschiedene Informationsaustausch-Modi auf die Performanz-Kennzahlen des Gesamtsystems auswirken. Dabei wurde in Abschnitt 8.5.1 zunächst das Zusammenspiel der Informationsaustausch-Modi mit verschiedenen ausgewählten Mutations-Einstellungen betrachtet. Es konnte gezeigt werden, dass sowohl bei der Verwendung von GMU als auch bei der Verwendung von SMU weder ein intensiver (100P100F) noch ein stark eingeschränkter Informationsaustausch (20P20F) zu einer guten System-Performanz führt. Stattdessen führt ein eher moderater Informationsaustausch (mit den Einstellungen 40P100F und 50P100F) zu wünschenswerten Ergebnissen. Über alle Informationsaustausch-Einstellungen hinweg stellte sich wie im vorhergehenden Abschnitt eine Parametrisierung der Mutation mit der Einstellung GMU(0,5-0,1) als empfehlenswert heraus.

In 8.5.2 wurde untersucht, wie sich eine in Bezug auf die Informationsaustausch- und Lern-Einstellungen uneinheitliche Parametrisierung der verschiedenen Agenten-Typen auf deren Performanz auswirkt. Nur die Prozessor-Produzenten-Agenten auf der mittleren Stufe des Wertschöpfungsnetzwerks wurden mit Lernfähigkeiten ausgestattet. Neben ihren Informationsaustausch-Einstellungen wurden auch die von ihnen verwendeten Fitness-Methoden und ProfitWish-Einstellungen variiert. Es zeigte sich, dass ein Informationsaustausch in beliebigem Umfang stets zu besseren Umsatz- und Gewinn-Kennzahlen führt als ein ganz ausgeschalteter Informationsaustausch (mit der Quote 0P0F). Aber es muss eine sinnvolle, an die Situation angepasste Einstellung für die Informationsaustausch-Quote verwendet werden, damit es beim Evolutionären Lernen zu keinem Overfitting-Effekt kommt. Daraus, dass die besten Werte der Performanz-Kennzahlen nicht wie in der vorhergehenden Testreihe von der Informationsaustausch-Einstellung 40P100F, sondern von der Einstellung 20P20F erzielt werden konnten, lässt sich ableiten, dass eine gute Parametrisierung der Agenten nicht isoliert betrachtet werden darf, sondern von Interdependenzen zwischen den Parameter-Einstellungen und vor allem von den Umweltbedingungen, in denen die Agenten handeln, abhängig ist.

In Abschnitt 8.6 wurde untersucht, welche Steigerung der System-Performanz beziehungsweise der Selbstkoordinations-Leistung im Vergleich zu den mit AVALANCHE durchgeführten Experimenten erreicht werden kann, wenn von den Geschäftsagenten in MACSIMA implementierte Erweiterungen der AVALANCHE-Konzepte genutzt werden. Es zeigte sich, dass die System-Performanz durch die Lernfähigkeit der Agenten signifikant gesteigert werden kann. Bereits die Verwendung des internen EA-Lernen (INT(G)) konnte die Gesamtleistung des Systems deutlich steigern. Durch die zusätzliche Verwendung des *Highest-Fitness-First-Verfahrens* (siehe Abschnitt 5.4.5.1) konnte die Leistung weiter verbessert werden. Durch die Nutzung von gemischtem Lernen (MIX(G)-Szenarien) bei gleichzeitiger Realisierung eines gruppenbezogenen Informationsaustauschs (mit der Einstellung 50P100F) innerhalb der verschiedenen Wertschöpfungsstufen konnte im Vergleich zum dem mit AVALANCHE-Einstellungen simulierten Eymann_1-Szenario sogar eine Leistungssteigerung von circa 1.000 % erreicht werden.

9 Zusammenfassung & Ausblick

Die wesentlichen Ergebnisse dieser Arbeit werden hier zusammengefasst und im Anschluss wird ein kurzer Ausblick auf weitere Fragestellungen gegeben, die auf Basis des MACSIMA-Systems untersucht werden können.

9.1 Zusammenfassung

In dieser Arbeit wurden agentenbasierte Mechanismen und Designprinzipien zur marktlichen, dezentralen Selbstkoordination von agentenbasierten Wertschöpfungsnetzwerken vorgestellt.

Da es sich um ein interdisziplinäres Thema handelt, wurden in der Arbeit zunächst die relevanten, das Kernthema betreffenden Ausschnitte der Forschungsgebiete Agententechnologie und Multiagenten-Systeme, Supply Chain Management und elektronische Märkte sowie ihre Implikationen für die geleistete Forschungsarbeit behandelt.

So führte beispielsweise die Analyse der in der Literatur genannten Supply Chain Modelle in Kapitel 3.2. dazu, sich im Rahmen der abschließenden Evaluation der entwickelten und im MACSIMA-Supply-Chain-Simulation-Manager umgesetzten Konzepte auf ein fünfstufiges Wertschöpfungsnetzwerk mit der in Kapitel 5 beschriebenen Basis-Topologie zu konzentrieren.

Im Anschluss daran wurden Beiträge der Arbeit im Detail dargelegt. Hierbei handelt es sich zum einen um das Konzept einer holonischen Koordinationsarchitektur für Wertschöpfungsnetzwerke (siehe Kapitel 4.3) und um das System MACSIMA (Multi Agent Supply Chain Simulation Manager) zur Simulation und Untersuchung von katallaktischen Selbstkoordinations-Prozessen in agentenbasierten Wertschöpfungsnetzwerken (siehe Kapitel 5).

Dabei fußen die Ursprünge des entwickelten MACSIMA-Systems auf den von Prof. Dr. Torsten Eymann entwickelten und im AVALANCHE-System implementierten Ansätzen. Seine Dissertation *Avalanche - ein agentenbasierter dezentraler Koordinationsmechanismus für elektronische Märkte* [Eymann 2000] wurde mit dem Forschungspreis „Technische Kommunikation“ der Alcatel-Lucent Stiftung ausgezeichnet. Sie steht mit vielen weiteren Veröffentlichungen im für diese Arbeit relevanten interdisziplinären Forschungsbereich in Verbindung (siehe u. a. [Eymann et al. 2003; Eymann & Padovan 1999; Eymann 2013]).

Die grundlegenden Ergebnisse von Professor Dr. Eymann bieten jedoch auch Raum für zusätzliche Erweiterungen und neue Fragestellungen. Wesentliche Erweiterungen, die im vorgestellten MACSIMA-System umgesetzt wurden, sind unter anderem die Erweiterung der Geschäftsagenten mit

- der Fähigkeit, aus gemachten Erfahrungen mit Hilfe *komplexer* evolutionärer Algorithmen zu lernen,
- sozialen Fähigkeiten zum gezielten Austausch von auf dem Markt gemachten Erfahrungen innerhalb von definierbaren Informationskreisen beziehungsweise Gruppen

von Agenten, wobei insbesondere auch der Umfang und die Häufigkeit des Informationsaustauschs eingestellt werden kann, sowie

- korrigierenden Elementen in der Agentenlogik, wie beispielsweise dem in Kapitel 5 vorgestellten Preisautomaten, wenn sich das erlernte Verhalten eines Agenten in eine Richtung entwickelt, die seinen ökonomischen Erfolg und gegebenenfalls sogar seine Existenz gefährdet.
- In Verbindung damit wurden in dieser Arbeit auch drei unterschiedliche Varianten der Deterministic Minority Selection (siehe Abschnitt 5.4.3.2), einer neuartigen Selektions-Methode für evolutionäre Algorithmen sowie das *Highest-Fitness-First-Verfahren (HFF)* (siehe Abschnitt 5.4.5.1) neu entwickelt.

Eine weitere wichtige Erweiterung, mit der die Adaptionfähigkeit der Geschäftsagenten gesteigert werden konnte, ist das Konzept des *2-Pool-separierten Lernens* (siehe Abschnitt 5.4.1 und insbesondere Abschnitt 5.4.1.2). Operiert das adaptive Verhandlungsmodul eines MACSIMA-Agenten im 2-Pool-Modus, so erlaubt dies dem Agenten, seine Verhandlungsstrategien in zwei disjunkte Klassen, nämlich Strategien für den Einkauf und den Verkauf von Gütern aufzuteilen. Die die Strategien spezifizierenden Genotypen, die ein Agent als Käufer oder Verkäufer einsetzt und die für sie auszuführenden Verwaltungs-, Selektions- und Adaption- beziehungsweise Lernprozesse werden dabei gemäß des Einsatzzweckes voneinander separiert. Hierdurch konnte eine spezifisch auf Kauf- oder Verkaufsprozesse ausgerichtete Adaptierung der Verhandlungsstrategien der Agenten realisiert werden.

Neben der Entwicklung und Beschreibung des MACSIMA-Systems wurden in den Kapiteln 6 und 7 auch eine dazugehörige Simulationsinfrastruktur sowie eine Evaluationsmethodik entwickelt, auf deren Basis die Evaluation von MACSIMA-Simulationen – wie in Kapitel 8 vorgestellt – durchgeführt wurde.

Die vorgestellte Evaluationsmethodik bietet dem Nutzer eine Vielzahl an Evaluationsmöglichkeiten, mit denen sich Hypothesen und Fragestellungen auf verschiedenen System- beziehungsweise Evaluationsebenen (siehe Abschnitt 7.2) formulieren und untersuchen lassen.

Die durch das Zusammenspiel von Simulationssystem, -infrastruktur und Evaluationsmethodik erzielten Ergebnisse wurden im abschließenden Evaluationsteil vorgestellt. Wesentliche Forschungsfrage war dabei, wie sich die in MACSIMA realisierten Erweiterungen der Ansätze von Eymann auf die Selbstkoordinationsfähigkeiten des gesamten Wertschöpfungsnetzwerks und seine Performanz auswirken - sowohl isoliert betrachtet, als auch in Ihrer Gesamtheit.

Die Evaluation beweist, dass die Systemerweiterungen im Vergleich zum AVALANCHE-System zu deutlich verbesserten Ergebnissen führen (siehe insbesondere Abschnitt 8.6). Die Selbstkoordinationsleistung konnte durch die entwickelten und im Detail evaluierten Konzepte für den Informationsaustausch und das evolutionäre Lernen sowie sonstige Erweiterungen stark, teilweise um das bis zu Zehnfache gesteigert werden (in Bezug auf Kennzahlen wie Gesamtumsatz, Gesamtgewinn und die Anzahl von erfolgreich abgeschlossenen Verhandlungen).

Die Evaluationsergebnisse geben insgesamt wertvolle Hinweise darauf, was zu beachten ist und welche Einstellungen bezüglich des Informationsaustauschs zwischen den Geschäftsagenten, ihrer Lernverfahren und sonstiger Verbesserungen für die Anwendungsdomäne empfehlenswert sind.

Eine Zusammenfassung sämtlicher Evaluationsergebnisse findet sich in Abschnitt 8.7. Im Folgenden wird auf einige der daraus ableitbaren wesentlichen Erkenntnisse gesondert hingewiesen.

9.2 Erzielte Erkenntnisse

Die Evaluationsergebnisse verdeutlichen, dass intelligent lernende Agenten, die sich gemäß der in MACSIMA umgesetzten Konzepte an sich dynamisch ändernde Marktsituationen anpassen, einen Vorteil gegenüber Agenten haben, die entweder nicht lernen oder dies mit beschränkteren Fähigkeiten beziehungsweise mit einer suboptimalen Parametrisierung ihres Lern- und Adaptionsprozesses tun.

Sie zeigen zudem, dass die marktliche Selbstkoordinationsleistung der Geschäftsagenten durch deren Ausstattung mit einer auf evolutionären Algorithmen beruhenden Lernfähigkeit signifikant gesteigert werden kann.

Die Ergebnisanalyse zeigt jedoch auch, dass eine prinzipielle Lernfähigkeit allein für eine effiziente Selbstkoordination nicht ausreicht. Wesentlich für den wirtschaftlichen Erfolg und eine messbar verbesserte Selbstorganisation ist vielmehr auch eine Feinjustierung und aufeinander abgestimmte Nutzung der für den Informationsaustausch und das Lernen einstellbaren Modi sowie der verwendeten Lernmechanismen und der weiteren in dieser Arbeit vorgestellten Methoden und Verfahren.

Hierbei war auffällig, dass der Designer eines sich dezentral koordinierenden marktlichen Multiagentensystems bei der Verwendung und der Parametrisierung des evolutionären Lernens sehr auf Feinheiten achten muss, um die Agenten mit einem gut an die Anwendungsdomäne angepassten Lern- und Adaptionsprozess auszustatten.

Handelt es sich bei dem verwendeten Lernmechanismus um ein Verfahren der evolutionären Algorithmen wie in MACSIMA, so übt die Art sowie die Parametrisierung der verwendeten Mutations-Methode in den untersuchten Simulationsszenarien einen besonders signifikanten Einfluss auf die Fähigkeit zur marktlichen Selbstorganisation und die Wohlfahrt des gesamten agentenbasierten Wertschöpfungsnetzwerks aus.

Bei der Auswahl der eingesetzten Mutations-Methode und seiner Parametrisierung ist wichtig, dass durch eine ausreichende „Seriosität“ bei der Mutation des Gen-Materials und der Ersetzung der Genotypen im Pool des Verhandlungsmoduls der Agenten im Verlauf der Simulation erzielte Adaptionserfolge nicht leichtfertig „verspielt“ werden.

Gleichzeitig muss aber auch eine gewisse Gen-Vielfalt in den Strategie- beziehungsweise Gen-Pools der Agenten erhalten bleiben, um die laufende Adaptionsfähigkeit der Agenten – auch unter extrem dynamischen Bedingungen, wie sie auf Märkten die Regel sind – gewährleisten zu können. Hierbei ist entscheidend den richtigen „Trade-off“ zu finden. Hierzu werden im Evaluationsteil auf Basis der qualitativen und quantitativen Analysen von Simulationsreihen entsprechende Empfehlungen gegeben (siehe die Abschnitte 8.4.7 und 8.5.1).

Darüber hinaus muss darauf geachtet werden, dass die Agenten durch eine gut gewählte Selektions-Methode und eine nicht zu übertriebene Rekombination eine gewisse Zielgerichtetheit entfalten.

Die Koordinationsleistung lässt sich zusätzlich durch die Ausstattung der Agenten mit relativ einfachen sozialen Fähigkeiten weiter verbessern. Befähigt man die Agenten dazu, Informationen über Verhandlungserfolge innerhalb von geeignet definierten Gruppen (zum Beispiel innerhalb einer Wertschöpfungsstufe) in einem sinnvollen Ausmaß miteinander auszutauschen und in ihrem Lernprozess zu nutzen, so lässt sich die Wohlfahrt des Systems beträchtlich steigern (siehe die Abschnitte 8.5 und 8.6).

Ähnlich wie beim Lernmechanismus zeigen aber auch die Evaluationsergebnisse zum Informationsaustausch, dass diese Fähigkeit allein kein Garant für eine effiziente Koordinationsleistung des Gesamtsystems ist. Ebenso wie beim Lernen ist auch hier eine Feinparametrisierung des Informationsaustausches beziehungsweise eine sinnvolle Einstellung seiner Quote (siehe Abschnitt 8.5) vonnöten, um ein optimales Ergebnis zu erzielen.

Denn ein Zuviel an ausgetauschten Informationen scheint beim evolutionären Lernen zu einer Art Overfitting zu führen. Dadurch, dass eine Strategie, die in einer Verhandlung zu einem weit überdurchschnittlichen Erfolg geführt hat, mit vielen anderen im Markt geteilt wird, pflanzt sich diese über mehrere Generationen hinweg überdurchschnittlich oft fort. Hierdurch werden andere nützliche Strategien beziehungsweise Genotypen, die sich die Agenten in ihrem Gen-Pool gemerkt haben verdrängt, was zu einer abnehmenden Performanz des einzelnen Agenten führen, aber auch durch Multiplikator-Effekte negative Auswirkungen auf die Performanz des Gesamtsystems haben kann.

Die Agenten scheinen dadurch an Anpassungsfähigkeit zu verlieren, die von der Vielfalt der Gen-Ausprägungen in ihrem Gen-Pool abhängig ist. Verändern sich anschließend die Marktbedingungen beziehungsweise die Strategien ihrer Verhandlungsopponenten, so scheinen sie sich daran nicht mehr zügig anpassen zu können. Beschränkt man jedoch den Informationsaustausch innerhalb einer Gruppe von Agenten sowohl in seiner Häufigkeit als auch in der Anzahl der Agenten, die adressiert und dadurch „ins Vertrauen gezogen“ werden, so tritt dieser hemmende Effekt nicht auf und die Koordinationsleistung wird deutlich gesteigert. Die diesbezüglich mit MACSIMA gemachten Beobachtungen weisen Ähnlichkeiten zu börsenpsychologischen Effekten auf, bei denen eine bestimmte Handelsstrategie nur so lange funktioniert, bis sie ein Großteil des Marktes übernimmt, was in der realen Welt früher oder später ein „Kippen“ der Börsenkurse zur Folge hat.

Die evolutionären Lernverfahren ermöglichen in ihrem dezentralen Einsatz innerhalb des MAS eine gute Selbstkoordinationsleistung, aber können bei eher ungünstigen Parametrisierungen dazu führen, dass einzelne Agenten Verhaltensweisen entwickeln, bei denen sie sich gewissermaßen „verrennen“. Hierfür muss es einen oder mehrere Verfahren geben, die dies bemerken und eingreifen, damit sie sich bei plötzlichen Änderungen von Umweltbedingungen (gegebenenfalls durch die Umkehrung von Trends in der Entwicklung von Güterpreisen) trotz einer möglicherweise geringen Vielfalt der in ihrem Gen-Pool enthaltenen Gen-Ausprägungen adaptieren können.

Eventuell dauert der Lern- und Adaptionsprozess bei ungünstigen Einstellungen zu lange, um mit solchen Geschehnissen gut umzugehen. Da im Simulationssystem ein menschlicher Überwacher der Agenten- beziehungsweise Systemperformanz fehlt, wurde zur Übernahme einer solchen korrigierenden Funktion neben dem Ticket-Limitierungs-Verfahren (siehe Abschnitt 5.4.5.2) in MACSIMA auch ein sogenannter Preisautomat integriert (siehe Abschnitt 5.3.3.7).

Dieser passt das Verhandlungsintervall eines Agenten beziehungsweise die Preisvorstellungen, mit denen ein Agent in eine neue Verhandlung startet, schrittweise an, falls der Agent mit seiner aktuellen Verhandlungsstrategie über einen längeren Zeitraum keinen erfolgreichen Verhandlungsabschluss erzielen konnte. Denn falls eine solche „Durststrecke“ auftritt, so sind oftmals nicht nur die im Pool enthaltenen Verhandlungsstrategien allein dafür verantwortlich, sondern es ist auch möglich, dass der Agent die aktuelle Marktsituation falsch einschätzt.

Hierzu kommt es, wie es bei Simulationen beobachtet werden kann, wenn der Agent aufgrund von ausbleibenden Verhandlungserfolgen nur in unzureichendem Ausmaß dazu Gelegenheit hat, seinen subjektiv wahrgenommenen durchschnittlichen Marktpreis für ein Gut

(das heißt seinen *weighted Average*) auf der Basis von Abschlusspreisen zu aktualisieren. In diesem Fall kann sich dieser schließlich so sehr von den momentan am Markt für ein Gut akzeptierten Preisen entfernen, dass ein auf seiner Basis bestimmtes Verhandlungsintervall (siehe Abschnitt 5.3.3.2) unabhängig von dem eingesetzten Genotyp kaum mehr dazu geeignet ist, Verhandlungen erfolgreich abzuschließen.

Ohne den Einsatz des Preisautomaten konnten daher in einigen Simulationsszenarien Entwicklungen beobachtet werden, die letztlich (aufgrund der weiterlaufenden Lebenshaltungskosten) zur Insolvenz vieler Agenten und zu einer starken Beeinträchtigung der Systemperformance geführt haben. Daher hat sich gezeigt, dass auch der Preisautomat eine wichtige Erweiterung der im AVALANCHE-System realisierten Ansätze darstellt.

9.3 Ausblick

In den mit MACSIMA durchgeführten Simulationen wurde der Informationsaustausch bislang in eher großen Gruppen untersucht, welche alle Agenten einer Wertschöpfungsstufe umfassen. Eine interessante Forschungsfrage besteht darin, zu untersuchen, welche Vorteile Agenten, die gemischtes Lernen verwenden, in einer nur einen Teil der Wertschöpfungsstufe umfassenden, miteinander Informationen austauschenden Gruppe gegenüber intern lernenden Agenten auf der gleichen Stufe erzielen können.

Darüber hinaus wurden bereits konzeptionelle Erweiterungen umgesetzt, durch die MACSIMA-Agenten in die Lage versetzt werden, nicht nur an bilateralen Verhandlungen, sondern auch an verschiedenen Auktionsformen wie zum Beispiel der Englischen oder der Holländischen Auktion teilzunehmen. Daher bietet es sich an, die mit verschiedenen Auktionsformen erreichbaren Selbstkoordinationsleistungen miteinander und mit den in dieser Arbeit erzielten Ergebnissen zu vergleichen.

Zudem ist vorstellbar, die entwickelten MACSIMA-Komponenten mit vergleichsweise geringem Anpassungsaufwand in weiteren Anwendungsszenarien zu nutzen. Die Ergebnisse und die beobachtbaren Phänomene könnten zum Beispiel für das Gebiet der Sozionik interessant sein. Kernfunktionalitäten wären relativ leicht auch für andere Simulationen einsetzbar, bei denen Agenten-Strategien in Vektoren beschreibbar und mit einem numerischen Nutzenwert bewertbar sind.

10 Literaturverzeichnis

- [Alexander 1987] Alexander, Jeffrey C. (Hrsg.): The Micro-Macro Link. Berkeley: University of California Press, 1987. 9780520060685
- [Arnold et al. 1995] Arnold, O.; Faisst, W.; Härtling, M.; Sieber, P.: Virtuelle Unternehmen als Unternehmenstyp der Zukunft? In: Handbuch der modernen Datenverarbeitung – Theorie und Praxis der Wirtschaftsinformatik, Band 185. Heidelberg: Hüthig-Verlag, 1995.
- [Arunachalam 2003] Arunachalam, R.: The Trading Agent Competition Supply Chain Management Game. Carnegie Mellon University, School of Computer Science, Institute for Software Research International, 2003.
- [Arunachalam 2004] Arunachalam, R.: The Supply Chain Management Game for the Trading Agent Competition. SICS technical report. Swedish Institute of Computer Science, 2004.
- [Baker et al. 1999] Baker, A.D.; Parunak, H.V.D.; Erol, K.: Internet-based Manufacturing – A Perspective from the AARIA Project. Cincinnati, Ohio: Enterprise Action Group, 1999.
- [Barbuceanu et al. 1993] Fox, M.S.; Chionglo, J.F.; Barbuceanu, M.: The Integrated Supply Chain Management System. University of Toronto, Department of Industrial Engineering, Enterprise Integration Laboratory, Internal Report, 1993.
- [Barbuceanu & Fox 1995a] Barbuceanu, M.; Fox, M.S.: The Architecture of an Agent Based Infrastructure for Agile Manufacturing. University of Toronto, Department of Industrial Engineering, Enterprise Integration Laboratory, Internal Report, 1995.
- [Barbuceanu & Fox 1995b] Barbuceanu, M.; Fox, M.S.: COOL: A Language for Describing Coordination in Multiagent Systems. In: Proceedings of the First International Conference on Multiagent Systems (ICMAS95). San Francisco, 1995, S. 14-24. – ISBN 978-0-262-62102-1
- [Barbuceanu & Fox 1997] Barbuceanu, M.; Fox, M.S.: Coordinating Multiple Agents in the Supply Chain. In: Proceedings of WET-ICE 97. Boston: IEEE Computer Society Press, 1997.
- [Beckmann 2003] Beckmann, Holger: Supply Chain Management. Heidelberg: Springer, 2003. – ISBN 3540443908, 9783540443902
- [Beleva 2007] Beleva, M.: Der Einsatz von Multi-Agenten-Systemen im Supply Chain Management. München: GRIN Verlag, 2007. – ISBN 9783638806268
- [Bibel et al. 2013] Bibel, W.; Nissen, V.; Kruse, R.: Einführung in Evolutionäre Algorithmen: Optimierung nach dem Vorbild der Evolution. Computational Intelligence. Wiesbaden: Vieweg+Teubner, 2013. – ISBN 9783322938619
- [Bleymüller et al. 1996] Bleymüller, J., Gehlert, G., Gülicher, H.: Statistik für Wirtschaftswissenschaftler. WiSt-Studienkurs. Kapitel 26. München: Vahlen, 1996. - ISBN 3-8006-2081-2

- [Bond & Gasser 1988] Bond, A. und Gasser, L. (Hrsg.): Readings in Distributed Artificial Intelligence. San Mateo: Morgan Kaufmann Publishers, 1988.
- [Bowersox et al. 2002] Bowersox, D.J.; Closs, D.J.; Cooper, B.: Supply Chain Logistics Management. Irwin/McGraw-Hill Series in Operations and Decision Sciences. New York: McGraw-Hill, 2002. – ISBN 0072351004, 9780072351002
- [Bratman 1987] Bratman, M.E.: Intentions, Plans, and Practical Reason. Cambridge, MA, USA: Harvard University Press, 1987.
- [Bratman et al. 1987] M. E. Bratman, D. J. Israel und M. E. Pollack. Toward an Architecture for Resource-Bounded Agents. Technical Report CSLI-87-104, Menlo Park, CA, USA: SRI and Stanford University, 1987.
- [Brenner et al. 1998] Brenner, W.; Zarnekow, R.; Wittig, H.: Intelligente Softwareagenten - Grundlagen und Anwendungen. Heidelberg: Springer, 1998.
- [Brenner et al. 2012] Brenner, W.; Schubert, C.; Hultsch, H.; Rudd, A.S.; Zarnekow, R.; Wittig, H.: Intelligent Software Agents: Foundations and Applications. Heidelberg: Springer, 2012. – ISBN 9783642804847
- [Brodersen 2012] Brodersen, Nina: e-Supply Chain: Wege der Optimierung. Hamburg: Bachelor+Master Publishing, 2012. – ISBN 3863413075, 9783863413071
- [Brugali et al. 1998] Brugali, D.; Menga, G.; Galarraga, S.: Inter-Company Supply Chains Integration via Mobile Agents. In: Proceedings of PROLAMAT '98. Kluwer Academic Publishers, 1998.
- [Bürckert et al. 1998] Bürckert, H.-J.; Fischer, K.; Vierke, G.: Transportation Scheduling with Holonic MAS - The TeleTruck Approach. In: Proceedings of the Third International Conference on Practical Applications of Intelligent Agents and Multiagents. PAAM'98. London: Practical Application Company, 1998.
- [Büttner 2010] Büttner, R.: Automatisierte Verhandlungen in Multi-Agenten-Systemen: Entwurf eines argumentationsbasierten Mechanismus für nur imperfekt beschreibbare Verhandlungsgegenstände. Gabler Research. Wiesbaden: Gabler, 2010. – ISBN 9783834921314
- [Bullinger et al. 2009] Bullinger, H.J.; Spath, D.; Warnecke, H.-J.; Westkämper, E.: Handbuch Unternehmensorganisation: Strategien, Planung, Umsetzung. Heidelberg: Springer-Verlag, 2009. - ISBN 3540875956, 9783540875956
- [Burkey & Anagnost 1995] Burkey, C.; Anagnost, T.: Determining What to Learn Through Component-Task Modeling. In: Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence. IJCAI-95, 1995.
- [Burkey & Anagnost 1996a] Burkey, C.; Anagnost, T.: The BargainFinder Agent: Comparison Price Shopping on the Internet. In: Williams, J. (Hrsg.): Bots and Other Internet Beasties. Indianapolis: Sams Publishing, 1996.
- [Burkey & Anagnost 1996b] Burkey, C.; Anagnost, T.: The ContactFinder Agent: Answering Bulletin Board Questions with Referrals. In: Proceedings of the 1996 National Conference on Artificial Intelligence. AAAI-96, 1996.
- [Bussmann et al. 2013] Bussmann, Stefan; Jennings, Nicolas R.; Wooldridge, M.J.: Multiagent Systems for Manufacturing Control: A Design Methodology. Springer Series on Agent Technology. Heidelberg: Springer Science & Business Media, 2013. - ISBN 366208872X, 9783662088722

- [Caglayan & Harrison 1998] A. K. Caglayan and C. G. Harrison. *Intelligente Software-Agenten: Grundlagen, Technik und praktische Anwendung im Unternehmen*. München: Carl Hanser Verlag, 1998.
- [Castellanos 2012] Castellanos, M.: *Agent Based Simulation Approach to Assess Supply Chain Complexity and Its Impact on Performance*. Supply Chain, logistics and operations management. Band 12. Tarpen: BoD – Books on Demand, 2012. - ISBN 3844102094, 9783844102093
- [Ceppi et al. 2014] Ceppi, S.; David, E.; Podobnik, V.; Robu, V.; Shehory, O.; Stein, S.; Vetsikas, I.A.: *Agent-Mediated Electronic Commerce. Designing Trading Strategies and Mechanisms for Electronic Markets: AMEC and TADA 2013. AMEC and TADA 2014. Lecture Notes in Business Information Processing*. Cham, Schweiz: Springer International Publishing, 2014. – ISBN 9783319132181
- [Chaib-draa & Müller 2006] Chaib-draa, B.; Müller, J.P.: *Multiagent Based Supply Chain Management. Studies in Computational Intelligence*. Heidelberg: Springer, 2006. – ISBN 9783540338758
- [Chavez & Maes 1996] Chavez, A; Maes, P.: *Kasbah: An Agent Marketplace for Buying and Selling Goods*. In: *Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology. PAAM'96*. London, 1996.
- [Chen et al. 1999a] Chen, Y.; Peng, Y.; Finin, T.; Labrou, Y.; Cost, S.: *A Negotiation-based Multi-agent System for Supply Chain Management*. In: *Working Notes of the Agents '99 Workshop on Agents for Electronic Commerce and Managing the Internet-Enabled Supply Chain*. Seattle, 1999.
- [Chen et al. 1999b] Chen, Ye; Peng, Yun; Labrou, Yannis K.; Cost, R. Scott: *Negotiating Agents for Supply Chain Management*. In: *Proceedings of the AAAI Workshop on Artificial Intelligence for Electronic Commerce*. AAAI Press, 1999.
- [Christopher 1992] Christopher, Martin: *Logistics and Supply Chain Management: Strategies for Reducing Costs and Improving Services*. Financial Times Logistics & Distribution Management Series. Financial Times, 1992. – ISBN 0273034154, 9780273034155
- [Christopher 2011] Christopher, Martin: *Logistics and Supply Chain Management: Creating Value-Adding Networks*, 4. Auflage. Pearson Education Limited, 2011. - ISBN 9780273731122
- [Coleman 1991] Coleman, James S.: *Grundlagen der Sozialtheorie: Handlungen und Handlungssysteme*. Band 1. München: Oldenbourg Verlag, 1991. - ISBN 3486558382, 9783486558388
- [Collins et al. 1999] Collins, J.; Sundareswara, R; Tsvetovat, M.; Gini, M.; Mobasher, B.: *Search Strategies for Bid Selection in Multi-Agent Contracting*. Agent-mediated Electronic Commerce. AmEC-99. IJCAI'99. Stockholm, 1999.
- [Collins et al. 2000a] Collins, J.; Sundareswara, R; Tsvetovat, M.; Gini, M.; Mobasher, B.: *Multi-Agent Contracting for Supply-Chain Management*. Technical Report 2000-10. University of Minnesota, 2000.
- [Collins et al. 2000b] Collins, J.; Bilot, M.; Gini, M.; Mobasher, B.: *Mixed-Initiative Decision Support in Agent-Based Automated Contracting*. In: *Proceedings of the Fourth International Conference on Autonomous Agents*. New York: ACM, 2000.

- [Collins et al. 2008] Collins, J.; Faratin, P.; Parsons, S.D.; Rodriguez-Aguilar, J.A.; Sadeh, N.M.; Shehory, O.; Sklar, E.: Agent-Mediated Electronic Commerce and Trading Agent Design and Analysis: AAMAS 2007. AMEC 2007. AAAI 2007. TADA 2007. Lecture Notes in Business Information Processing. Heidelberg: Springer, 2008. – ISBN 9783540887133
- [Corsten & Gössinger 1997a] Corsten, H.; Gössinger, R.: Multiagentensystem zur Unterstützung der Produktionsplanung und -steuerung. In: Information Management (1997) 3, S. 65-75.
- [Corsten & Gössinger 1997b] Corsten H.; Gössinger, R.: Entwurf eines konzeptionellen Rahmens für ein Multiagentensystem zur integrativen Unterstützung der Produktionsplanung und -steuerung. Technical Report Nr. 13. Universität Kaiserslautern, 1997.
- [CSTaR 1997] Andersen Consulting Center for Strategic Technology Research CSTaR. CStar intelligent agents project. Andersen Consulting: <http://www.ac.com/cstar/>, 1997.
- [Dangelmaier et al. 2013] Dangelmaier, W.; Pape, U.; Rüther, M.: Agentensysteme für das Supply Chain Management: Grundlagen – Konzepte – Anwendungen. Wirtschaftsinformatik. Wiesbaden: Deutscher Universitätsverlag, 2013. – ISBN 3322812375, 9783322812377
- [Dasgupta & Michalewicz 1997] Dasgupta, D.; Michalewicz, Z.: Evolutionary Algorithms in Engineering Applications. Heidelberg: Springer Verlag, 1997. – ISBN 3-540-62021-4
- [David et al. 2010] David, E.; Gerding, E.; Sarne, D.; Shehory, O.: Agent-Mediated Electronic Commerce. Designing Trading Strategies and Mechanisms for Electronic Markets: AAMAS Workshop. AMEC 2009. IJCAI Workshop, TADA 2009. Lecture Notes in Business Information Processing Selected and Revised Papers. Heidelberg: Springer, 2010. – ISBN 9783642151163
- [David et al. 2012] David, E.; Larson, K.; Rogers, A.; Shehory, O.; Stein, S.: Agent-Mediated Electronic Commerce. Designing Trading Strategies and Mechanisms for Electronic Markets: AMEC 2010. TADA 2010. Revised Selected Papers. Lecture Notes in Business Information Processing. Heidelberg: Springer, 2012. – ISBN 9783642342004
- [Davis 1991] Davis, L.D.: Handbook of Genetic Algorithms. New York: Van Nostrand Reinhold, 1991
- [Dawid 1995] Dawid, H.: Genetic Learning in Economic Systems: Analytical Results and Applications. 1995.
- [Dawid 1999] Dawid, Herbert: Adaptive Learning by Genetic Algorithms. Heidelberg: Springer, 1999. – ISBN 3-540-65607-3
- [De Paula et al. 2001] De Paula, Gustavo E.; Ramos, Francisco S.; Ramalho, Geber L.: Bilateral Negotiation Model for Agent-Mediated Electronic Commerce. In: Dignum, Frank; Cortés, Ulises (Hrsg.): Agent-Mediated Electronic Commerce III – Current Issues in Agent-Based Electronic Commerce Systems. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Science & Business Media, 2001, S. 1-14. – DOI 10.1007/3-540-44723-7_1, http://dx.doi.org/10.1007/3-540-44723-7_1 – ISBN 3540417494, 9783540417491

- [Deßloch 2002] Deßloch, Jörg: Implementierung einer agentenbasierten Koordinationsarchitektur für Supply Webs. Diplomarbeit am Lehrstuhl Professor Dr. Jörg H. Siekmann an der Universität des Saarlandes. 2002.
- [De Wilde et al. 2003] De Wilde, P.; Chli, M.; Correia, L.; Ribeiro, R.; Mariano, P.; Abramov, V.; Goossenaerts, J.: Adapting Populations of Agents. In: Adaptive Agents and Multi-Agent Systems (Lecture Notes in Computer Science, Springer Series Volume 2636, Series ISSN 0302-9743). Berlin Heidelberg: Springer-Verlag, 2003, S. 110-124. – DOI 10.1007/3-540-44826-8_7. – ISBN 978-3-540-40068-4, 978-3-540-44826-6
- [Esser 1993] Hartmut Esser. Soziologie: allgemeine Grundlagen. Campus Verlag, 1993. – ISBN 3593349604, 9783593349602.
- [Eymann & Padovan 1999] Eine Multi-Agenten-Simulation zur ökonomischen Analyse der dezentralen Koordination von Wertschöpfungsketten. In: A.-W. Scheer, M. Nüttgens (Hrsg.): Electronic Business Engineering. Konferenzband zur 4. Internationalen Tagung Wirtschaftsinformatik. WI99. Heidelberg: Physica Verlag, 1999, S. 625- 641.
- [Eymann 2000] Eymann, Thorsten: AVALANCHE – Ein agentenbasierter dezentraler Koordinationsmechanismus für elektronische Märkte. Dissertation an der Albert-Ludwigs-Universität Freiburg im Breisgau, 2000. <http://www.freidok.unifreiburg.de/volltexte/147/>.
- [Eymann 2003] Eymann, Thorsten: Digitale Geschäftsagenten – Softwareagenten im Einsatz. Heidelberg: Springer, 2003.
- [Eymann et al. 2003] Eymann, T.; Sackmann, S.; Müller, G.; Pippow, I.: Hayek's Catallaxy – A Forward Looking Concept for Information Systems? AMCIS-2003. <http://ssrn.com/abstract=1465491>.
- [Eymann 2013] Eymann, Torsten: Digitale Geschäftsagenten: Softwareagenten im Einsatz. Heidelberg: Springer, 2013. – ISBN 3642556221, 9783642556227
- [Falk et al. 1993] Falk, J.; Spieck, S.; Mertens, P.: Unterstützung der Lager- und Transportlogistik durch Teilintelligente Agenten. In: Information Management, 1993, S. 26-31.
- [Ferber 2001] J. Ferber. Multiagentensysteme: Eine Einführung in die Verteilte Künstliche Intelligenz. Addison-Wesley, 2001.
- [Ferruci et al. 2012] Ferrucci, D.; Brown, E.; Chu-Carroll, J.; Fan, J.; Gondek, D.; Kalyanpur, A.; Lally, A.; Murdock, J.; Nyberg, E.; Prager, J.; Schlaefler, N.; Welty, C.: The AI behind Watson - The Technical Article. In: AI Magazine. AAAI, 2010. <http://www.aaai.org/Magazine/Watson/watson.php>, abgerufen am 18. Februar 2015.
- [Ferschl 1985] F. Ferschl: Deskriptive Statistik. 3. Auflage. Heidelberg: Physica-Verlag, 1985. - ISBN 3-7908-0336-7, 978-3790803365.
- [Finin et al. 1994] Finin, Tim et al.: Specification of the KQML Agent-Communication Language – Plus Example Agent Policies and Architectures. 1994.
- [Finin et al. 1995] Finin, T.; Fritzson, R.; McKay, D.; McEntire, R.: KQML as an Agent Communication Language. In: Adam, N. (Hrsg.); Bhargava, B. (Hrsg.); Yesha, Y. (Hrsg.): Proceedings of the 3rd International Conference on Information and Knowledge Management. CIKM'94. Gaithersburg, MD, USA: ACM Press, 1995, S. 456–463

- [Fischer 1998] K. Fischer. An Agent-Based Approach to Holonic Manufacturing Systems. In: Camarinha-Malos, L. M.; Afsarmanesh, H.; Marik, V. (Hrsg.): Intelligent Systems for Manufacturing. Boston: Kluwer Academic Publishers, 1998.
- [Fischer 1999] Fischer, K.: Agent-Based Design of Holonic Manufacturing Systems. Journal of Robotics and Autonomous Systems, 27 (1999), S. 3-13.
- [Fischer 2013] Fischer, T.: Multi-Agenten-Systeme im Fahrzeugumschlag: Agentenbasierte Planungsunterstützung für Seehafen-Automobilterminals. Deutscher Universitätsverlag, 2013. – ISBN 9783322816887
- [Fischer & Ruß 1998] Fischer, K.; Ruß, C.: Agent Technologies in Virtual Enterprises. In: Wittig, W. S.; Grieser, G. (Hrsg.): LIT'98. Tagungsband der 6. Leipziger Informatik-Tage. Forschungsinstitut für Informationstechnologien Leipzig e.V., 1998.
- [Fischer & Ruß 1999] Fischer, K.; Ruß, C.: MAS Technologies for Supply Chain Management in Virtual Enterprises. In: Kirn, Stefan; Petsch, Mathias (Hrsg.): Workshop Intelligente Softwareagenten und betriebswirtschaftliche Anwendungsszenarien. Arbeitsbericht Nr. 14 der Fakultät für Wirtschaftswissenschaften. Lehrstuhl für Wirtschaftsinformatik 2 der Technischen Universität Ilmenau, 1999.
- [Fischer et al. 1993] Fischer, K.; Kuhn, N.; Müller, H.-J.; Müller, J.P.; Pischel, M.; Schroth, A.: Verteiltes Problemlösen im Transportwesen. In: Information Management 8 (1993) 2, S. 32- 40.
- [Fischer et al. 1996a] Fischer, K.; Müller, J. P.; Pischel, M.: A Pragmatic BDI Architecture. In: Wooldridge, M.; Müller, J.P.; Tambe, M. (Hrsg.): Intelligent Agents II. LNAI Volume 1037. Heidelberg: Springer-Verlag, 1996. S. 203–218.
- [Fischer et al. 1996b] Fischer, K.; Heimig, I.; Kocian, C.; Müller, J.: Intelligente Agenten für das Management Virtueller Unternehmen. Information Management 1 (1996), S. 38 – 45.
- [Fischer et al. 1996c] Fischer, K.; Müller, J.; Pischel, M.: Cooperative Transportation Scheduling: An Application Domain for DAI. In: Journal of Applied Artificial Intelligence. Special Issue on Intelligent Agents. 10 (1), 1996.
- [Fischer et al. 1998] Fischer, K.; Ruß, C.; Vierke, G.: Decision Theory and Coordination in Multiagent Systems. DFKI Research Report. RR-98-02. Deutsches Forschungszentrum für Künstliche Intelligenz, 1998.
- [Fischer et al. 2003] Fischer, K.; Schillo, M.; Siekmann, J.: Holonic Multiagent Systems: The Foundation for the Organization of Multiagent Systems. Proceedings of the First International Conference on Applications of Holonic and Multiagent Systems. HoloMAS'03. Lecture Notes in Artificial Intelligence. Ausg. 2744. Heidelberg: Springer, 2003, S. 71-80.
- [Fischer et al. 2005] Fischer, K.; Florian, M.; Malsch, T.: Socionics: Scalability of Complex Social Systems. Lecture Notes in Computer Science. Heidelberg: Springer, 2005. – ISBN 9783540307075
- [Fleisch 2013] Fleisch, Elgar: Das Netzwerkunternehmen: Strategien und Prozesse zur Steigerung der Wettbewerbsfähigkeit in der Networked Economy. Business Engineering. Heidelberg: Springer-Verlag, 2013. – ISBN 3642567282, 9783642567285
- [Foner 1999] Foner, L.N.: Political Artifacts and Personal Privacy: The Yenta Multi-Agent Distributed Matchmaking System. PhD Thesis. MIT, 1999. <http://foner.www.media.mit.edu/people/foner/PhD-Thesis/Dissertation/>, letzter Abruf: 14.05.2007.

- [Franklin & Graesser 1996] Franklin, S.; Graesser, A.: Is It an Agent, or Just a Program: A Taxonomy for Autonomous Agents. Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages. ATAL, 1996.
- [Fujita et al. 2015] Fujita, K.; Ito, T.; Zhang, M.; Robu, V.: Next Frontier in Agent-based Complex Automated Negotiation. Studies in Computational Intelligence. Springer Japan, 2015. – ISBN 9784431555254
- [Gabler 2015a] Springer Gabler Verlag (Hrsg.): Gabler Wirtschaftslexikon, Stichwort: Netzwerkorganisation, online im Internet: <http://wirtschaftslexikon.gabler.de/Archiv/6925/netzwerkorganisation-v8.html>, abgerufen am 22.06.2015.
- [Gabler 2015b] Springer Gabler Verlag (Hrsg.): Gabler Wirtschaftslexikon, Stichwort: spontane Ordnung, online im Internet: <http://wirtschaftslexikon.gabler.de/Archiv/8457/spontane-ordnung-v8.html>, abgerufen am 22.06.2015.
- [Ganeshan & Harrison 1995] Ganeshan, R.; Harrison, T.P.: An Introduction to Supply Chain Management. Penn State University, 1995.
- [Gerber et al. 1999a] Gerber, C.; Siekmann, J.; Vierke, G.: Flexible Autonomy in Holonic Agent Systems. In: Proceedings of the 1999 AAAI Spring Symposium on Agents with Adjustable Autonomy. AAAI, 1999.
- [Gerber et al. 1999b] Gerber, C.; Siekmann, J.; Vierke, G.: Holonic Multi-Agent Systems. Research Report. RR-99-03. Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI), 1999. - ISSN 0946-008X
- [Gerber 2005] Gerber, Andreas: Flexible Kooperation zwischen autonomen Agenten in dynamischen Umgebungen. Hochschulschrift Saarbrücken. Univ. Diss., 2005. Persistent Identifier urn:nbn:de:bsz:291-scidok-4571. <http://scidok.sulb.uni-saarland.de/volltexte/2005/457/index.html>.
- [Gomber et al. 1997] Gomber, Peter; Schmidt, Claudia; Weinhardt, Christof: Elektronische Märkte für die dezentrale Transportplanung. In: Wirtschaftsinformatik 39, 2, 1997, S. 137 - 145.
- [Gomber et al. 1998] Gomber, P.; Schmidt, C.; Weinhardt, C.: Efficiency Incentives and Computational Tractability in the Coordination of Multi-Agent Systems. Proceedings of the Workshop Kooperationsnetze und Elektronische Koordination. 1998.
- [Gomber et al. 2000] Gomber, Peter; Schmidt, Claudia; Weinhardt, Christof: Pricing in Multi-Agent Systems for Transportation Planning In: Journal of Organizational Computing and Electronic Commerce, 10, 4, 2000, S. 271-280.
- [Gomber 2013] Gomber, Peter: Elektronische Handelssysteme: Innovative Konzepte und Technologien im Wertpapierhandel. Information Age Economy. Physica-Verlag HD, 2013. – ISBN 3642576435, 9783642576430
- [Gomber 2014] Gomber, Peter; Zimmermann, Kai: Algorithmic Trading. In: Shu-Heng Chen and Mak Kaboudan. Forthcoming. OUP Handbook on Computational Economics and Finance. Oxford University Press, 2014.
- [Gomolka 2011] Gomolka, J.: Algorithmic Trading – Analyse von computergesteuerten Prozessen im Wertpapierhandel unter Verwendung der Multifaktorenregression. Universitätsverlag Potsdam, 2011. – ISBN 3869561254, 9783869561257
- [Graeber 2006] Gräber, Heiko. Quantitative Analyse von Agenten-basierten Supply-Chain-Marktszenarien. Diplomarbeit am Lehrstuhl Professor Dr. Jörg H. Siekmann. Saarbrücken, Universität des Saarlandes, 2006.

- [Greve et al. 2009] Greve, J.; Schnabel, A.; Schützeichel, R.: Das Mikro-Makro-Modell der soziologischen Erklärung: Zur Ontologie, Methodologie und Metatheorie eines Forschungsprogramms. In: Das Mikro-Makro-Modell der soziologischen Erklärung. VS Verlag für Sozialwissenschaften, 2009, S. 7-17. – ISBN 978-3-531-15436-7, 978-3-531-91774-0
- [Grolik et al. 2001] Grolik, Sven; Stockheim, Tim; Wendt, Oliver; Albayrak, Sahin; Fricke, Stefan: Dispositive Supply-Web-Koordination durch Multiagentensysteme. In: Wirtschaftsinformatik 43, 2001, Nr. 2, S. 143–155. <http://dx.doi.org/10.1007/BF03250791>. – DOI 10.1007/BF03250791. – ISSN 0937–6429
- [Guttman & Maes 1998] Guttman, R.; Maes, P.: Agent-Mediated Integrative Negotiation for Retail Electronic Commerce. In: Proceedings of the Workshop on Agent-Mediated Electronic Trading AMET'98. Minneapolis, 1998.
- [Guttman et al. 1997] Guttman, R.; Maes, P.; Chavez, A.; Dreilinger, D.: Results from a Multi-Agent Electronic Marketplace Experiment. In: Proceedings of Modeling Autonomous Agents in a Multi-Agent World. MAAMAW'97. Ronneby, Sweden, 1997.
- [Hayek 1996] Hayek, F. A.: Die verhängnisvoll Anmaßung – Die Irrtümer des Sozialismus. Tübingen: Mohr. 1996.
- [Hayek 1988] Hayek, F. A. (1988): The Fatal Conceit: The Errors of Socialism. Chicago: Univ. of Chicago Press, 1988.
- [Hehenkamp 2007] Hehenkamp, Burkhard: Die Grundlagen der Mechanismus-Design-Theorie. In: Wirtschaftsdienst. 87, 1, 2007. S. 768-772. - ISSN 0043-6275
- [Herlyn 2014] Herlyn, W. The Bullwhip Effect in expanded Supply Chains and the Concept of Cumulative Quantities, 2014, S. 513-528. - ISBN 978-3-8442-9878-9
- [Hertz 2012] Hertz, S.: An Empirical Study of the Ad Auction Game in the Trading Agent Competition. Tel Aviv University, 2012.
- [Heun 2008] Heun, Michael et al.: Finanzmarktsimulation mit Multiagentensystemen: Entwicklung eines methodischen Frameworks. Gabler Edition Wissenschaft. Springer-Verlag, 2008. - ISBN 3835055135, 9783835055131
- [Hinkkanen et al. 1997] Hinkkanen, A.; Kalakota, R.; Saengcharoenrat, P.; Stallaert, J.; Whinston, A.B.: Distributed Decision Support Systems for Real Time Supply Chain Management using Agent Technologies. http://yama.bus.utexas.edu/ejou/articles/art_1.html, Abruf am 9.10.1999.
- [Hohrath 2013] Hohrath, Philipp Alexander: Analyse der strategisch und strukturell induzierten Verwundbarkeit von Wertschöpfungsnetzwerken: Eine empirische Untersuchung am Beispiel der Windenergieanlagenindustrie. Band 13 der Reihe: Supply Chain, Logistics and Operations Management. Josef Eul Verlag GmbH, 2013. – 3844102396, ISBN 9783844102390
- [Hoppmann 1999] Hoppmann, E.: Unwissenheit, Wirtschaftsordnung und Staatsgewalt. In: Vanberg, V. (Hrsg.): Freiheit, Wettbewerb und Wirtschaftsordnung, Freiburg: Haufe Verlag, 1999.
- [Houlihan 1985] Houlihan, J. B. (1985): International Supply Chain Management. In: International Journal of Physical Distribution and Materials Management, 1(1985), S. 22-38.
- [Hsu 2002] Hsu, Feng-hsiung: Behind Deep Blue: Building the Computer that Defeated the World Chess Champion. 2002. - ISBN 0-691-09065-3

- [Hu & Watt 2014] Hu, Rui; Watt, Stephen M.: An Agent-Based Financial Market Simulator for Evaluation of Algorithmic Trading Strategies. In: Proceedings of the Sixth International Conference on Advances in System Simulation. SIMUL 2014, 2014, S. 221-227.
- [Hülsmann 2008] Hülsmann, M.: Kontinuitätsorientierte Koordination dynamischer Kooperationen. Gabler-Edition Wissenschaft. Gabler Verlag, 2008. – ISBN 9783834909459
- [Ickerott 2007] Ickerott, Ingmar: Agentenbasierte Simulation für das Supply Chain Management. In: Hoppe, U.; Rieger, B.; Teuteberg, F.; Witte, T. (Hrsg.): Schriften des Instituts für Informationsmanagement und Unternehmensführung (IMU), Band 2. Köln: Josef Eul Verlag GmbH, 2007. - ISBN3899366107, 9783899366105
- [Ito 2008] Ito, T.: Rational, Robust, and Secure Negotiations in Multi-Agent Systems. Studies in Computational Intelligence. Springer, 2008. – ISBN 9783540762812
- [Ito et al. 2009] Ito, T.; Zhang, M.; Robu, V.; Fatima, S.; Matsuo, T.: Advances in Agent-Based Complex Automated Negotiations. Studies in Computational Intelligence. Springer, 2009. – ISBN 9783642031892
- [Ito et al. 2010] Ito, T.; Zhang, M.; Robu, V.; Fatima, S.; Matsuo, T.; Yamaki, H.: Innovations in Agent-Based Complex Automated Negotiations. Studies in Computational Intelligence. Springer, 2010. – ISBN 9783642156113
- [Ito et al. 2011] Ito, T.; Zhang, M.; Robu, V.; Fatima, S.; Matsuo, T.: New Trends in Agent-Based Complex Automated Negotiations. Studies in Computational Intelligence Springer, 2011. – ISBN 9783642246951
- [Ivanov & Sokolov 2009] Ivanov, Dmitry; Sokolov, Boris: Adaptive Supply Chain Management. London: Springer Science & Business Media, 2009. – ISBN 1848829523, 9781848829527
- [Jennings & Wooldridge 1998] Jennings, N.; Wooldridge, M.J.: Agent Technology: Foundations, Applications, and Markets. Springer Verlag, 1998. – ISBN 9783540635918
- [Jennings 1999] N. R. Jennings. Agent-based Computing: Promise and Perils. Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI), 1999.
- [Kalakota et al. 1996] Kalakota, R.; Stallaert, J.; Whinston, A.B.: Implementing Real-time Supply Chain Optimization Systems.
- [Kassel 1996] Kassel, S.: Multiagentensysteme als Ansatz zur Produktionsplanung und -steuerung. In: Information Management 1 (1996), S. 46-50.
- [Ketter et al. 2010] Ketter, W.; Poutré, H.L.; Sadeh, N.M.; Shehory, O.; Walsh, W.: Agent-Mediated Electronic Commerce and Trading Agent Design and Analysis: AAMAS Workshop, AMEC 2008 and AAI Workshop, TADA 2008. Revised, Selected Papers. Lecture Notes in Business Information Processing. Heidelberg: Springer, 2010. – ISBN 9783642152375
- [Kjenstad 1998] Kjenstad, Dag: Coordinated supply chain scheduling. Ph. D. Thesis and NTNU Report 1998:24. Trondheim: Norwegian University of Science and Technology (NTNU), Department of Production and Quality Engineering, 1998.
- [Klügl 2004] Franziska Klügl: Simulation von Selbstorganisation und Evolution in Multiagentensystemen. In: Florian, M.; Hillebrandt, F.; (Hrsg.): Adaption und Lernen von und in Organisationen, Beiträge aus der Sozionik. Wiesbaden: VS Verlag für Sozialwissenschaften, 2004, S. 141-161. - ISBN 978-3-531-14164-0

- [Koestler 1967] Koestler, A.: The Ghost in the Machine. London: Hutchinson & Co, 1967.
- [Kohn 2004] Wolfgang Kohn: Statistik: Datenanalyse und Wahrscheinlichkeitsrechnung. Springer, 2004, S. 81. - ISBN 978-3540216773.
- [Kolesnewa 2009] Kolesnewa, J.: Koordinationsmechanismen im Supply Chain Management. Diplom.de, 2009. – ISBN 9783836634328
- [Kortus-Schultes & Ferfer 2015] Kortus-Schultes, D.; Ferfer, U.: Logistik und Marketing in der Supply Chain: Wertsteigerung durch virtuelle Geschäftsmodelle. Gabler Lehrbuch. Springer-Verlag, 2015. – ISBN 3322823016, 9783322823014
- [Kraus 1997] Kraus, S.: Negotiation and cooperation in multi-agent environments, Artificial Intelligence 94 (1997).
- [Lambert et al.1998] Lambert, Douglas M.; Cooper, Martha C.; Pagh, Janus D.: Supply Chain Management: Implementation Issues and Research Opportunities. In: The International Journal of Logistics Management. Band 9, Ausgabe 2, MCB UP Ltd., 1998, S. 1-20.
- [Lang 2008] Lang, Florian: Wissensbasierte Verhandlungsautomatisierung auf elektronischen Echtzeit-Märkten. Gabler Edition Wissenschaft. Springer-Verlag, 2008. - ISBN 383505564X, 9783835055643
- [Laseter 1998] Laseter, T. M.: Balanced Sourcing: Cooperation and Competition in Supplier Relationships. Jossey-Bass, 1998.
- [Laudon et al. 2010] Laudon, K.C.; Laudon, J.P.; Schoder, D.: Wirtschaftsinformatik: eine Einführung. Always learning. Pearson Studium, 2010. – ISBN 3827373484, 9783827373489
- [Lawrenz et al. 2013a] Lawrenz, O.; Hildebrand, K.; Nenninger, M.; Hillek, T.: Supply Chain Management: Konzepte, Erfahrungsberichte und Strategien auf dem Weg zu digitalen Wertschöpfungsnetzen. Reihe XBusiness Computing. 2. Ausgabe. Springer-Verlag, 2013 – ISBN 3663078388, 9783663078388
- [Lawrenz et al. 2013b] Lawrenz, Oliver; Hildebrand, Knut; Nenninger, Michael: Supply Chain Management: Strategien, Konzepte und Erfahrungen auf dem Weg zu E-Business Networks. XBusiness Computing. Springer-Verlag, 2013. - ISBN 3663014622, 9783663014621
- [Lee 1997] Lee, Hau L. et al.: Information Distortion in a Supply Chain: The Bullwhip Effect. In: Management Science. 43 (1997) 4, S. 546-558.
- [Lee 2000] Lee, H.L.: Creating Value through Supply Chain Integration. In: Supply Chain Management Review. Band 4, Nummer 4, Juli/August 2000, S. 30-36.
- [Lee & Billington 1995] Lee, H. L.; Billington, C.: The Evolution of Supply-Chain-Management Models and Practice at Hewlett-Packard. Interfaces 25 (1995) 5, S. 42-63.
- [Lin 1996] Lin, F.: Reengineering the Order Fulfillment Process in Supply Chain Networks: A Multiagent Information Systems Approach. Ph. D. Thesis. University of Illinois at Urbana-Champaign.
- [Lin et al. 1996] Lin, F.; Tan, G. W.; Shaw, H. J.: Multi-Agent Enterprise Modeling. University of Illinois at Urbana-Champaign, College of Commerce and Business Administration, Office of Research, Working Paper 96-0134.
- [Lin et al. 2000] Lin, F., Strader, T. & Shaw, M.: Using Swarm for Simulating the Order Fulfillment Process in Divergent Assembly Supply Chains. In: Luna, F. & Stefansson,

- B. (Hrsg.): *Economic Simulations in Swarm: Agent-Based Modeling and Object-Oriented Programming*. Boston: Kluwer Academic Publishers, 2000, S. 225-249.
- [Lin et al. 2005] Lin, F.; Sung, Y.; Lo, Y.: Effects of Trust Mechanisms on Supply-Chain Performance: A Multi-Agent Simulation Study. In: *International Journal of Electronic Commerce*, 9 (4), 2005, S. 91-112.
- [Liu 2005] Liu, D.: *An Investigation of the Trading Agent Competition: A Thesis Presented in Partial Fulfilment of the Requirements for the Degree of Master of Science in Computer Science at Massey University, Albany, New Zealand*. Massey University, Albany, 2005.
- [Malone 1987a] Malone, Thomas W.: Modeling Coordination in Organizations and Markets. In: *Management Science*, 33 (10): 1987, S. 1317-1332.
- [Malone 1987b] Malone, T.; Yates, J.A.; Benjamin, R.: Electronic Markets and Electronic Hierarchies. *CACM* 30, 1987, S. 484-497.
- [Malone & Crowston 1992] Malone, T. W.; Crowston, K. (1992): *Towards an Interdisciplinary Theory of Coordination*. Technical Report CSS TR#120, Cambridge: Center for Coordination Science, Sloan School of Management, Massachusetts Institut of Technology.
- [Malone & Crowston 1994] Malone, T. W., Crowston, K. (1994): *The Interdisciplinary Study of Coordination*. In: *ACM Computing Surveys*, Ausg. 26, Nummer. 1, März 1994, S. 87-119.
- [Malone et al. 1988] Malone, Thomas W.; Grant, Kenneth R.; Fikes, Richard E.; Howard, Michael T.: *Enterprise: A Market-Like Task Scheduler for Distributed Computing Environments*. In: Huberman, B. A. (Hrsg.): *The Ecology of Computation*. Amsterdam: North-Holland Publishing Company, 1988, S. 177-205.
- [Malsch 1997] Malsch, Thomas: Die Provokation der "Artificial Societies". Warum die Soziologie sich mit den Sozialmetaphern der Verteilten Künstlichen Intelligenz beschäftigen sollte. In: *ZfS Zeitschrift für Soziologie*, Jg. 26, Heft 1, Februar 1997. Enke Verlag Stuttgart, 1997, S. 3-21.
- [Malsch 1998] Malsch, Thomas; Müller, J.H.; Schulz-Schaeffer, I.: *Socionics: Introduction and Potential*. In: *Journal of Artificial Societies and Social Simulation* 1(3) 1998.
- [Malsch 1999] Malsch, Thomas: „Artificial Societies“ außer Kontrolle? Notizen zum Forschungsprogramm der Sozionik. Vortrag auf der Veranstaltung "Zwölfter Stuttgarter Filmwinter", Stuttgart, 14.-17. Januar 1999. <http://www.wand5.de/fiwi99/malsch%20text.htm>, letzter Abruf: 24.09.2015.
- [Malsch 2001] Malsch, Thomas: Naming the Unnamable: Socionics or the Sociological Turn of/to Distributed Artificial Intelligence. In: *Autonomous Agents and Multi-Agent Systems* 3/2001, S. 155-187.
- [Malsch 2002] Malsch, Thomas; Albrecht, Steffen: The Research Programme of 'Socionics - Investigating and Modelling Artificial Societies'. In: *Künstliche Intelligenz*, 3, 2002.
- [Marik et al. 2009] Marik, V.; Strasser, T.; Zoitl, A.: *Holonic and Multi-Agent Systems for Manufacturing: 4th International Conference on Industrial Applications of Holonic and Multi-Agent Systems. HoloMAS 2009. Proceedings. Lecture Notes in Computer Science*. Heidelberg: Springer, 2009. – ISBN 9783642036682
- [Marik et al. 2011] Marik, V.; Vrba, P.; Leitão, P.: *Holonic and Multi-Agent Systems for Manufacturing: 5th International Conference on Industrial Applications of Holonic*

- and Multi-Agent Systems. HoloMAS 2011. LNCS sublibrary: Artificial Intelligence. Springer, 2011. – ISBN 9783642231803
- [Marik et al. 2013] Marik, V.; Lastra, J.L.M.; Skobelev, P.: Industrial Applications of HoloMAS 2013. Proceedings. Lecture Notes in Computer Science. Heidelberg: Springer, 2013. – ISBN 9783642400902
- [Mattern 1998] Mattern, F.: Mobile Agenten. Informationstechnik und Technische Informatik, Nr. 4. Oldenbourg Verlag, 1998.
- [Mentzer et al. 2001] Mentzer, J. T.; DeWitt, W.; Keebler, J. S.; Min, S.; Nix, N. W.; Smith, C. D.; Zacharia, Z. G.: Defining Supply Chain Management. Journal of Business Logistics 22, S. 1 - 25. – DOI:10.1002/j.2158-1592.2001.tb00001.x
- [Möhle et al. 1996] Möhle, S.; Weigelt, M. et al.: Dezentrale Produktionsplanungs- und -steuerungs-Experten: Kombination Wissensbasierter Ansätze mit ComponentWare. In: IM Information Management 1, 1996, S. 30-37.
- [Moulin 1995] Moulin, H.: Cooperative Microeconomics – A Game-Theoretic Introduction. London: Prentice Hall Harvester Wheatsheaf, 1995.
- [Müller 1996a] Müller, J. P.: The Design of Autonomous Agents – A Layered Approach, Volume 1177 of Lecture Notes in Artificial Intelligence. Springer-Verlag, 1996.
- [Müller 1996b] Müller, J. P.: An Architecture for Dynamically Interacting Agents. PhD Thesis. Saarbrücken: Universität des Saarlandes, 1996.
- [Müller 2004] Müller, H.: Gestaltungsanforderungen Internettechnologie-basierter Marktplätze aus Sicht der industriellen Beschaffung : Entwurf eines entscheidungsorientierten objektzentrierten Analyseschemas. Dissertation an der Bayerischen Julius-Maximilians-Universität Würzburg, 2004.
- [Müller & Pischel 1993a] Müller, J. P.; Pischel, M.: InteRRaP: eine Architektur zur Modellierung flexibler Agenten. In: Müller, H. J. (Hrsg.): Verteilte Künstliche Intelligenz – Methoden und Anwendungen. (Distributed Artificial Intelligence – Methods and Applications). Mannheim: BI Wissenschaftsverlag, 1993, S. 45–54.
- [Müller & Pischel 1993b] Müller, J.P.; Pischel, M.: The Agent Architecture InteRRaP. DFKI Research Report (RR-93-26). 1993.
- [Nenninger & Lawrenz 2013] Nenninger, M.; Lawrenz, O.: B2B-Erfolg durch eMarkets – Best Practice: Von der Beschaffung über eProcurement zum Net Market Maker. XBusiness Computing. Springer-Verlag, 2013 – ISBN 3322916057, 9783322916051
- [Newell & Simon 1976] Newell, A.; Simon, H. A.: Computer Science as Empirical Enquiry: Symbols and Search. Communications of the ACM No. 19, 1976.
- [Nissen 2013] Nissen, V.: Einführung in Evolutionäre Algorithmen. Springer-Verlag, 2013. – ISBN 3322938611, 9783322938619
- [Nissen 2000] Nissen, M. E.: Supply Chain Process and Agent Design for E-Commerce. Proceedings of the 33rd Hawaii International Conference on System Science, 2000.
- [Nissen 2001] Nissen, M. E.: Agent-Based Supply Chain Integration. In: Information Technology & Management, Band 2, Nummer 3, 2001, S. 289-312.
- [Nissen 2006] Nissen, V.: Softwareagenten und Soft Computing im Geschäftsprozessmanagement: Innovative Methoden und Werkzeuge zur Gestaltung, Steuerung und Kon-

- trolle von Geschäftsprozessen in Dienstleistung, Verwaltung und Industrie. Tagungsband zum 9. Symposium Soft Computing am 30.11.2006 an der TU Ilmenau. Cuvillier, 2006. – ISBN 9783867270625
- [Nissen 2013] Nissen, V.: Evolutionäre Algorithmen: Darstellung, Beispiele, betriebswirtschaftliche Anwendungsmöglichkeiten. Deutscher Universitätsverlag, 2013. – ISBN 9783322834300
- [Noriega 1997] Noriega, P.: Agent-Mediated Auctions: The Fishmarket Metaphor. PhD Thesis. Universitat Autònoma de Barcelona, 1997.
- [Nwana et al. 1996] Nwana, H. S.; Lee, L.; Jennings, N.: Coordination in Software Agent Systems. *BT Technology Journal*, 14(4), 1996, S. 79-88.
- [Ollmert & Schinzer 2001] Ollmert, Clemens; Schinzer, Heiko: Multiagentensysteme – Aufbau und Anwendungsbereiche in der Wirtschaftsinformatik. In: Meier, A. (Hrsg.): HMD. Heidelberg, (2001) 8, S. 92-102.
- [Papaioannou & Edwards 1998] Papaioannou, T.; Edwards, J.: Mobile Agent technology Enabling the Virtual Enterprise: A Pattern for Database Query. In: Working Notes of the Agent-Based Manufacturing Workshop, Minneapolis, 1998.
- [Papaioannou & Edwards 2000] Papaioannou, T.; Edwards, J.: Manufacturing Systems Integration and Agility: Can Mobile Agents Help?
- [Parunak 1998] Parunak, H. Van Dyke: The DASCh Experience: How to model a Supply Chain.
- [Parunak 1999] Parunak, H. Van Dyke: DASCh: Dynamic Analysis of Supply Chain, 1999.
- [Parunak et al. 1998] Parunak, H. Van Dyke; Baker, A. D.; Clark, S. J.: The AARIA Agent Architecture: From Manufacturing Requirements to Agent- Based System Design. Workshop on Agent-Based Manufacturing. ICAA'98. Minneapolis, 1998.
- [Parunak et al. 1999] Sauter, J.A.; Parunak, H. Van Dyke: ANTS in the Supply Chain. Presented at the Workshop on Agent based Decision Support for Managing the Internet-Enabled Supply Chain. Agents 99. Seattle, 1999.
- [Peters 2013] Peters, Ralf: Elektronische Märkte: Spieltheoretische Konzeption und agentenorientierte Realisierung. Reihe Information Age Economy. Springer-Verlag, 2013. - ISBN 3642575080, 9783642575082
- [Pippow 2004] Pippow, Ingo: Software-Agenten in Distributionsnetzen – Potenziale vertikaler Informationsteilung zur Senkung von Transaktionskosten. Springer-Verlag, 2013. - ISBN 3322816842, 9783322816849.
- [Pippow 2013] Pippow, Ingo: Software-Agenten in Distributionsnetzen: Potenziale vertikaler Informationsteilung zur Senkung von Transaktionskosten (Markt- und Unternehmensentwicklung Markets and Organisations). Springer-Verlag, 2013. - ISBN 3322816842, 9783322816849
- [Pohlheim 2013] Pohlheim, Hartmut: Evolutionäre Algorithmen - Verfahren, Operatoren und Hinweise für die Praxis. Springer-Verlag, 2013. - ISBN 3642571379, 9783642571374
- [Ponomareva & Calinescu 2012] Ponomareva, Natalia; Calinescu, Anisoara: Extending and Evaluating Agent-Based Models of Algorithmic Trading Strategies. In: ICECCS 2012, S. 351-360.

- [Ponomareva & Calinescu 2014] Ponomareva, Natalia; Calinescu, Anisoara: Revisiting Agent-Based Models of Algorithmic Trading Strategies. Transactions on Computational Collective Intelligence XVI Lecture Notes in Computer Science 2014, S. 92-121.
- [Poutré et al. 2006] Poutré, H.L.; Sadeh, N.; Janson, S.: Agent-Mediated Electronic Commerce. Designing Trading Agents and Mechanisms: AAMAS 2005 Workshop, AMEC 2005 and IJCAI 2005 Workshop, TADA 2005. Selected and Revised Papers. Lecture Notes in Computer Science. Heidelberg: Springer, 2006. – ISBN 9783540462439
- [Preist 1998] Preist, C.: Economic Agents for Automated Trading. HP Technical Reports HPL-98-77. Bristol: Hewlett Packard Laboratories, 1998.
- [Pruitt 1981] Pruitt, Dean G.: Negotiation Behaviour. New York: Academic Press, 1981.
- [Puin 2004] Puin, Rolf-Armin: Development and Analysis of a Multiagent-based Architecture for Supply Chain Simulations.
- [Ram & Lui 2005] Ram, S.; Liu, J.: An Agent-Based Approach for Sourcing Business Rules in Supply Chain Management. In: International Journal of Intelligent Information Technologies. I (1), 2005, S. 1-16.
- [Rao & Georgeff 1991] Rao, A. S.; Georgeff, M. P.: Modeling Rational Agents Within a BDI-Architecture. In: Fikes, E. S. R. (Hrsg.); Knowledge Representation and Reasoning (KR&R-91). San Mateo, CA, USA: Morgan Kaufman Publishers, 1991, S. 473 - 484.
- [Rao & Georgeff 1995] Rao, Anand S.; Georgeff, Michael P.: BDI Agents: From Theory to Practice. In: Proceedings of the First International Conference on Multi-Agent Systems. ICMAS-9, 1995, S. 312-319.
- [Rodríguez et al. 1998] Rodríguez-Aguilar. J. A.; Martin, F. J.; Noriega P.; Garcia, P.; Sierra, C.: Competitive Scenarios for Heterogeneous Trading Agents. In: Proceedings of the Second International Conference on Autonomous Agents. AGENTS'98. 1998.
- [Rosenschein & Zlotkin 1994] Rosenschein, Jeffrey S.; Zlotkin, Gilad: Rules of Encounter: Designing Conventions for automated negotiation among Computers. Boston: MIT Press, 1994.
- [Rupprecht-Däullary 1994] Rupprecht-Däullary, Marita: Zwischenbetriebliche Kooperation - Möglichkeiten und Grenzen durch neue Informations- und Kommunikationstechnologien. Gabler Edition Wissenschaft. Wiesbaden: Dt. Univ.-Verl. Gabler, 1994. – ISBN: 3-8244-6109-9, 978-3-8244-6109-7.
- [Russell & Norvig 1995] Russell, Stuart; Norvig, Peter: Artificial Intelligence – A Modern Approach. Englewood Cliffs, NJ, USA: Prentice Hall, 1995
- [Russell & Norvig 2003] Russell, S. J., Norvig, P.: Artificial Intelligence. A Modern Approach. Second Edition. 2. Auflage. Prentice Hall International, 2003.
- [Russell & Norvig 2004] Russell, Stuart; Norvig, Peter: Künstliche Intelligenz – Ein moderner Ansatz. Pearson Studium – IT, 2. Auflage. München: Pearson Education Deutschland, 2004. – ISBN 3-8273-7089-2
- [Ruß 1997] Ruß, Christian: Economic Mechanism Design for the Auction-Based Coordination of Self-Interested Agents. Diplomarbeit in Informatik. Saarbrücken, Universität des Saarlandes, 1997.

- [Ruß & Gerber 2000] Gerber, Andreas; Ruß, Christian: A Holonic Coordination Infrastructure for Agent-based Supply Webs. In: Proceedings of the Eighth Annual Workshop on Information Technologies and Systems WITS'2000. 2000.
- [Ruß & Gerber 2001a] Gerber, Andreas; Ruß, Christian: Holonic Agents for the Simulation of Supply Webs. In: Proceedings of the International Workshop Agent-based Simulation II, April 2-4, Passau (Germany), SCS Publishing House, 2001.
- [Ruß & Gerber 2001b] Gerber, Andreas; Ruß, Christian: A Holonic Multi-agent Infrastructure for Electronic Procurement. In: Proceedings of the Fourteenth Canadian Conference on Artificial Intelligence (AI 2001), June 7-9, Ottawa (Canada), 2001.
- [Ruß & Gerber 2001c] Gerber, Andreas; Ruß, Christian: A Holonic Multi-agent Co-ordination Server. In: Proceedings of the 14th International FLAIRS Conference. 2001.
- [Ruß & Vierke 1998a] Ruß, C.; Vierke, G.: Agent-Based Configuration of Virtual Enterprises. In: Proceedings of the KI'98 Workshop: Intelligente Agenten im Informations- und Prozeßmanagement, 1998.
- [Ruß & Vierke 1998b] Ruß, C.; Vierke, G.: The Matrix Auction: An Efficient Market Mechanism for Virtual Enterprises. In: Proceedings of the Eight Annual Workshop on Information Technologies and Systems WITS'98, 1998. – ISBN 951-39-0364-8
- [Ruß & Vierke 1999a] Ruß, C.; Vierke, G.: Die Matrix Auktion: Ein Marktmechanismus zur Koordination von Virtuellen Unternehmen. In: Tagungsband Workshop Sicherheit und Electronic Commerce. WSSEC'98. Vieweg Verlag, 1999.
- [Ruß & Vierke 1999b] Ruß, C.; Vierke, G.: The Matrix Auction: A Mechanism for the Market-Based Coordination of Enterprise Networks. Research Report. Deutsches Forschungszentrum für Künstliche Intelligenz, 1999.
- [Ruß & Walz 2009a] Ruß, C.; Walz, A.: An Interoperability Framework for the Negotiation-Based Coordination of Adaptive Supply Web Agents. In: Proceedings of ATOP 2009 (Agent-based Technologies and applications for enterprise interoperability) at the 8th International Joint Conference on Autonomous Agents & Multi-Agent Systems (AAMAS 2009). Springer, 2009.
- [Ruß & Walz 2009b] Ruß, C.; Walz, A.: MACSIMA: An Agent Framework for Simulating the Evolution of Negotiation Strategies in B2B-Networked Economies. In: Proceedings of the 23rd European Conference on Modelling and Simulation. ECMS 2009, Track: Agent-Based Simulation (ABS). Madrid, Spain, 2009.
- [Ruß & Walz 2009c] Ruß, C.; Walz, A.: MACSIMA: On the Effects of Adaptive Negotiation Behavior in Agent-Based Supply Networks. In: Proceedings of the 7th German Conference Series on Multiagent System Technologies. MATES 2009. Springer, 2009.
- [Ruß & Walz 2009d] Ruß, C.; Walz, A.: MACSIMA: Simulating the Co-Evolution of Negotiation Strategies in Agent-Based Supply Networks. In: Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology. IAT 2009. Milan, Italy, 2009.
- [Ruß & Walz 2012] Russ, C.; Walz, A.: MACSIMA: An Agent Framework for Simulating the Evolution of Negotiation Strategies in B2B-Networked Economies. In: Lecture Notes in Business Information Processing (LNBIP 98) of the ATOP 2009 and ATOP 2010 Workshops for Agent-based Technologies and Applications for Enterprise Interoperability. Heidelberg, Springer-Verlag, 2012. – ISBN: 978-3-642-28562-2, DOI: 10.1007/978-3-642-28563-9

- [Ruß et al. 1999] Gerber, C.; Ruß, C.; Vierke, G.: On the Suitability of Market-Based Mechanisms for Telematics Applications. In: Proceedings of the International Conference on Autonomous Agents (Agents'99). 1999.
- [Ruß et al. 2001] Gerber, Andreas; Klusch, Matthias; Ruß, Christian; Zinnikus, Ingo: Hologonic Agents for the Coordination of Supply Webs. In: Proceedings of the 5th International Conference on Autonomous Agents (Agents'2001). Montreal, Canada, 2001.
- [Ruß et al. 2002] Gerber, Andreas; Ruß, Christian; Klusch, Matthias. An agent-based approach for integrating logistics services into a B2B marketplace. Proceedings of the International Conference of Internet Computing (IC'02) (Volume III), (ISBN 1-892512-37-8), pages 705-712, 2002.
- [Ruß et al. 2003] Gerber, Andreas; Klusch, Matthias; Ruß, Christian: Supply Web Coordination by an Agent-based Trading Network with Integrated Logistics Services. International Journal of Electronic Commerce Research and Applications. Band 2. Elsevier, 2003. - ISSN 1567-4223
- [Sackmann 2003] Sackmann, Stefan: Bilaterale Preisverhandlungen von Software-Agenten: Ein Modell und System zur Analyse des marktplatzspezifischen Verhandlungsspielraumes. Markt- und Unternehmensentwicklung, Markets and Organisations. Deutscher Universitätsverlag, 2003. – ISBN 3824478536, 9783824478538
- [Sackmann 2013] Sackmann, S.: Bilaterale Preisverhandlungen von Software-Agenten: Ein Modell und System zur Analyse des marktplatzspezifischen Verhandlungsspielraumes. Markt- und Unternehmensentwicklung Markets and Organisations. Deutscher Universitätsverlag, 2013. – ISBN 9783322815330
- [Sadeh et al. 1999] Sadeh-Konieczpol, N.; Hildum, D.; Kjenstad, D.; Tseng, A.: MASCOT: An Agent- Based Architecture for Coordinated Mixed-Initiative Supply Chain Planning and Scheduling. In: Workshop Notes, Agent-Based Decision Support for Managing the Internet-Enabled Supply Chain. Third International Conference on Autonomous Agents. Agents '99. 1999.
- [Sadeh et al. 2001] Sadeh-Konieczpol, N. M.; Hildum, D.; Kjenstad, D.; Tseng, A.: Mascot: An Agent-Based Architecture for Dynamic Supply Chain Creation and Coordination in the Internet Economy. Production Planning and Control. I2(3), 2001. S. 212-223. - DOI: 10.1080/095372801300107680
- [Sadeh et al. 2003] Sadeh, N. M.; Hildum, D. W.; Kjenstad, D.: Agent-Based E-Supply Chain Decision Support. In: Journal of Organizational Computing and Electronic Commerce, I3(3/4), 2003, S. 225-241. - DOI: 10.1207/S15327744JOCE133&4_05
- [Sandholm 1993] Sandholm, Tuomas W.: An Implementation of the Contract Net Protocol Based on Marginal Cost Calculations. In: Proceedings of the Eleventh National Conference on Artificial Intelligence. AAAI'93. 1993.
- [Sandholm 1999] Sandholm, T.W.: An Algorithm for Optimal Winner Determination in Combinatorial Auctions. International Joint Conference on Artificial Intelligence (IJCAI). 1999, S. 542-547.
- [Sandholm & Lesser 1995a] Sandholm Tuomas; Lesser, Victor: Advantages of a Leveled Commitment Contracting Protocol. In: University of Massachusetts at Amherst, Computer Science Technical report TR 95-72, 1995.
- [Sandholm & Lesser 1995b] Sandholm, Tuomas W.; Lesser, Victor R.: Issues in Automated Negotiation and Electronic Commerce: Extending the Contract Net Framework. In

- Proceedings of the First International Conference on Multiagent Systems. ICMAS '95. 1995.
- [Sandholm & Lesser 1995c] Sandholm, Tuomas; Lesser, Victor: On Automated Contracting in Multi-enterprise manufacturing. In: Proc. Improving Manufacturing Performance in a Distributed Enterprise: Advanced Systems and Tools. Edinburgh, 1995, S. 33-42.
- [SAP 2002] SAP: SAP White Paper: Adaptive Supply Chain Networks. 2002.
- [Schillo 2004] Schillo, M.: Multiagent Robustness: Autonomy vs. Organization. Dissertation. Universität des Saarlandes, 2004.
- [Schimmel & Zelewski 1996] Schimmel, Katrin; Zelewski, Stefan: Untersuchung alternativer Auktionsformen hinsichtlich ihrer Eignung zur Koordination verteilter Agenten auf Elektronischen Märkten. Universität Leipzig: Arbeitsbericht Nr. 19 des Instituts für Produktionswirtschaft und Industrielle Informationswirtschaft, 1996.
- [Schmid 1993] Schmid, B.: Elektronische Märkte; In: Wirtschaftsinformatik Vol. 35, Nr. 5, Vieweg Verlag 1993, S. 465-480.
- [Schmidt 2013] Schmidt, Claudia: Marktliche Koordination in der dezentralen Produktionsplanung: Effizienz — Komplexität — Performance. Springer-Verlag, 2013. — ISBN 3663087336, 9783663087335
- [Seifert 2006] Seifert, D.: Konzepte des Supply Chain Managements – CPFR als unternehmensübergreifende Lösung. In: Zentes, J. (Hrsg.): Handbuch Handel. Wiesbaden, 2006, S. 781–794.
- [Sikora & Shaw 1998] Sikora, R.; Shaw, M. J.: A Multi-Agent Framework for the Coordination and Integration of Information Systems. In: Management Science 44 (11-part-2), 1998, S. 65-78.
- [Shen & Norrie 1998a] Shen, W.; Norrie, D.H.: An Agent- Based Approach for Manufacturing Enterprise Integration and Supply Chain Management. In: Jacucci, G. et al. (Hrsg.): Globalization of Manufacturing in the Digital Communications Era of the 21st Century: Innovation, Agility, and Virtual Enterprise. Kluwer Academic Publisher, 1998, S. 579- 590.
- [Shen & Norrie 1998b] Shen, W.; Norrie, D.H.: A Hybrid Agent-Oriented Infrastructure for Modeling Manufacturing Enterprises. In: Proceedings of KAW'98. Agent-5. 1998, S. 1-19.
- [Shen et al. 1999] Shen, W.; Norrie, D.H.; Kremer, R.: Developing Intelligent Manufacturing Systems Using Collaborative Agents. In: Proceedings of the 2nd International Workshop on Intelligent Manufacturing Systems. Leuven. 1999, S. 157-166.
- [Smith 1776] Smith, Adam: An Inquiry into the Nature and Causes of the Wealth of Nations. Vol. I/ Vol. II. Printed for W. Strahn; and T. Cadell, in the Strand, 1776. München: IDION-Verlag, 1976 (als Vorlage diente eine sich in der Universitätsbibliothek Heidelberg befindende Originalausgabe).
- [Smith 1980] Smith, Reid G.: The Contract Net Protocol: High Level Communication and Control in a Distributed Problem Solver. In: IEEE Transactions on Computers, C-29, Nr. 12, Dezember 1980, S. 1104–1113.
- [Smith & Davis 1988]. Smith, Reid G; Davis, R. G.: Negotiation as a Metaphor for Distributed Problem Solving. In: Bond, A.; Gassor, L. (Hrsg.): Readings in DAI. 333-356, 1988.

- [Smith et al.1999] Smith, R.E.; Kearney, P.J.; Merlat, W.: Evolutionary Adaptation in Autonomous Agent Systems - A Paradigm for the Emerging Enterprise. In: *BT Technology Journal*. Springer Netherlands. Ausgabe 17, Nummer 4, Oktober 1999, S. 157-167.
- [Smith & Taylor 1998]. Smith, Robert E.; Taylor, Nick: A Framework for Evolutionary Computation in Agent-Based Systems. In: Looney, D., Castaing, J.: *Proceedings of the 1998 International Conference on Intelligent Systems*, S. 221-224. <http://www.cems.uwe.ac.uk/~rsmith/fecabs.pdf>
- [Spriestersbach 1999] Spriestersbach, Axel: Analyse und Einsatz von Negotiations-Verfahren für Mobile Agenten im Electronic Commerce. Diplomarbeit an der Technischen Universität Darmstadt, Fachbereich Informatik, Informationstechnologie. Transfer Office, Prof. Dr. Kammerer, 1999.
- [Stadtler 2010] Stadtler, H.; Kilger, C.; Meyr, H. (Hrsg.): *Supply Chain Management und Advanced Planning: Konzepte, Modelle und Software*. Heidelberg: Springer, 2010. - ISBN 3642141307, 9783642141300
- [Steels & Brooks 1995] Steels, L.; Brooks, R.A.: *The Artificial Life Route to Artificial Intelligence: Building Embodied, Situated Agents*. (The Artificial Life Route to Artificial Intelligence: Building Embodied, Situated Agents). L. Erlbaum Associates, 1995. - ISBN 9780805815184
- [Stickel et al. 1998] Stickel, E.; Groffmann, H.-D.; Rau, K.-H.: *Gabler Wirtschafts-*[Steels 1994a] Steels, Luc: Building Agents with Autonomous Behavior Systems. In: Steels, Luc; Brooks, Rodney (Hrsg.): *The Artificial Life Route to Artificial Intelligence: Building Situated Embodied Agents*. New Haven: Lawrence Erlbaum Associates, 1994.
- [Steels 1994] Steels, Luc: *The Artificial Life Roots of Artificial Intelligence*. *Artificial Life Journal*, 1(1), 1994.
- [Steels & Brooks 1995] Steels, Luc; Brooks, Rodney Allen; (Hrsg.): *The Artificial Life Route to Artificial Intelligence: Building Embodied, Situated Agents*. L. Erlbaum Associates, 1995. ISBN 080581518X, 9780805815184.
- [Stuedter 2010] Stuedter, Sven: Design and Analysis of a Generic 2-Pool-Adaptation-Process for Buy and Sell Negotiations. Bachelor-Arbeit am Lehrstuhl von Professor Dr. Jörg H. Siekmann. Saarbrücken, Universität des Saarlandes, 2010.
- [Stuedter 2013] Stuedter, Sven: Implementierung und Evaluation von Koordinationsmechanismen in der Multiagenten-Supply-Chain-Simulationsumgebung MACSIMA. Master-Arbeit am Lehrstuhl von Professor Dr. Jörg H. Siekmann. Saarbrücken, Universität des Saarlandes, 2013.
- [Strader et al. 1998] Strader, T. J.; Lin, F.-R.; Shaw, M. J.: Simulation of Order Fulfillment in Divergent Assembly Supply Chains. In: *Journal of Artificial Societies and Social Simulation* 1 (1998) 2.
- [Supply Chain Council 2005] Supply Chain Council _Supply-chain Operations Reference Model: SCORVersion 7.0 Overview".
- [Svensson 2004] Svensson, Göran: Key Areas, Causes and Contingency Planning of Corporate Vulnerability in Supply Chains. In: *International Journal of Physical Distribution & Logistics Management*. 34 (9), S. 728-48.

- [Swaminathan et al. 1998] Swaminathan, J.M.; Smith, S.F.; Sadeh, N.M.: Modeling Supply Chain Dynamics: A Multi-Agent Approach. In: Decision Sciences 29 (1998) 3.
- [Szapiro et al. 1991] Szapiro, T.; Haigh, K.; Matwin, S.: Genetic Algorithms Approach to a Negotiation Support System. In: IEEE Transactions on Systems, Man and Cybernetics, Ausgabe 21, Nummer 1, Januar/Februar 1991.
- [Szirbik et al. 1999] Szirbik, N.B.; Hammer, D.K.; Goossenaerts, J.B.M.; Aerts, A.T.M.: Mobile Agent Support for Tracking Products in Virtual Enterprises. In: The working papers of the Workshop on Agent-Based Decision-Support for Managing the Internet Enabled Supply-Chain. AGENTS'99 Conference. Seattle, S. 93-100, 1999.
- [Szirbik et al. 2000] Szirbik, N. B.; Wortmann, J.C.; Hammer, D.K.; Goossenaerts, J.B.M.; Aerts, A.T.M.: Mediating Negotiations in a Virtual Enterprise via Mobile Agents. Mobile Technologies and Virtual Enterprises. AIWoRC'2000. 2000.
- [Teuteberg 2005] Teuteberg, F.: Realisierung ubiquitärer Supply Networks auf Basis von Auto-ID- und Agenten-Technologien – Evolution oder Revolution? In: Ferstl, O. K.; Sinz, E. J.; Eckert, S.; Isselhorst, T. (Hrsg.): Wirtschaftsinformatik 2005: eEconomy, eGovernment, eSociety; Heidelberg: Physica-Verlag, 2005, S. 3-22.
- [Teuteberg 2007] Teuteberg, F.: Analyse, Design und Realisierung agentenbasierter Marktplätze – Referenzmodellierung und Simulation multidimensionaler Verhandlungen; Osnabrück, 2007.
- [Tsvetovaty et al. 1997] Tsvetovaty, M.; Gini, M.; Mobasher, B.; Wieckowski, Z.: MAGMA: An Agent-Based Virtual Market for Electronic Commerce. Department of Computer Science University of Minnesota, Minneapolis, MN, USA, 1997.
- [Tsyganova 2000] Tsyganova, Elena: Supply Chain Management als Ansatz zur Effizienzsteigerung in der Logistikkette. Verlag Diplom.de, 2000. - ISBN 3832421793, 9783832421793
- [Turing 1950] Turing, A., Computing Machinery and Intelligence. Mind, 1950. <http://www.loebner.net/Prizef/TuringArticle.html>, letzter Abruf: 15.05.2007.
- [Vahrenkamp 1996] Vahrenkamp, Richard: Logistikmanagement. 3. Auflage. München: Oldenbourg Verlag, 1996.
- [Vahrenkamp 2008] Vahrenkamp, Richard: Produktionsmanagement. München: Oldenbourg Verlag, 2008. – ISBN 3486587846, 9783486587845
- [van Dinther 2007] van Dinther, C.: Adaptive Bidding in Single-Sided Auctions under Uncertainty: An Agent-based Approach in Market Engineering. Whitestein Series in Software Agent Technologies and Autonomic Computing. Basel: Birkhäuser, 2007. – ISBN 9783764381134
- [Varian 1999] Varian, H. R. Grundzüge der Mikroökonomik. 4. Auflage. München: Oldenbourg, 1999.
- [Vetter 2006] Vetter, Michael: Ein Multiagentensystem zur Verhandlungsautomatisierung in elektronischen Märkten. IPA-IAO-Forschung und Praxis, Ausgabe 446. Jost-Jetter Verlag, 2006. - ISBN 3936947996, 9783936947991
- [Vickrey 1961] Vickrey, W.: Counterspeculation, auctions and competitive sealed tenders. In: Journal of Finance, 16, 1961, S. 8–37.
- [Vierke 2000] Vierke, G.: TELETRUCK - A Holonic Multi-Agent System for Telematics. PhD Thesis. Saarbrücken: Universität des Saarlandes, 2000.

- [Vogt 2010] Vogt, Franziska: Evolutionäre Algorithmen am Beispiel einer Supply Chain Problematik. GRIN Verlag, 2010. – ISBN 9783640555239
- [von Martial 1992] Frank von Martial: Einführung in die Verteilte Künstliche Intelligenz. In: KI-Künstliche Intelligenz 1/92, 1992, S. 6-11.
- [Vorobeychik 2008] Vorobeychik, Y.: Mechanism Design and Analysis Using Simulation-based Game Models. University of Michigan, 2008. – ISBN 9780549820529
- [Walras 1874] Walras, L.: *Éléments d'économie politique pure: ou théorie de la richesse sociale*. Paris: Economica, 1988. Nachdruck von Lausanne: L. Corbaz, 1874.
- [Walz 2007] Walz, A.: Design, Implementierung und Analyse von Verhandlungsstrategien und Koordinationsmechanismen für Geschäftsagenten in Wertschöpfungsnetzwerke. Diplomarbeit am Lehrstuhl von Professor Dr. Jörg H. Siekmann. Saarbrücken, Universität des Saarlandes, 2007.
- [Wannenwetsch & Nicolai 2013] Wannenwetsch, Helmut H.; Nicolai, Sascha.: E-Supply-Chain-Management: Grundlagen — Strategien — Praxisanwendungen. Springer-Verlag, 2013. - ISBN 3322828735, 9783322828736
- [Weicker 2007] Weicker, K.: Evolutionäre Algorithmen. Leitfäden der Informatik. Vieweg+Teubner Verlag, 2007. – ISBN 9783835102194
- [Weicker 2015] Weicker, Karsten: Evolutionäre Algorithmen. Springer-Verlag, 2015. – ISBN 3658099585, 9783658099589
- [Weinhardt & Gomber 1995] Weinhardt, Christof; Gomber, Peter: Domänenunabhängige Koordinationsmechanismen für die dezentrale betriebliche Planung. 1995.
- [Weinhardt et al. 1996] Gomber, Peter; Schmidt, Claudia; Weinhardt, Christof. Synergie und Koordination in dezentral planenden Organisationen. In: Wirtschaftsinformatik 38, 3, 1996, S. 299 – 307.
- [Weiss 1999] Weiss, G.: Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence. MIT Press, 1999.
- [Weiss 2000] Weiss, Gerhard.: Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. Cambridge, Massachusetts, USA: MIT Press, 2000.
- [Wellman & Wurman 1997] Wellman, M. P., Wurman, P.R.: Market-Aware Agents for a Multiagent World. In: Boman, Magnus; Van de Velde, Walter (Hrsg.): Multi-Agent Rationality: Proceedings of Eight European Workshop on Modelling Autonomous Agents in a Multi-Agent World. MAAMAW-97. Springer Notes in Artificial Intelligence. Heidelberg: Springer, 1997.
- [Wellman 2011] Wellman, M.: Trading Agents. Morgan & Claypool Publishers, 2011 (Synthesis Lectures on Artificial Intelligence & Machine Learning). – ISBN 9781598296068
- [Wellman et al. 2007] Wellman, M.P.; Greenwald, A.; Stone, P.: Autonomous Bidding Agents: Strategies and Lessons from the Trading Agent Competition. Intelligent Robotics and Autonomous Agents. MIT Press, 2007. – ISBN 9780262232609
- [Wikipedia 2015] Wikipedia, die freie Enzyklopädie. Kooperation. <http://de.wikipedia.org/w/index.php?title=Kooperation&oldid=138795735>". Online abgerufen am 20. Februar 2015.

-
- [Wooldridge & Jennings 1995] Wooldridge, M. J.; Jennings N. R. (Hrsg.): Intelligent Agents - Theories, Architectures, and Languages, Band 890: Lecture Notes in Artificial Intelligence. Springer-Verlag, 1995.
- [Wooldridge & Jennings 1995] Wooldridge, M.J.; Jennings, N.R. (Hrsg.): Intelligent Agents: Theory and practice. In: Knowledge Engineering Review 1995, Nr. 10, Januar 1995, S. 115–152.
- [Wooldridge 2002] Wooldridge, M.: An Introduction to Multiagent Systems. John Wiley & Sons, 2002.
- [Wooldridge 2009] Wooldridge, M.: An Introduction to MultiAgent Systems. Wiley, 2009. – ISBN 9780470519462
- [Yung et al. 2000] Yung, S. K.; Yang, C. C.; Yen, J.: Applying Multi-Agent Technology to Supply Chain Management. In: Journal of Electronic Commerce Research, Special Issue on Intelligent Agents in E-Commerce, Band 1, Nummer 4, November 2000.
- [Zarnekow 2013] Zarnekow, R.: Softwareagenten und elektronische Kaufprozesse: Referenzmodelle zur Integration. Deutscher Universitätsverlag, 2013. – ISBN 9783322952059
- [Zelewski 1995] Zelewski, S.: Multi-Agenten-Systeme zur Koordination von Produktionsprozessen. In: Scheer, A.-W. (Hrsg.): 16. Saarbrücker Arbeitstagung, 1995, S. 123-150.
- [Zelewski 1997] Zelewski, S.: Elektronische Märkte zur Prozeßkoordinierung in Produktionsnetzwerken, Wirtschaftsinformatik, 39 (1997), S. 231-243.
- [Zelewski 1998] Zelewski, S.: Auktionsverfahren zur Koordinierung von Agenten auf elektronischen Märkten. In: Becker, M.; Kloock, J.; Schmidt, R.; Wäscher, G. (Hrsg.): Unternehmen im Wandel und Umbruch, Stuttgart, 1998, S. 305-337.
- [Zeng & Sycara1998] Zeng, D.; Sycara, K.: Agent-Facilitated Real-Time Flexible Supply Chain Structuring.
- [Zenner 2004] Zenner, Christian: Agentenbasierte Verhandlungsmechanismen für die Bildung Virtueller Unternehmensstrukturen. Diplomarbeit am Lehrstuhl von Prof. Dr. Jörg H. Siekmann. Saarbrücken, Universität des Saarlandes, 2004.
- [Zimmermann 2006] Zimmermann, Roland: Agent-based Supply Network Event Management. Whitestein Series in Software Agent Technologies and Autonomic Computing. Springer Science & Business Media, 2006. - ISBN 376437487X, 9783764374877
- [Zlotkin & Rosenschein 1994] Rosenschein, Jeffrey S.; Zlotkin, Gilad: Rules of Encounter: Designing Conventions for automated negotiation among Computers. Boston: MIT Press, 1994.

11 Anhang

11.1 Testreihen

Im Folgenden werden die in Kapitel 8 durchgeführten Testreihen im Detail beschrieben. Dadurch kann genau nachvollzogen werden, wie die Simulations- und Evaluationsergebnisse erzielt wurden. Darüber hinaus ist aus der Beschreibung der Testreihen ersichtlich, dass jeweils eine große Zahl an Szenarien mit vielen Wiederholungen simuliert und ausgewertet wurde, um die Evaluationsergebnisse durchgängig statistisch abzusichern. Insgesamt basieren die in Kapitel 8 vorgestellten Evaluationsergebnisse daher auf einer statistisch gut abgesicherten Datenbasis.

Erläuterungen zu den Namenskomponenten der genannten Szenarien und Parametrisierungen finden sich in Abschnitt 6.3.

Im folgenden Abschnitt werden zunächst die in [Eymann 2000] mit dem AVALANCHE-System durchgeführten Experimente erläutert. Dies erleichtert zum einen das Verständnis von Abschnitt 8.2, in dem das Systemverhalten von AVALANCHE und MACSIMA miteinander verglichen wird. Zum anderen sind in einigen der mit MACSIMA durchgeführten Testreihen Simulationsszenarien enthalten, die Parameter-Einstellungen der AVALANCHE-Experimente verwenden oder stark an diese angelehnt sind. Daher erleichtern die in Tabelle 14 und Tabelle 15 angegebenen Informationen und Szenario-Benennungen auch die Beschreibung und das Verständnis der mit dem MACSIMA-System im Rahmen dieser Arbeit durchgeführten und in diesem Anhang beschriebenen Testreihen.

11.1.1 AVALANCHE-Experimente

In [Eymann 2000] wurde mit dem AVALANCHE-System eine Reihe von Experimenten durchgeführt, um das Koordinationsverhalten von Agenten auf elektronischen Marktplätzen zu untersuchen. In diesem Abschnitt wird ein Überblick dieser Experimente mit ihren genauen Parameter-Einstellungen gegeben.

Eine Übersicht der mit AVALANCHE durchgeführten Experimente, die aus [Eymann 2000, S. 176] stammt, gibt Tabelle 14. Es handelt sich dabei um sechs grundlegende Testreihen. Da zwei davon mit Variationen durchgeführt wurden, ergeben sich daraus neun verschiedene Untersuchungen. Tabelle 15 gibt anschließend die genauen Gen-Wert-Einstellungen für die mit AVALANCHE durchgeführten Experimente an. Die Einträge der Form [Eymann-Nr.] in der ersten Spalte von Tabelle 14 referenzieren dabei auf die entsprechenden Zeilen in Tabelle 15.

Das erste in Tabelle 14 beschriebene Experiment (Eymann_1) sollte zeigen, dass das Multiagenten-System den Markt koordiniert. Es wurden 50 Agenten je Typ, das heißt insgesamt 250 Agenten 600 Sekunden lang simuliert. Dabei waren die Strategien der Agenten homogen und die Startpreise vorgegeben (siehe [Eymann 2000, S. 177 ff.]).

Die zweite Testreihe (Eymann_2 und Eymann 2.2, siehe [Eymann 2000, S. 181 f.]) hat die Auswirkungen des AQ-Gens untersucht – insbesondere in Bezug auf die Anzahl der Transaktionen und sein Zusammenwirken mit dem SF-Gen.

Die dritte Testreihe (Eymann_3, vgl. [Eymann 2000, S. 185 f.]) diente der Untersuchung des Zusammenspiels der Gene AQ und DJ. Dazu wurden die Agenten leicht unterschiedlich parametrisiert. Die Zimmermann-Agenten in der Mitte der AVALANCHE-Wertschöpfungskette erhielten mit 0,6 einen höheren AQ-Wert als alle anderen Agenten mit lediglich 0.4.

In der vierten Experiment-Reihe (Eymann_4.1 – 4.3, siehe [Eymann 2000, S. 185 ff.]) wurden die Agenten mit der Fähigkeit ausgestattet, ihr Verhalten durch Nutzung des STDEA-Verfahrens während der Simulation anzupassen. Zusätzlich wurde im Test Eymann_4.1 das AQ-Gen als kontinuierliche Zufallsvariable mit einem Mittelwert von 0.5 und einer Breite der Gleichverteilung von 0.9 erzeugt; die Werte liegen also zumeist im Intervall [0.05; 0.95].

Dieser Test wurde mit dem Mittelwert 0,6 für das AQ- Gen wiederholt (s. Eymann_4.2). Ein erster Stufentest folgt, bei dem die Zimmermann-Agenten mit einem AQ-Wert von 0,6 und alle anderen mit einem AQ-Wert von 0,5 initialisiert werden (Eymann_4.3).

Beim fünften von Eymann durchgeführten Test (Eymann_5) verkauft jeder Agent sein Gut am Anfang der Simulation zu einem zufälligen Preis zwischen 10 und 120 Geldeinheiten.

Kapitelnummer und Grafik [Eymann-Nr.]	Konzessionsbereitschaft (p_acq bzw. AC)	Konzessionshöhe (del_chg bzw. DC)	Zufriedenheit (p_sat bzw. SF)	Adaption durch evolutionären Algorithmus	Startpreise
5.1 [Eymann_1]	homogen über die gesamte Population	homogen über die gesamte Population	homogen über die gesamte Population	Nein	vorgegeben
5.2 [Eymann_2] [Eymann_2.2]	homogen über die gesamte Population	homogen über die gesamte Population	homogen über die gesamte Population	Nein	vorgegeben
5.3 [Eymann_3]	homogen innerhalb eines Agenten-Typs	homogen über die gesamte Population	homogen über die gesamte Population	Nein	vorgegeben
5.4 [Eymann_4.1] [Eymann_4.2] [Eymann_4.3]	Zufallsverteilung	homogen über die gesamte Population	homogen über die gesamte Population	Ja	vorgegeben
5.5 [Eymann_5]	homogen über die gesamte Population	homogen über die gesamte Population	homogen über die gesamte Population	Nein	zufällig
5.6 Abbildung 39 [Eymann_6]	homogen über die gesamte Population	Zufallsverteilung	homogen über die gesamte Population	Ja	vorgegeben
5.6 Abbildung 42 + 43 [Eymann_7]	Zufallsverteilung	Zufallsverteilung	homogen über die gesamte Population	Ja	vorgegeben
5.6 Abbildung 44 [Eymann_8]	Zufallsverteilung	Zufallsverteilung	homogen über die gesamte Population	Ja	zufällig
5.6 Abbildung 45 [Eymann_9]	Zufallsverteilung	Zufallsverteilung	Zufallsverteilung	Ja	zufällig

Tabelle 14: Übersicht der mit AVALANCHE durchgeführten Experimente⁷⁶

⁷⁶ Quelle: [Eymann 2000, S. 176]

Im Experiment Eymann_6 wurde untersucht, wie sich Variationen der Konzessions-Schritte auf die Ergebnisse auswirken. Der Wert des DC-Gen wurde dazu zufällig im Intervall [0.05; 0.65] initialisiert.

Im Test Eymann_7 werden zwei Gen-Werte zufällig initialisiert, nämlich das AQ-Gen mit einem Mittelwert von 0.6 und einer Breite der Gleichverteilung von 0.3 (also im Intervall [0.3; 0.9]) und der DC-Wert mit einem Mittelwert von 0.35 und einer Breite von 0.3 (also im Intervall [0.05; 0.65]). Die Startpreise der Güter sind zum Simulationsstart fest vorgegeben.

Dieses Experiment wurde durch den Test Eymann_8 um den Aspekt zufälliger Startpreise erweitert.

In Eymann_9 wird neben dem AQ- und dem DC-Gen auch das SF-Gen mit einer Zufallsverteilung initialisiert.

Eymann-Test	AQ	DC	DJ	SF	WM	RP	STDEA	Agenten
Eymann_1	0,5	0,25	0,15	0,75	0,2	1		5 x 50
Eymann_2	0,9	0,25	0,15	0,75	0,2	1		5 x 50
Eymann_2.2	variabel	0,25	0,15	0,75	0,2	1		5 x 50
Eymann_3	0,4/ 0,6	0,25	0,15	0,75	0,2	1		5 x 50
Eymann_4.1	0,5 +/-0,45	0,25	0,15	0,75	0,2	1		5 x 50
Eymann_4.2	0,6 +/- 0,3	0,25	0,15	0,75	0,2	1		5 x 50
Eymann_4.3	0,5 / 0,6 +/- 0,3	0,25	0,15	0,75	0,2	1	An	5 x 50
Eymann_5	0,75	0,25	0,15	0,75	0,2	1		5 x 20
Eymann_6	0,6	0,25 / 0,35 +/- 0,3	0,15	0,75	0,2	1	An	5 x 20
Eymann_7	0,6 +/- 0,3	0,35 +/- 0,3	0,15	0,75	0,2	1	An	5 x 20
Eymann_8	0,6 +/- 0,3	0,35 +/- 0,3	0,15	0,75	0,2	1	An	5 x 20
Eymann_9	0,6 +/- 0,3	0,35 +/- 0,3	0,15	0,6 +/-0,3	0,2	1	An	5 x 20

Tabelle 15: Gen-Wert-Einstellungen für die AVALANCHE-Experimente

11.1.2 Testreihe zu Startwert-Effekten

Im Rahmen dieser Testreihe wurden fast 200 Szenarien mit jeweils fünfzig initialisierten Agenten untersucht. Jedes dieser Szenarien wurde viermal simuliert.

Die Agenten waren dabei auf die gleichen Fähigkeiten beschränkt, mit denen auch die Agenten im AVALANCHE-System (vgl. [Eymann 2000]) ausgestattet waren. Die in Kapitel 5 beschriebenen Erweiterungen von MACSIMA wurden bewusst ausgeschaltet, um das Verhalten der Agenten bei variierenden Gen-Wert-Initialisierungen zum Simulationsstart unabhängig von sonstigen Einflüssen untersuchen zu können.

Die Definition von zweiundzwanzig dieser Szenarien orientierte sich direkt an den Gen-Startwert-Parametrisierungen und sonstigen Einstellungen der in [Eymann 2000] durchgeführten Experimente (vgl. Abschnitt 11.1.1). Die entsprechenden Szenarien sind in Tabelle 16 aufgeführt. In allen Szenarien – bis auf die Szenarien, die das Verfahren von [Smith &

Taylor 1998]⁷⁷ nutzen – waren die Lernfähigkeiten der Agenten und der Informationsaustausch zwischen ihnen ausgeschaltet.

Eymann_1-NoLearning	Eymann_2-AQ(0,5)-NoLearning-	Eymann_3-NoLearning	Eymann_5-DJstatic-NoLearning
Eymann_2-NoLearning	Eymann_2-AQ(0,6)-NoLearning-	Eymann_4_1-Smith98	Eymann_6-Smith98
Eymann_2-AQ(0,1)-NoLearning	Eymann_2-AQ(0,7)-NoLearning	Eymann_4_2-Smith98	Eymann_7-Smith98
Eymann_2-AQ(0,2)-NoLearning-	Eymann_2-AQ(0,8)-NoLearning-	Eymann_4_3-Smith98	Eymann_9-Smith98
Eymann_2-AQ(0,3)-NoLearning	Eymann_2-AQ(0,85)-NoLearning	Eymann_5-NoLearning	
Eymann_2-AQ(0,4)-NoLearning	Eymann_2-AQ(0,9)-NoLearning-	Eymann_5-DJrandom-NoLearning	

Tabelle 16: An [Eymann 2000] orientierte Szenario-Definitionen zur Untersuchung von Gen-Startwert-Effekten

Darüber hinaus wurde eine Reihe von Szenarien definiert, in denen die Lernfähigkeiten der Agenten und der Informationsaustausch zwischen ihnen ebenfalls ausgeschaltet waren und die Startwerte der Gene jeweils mit $0,2$, $0,5$ oder $0,8$ initialisiert wurden. Insgesamt wurden auf diese Weise weitere ca. 170 Szenarien mit variierenden Gen-Startwert-Kombinationen definiert.

Dabei wurde darauf geachtet, die Gen-Wert-Initialisierungen so auszuwählen, dass eine breite Abdeckung des Ergebnisraums für wesentliche Ergebnisvariablen, insbesondere Gesamtumsatz und Gesamtgewinn, aber auch die Anzahl an Verkaufsabschlüssen (beziehungsweise Verkaufs-Deals $\#V-D$) und Verhandlungsabbrüchen (beziehungsweise *Verkaufs-Rejects* $\#V-R$) zu erwarten war. Einen Ausschnitt der sich ergebenden Szenario-Definitionen gibt Tabelle 17 wieder.

N-NoLearning-2-2-2-2-5-1	...	N-NoLearning-8-5-5-5-2-1	N-NoLearning-8-8-2-8-2-1
N-NoLearning-2-2-2-2-8-1	N-NoLearning-5-2-5-5-5-1	N-NoLearning-8-5-5-5-5-1	N-NoLearning-8-8-2-8-5-1
N-NoLearning-2-2-2-5-2-1	N-NoLearning-5-2-5-5-8-1	N-NoLearning-8-5-5-8-2-1	N-NoLearning-8-8-5-2-5-1
N-NoLearning-2-2-2-5-5-1	N-NoLearning-5-2-5-8-8-1	N-NoLearning-8-5-8-2-8-1	N-NoLearning-8-8-5-2-8-1
N-NoLearning-2-2-2-5-8-1	N-NoLearning-5-2-8-2-2-1	N-NoLearning-8-8-2-2-5-1	N-NoLearning-8-8-5-5-8-1
N-NoLearning-2-2-2-8-2-1	N-NoLearning-5-2-8-2-5-1	N-NoLearning-8-8-2-2-8-1	N-NoLearning-8-8-5-8-2-1
N-NoLearning-2-2-2-8-5-1	N-NoLearning-5-2-8-2-8-1	N-NoLearning-8-8-2-5-2-1	N-NoLearning-8-8-5-8-8-1
N-NoLearning-2-2-5-2-5-1	...	N-NoLearning-8-8-2-5-8-1	N-NoLearning-8-8-8-8-8-1

Tabelle 17: Ausschnitt der Szenario-Definitionen mit einer breiten Verteilung von Gen-Startwerten

11.1.3 Testreihe zur Relation von Pool- und Populations-Größe

Innerhalb dieser Testreihe wurden die in Tabelle 18 angegebenen dreißig Kombinationen von Pool-Größe (P) und Populations-Größe (P_n) untersucht.

Mit den entsprechenden P:P_n-Kombinationen wurden die in Tabelle 19 genannten 27 Basis-Parametrisierungen zu Simulationsszenarien vervollständigt. Hieraus ergaben sich 810 Szenarien, die simuliert und ausgewertet wurden.

⁷⁷ in der Tabelle bezeichnet als Smith98.

Die ersten zehn der in Tabelle 19 aufgelisteten Szenarien orientieren sich dabei in Bezug auf die Einstellungen der Gen-Werte zu Simulationsstart an den wesentlichen in [Eymann 2000] untersuchten Szenarien.

P	Pn																
	3	5	10	12	15	18	20	25	30	35	40	42	45	50	60	70	80
3	X	X	X	X	X	X	X	X			X						
5		X			X		X	X	X	X	X						
7										X	X	X	X	X	X		
10			X				X		X		X			X	X	X	X

Tabelle 18: Untersuchte Kombinationen von Pool- und Populations-Größe

Die weiteren siebzehn Szenarien weisen variierende Gen-Startwert-Kombinationen für die Strategie-Parameter auf. Die verwendeten Gen-Startwert-Variationen wurden anhand der Ergebnisse der in Abschnitt 11.1.2 beschriebenen Testreihen so selektiert, dass hinsichtlich ihrer Effekte auf die Umsätze und Gewinne des Gesamtsystems sowie weiterer Kennzahlen relativ gleichverteilte Ergebnisse zu erwarten waren.

Simulierte und ausgewertete Szenarien:
Eymann_1-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2Pool
Eymann_2-50A-Aq(0,5)-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2Pool
Eymann_2-50A-Aq(0,6)-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2Pool
Eymann_2-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2Pool
Eymann_3-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2Pool
Eymann_4_1-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2Pool
Eymann_5-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2Pool
Eymann_6-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2Pool
Eymann_7-50A-MX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2Pool
Eymann_9-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2-2-2-2-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2-5-5-5-5-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2-5-5-8-2-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2-5-5-8-5-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2-5-8-2-2-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2-8-5-8-5-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-2-8-8-8-8-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-5-2-2-8-2-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-5-2-5-5-5-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-5-2-5-5-8-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-5-2-8-5-8-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-5-5-2-8-2-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-5-5-5-5-5-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-5-8-2-5-2-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-5-8-5-8-5-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-8-2-2-5-5-10-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-8-8-8-8-8-10-50A-2Pool

Tabelle 19: Basis-Parametrisierungen zur Testreihe „Relation von Pool- und Populations-Größe“

Alle Szenarien der Testreihe weisen als Grundmuster die Parameter-Einstellungen

$$MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-AvP-50A-2Pool$$

auf. Dieses Grundmuster hatte sich in vorhergehenden Testreihen (s. [Ruß & Walz 2009a; Ruß & Walz 2009b; Ruß & Walz 2009c; Ruß & Walz 2009d; Ruß & Walz 2012]) als für die Gesamtperformanz von agentenbasierten Wertschöpfungsnetzwerken vorteilhaft erwiesen. Jedes der 810 Szenarien wurde jeweils viermal simuliert, um trotz des nichtdeterministischen Systemverhaltens Analyseergebnisse mit statistischer Aussagekraft zu erhalten.

11.1.4 Testreihe zu den Effekten der Fitness-Methoden

In dieser Testreihe wurden die siebenundzwanzig in Tabelle 20 aufgeführten Szenario-Grundmuster durch die Initialisierung ihrer Variable *[Fitness-Methode]* mit den Fitness-Methoden *PmA*, *RMF*, *BaP*, *MdP* und *AvP* sowie der Variable *[P-Pn]* mit den Einstellungen P5Pn30 und P5Pn35 vervollständigt und jeweils viermal simuliert. Anschließend wurden die Simulationsergebnisse statistisch ausgewertet.

Die Evaluationsergebnisse für variierende Relationen von Pool und Population (s. die in Abschnitt 11.1.3 beschriebene Testreihe und ihre Evaluationsergebnisse in Abschnitt 8.4.1) hatten ergeben, dass eine Relation von 1:6 oder 1:7 empfehlenswert ist. Daher wurden sowohl Tests für die Einstellung P5Pn30 als auch für P5Pn35 durchgeführt.

Simulierte und ausgewertete Szenarien:
Eymann_1-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2Pool
Eymann_2-50A-Aq(0,5)-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2Pool
Eymann_2-50A-Aq(0,6)-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2Pool
Eymann_2-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2Pool
Eymann_3-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2Pool
Eymann_4_1-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2Pool
Eymann_5-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2Pool
Eymann_6-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2Pool
Eymann_7-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2Pool
Eymann_9-50A-MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2-2-2-2-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2-5-5-5-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2-5-5-8-2-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2-5-5-8-5-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2-5-8-2-2-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2-8-5-8-5-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-2-8-8-8-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-5-2-2-8-2-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-5-2-5-5-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-5-2-5-5-8-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-5-2-8-5-8-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-5-5-2-8-2-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-5-5-5-5-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-5-8-2-5-2-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-5-8-5-8-5-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-8-2-2-5-5-1-50A-2Pool
MIX(G)-[P-Pn]-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(1)-[Fitness-Methode]-8-8-8-8-1-50A-2Pool

Tabelle 20: Szenario-Grundmuster der Testreihe zu den Effekten der Fitness-Methoden

Die Agenten haben in allen Szenarien der Testreihe homogen die in Tabelle 20 genannten Parametrisierungen verwendet.

11.1.5 Testreihe zu den Selektions-Methoden

11.1.5.1 Statische Initialisierung der Gen-Startwerte

Alle Szenarien dieser Testreihe beinhalten 2-Pool-separiertes Lernen, die Initialisierung von jeweils 50 Agenten und bauen auf dem Szenario-Grundmuster

MIX(G)-P5-Pn30-[Selektion]-nPC(0,5-1-0)-GMU(0,5-0,01)-[Ersetzung]-AvP-[Startwerte]

auf. Die Einstellung *P5-Pn30* ergab sich aus der Erkenntnis, dass für die Pool- und Populations-Größen Relationen von 1:6 oder 1:7 empfehlenswert sind (s. Abschnitt 8.4.1). Die Mutations-Einstellung *GMU(0,5-0,01)* wurde gewählt, da so wünschenswerte Mutations-Effekte zu erwarten waren, ohne den Einfluss der Mutation auf das Simulationsergebnis zu groß werden zu lassen.

Die übrigen Einstellungen der Basis-Parametrisierung ergaben sich, da aufgrund der in [Ruß & Walz 2009a; Ruß & Walz 2009b; Ruß & Walz 2009c; Ruß & Walz 2009d; Ruß & Walz 2012] erzielten Evaluationsergebnisse zu erwarten war, dass es sich dabei um Einstellungen handelt, die in Bezug auf die Gesamtperformanz des Multiagenten-basierten Wertschöpfungsnetzwerks zu guten Ergebnissen führen.

Die Variable *[Selektion]* des Szenario-Grundmusters wurde mit den acht Selektions-Einstellungen *BiC*, *DMS*, *DMS2*, *DMS3*, *DtA*, *DtS*, *DWS* und *RWS* initialisiert. Die daraus entstehenden Szenario-Parametrisierungen wurden dann in Kombination mit den vier Ersetzungsschema-Varianten *WE(0)*, *WE(1)*, *WE(2)* und *WE(3)* getestet.

Die Gen-Startwerte der Variablen *[Startwerte]* wurden randomisiert initialisiert, wie im Abschnitt 11.1.5.2 erläutert wird. Neben den Szenarien mit randomisierten Gen-Startwerten wurden für jede Kombination von *[Selektion]* und *[Ersetzung]* weitere neun Szenarien definiert und simuliert, bei denen die Gen-Startwerte mit den statischen Einstellungen *2-5-5-8-5-1*, *2-8-5-8-5-1*, *2-8-8-8-8-1*, *5-2-5-5-8-1*, *5-5-2-8-2-1*, *5-5-5-5-5-1*, *5-8-2-5-2-1*, *8-2-2-5-5-1* und den Einstellungen von *Eymann-Test1* initialisiert wurden.. Jedes dieser Szenarien wurde viermal simuliert.

Hinzu kamen einige Simulationsläufe, bei denen das Lernen ausgeschaltet war (*NoLearning*) und daher von den Agenten keine Selektions-Methode verwendet wurde.

Insgesamt wurden 128 Simulationsläufe mit einer statischen Initialisierung der Gen-Werte zum Simulationsstart durchgeführt.

11.1.5.2 Randomisierte Initialisierung der Gen-Startwerte

Jedes der auf Basis des Grundmusters

MIX(G)-P5-Pn30-[Selektion]-nPC(0,5-1-0)-GMU(0,5-0,01)-[Ersetzung]-AvP-[Startwerte]

bis auf die Einstellungen für die Variable *[Startwerte]* vollständig definierten Szenarien erhält randomisierte Gen-Startwert-Initialisierungen. Dies bedeutet, dass die Gen-Werte jedes Agenten zum Simulationsstart (unabhängig voneinander) randomisiert initialisiert werden. Die dabei in der Testreihe verwendeten Randomisierungseinstellungen sind

- *TOP*-Randomisierung (bzw. *(35w5)-3x(65w5)-(35w5)-(5w9)*)

Hierbei wird das AQ-Gen zufällig unter Verwendung der Java-Methode *nextGaussian* aus einer Gauß'schen Normalverteilung mit einer Breite von 0,5 um den Gen-

Wert 0,35, der den Mittelwert der Gauß'schen Normalverteilung darstellt, gezogen. In der Notation wird dieser Sachverhalt durch das Kürzel $(35w5)$ bezeichnet. Der Wert, mit dem das Gen AQ des Agenten initialisiert wird, liegt daher im Intervall $[0,1; 0,6]$.

Die Initialisierungs-Werte der Gene DC, DJ und SF liegen gemäß der Einstellung im Intervall $[0,4; 0,9]$. Und der Initialwert des Gens WM wird zufällig aus dem Intervall $[0,05; 0,95]$ ausgewählt.

- *MID*-Randomisierung (bzw. $All(5w9)$)
Bei dieser Randomisierungs-Einstellung liegen die Initialisierungs-Werte aller Gene eines Agenten im Intervall $[0,05; 0,95]$.
- *LOW*-Randomisierung (bzw. $(65w5)-3x(35w5)-(65w5)-(5w9)$)

Die drei verwendeten Randomisierungs-Einstellungen haben sich aus der Analyse der Testreihe zu den Effekten von Startwerten ergeben (s. die Abschnitte 11.1.2 und 8.3.3). Die detaillierte Evaluation der Simulationsergebnisse hatte gezeigt, dass die homogene Parametrisierung der Agenten-Gesellschaft mit Gen-Startwerten gemäß der *TOP*-Einstellung (bei sonst gleichen Einstellungen) zu einer überdurchschnittlichen Systemperformanz führen. Umgekehrt war aus den Ergebnissen ersichtlich, dass die Verwendung der Randomisierungs-Einstellung *LOW* in der Regel zu einer unterdurchschnittlichen Systemperformanz führt.

Eine Randomisierung der Gen-Initialwerte gemäß der Einstellung *MID* kann im Einzelfall in Bezug auf Performanz-Kriterien wie Gesamtumsatz, Gesamtgewinn oder die Anzahl von erzielten Verkaufsabschlüssen zu über- oder unterdurchschnittlichen System-Ergebnissen führen. In der Regel führt eine solche zufällige Belegung der Gen-Werte zum Simulationsstart aber zu durchschnittlichen System-Ergebnissen.

Da die drei benutzten Randomisierungs-Einstellungen jeweils zu einer zufälligen Initialisierung der Gen-Werte im Rahmen der spezifizierten Verteilungen bzw. Intervalle führen, wurden für jedes mit diesen Einstellungen versehene Szenario sechzehn Simulationsläufe durchgeführt.

In den folgenden Testreihen findet eine randomisierte Initialisierung der Gen-Startwerte häufiger Verwendung. Sofern die drei Randomisierungs-Einstellungen *TOP*, *MID* und *LOW* benutzt werden, wird dies daher im Folgenden durch das Kürzel bzw. den Namenszusatz [Rnd] angezeigt.

Auf der Grundlage der 32 getesteten Kombinationen von Selektions-Einstellungen (*BiC*, *DMS*, *DMS2*, *DMS3*, *DtA*, *DtS*, *DWS* und *RWS*) und Ersetzungsschema-Varianten (*WE(0)*, *WE(1)*, *WE(2)* und *WE(3)*) ergaben sich so 512 Simulationsläufe mit einer Gen-Startwert-Randomisierung.

Zusammen mit den 128 Simulationsläufen, die mit einer statischen Startwert-Einstellung vorgenommen wurden, bildeten somit knapp über 640 Simulationsläufe die Grundlage für die in Abschnitt 8.4.3 vorgestellten Evaluationsergebnisse.

11.1.6 Testreihe zu Rekombinations-Einstellungen

Die in der Testreihe simulierten Szenarien für die Rekombinations-Methode *nPC* und *RaC* (siehe Abschnitt 5.4.3.3) basieren auf den fünf Grundmustern

- *MIX(G)-P5-Pn30-BiC-[Rekombination]-GMU(0,5-0,01)-WE(2)-AvP-[Rnd]-50A-2Pool*
- *MIX(G)-P5-Pn30-DMS-[Rekombination]-GMU(0,5-0,01)-WE(3)-AvP-[Rnd]-50A-2Pool*

- *MIX(G)-P5-Pn30-DtS-[Rekombination]-GMU(0,5-0,01)-WE(2)-AvP-[Rnd]-50A-2Pool*
- *MIX(G)-P5-Pn30-DWS [Rekombination]-GMU(0,5-0,01)-WE(3)-AvP-[Rnd]-50A-2Pool*
- *MIX(G)-P5-Pn30-RWS-[Rekombination]-GMU(0,5-0,01)-WE(3)-AvP-[Rnd]-50A-2Pool*

Der Test der genannten Rekombinations-Einstellungen für *nPC* und *RaC* wurde in Kombination mit den fünf Selektions-Einstellungen *BiC*, *DMS*, *DtS*, *DWS* und *RWS* vorgenommen, da sich die genetischen Operatoren mit ihren Einstellungen gegenseitig beeinflussen und die von anderen Operatoren bewirkten Effekte sowohl hemmen als auch vergrößern können (vgl. [Weicker 2015, S. 116 f.]).

Aus den genannten fünf Grundmustern ergeben sich für die Rekombinations-Methode *nPC* fünfundvierzig Basis-Simulationsszenario-Definitionen, indem ihre Lern-Einstellungen mit den Rekombinations-Einstellungen *nPC(0,3-1-X)*, *nPC(0,3-2-X)*, *nPC(0,3-3-X)* und *nPC(0,5-1-X)*, *nPC(0,5-2-X)*, *nPC(0,5-3-X)* sowie *nPC(1-1-X)*, *nPC(1-2-X)* und *nPC(1-3-X)* kombiniert werden. Weitere zehn Basis-Simulationsszenario-Definitionen ergeben sich aus der Kombination der Grundmuster mit den Einstellungen *RaC(X-X-0,5)* und *RaC(X-X-0,8)* der *RaC*-Rekombinations-Methode.

Aufgrund der durch den Namenszusatz bzw. Grundmuster-Variablen [*Rnd*] angezeigten darauffolgenden Randomisierung der Gen-Startwerte anhand der Randomisierungseinstellungen *TOP*, *MID* und *LOW* ergeben sich 165 vollständig definierte Szenarien. Für diese Szenarien werden jeweils 16 Simulationslauf-Wiederholungen durchgeführt, um die Ergebnisse statistisch abzusichern. Insgesamt basieren die in Abschnitt 8.4.4 vorgestellten Evaluationsergebnisse daher auf einer statistisch gut abgesicherten Datenbasis.

Dies war sinnvoll, da bei jedem Simulationslauf ein Multiagenten-System initialisiert wird, dessen Gesamtverhalten nichtdeterministisch ist und sich so – ähnlich wie an der Börse – für die gehandelten Güter dynamisch Preiskurven bilden, die volatil sein können.

11.1.7 Testreihe zur Pool-Separierung

Diese Testreihe hatte den Zweck, zu untersuchen, wie sich die Wahl des 1-Pool- oder 2-Pool-Modus im Wechselspiel mit variierenden Selektions- und Rekombinations-Einstellungen auf die Gesamtperformanz des Wertschöpfungsnetzwerks auswirkt.

Hierzu wurden fast die gleichen fünf Szenario-Grundmuster wie im vorhergehenden Abschnitt verwendet, nämlich

- *MIX(G)-P5-Pn30-BiC-[Rekombination]-GMU(0,5-0,01)-WE(2)-AvP-[Rnd]-50A-[Pool]*
- *MIX(G)-P5-Pn30-DMS-[Rekombination]-GMU(0,5-0,01)-WE(3)-AvP-[Rnd]-50A-[Pool]*
- *MIX(G)-P5-Pn30-DtS-[Rekombination]-GMU(0,5-0,01)-WE(2)-AvP-[Rnd]-50A-[Pool]*
- *MIX(G)-P5-Pn30-DWS [Rekombination]-GMU(0,5-0,01)-WE(3)-AvP-[Rnd]-50A-[Pool]*
- *MIX(G)-P5-Pn30-RWS-[Rekombination]-GMU(0,5-0,01)-WE(3)-AvP-[Rnd]-50A-[Pool]*

Im Unterschied zu Abschnitt 11.1.6 war der Pool-Modus nicht auf *2-Pool* festgelegt, sondern die Variable [*Pool*] wurde mit den Pool-Modi *1-Pool* und *2-Pool* initialisiert. Als weiterer Unterschied waren die Initialisierungen für die Variable [*Rekombination*] auf die Einstellungen *nPC(0,5-1-X)* und *nPC(1-1-X)* beschränkt. Das sonstige Vorgehen bei dieser Testreihe verlief analog zu den Ausführungen in Abschnitt 11.1.6.

11.1.8 Testreihe zu Ersetzungsschema-Einstellungen

Die Auswirkungen der Nutzung verschiedener Ersetzungsschemata (siehe Abschnitt 5.4.4) wurden auf der Basis des Szenario-Grundmusters

MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-[Ersetzung]-AvP-[Rnd]-50A-2Pool

untersucht. Dieses Grundmuster wurde durch die Initialisierung seiner Variablen *[Ersetzung]* mit den zehn Ersetzungsschema-Einstellungen $E(0)$, $E(1)$, $E(2)$, $E(3)$, $E(4)$ sowie $WE(0)$, $WE(1)$, $WE(2)$, $WE(3)$, $WE(4)$ weiter vervollständigt, was zu den zehn untersuchten Basis-Szenarien führt. *[Rnd]* steht als Variable für eine Randomisierung der Startwerte mit den Randomisierungseinstellungen *TOP*, *MID* und *LOW* (vgl. Abschnitt 11.1.5.2).

Durch die Randomisierung ergeben sich 30 vollständige Simulationsszenario-Definitionen, von denen jede sechzehnmal simuliert wurde, um die Evaluationsergebnisse (siehe Abschnitt 8.4.6) statistisch gegen „Ausreißer“ abzusichern.

11.1.9 Testreihe zu Mutations-Methoden

Die Auswirkungen der Nutzung verschiedener Mutations-Methoden wurden auf der Basis des Szenario-Grundmusters

MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-[Mutation]-WE(3)-AvP-[Rnd]-50A-2Pool

untersucht. Dieses Grundmuster wurde durch die Initialisierung seiner Variablen *[Mutation]* mit den achtunddreißig in Tabelle 21 angegebenen Mutations-Parameter-Einstellungen der Mutations-Methoden *General Mutation (GMU)* und *Single Mutation (SMU)* (siehe 5.4.3.4) weiter vervollständigt, was zu gleichfalls achtunddreißig Mutations-Basis-Szenarien führt, die simulativ untersucht wurden. Zusätzlich wurde ein weiteres Szenario definiert, das dem Szenario-Grundmuster entspricht, bei dem aber die Mutation ausgeschaltet war.

[Rnd] steht als Variable für eine Randomisierung der Startwerte mit den Randomisierungseinstellungen *TOP*, *MID* und *LOW* (vgl. Abschnitt 11.1.5.2).

Durch die Randomisierung ergeben sich aus den 39 Basis-Szenarien 117 vollständige Simulationsszenario-Definitionen, von denen jede sechzehnmal simuliert wurde, um die Analyseergebnisse (s. Abschnitt 8.4.7) statistisch gegen „Ausreißer“ abzusichern.

GMU(0,5-0,01)	GMU(1,0-0,01)	SMU(0,5-0,01)	SMU(1,0-0,01)
GMU(0,5-0,02)	GMU(1,0-0,02)	SMU(0,5-0,03)	SMU(1,0-0,03)
GMU(0,5-0,03)	GMU(1,0-0,03)	SMU(0,5-0,05)	SMU(1,0-0,05)
GMU(0,5-0,05)	GMU(1,0-0,05)	SMU(0,5-0,07)	SMU(1,0-0,07)
GMU(0,5-0,07)	GMU(1,0-0,07)	SMU(0,5-0,10)	SMU(1,0-0,10)
GMU(0,5-0,09)	GMU(1,0-0,09)	SMU(0,5-0,30)	SMU(1,0-0,30)
GMU(0,5-0,10)	GMU(1,0-0,10)	SMU(0,5-0,50)	SMU(1,0-0,50)
GMU(0,5-0,11)	GMU(1,0-0,11)		
GMU(0,5-0,13)	GMU(1,0-0,13)		
GMU(0,5-0,15)	GMU(1,0-0,15)		
GMU(0,5-0,30)	GMU(1,0-0,30)		
GMU(0,5-0,50)	GMU(1,0-0,50)		

Tabelle 21: Untersuchte Mutations-Parameter-Einstellungen

Der Informationsaustausch zwischen den Agenten einer Wertschöpfungsstufe (beziehungsweise zwischen den Agenten des gleichen Typs) war bei der Durchführung der Testreihe so

eingestellt, dass alle Agenten Informationen über einen erfolgreichen Verkaufsabschluss immer mit allen anderen Agenten der gleichen Stufe geteilt haben (Einstellung $100P100F^{78}$).

11.1.10 Testreihe zu Informationsaustausch und Mutation

Bei dieser Testreihe wurde untersucht, wie unterschiedliche Informationsaustausch-Einstellungen die Auswirkungen verschiedener Mutations-Methoden auf die Performanz des Gesamtsystems beeinflussen.

Dazu wurde das Szenario-Grundmuster

MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-[Mutation]-WE(3)-AvP-[Rnd]-50A-2Pool

durch die Initialisierung seiner Variablen *[Mutation]* mit den Mutations-Parameter-Einstellungen *GMU(0,5-0,01)*, *GMU(0,5-0,1)*, *GMU(1,0-0,1)*, *SMU(0,5-0,1)* und *SMU(1,0-0,1)* zu fünf Mutations-Basis-Szenarien vervollständigt, die wie in Abschnitt 11.1.9 randomisiert wurden. Dadurch ergaben sich 15 Mutations-Szenarien, die mit den Informationsaustausch-Einstellungen *20P20F*, *20P50F*, *20P100F*, *40P100F*, *50P50F*, *50P75F*, *50P100F* und *100P100P* mit jeweils 16 Wiederholungen simuliert wurden.

Bei der Notation für die Informationsaustausch-Einstellungen gibt die Zahl vor dem *P* an, mit wieviel Prozent der Agenten in der eigenen Agentengruppe die Informationen über einen Verkaufsabschluss geteilt werden und die Zahl vor dem *F* gibt an, in wieviel Prozent der Fälle dies geschieht.

11.1.11 Testreihe zum Informationsaustausch bei heterogenem Lernen

Diese Testreihe hatte den Zweck, zu beobachten, wie sich eine in Bezug auf die Informationsaustausch- und Lern-Einstellungen heterogene Parametrisierung der Agenten-Typen auf deren Performanz auswirkt.

Hierzu wurden die Prozessor-Produzenten-Agenten auf der dritten beziehungsweise mittleren Stufe des Wertschöpfungsnetzwerks mit Lernfähigkeiten ausgestattet, während bei allen übrigen Agenten-Typen die Lern-Komponente abgeschaltet wurde.

Bei der Testreihe wurde das Szenario-Grundmuster

S3-MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,01)-WE(3)-[Fitness]-[Rnd]-50A-2Pool-[ProfitWish]-G-NoLearning-[Rnd]

genutzt. Dessen Variablen *[Fitness]* und *[ProfitWish]* wurden für die Prozessor-Produzenten variiert. Dabei wurde die Variable *[Fitness]* mit den Fitness-Methoden *AvP* und *PmA* in Kombination mit *[ProfitWish]*-Einstellungen von 0% und 10% initialisiert. Entsprechende Einstellungen wurden für die übrigen Agenten nicht vorgenommen.

Die sich ergebenden Basis-Szenario-Definitionen wurden anschließend wie in 11.1.5.2 beschrieben randomisiert. Die zwölf resultierenden, vollständig definierten Szenarien wurden dann mit den Informationsaustausch-Einstellungen *0P0F*, *10P10F*, *10P20F*, *10P50F*, *20P20F*, *20P50F*, *40P100F* und *100P100P* mit jeweils 16 Wiederholungen simuliert.

Zu betonen ist hierbei, dass ausgetauschte Informationen nur von den Prozessor-Produzenten ordentlich genutzt werden können, da bei den übrigen Agenten die Lernfähigkeiten abgeschaltet sind.

⁷⁸ *100P100F* bedeutet, dass die Agenten in 100 Prozent der Fälle Informationen mit 100 Prozent der Agenten auf der gleichen Wertschöpfungsstufe teilen.

Von Interesse war dabei, zu beobachten, wie sich die heterogen verteilten Fähigkeiten der Agenten auf die Performanz-Kennzahlen der Prozessor-Produzenten-Agenten auswirken.

11.1.12 Testreihe zur Selbstkoordination mit MACSIMA-Erweiterungen

Zweck dieser Testreihe war, die Gesamtperformanz des agentenbasierten Wertschöpfungsnetzwerks bei verschiedenen Parameter-Einstellungen zu vergleichen. Dabei sollten insbesondere die Auswirkungen des Ein- und Ausschaltens verschiedener MACSIMA-Erweiterungen untersucht werden.

Tabelle 22 bietet eine Übersicht aller definierten und getesteten Szenarien und ihrer detaillierten Parameter-Einstellungen.

In den Szenarien, deren Namen mit *Eymann_* beginnen, werden die Agenten zum Simulationsstart homogen mit Gen-Werten initialisiert, die den Erläuterungen in Abschnitt 11.1.1 zu entnehmen sind. Dabei handelt es sich um Gen-Startwerte, mit denen die Agenten analog zu den in [Eymann 2000] vorgenommenen Experimenten parametrisiert wurden.

Die Simulationsszenarien der Testreihe bauen auf dem Szenario-Grundmuster

[Lern-Modus]-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-[Startwerte]-50A-2Pool-[HFF]

auf. Die Grundmuster-Variable *[Lern-Modus]* wurde entweder mit *INT(G)* oder *MIX(G)* initialisiert (*internes Lernen* oder *gemischtes Lernen*, siehe Abschnitt 6.3).

Szenario-Name	Parameter-Einstellungen
Eymann_1	Eymann_1-NoLearning-50Agenten
Eymann_2	Eymann_2-NoLearning-50Agenten
INT(G)-Eymann_1	INT(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann_1-50A-2Pool
INT(G)-Eymann_1-HFF	INT(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann_1-50A-2Pool-HFF
INT(G)-Eymann_2	INT(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann_2-50A-2Pool
INT(G)-Eymann_2-HFF	INT(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann_2-50A-2Pool-HFF
INT(G)-TopRnd	INT(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-(35w5)-3x(65w5)-(35w5)-(5w9)-50A-2Pool
INT(G)-TopRnd-HFF	INT(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-(35w5)-3x(65w5)-(35w5)-(5w9)-50A-2Pool-HFF
MIX(G)-Eymann_1	MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann_1-50A-2Pool
MIX(G)-Eymann_1-HFF	MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann_1-50A-2Pool-HFF
MIX(G)-Eymann_2	MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann_2-50A-2Pool
MIX(G)-Eymann_2-HFF	MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann_2-50A-2Pool-HFF
MIX(G)-TopRnd	MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-(35w5)-3x(65w5)-(35w5)-(5w9)-50A-2Pool
MIX(G)-TopRnd-HFF	MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-(35w5)-3x(65w5)-(35w5)-(5w9)-50A-2Pool-HFF

Tabelle 22: Innerhalb der Testreihe „Performanzvergleich“ untersuchte Szenarien

Die in der Tabelle 22 genannten Szenarien der Testreihe wurden mit jeweils 16 Wiederholungen simuliert. Die Variable *[Startwerte]* wurde entweder mit den Einstellungen *Eymann_1* und *Eymann_2* oder mit der in Abschnitt 11.1.5.2 beschriebenen TOP-Randomisierung belegt. Die Einstellung *[HFF]* wurde ein- oder ausgeschaltet. Sie wird in den Parameter-Einstellungen nur genannt, wenn sie eingeschaltet ist.

11.2 Evaluations-Diagramme

Im Folgenden finden sich einige Abbildungen, die bei der Evaluation der im vorhergehenden Abschnitt beschriebenen Testreihen entstanden sind, aber in Kapitel 8 nicht verwendet wurden. Abbildung 132 bis Abbildung 135 ergänzen die Abbildungen in Abschnitt 8.4.7, in dem die Effekte der Mutationsmethoden untersucht wurden. Abbildung 136 bietet eine Gesamt-sicht auf die kombinierten Effekte von verschiedenen Mutations- und Informationsaus-tausch-Einstellungen, die in Abschnitt 8.5.1 untersucht wurden. Bei ihr handelt es sich um eine Vergrößerung von Abbildung 123.

Bei Abbildung 132 handelt es sich um eine nach dem durchschnittlichen Gesamtumsatz sor-tierte Variante von Abbildung 124. Die abgebildeten Werte wurden über die für alle Infor-mationsaustausch-Einstellungen generierten Simulationsergebnisse aggregiert. Es zeigt sich, dass die Parametrisierung der Mutation mit der Einstellung GMU(0,5-0,1) eine gute Wahl darstellt.

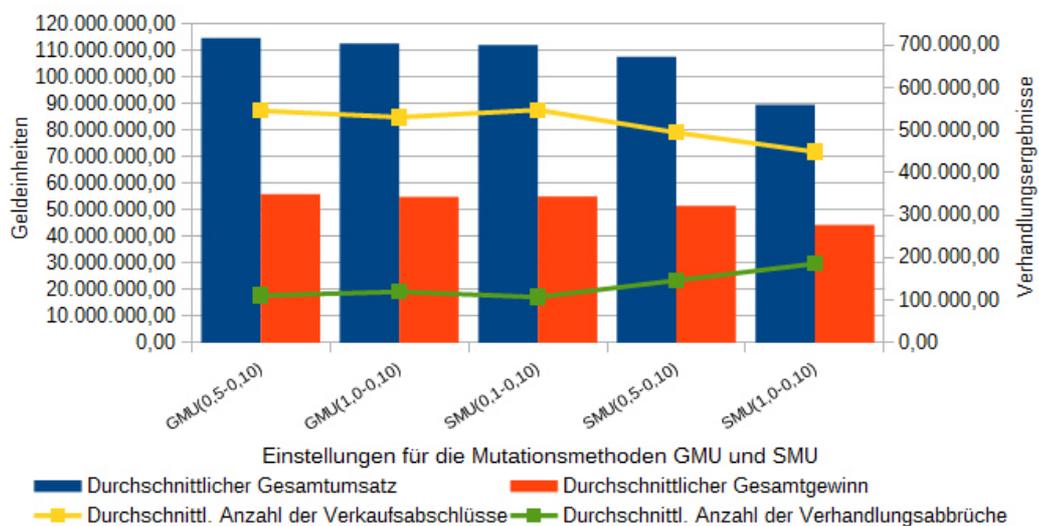


Abbildung 132: Gesamtumsatz, Gesamtgewinn und Verhandlungsergebnisse für GMU und SMU bei Variation von Mutations-Rate u. –Weite

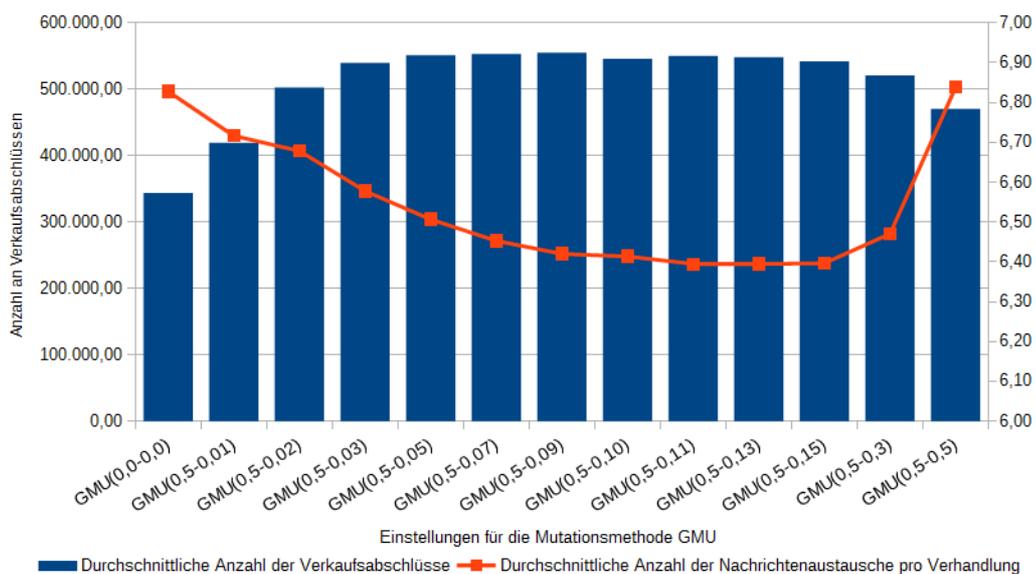


Abbildung 133: Anzahl an Verhandlungsabschlüssen und pro Verhandlung ausgetauschten Nachrichten für GMU(0,5-X) bei Variation der Mutations-Weite

Abbildung 133 enthält einige Ergebnisse für Einstellungen der Mutations-Weite, die aus Gründen der Übersichtlichkeit in Abbildung 112 nicht dargestellt wurden. Darüber hinaus fokussiert sich Abbildung 133 auf die Darstellung von Ergebnissen für die Einstellung GMU(0,5-X).

Abbildung 134 enthält im Vergleich zu Abbildung 117 aus Abschnitt 8.4.7 zusätzlich Ergebnisse für die Mutationsweite 0,05. Ansonsten stellt sie weniger Ergebniswerte dar und bietet daher eine gefilterte Übersicht der Ergebnisse.

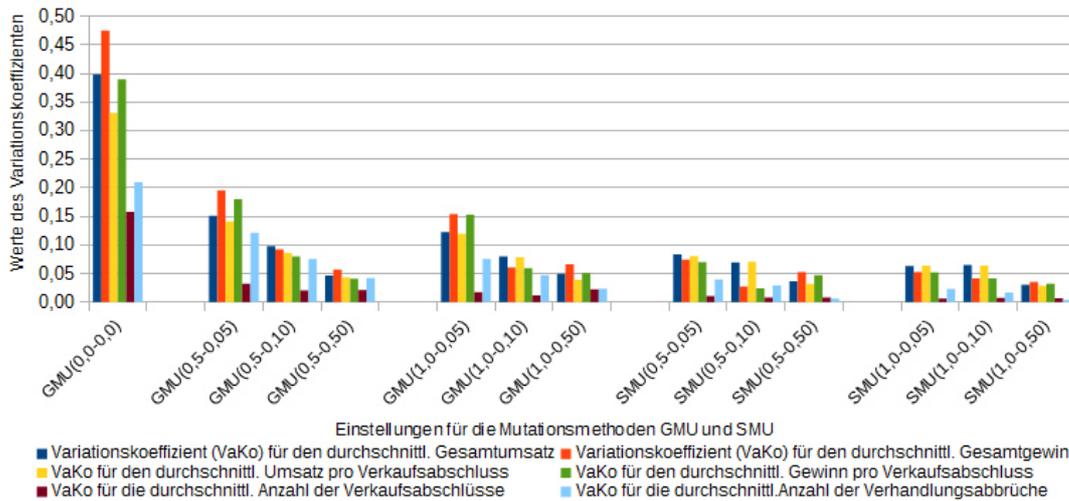


Abbildung 134: Variationskoeffizienten für die Mutations-Methoden GMU und SMU bei Variation von Mutations-Rate und –Weite

Abbildung 135 stellt einen Ausschnitt aus Abbildung 118 vergrößert dar, da die genauen Ergebniswerte dieses Ausschnitts aus Abbildung 118 aufgrund ihrer Skalierung nicht abgelesen werden können.

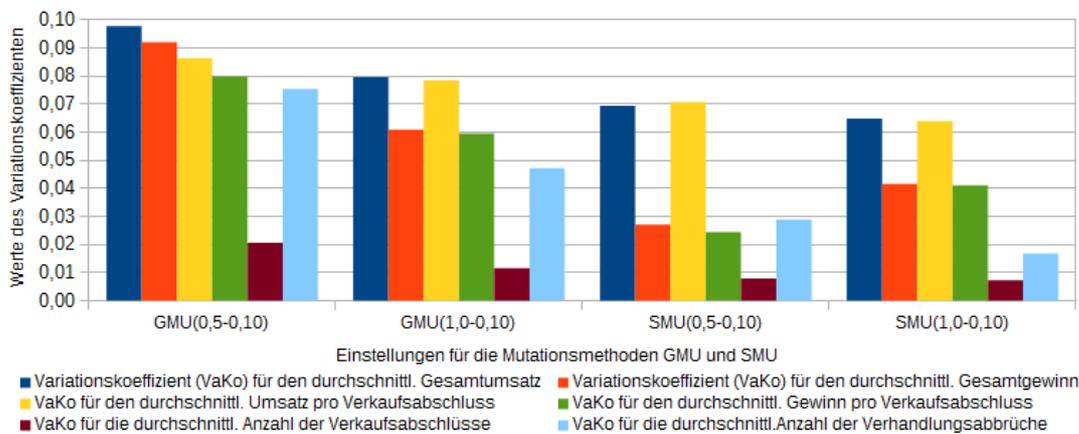


Abbildung 135: Variationskoeffizienten für die Mutations-Methoden GMU und SMU bei ausgewählten Parameter-Einstellungen

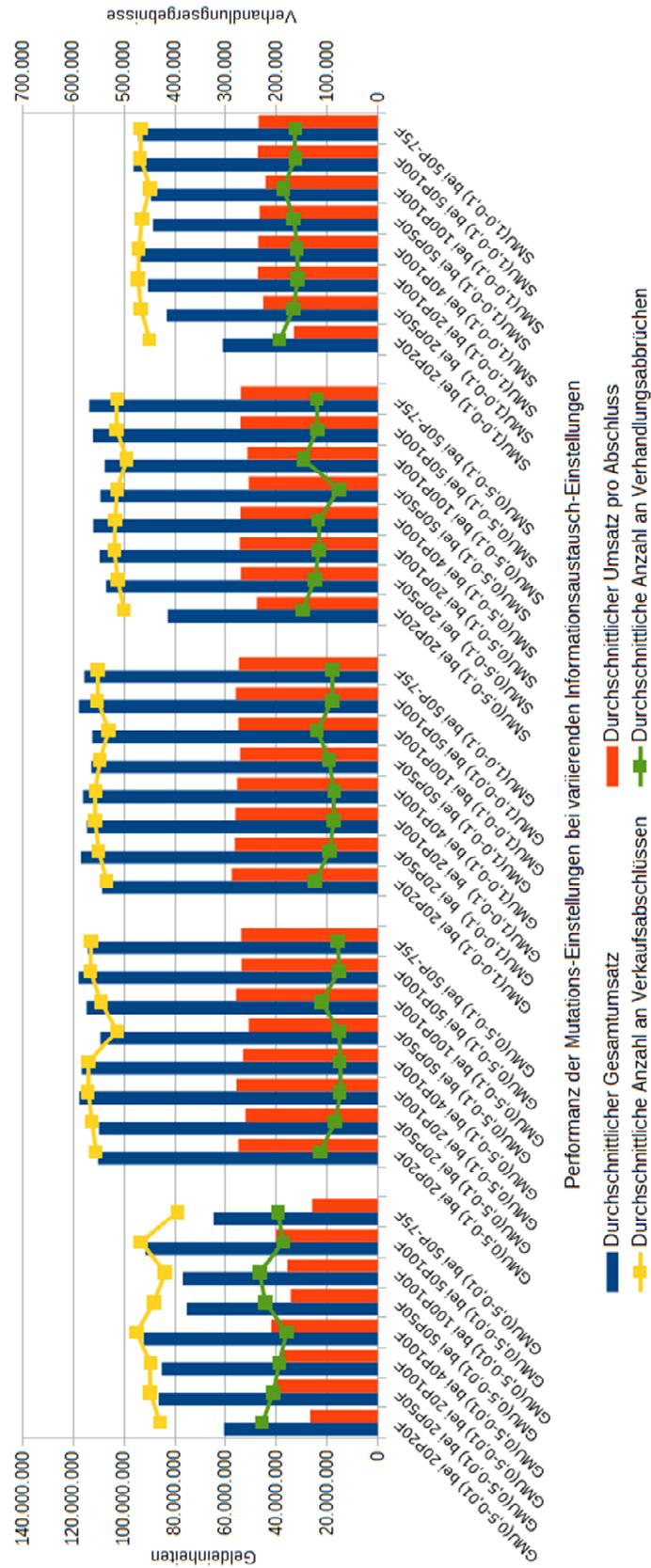


Abbildung 136: Gesamtübersicht kombinierter Effekte von variierenden Informationsaus-tausch- und Mutations-Einstellungen im Detail

11.3 Makro-Ergebnistabellen

11.3.1 Makro-Tabellen

Szenario: MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann-Test1-50A-2Pool-HFF

Legende:			
Li	Lauf i	SP	Startpreis
GU	Gesamt-Umsatz	EP	Endpreis
GG	Gesamt-Gewinn	HP	Höchstpreis
U/D	Umsatz/ Deal	TP	Tiefstpreis
G/D	Gewinn/ Deal	ME	Median
#D	Anzahl der Deals	OQ	Oberes Quartil
#R	Anzahl der Rejects	UQ	Unteres Quartil
#Ru	Anzahl der Runden	KV	Konvergiert
MW	Mittelwert	MWP	Mittelwert des Preises pro Gut
SA	Standardabweichung	MAM	Mittlere Abweichung vom Median
VK	Variationskoeffizient		
G ²	Varianz		

Gut	Silicium				Chip				Processor				Computer			
	Lauf 1	Lauf 2	Lauf 3	Lauf 4	Lauf 1	Lauf 2	Lauf 3	Lauf 4	Lauf 1	Lauf 2	Lauf 3	Lauf 4	Lauf 1	Lauf 2	Lauf 3	Lauf 4
Verkaufs-Deals	1391147	136657	136994	137810	139143	136954	139390	137803	139136	139866	139397	137797	139121	136883	136833	137792
Mittelwert des Preises	57.8896	71.8895	24.4732	50.8382	99.6989	148.058	187.185	189.069	224.466	302.701	284.148	267.134	418.824	441.527	444.639	361.121
Standardabweichung	24.7129	26.9471	20.7115	31.8122	33.2379	38.7391	38.9083	38.9444	49.3266	58.299	67.0116	91.6524	74.4026	77.0482	110.402	
Variationskoeffizient	0.4269	0.3748	0.3038	0.6358	0.3334	0.3731	0.297	0.2958	0.4386	0.163	0.2509	0.2188	0.1685	0.1733	0.2897	
Varianz	610.731	726.146	428.958	1012.01	1104.76	3090.92	1500.72	1513.85	9891.22	2433.02	3391.78	4490.55	8401.07	5535.74	5936.43	12188.7
Startpreis	50	50	51.875	50	100	100	98.125	101.875	152.812	152.812	150	152.812	195.25	200	203.75	192.5
Höchstpreis	34.8572	16.9336	16.3423	63.485	66.8611	75.6269	220.075	211.68	293.238	171.788	236.329	346.526	475.221	486.419	487.51	486.804
Tiefstpreis	120.937	125	112.873	124.002	230.497	250	250	375	375	375	375	375	500	500	500	500
Median	52.9021	75.0876	27.7676	46.1762	93.2782	140.491	200.575	203.627	265.435	318.282	308.201	288.299	464.055	474.832	474.725	441.053
Oberes Quartil	75.2483	94.4263	44.7351	73.1863	108.94	199.61	216.552	218.716	319.337	336.482	328.411	325.487	482.288	483.336	482.958	461.248
Unteres Quartil	39.6854	48.9361	17.8155	21.4002	80.6321	104.364	169.536	164.527	114.427	281.988	243.705	214.569	382.736	446.946	452.59	290.572
Verkaufs-Deals	553752				553730				553706							553689
Mittelwert des Preises	53.8826				155.78				269.601							421.559
Standardabweichung	29.6098				55.8639				76.4751							93.0146
Variationskoeffizient	0.5494				0.3586				0.2837							0.2206
Varianz	876.212				3120.78				5948.45							8651.72
Startpreis	50				100				152.812							196.25
Höchstpreis	63.485				211.68				346.526							486.804
Tiefstpreis	125				250				375							500
Median	11.0746				50.3794				84.1681							166.76
Oberes Quartil	49.6114				165.579				301.564							468.692
Unteres Quartil	77.8332				209.985				325.489							482.654
Herfindahl-Index	27.5952				102.725				222.802							402.001

Gut	Silicium				Chip				Processor				Computer			
	Lauf 1	Lauf 2	Lauf 3	Lauf 4	Lauf 1	Lauf 2	Lauf 3	Lauf 4	Lauf 1	Lauf 2	Lauf 3	Lauf 4	Lauf 1	Lauf 2	Lauf 3	Lauf 4
GesU	111430000	58271400	61762200	241.04	110.376	559560	78996	4405320	2.7978	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108
GesG	122384000	52515500	61762200	241.04	111.154	559560	89366	4415380	2.7578	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108
UvD	200.21	104.699	556557	74334	556557	74334	4397190	2.7978	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108	29.9085
GvD	200.21	104.699	556557	74334	556557	74334	4397190	2.7978	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108	29.9085
#V/D	200.21	104.699	556557	74334	556557	74334	4397190	2.7978	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108	29.9085
#R	200.21	104.699	556557	74334	556557	74334	4397190	2.7978	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108	29.9085
#Proposals	200.21	104.699	556557	74334	556557	74334	4397190	2.7978	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108	29.9085
#V/D #R	200.21	104.699	556557	74334	556557	74334	4397190	2.7978	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108	29.9085
Umsatzanteil Gruppe in %	0.0711	0.0618	0.0716	0.0618	0.0716	0.0618	0.0716	0.0618	0.0716	0.0618	0.0716	0.0618	0.0716	0.0618	0.0716	0.0618
Gewinnanteil Gruppe in %	0.0711	0.0618	0.0716	0.0618	0.0716	0.0618	0.0716	0.0618	0.0716	0.0618	0.0716	0.0618	0.0716	0.0618	0.0716	0.0618
Startwert	111430000	58271400	61762200	241.04	110.376	559560	78996	4405320	2.7978	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108
Endwert	122384000	52515500	61762200	241.04	111.154	559560	89366	4415380	2.7578	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108
Minimum	111430000	52515500	61762200	241.04	111.154	559560	89366	4415380	2.7578	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108
Maximum	134875000	61762200	61762200	241.04	111.154	559560	89366	4415380	2.7978	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108
Median	122384000	58271400	61762200	241.04	110.699	556557	78996	4405320	2.756	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108
Oberes Quartil	130099000	60863500	61762200	237.6	110.376	556557	85124	4410420	2.7828	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108
Unteres Quartil	111430000	52515500	61762200	200.21	95.2745	547558	74334	4397190	2.7378	3.6275	19.6934	29.897	46.7821	7.7539	34.3421	21.8108
Herfindahl-Index	0.2513	0.251	0.2513	0.251	0.2513	0.251	0.2513	0.251	0.2513	0.251	0.2513	0.251	0.2513	0.251	0.2513	0.251

Tabelle 23: Makro-Preis-Tabelle (links) und Makro-Umsatz-Tabelle (rechts)

11.3.2 Medio-Tabelle

Szenario: MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann-Test1-50A-2Pool-HFF

Gr	StU	Gr-LAn 3U		Gr-GAt 5h		Gr-GAn 6Gr		# DGr		# RGr		# ARQGr		# EDGr		# E-RGr		# E-ABQGr		# V-RGr		# VARQGr		# QV-DGr		# UV-DGr		# Abt GU		# Abt GGr		SI			
		Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2				
	Gr SdS	895517	1072118	100,0%	100,0%	100,0%	100,0%	0,1392	0,1638	180,48	185892	17245	20131	0,8403	0,8284	0	0	0	0	0	130,148	139592	2041	20131	0,8403	0,8284	34,4722	34,4722	56,8826	57,8826	855171	1057118	100,0%	100,0%	
	Gr Dsch	13872503	20724200	100,0%	100,0%	100,0%	100,0%	0,0775	0,1334	136896	17913	26252	28548	0,8748	0,8811	139856	-7041	20137	0,8509	0,8742	138144	139894	27794	17225	0,8593	0,8688	41,8065	41,8065	13872503	20724200	100,0%	100,0%	100,0%	100,0%	
	Gr DHP	2521103	25242300	100,0%	100,0%	100,0%	100,0%	0,3476	0,8277	272729	278513	42535	36258	0,8573	0,9411	137816	70251	29258	0,8403	0,8284	138144	139894	27794	17225	0,8593	0,8688	41,8065	41,8065	2521103	25242300	100,0%	100,0%	100,0%	100,0%	
	Gr Dico	38271400	51762200	100,0%	100,0%	100,0%	100,0%	0,4854	0,3144	73821	27977	34563	41903	0,8704	0,9063	136896	137800	11888	0,8593	0,8742	138144	139894	27794	17225	0,8593	0,8688	41,8065	41,8065	38271400	51762200	100,0%	100,0%	100,0%	100,0%	
	Gr EFCO	6985200	52515500	100,0%	100,0%	100,0%	100,0%	0,3028	0,7991	273771	277550	42781	33068	0,8521	0,9279	136896	17798	18395	0,8593	0,8742	138144	139894	27794	17225	0,8593	0,8688	41,8065	41,8065	6985200	52515500	100,0%	100,0%	100,0%	100,0%	
	Lauf 1-4	SA	MW	SW	SA	MW	SW	SA	MW	SW	SA	MW	SW	SA	MW	SW	SA	MW	SW	SA	MW	SW	SA	MW	SW	SA	MW	SW	SA	MW	SW	SA	MW	SW	
	Gr SdS	7428410	10246200	1	1	1	1	0,1128	0,0312	138438	1162,38	22633,4	4570,31	0,8584	0,9353	136896	14673	22953	0,8584	0,9353	136896	14673	22953	14673	0,8584	0,9353	136896	14673	22953	14673	22953	14673	22953	14673	22953
	Gr Dsch	0,7158	0,258	0	0	0	0	0,6654	0,2437	0,0217	0,0084	0,9014	0,1192	0,0728	0,0253	0,6958	0,2506	0,2506	0,0728	0,0253	0,6958	0,2506	0,2506	0,0728	0,0253	0,6958	0,2506	0,2506	0,0728	0,0253	0,6958	0,2506	0,2506	0,0728	0,0253
	Gr DHP	21953103	49118900	1	1	1	1	0,2448	0,1117	77697	2295,88	43724,5	5639,71	0,8638	0,9158	138438	1162,38	22633,4	0,8638	0,9158	138438	1162,38	22633,4	1162,38	0,8638	0,9158	138438	1162,38	22633,4	1162,38	22633,4	1162,38	22633,4	1162,38	22633,4
	Gr Dico	3219900	4027700	1	1	1	1	0,2678	0,6995	276866	2325,15	41102,3	1028,44	0,8703	0,9353	138438	1162,38	22633,4	0,8703	0,9353	138438	1162,38	22633,4	1162,38	0,8703	0,9353	138438	1162,38	22633,4	1162,38	22633,4	1162,38	22633,4	1162,38	22633,4
	Gr EFCO	88233103	3655600	1	1	1	1	0,4597	0,9644	276895	2294,35	38220,3	4534,31	0,8798	0,9129	138438	1162,38	22633,4	0,8798	0,9129	138438	1162,38	22633,4	1162,38	0,8798	0,9129	138438	1162,38	22633,4	1162,38	22633,4	1162,38	22633,4	1162,38	22633,4
	Gr EFCO	0,1588	0,0618	0	0	0	0	0,4597	0,1792	390427	1102,48	13013,3	3244,67	0,8653	0,9196	138438	1162,38	22633,4	0,8653	0,9196	138438	1162,38	22633,4	1162,38	0,8653	0,9196	138438	1162,38	22633,4	1162,38	22633,4	1162,38	22633,4	1162,38	22633,4

Tabelle 24: Medio-Tabelle – Überblick

Szenario: MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann-Test1-50A-2Pool-HFF

Gr	St-U		Gr-UA n StU		StG		Gr-GAn St		GrGAn GesG		# D/Gr	
	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2
	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4
Gr SuSi	8055170	10057100	100.00%	100.00%	8055170	10057100	100.00%	100.00%	0.1382	0.1628	139148	139899
	4719320	7006010	100.00%	100.00%	4719320	7006010	100.00%	100.00%	0.0775	0.1334	136898	137810
Gr DeCh	13872500	20712400	100.00%	100.00%	5817380	10655300	100.00%	100.00%	0.0998	0.1725	278292	279793
	25621100	26054300	100.00%	100.00%	20901700	19048300	100.00%	100.00%	0.3434	0.3627	273789	275613
Gr DePr	31231300	42343600	100.00%	100.00%	17358800	21631100	100.00%	100.00%	0.2979	0.3502	278282	279781
	38895900	36808900	100.00%	100.00%	13274800	10754500	100.00%	100.00%	0.2181	0.2048	273779	275601
Gr DeCo	58271400	61762200	100.00%	100.00%	27040100	19418500	100.00%	100.00%	0.464	0.3144	278271	279770
	60863500	52515500	100.00%	100.00%	21967500	15706600	100.00%	100.00%	0.3609	0.2991	273771	275590
Gr BrCo	0	0	NaN	NaN	-58271400	-61762200	1	1	-100.00%	-100.00%	139133	139883
	0	0	NaN	NaN	-60863500	-52515500	1	1	-100.00%	-100.00%	136883	137792
Lauf 1-4												
	MW	SA	MW	SA	MW	SA	MW	SA	MW	SA	MW	SA
	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK
Gr SuSi	7459410	1924620	1	0	7459410	1924620	1	0	0.128	0.0312	138438	1162.39
	0.7156	0.258	0	0	0.7156	0.258	0	0	0.6664	0.2437	0.0217	0.0084
Gr DeCh	21565100	4911890	1	0	14105700	6148520	1	0	0.2446	0.1117	276871	2325.86
	0.5649	0.2278	0	0	1.0694	0.4359	0	0	1.0747	0.4564	0.0217	0.0084
Gr DePr	37319900	4032770	1	0	15754800	4130900	1	0	0.2678	0.0595	276860	2325.5
	0.2978	0.1081	0	0	0.6904	0.2622	0	0	0.5432	0.2221	0.0217	0.0084
Gr DeCo	58353100	3605820	1	0	21033200	4121140	1	0	0.3596	0.0644	276850	2324.51
	0.1585	0.0618	0	0	0.5388	0.1959	0	0	0.4587	0.1792	0.0217	0.0084
Gr BrCo	0	0	nan	nan	-58353100	3605820	1	0	-1	0	138422	1162.48
	NaN	NaN	nan	nan	-0.1585	-0.0618	0	0	0	0	0.0217	0.0084
Gr	# R/Gr	AbQ/Gr	# E-D/Gr	# E-R/Gr	E-AbQ/Gr	# V-D/Gr	# V-R/Gr					
	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2
	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4
Gr SuSi	17047	20137	0.8909	0.8742	0	0	0	0	NaN	NaN	139148	139899
	26025	28548	0.8403	0.8284	0	0	0	0	NaN	NaN	136898	137810
Gr DeCh	39831	37466	0.8748	0.8819	139148	139899	17047	20137	0.8909	0.8742	139144	139894
	45585	52056	0.8573	0.8411	136898	137810	26025	28548	0.8403	0.8284	136891	137803
Gr DePr	42176	39582	0.8684	0.8761	139144	139894	22784	17329	0.8593	0.8898	139138	139887
	40748	41903	0.8704	0.868	136891	137803	19560	23508	0.875	0.8543	136888	137798
Gr DeCo	34503	41530	0.8897	0.8707	139138	139887	19392	22253	0.8777	0.8628	139133	139883
	43781	33068	0.8621	0.8929	136888	137798	21188	18395	0.866	0.8822	136883	137792
Gr BrCo	15111	19277	0.902	0.8789	139133	139883	15111	19277	0.902	0.8789	0	0
	22593	14673	0.8583	0.9038	136883	137792	22593	14673	0.8583	0.9038	0	0
Lauf 1-4												
	MW	SA	MW	SA	MW	SA	MW	SA	MW	SA	MW	SA
	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK
Gr SuSi	22939.2	4570.31	0.8584	0.0252	0	0	0	0	0	0	138438	1162.39
	0.5014	0.1992	0.0728	0.0293	NaN	NaN	NaN	NaN	nan	nan	0.0217	0.0084
Gr DeCh	43734.5	5639.21	0.8638	0.0159	138438	1162.39	22939.2	4570.31	0.8584	0.0252	138433	1163.47
	0.3336	0.1289	0.0472	0.0184	0.0217	0.0084	0.5014	0.1992	0.0728	0.0293	0.0217	0.0084
Gr DePr	41102.2	1028.44	0.8707	0.0032	138433	1163.47	20795.2	2492.73	0.8696	0.0139	138427	1162.03
	0.0631	0.025	0.0092	0.0037	0.0217	0.0084	0.2971	0.1199	0.0408	0.016	0.0217	0.0084
Gr DeCo	38220.5	4534.31	0.8789	0.0128	138427	1162.03	20307	1504.66	0.8722	0.008	138422	1162.48
	0.2803	0.1186	0.035	0.0146	0.0217	0.0084	0.19	0.0741	0.0223	0.0092	0.0217	0.0084
Gr BrCo	17913.5	3244.67	0.8858	0.0186	138422	1162.48	17913.5	3244.67	0.8858	0.0186	0	0
	0.4421	0.1811	0.0513	0.021	0.0217	0.0084	0.4421	0.1811	0.0513	0.021	NaN	NaN
Gr	V-AbQ/Gr	G/V-D/Gr	UV-D/Gr	Abs GrU	Abs GrG	Sf						
	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2	Lauf1	Lauf2
	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4	Lauf3	Lauf4
Gr SuSi	0.8909	0.8742	57.8896	71.8895	57.8896	71.8895	8055170	10057100	8055170	10057100	0	0
	0.8403	0.8284	34.4732	50.8382	34.4732	50.8382	4719320	7006010	4719320	7006010	0	0
Gr DeCh	0.8593	0.8898	41.8086	76.1673	99.6999	148.058	13872500	20712400	5817380	10655300	0	0
	0.875	0.8543	152.69	138.228	187.165	189.069	25621100	26054300	20901700	19048300	0	0
Gr DePr	0.8777	0.8628	124.761	154.634	224.466	302.701	31231300	42343600	17358800	21631100	0	0
	0.866	0.8822	96.9765	78.0464	284.146	267.124	38895900	36808900	13274800	10754500	0	0
Gr DeCo	0.902	0.8789	194.349	138.819	418.824	441.527	58271400	61762200	27040100	19418500	0	0
	0.8583	0.9038	160.484	113.988	444.639	381.121	60863500	52515500	21967500	15706600	0	0
Gr BrCo	NaN	NaN	NaN	NaN	NaN	NaN	0	0	-58271400	-61762200	0	0
	NaN	NaN	NaN	NaN	NaN	NaN	0	0	-60863500	-52515500	0	0
Lauf 1-4												
	MW	SA	MW	SA	MW	SA	MW	SA	MW	SA	MW	SA
	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK	Rel. SW	VaK
Gr SuSi	0.8584	0.0252	53.7726	13.4744	53.7726	13.4744	7459410	1924620	7459410	1924620	0	0
	0.0728	0.0293	0.6958	0.2506	0.6958	0.2506	0.7156	0.258	0.7156	0.258	NaN	NaN
Gr DeCh	0.8696	0.0139	102.223	45.1999	155.998	36.3925	21565100	4911890	14105700	6148520	0	0
	0.0408	0.016	1.0847	0.4422	0.5729	0.2333	0.5649	0.2278	1.0694	0.4359	NaN	NaN
Gr DePr	0.8722	0.008	113.604	28.9343	269.609	28.9414	37319900	4032770	15754800	4130900	0	0
	0.0223	0.0092	0.6742	0.2547	0.2902	0.1073	0.2978	0.1081	0.6904	0.2622	NaN	NaN
Gr DeCo	0.8858	0.0186	151.91	29.5131	421.528	25.3676	58353100	3605820	21033200	4121140	0	0
	0.0513	0.021	0.529	0.1943	0.1507	0.0602	0.1585	0.0618	0.5388	0.1959	NaN	NaN
Gr BrCo	nan	nan	nan	nan	nan	nan	0	0	-58353100	3605820	0	0
	nan	nan	nan	nan	nan	nan	NaN	NaN	-0.1585	-0.0618	NaN	NaN

Tabelle 25: Medio-Tabelle – Details

11.3.3 Mikro-Tabellen

Szenario: MIX(G)-P5-Pn30-RWS-nPC(0,5-1-0)-GMU(0,5-0,1)-WE(3)-AvP-Eymann-Test1-50A-2Pool-HFF

	Gr	# Ag	pM	G	U	D	E-D	V-D	G/V-D	U/V-D
	SuSi	10	10%	8055170	8055170	139147	0	139147	57.8896	57.8896
Maximum				812049	812049	14027	0	14027	58.0225	58.0225
Minimum				798112	798112	13816	0	13816	57.7672	57.7672
Mittelwert				805517	805517	13914.7	0	13914.7	57.8896	57.8896
Median				804742	804742	13895	0	13895	57.876	57.876
relative Spannweite				0.0173	0.0173	0.0152	NaN	0.0152	0.0044	0.0044
Standardabweichung				3688.84	3688.84	61.2797	0	61.2797	0.0861	0.0861
Herfindahl-Index				0.1	0.1	0.1	NaN	0.1	0.1	0.1
	DeCh	10	10%	5817380	13872500	278290	139147	139143	41.8086	99.6999
Maximum				583571	1389280	27932	13966	13966	42.1163	99.9198
Minimum				580136	1382690	27677	13839	13838	41.5847	99.4762
Mittelwert				581738	1387250	27829	13914.7	13914.3	41.8089	99.7002
Median				581307	1387600	27827	13914	13913	41.7406	99.667
relative Spannweite				0.0059	0.0048	0.0092	0.0091	0.0092	0.0127	0.0044
Standardabweichung				1148.66	1966.5	66.3068	33.0758	33.2326	0.1503	0.1356
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	DePr	10	10%	17358800	31231300	278279	139143	139136	124.761	224.466
Maximum				1747740	3143720	27940	13970	13970	125.237	225.033
Minimum				1725700	3111340	27757	13879	13878	124.183	223.91
Mittelwert				1735880	3123130	27827.9	13914.3	13913.6	124.761	224.466
Median				1732950	3120610	27810	13905	13905	124.731	224.415
relative Spannweite				0.0127	0.0104	0.0066	0.0065	0.0066	0.0084	0.005
Standardabweichung				6323.52	8461.47	50.1466	24.9761	25.1722	0.362	0.3404
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	DeCo	10	10%	27040100	58271400	278267	139136	139131	194.349	418.824
Maximum				2714480	5840350	27917	13959	13958	195.328	419.117
Minimum				2694220	5804230	27699	13850	13849	193.314	418.423
Mittelwert				2704010	5827140	27826.7	13913.6	13913.1	194.35	418.824
Median				2704010	5824470	27819	13910	13909	194.357	418.747
relative Spannweite				0.0075	0.0062	0.0078	0.0078	0.0078	0.0104	0.0017
Standardabweichung				6045.58	9901.57	55.9536	27.9864	27.9694	0.4753	0.1978
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	BrCo	10	10%	-58271400	0	139131	139131	0	NaN	NaN
Maximum				-5764100	0	14029	14029	0	NaN	NaN
Minimum				-5878420	0	13755	13755	0	NaN	NaN
Mittelwert				-5827140	0	13913.1	13913.1	0	nan	nan
Median				-5836880	0	13900	13900	0	NaN	NaN
relative Spannweite				-0.0196	NaN	0.0197	0.0197	NaN	nan	nan
Standardabweichung				33539.8	0	84.7684	84.7684	0	nan	nan
Herfindahl-Index				0.1	NaN	0.1	0.1	NaN	nan	nan

Tabelle 26: Mikro-Tabelle – Simulationslauf 1

	Gr	# Ag	pM	G	U	D	E-D	V-D	G/V-D	U/V-D
	SuSi	10	10%	10057100	10057100	139897	0	139897	71.8895	71.8895
Maximum				1013020	1013020	14102	0	14102	72.11	72.11
Minimum				1001440	1001440	13904	0	13904	71.5619	71.5619
Mittelwert				1005710	1005710	13989.7	0	13989.7	71.89	71.89
Median				1004710	1004710	13972	0	13972	71.9226	71.9226
relative Spannweite				0.0115	0.0115	0.0142	NaN	0.0142	0.0076	0.0076
Standardabweichung				3405.49	3405.49	61.6863	0	61.6863	0.171	0.171
Herfindahl-Index				0.1	0.1	0.1	NaN	0.1	0.1	0.1
	DeCh	10	10%	10655300	20712400	279791	139897	139894	76.1673	148.058
Maximum				1068460	2076400	28052	14026	14026	76.406	148.3
Minimum				1062490	2068560	27930	13965	13965	75.9741	147.767
Mittelwert				1065530	2071240	27979.1	13989.7	13989.4	76.1674	148.058
Median				1065380	2070800	27968	13984	13984	76.1371	148.024
relative Spannweite				0.0056	0.0038	0.0044	0.0044	0.0044	0.0057	0.0036
Standardabweichung				2067.44	2165.2	37.6389	18.8523	18.7893	0.1473	0.1538
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	DePr	10	10%	21631100	42343600	279780	139894	139886	154.634	302.701
Maximum				2173110	4246460	28051	14026	14025	154.945	303.025
Minimum				2152740	4212310	27864	13932	13932	154.25	302.298
Mittelwert				2163110	4234360	27978	13989.4	13988.6	154.634	302.701
Median				2163350	4237920	27989	13995	13994	154.629	302.698
relative Spannweite				0.0094	0.0081	0.0067	0.0067	0.0066	0.0045	0.0024
Standardabweichung				5453.61	11105.2	66.1331	33.1366	32.9975	0.1904	0.2207
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	DeCo	10	10%	19418500	61762200	279769	139886	139883	138.819	441.527
Maximum				1952920	6199190	28073	14037	14036	139.744	441.916
Minimum				1933620	6155480	27868	13934	13934	138.375	441.077
Mittelwert				1941850	6176220	27976.9	13988.6	13988.3	138.82	441.527
Median				1941260	6171890	27959	13980	13979	138.844	441.474
relative Spannweite				0.0099	0.0071	0.0073	0.0074	0.0073	0.0099	0.0019
Standardabweichung				5677.88	12353.8	57.5837	28.8485	28.7369	0.4265	0.24
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	BrCo	10	10%	-61762200	0	139883	139883	0	NaN	NaN
Maximum				-6049630	0	14129	14129	0	NaN	NaN
Minimum				-6244010	0	13742	13742	0	NaN	NaN
Mittelwert				-6176220	0	13988.3	13988.3	0	nan	nan
Median				-6186330	0	13973	13973	0	NaN	NaN
relative Spannweite				-0.0315	NaN	0.0277	0.0277	NaN	nan	nan
Standardabweichung				50939.8	0	106.062	106.062	0	nan	nan
Herfindahl-Index				0.1	NaN	0.1	0.1	NaN	nan	nan

Tabelle 27: Mikro-Tabelle – Simulationslauf 2

	Gr	# Ag	pM	G	U	D	E-D	V-D	G/V-D	U/V-D
	SuSi	10	10%	4719320	4719320	136898	0	136898	34.4732	34.4732
Maximum				475693	475693	13750	0	13750	34.5958	34.5958
Minimum				467448	467448	13633	0	13633	34.2879	34.2879
Mittelwert				471932	471932	13689.8	0	13689.8	34.4731	34.4731
Median				472570	472570	13671	0	13671	34.4697	34.4697
relative Spannweite				0.0175	0.0175	0.0085	NaN	0.0085	0.0089	0.0089
Standardabweichung				2433.36	2433.36	39.4076	0	39.4076	0.1003	0.1003
Herfindahl-Index				0.1	0.1	0.1	NaN	0.1	0.1	0.1
	DeCh	10	10%	20901700	25621100	273788	136898	136890	152.69	187.165
Maximum				2097870	2571370	27462	13731	13731	152.988	187.421
Minimum				2082470	2552990	27261	13631	13630	152.414	186.881
Mittelwert				2090170	2562110	27378.8	13689.8	13689	152.69	187.165
Median				2090880	2563830	27394	13697	13697	152.714	187.221
relative Spannweite				0.0074	0.0072	0.0073	0.0073	0.0074	0.0038	0.0029
Standardabweichung				4743.51	5606.92	63.6314	31.7389	31.8935	0.1738	0.1662
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	DePr	10	10%	13274800	38895900	273777	136890	136887	96.9765	284.146
Maximum				1335080	3904950	27454	13727	13727	97.6377	284.649
Minimum				1316600	3867510	27270	13635	13635	96.1379	283.563
Mittelwert				1327480	3889590	27377.7	13689	13688.7	96.9766	284.146
Median				1325500	3890230	27402	13701	13701	96.9694	284.062
relative Spannweite				0.0139	0.0096	0.0067	0.0067	0.0067	0.0155	0.0038
Standardabweichung				6539.49	10600.1	67.1759	33.5857	33.5918	0.4363	0.3512
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	DeCo	10	10%	21967500	60863500	273770	136887	136883	160.484	444.639
Maximum				2202560	6108960	27479	13740	13739	160.842	444.891
Minimum				2189690	6068310	27281	13641	13640	160.252	444.223
Mittelwert				2196750	6086350	27377	13688.7	13688.3	160.484	444.639
Median				2197210	6083210	27372	13686	13686	160.434	444.644
relative Spannweite				0.0059	0.0067	0.0072	0.0072	0.0072	0.0037	0.0015
Standardabweichung				4764.05	12226.3	61.4865	30.7572	30.7312	0.1774	0.2034
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	BrCo	10	10%	-60863500	0	136883	136883	0	NaN	NaN
Maximum				-6023240	0	13897	13897	0	NaN	NaN
Minimum				-6175120	0	13542	13542	0	NaN	NaN
Mittelwert				-6086350	0	13688.3	13688.3	0	nan	nan
Median				-6083830	0	13669	13669	0	NaN	NaN
relative Spannweite				-0.025	NaN	0.0259	0.0259	NaN	nan	nan
Standardabweichung				44487	0	98.0541	98.0541	0	nan	nan
Herfindahl-Index				0.1	NaN	0.1	0.1	NaN	nan	nan

Tabelle 28: Mikro-Tabelle – Simulationslauf 3

	Gr	# Ag	pM	G	U	D	E-D	V-D	G/V-D	U/V-D
	SuSi	10	10%	7006010	7006010	137810	0	137810	50.8382	50.8382
Maximum				708174	708174	13858	0	13858	51.1021	51.1021
Minimum				694023	694023	13655	0	13655	50.6112	50.6112
Mittelwert				700601	700601	13781	0	13781	50.8379	50.8379
Median				701271	701271	13767	0	13767	50.8256	50.8256
relative Spannweite				0.0202	0.0202	0.0147	NaN	0.0147	0.0097	0.0097
Standardabweichung				4226.27	4226.27	60.0016	0	60.0016	0.1377	0.1377
Herfindahl-Index				0.1	0.1	0.1	NaN	0.1	0.1	0.1
	DeCh	10	10%	19048300	26054300	275613	137810	137803	138.228	189.069
Maximum				1911080	2613800	27637	13819	13818	138.666	189.359
Minimum				1894390	2591970	27443	13722	13721	137.786	188.8
Mittelwert				1904830	2605430	27561.3	13781	13780.3	138.228	189.069
Median				1904640	2604500	27558	13779	13779	138.116	188.978
relative Spannweite				0.0088	0.0084	0.007	0.007	0.007	0.0064	0.003
Standardabweichung				5084.08	6189.45	54.6206	27.247	27.3753	0.2719	0.2018
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	DePr	10	10%	10754500	36808900	275600	137803	137797	78.0464	267.124
Maximum				1080570	3690340	27610	13805	13805	78.3858	267.435
Minimum				1067520	3658240	27394	13697	13697	77.4521	266.663
Mittelwert				1075450	3680890	27560	13780.3	13779.7	78.0464	267.124
Median				1076810	3682850	27566	13783	13783	78.0738	267.083
relative Spannweite				0.0121	0.0087	0.0078	0.0078	0.0078	0.012	0.0029
Standardabweichung				4704.24	9487.48	60.0915	30.1033	29.9901	0.2886	0.2276
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	DeCo	10	10%	15706600	52515500	275589	137797	137792	113.988	381.121
Maximum				1583870	5270940	27653	13827	13826	114.765	381.276
Minimum				1561680	5235170	27475	13738	13737	113.489	380.811
Mittelwert				1570660	5251550	27558.9	13779.7	13779.2	113.987	381.121
Median				1569310	5249790	27552	13776	13776	113.832	381.156
relative Spannweite				0.0141	0.0068	0.0065	0.0065	0.0065	0.0112	0.0012
Standardabweichung				6861.54	9939.02	53.6142	26.8031	26.8134	0.4227	0.1643
Herfindahl-Index				0.1	0.1	0.1	0.1	0.1	0.1	0.1
	BrCo	10	10%	-52515500	0	137792	137792	0	NaN	NaN
Maximum				-5199660	0	13888	13888	0	NaN	NaN
Minimum				-5293120	0	13675	13675	0	NaN	NaN
Mittelwert				-5251550	0	13779.2	13779.2	0	nan	nan
Median				-5265570	0	13767	13767	0	NaN	NaN
relative Spannweite				-0.0178	NaN	0.0155	0.0155	NaN	nan	nan
Standardabweichung				30869.2	0	69.7277	69.7277	0	nan	nan
Herfindahl-Index				0.1	NaN	0.1	0.1	NaN	nan	nan

Tabelle 29: Mikro-Tabelle – Simulationslauf 4