

---

---

# Kalibrierung Agenten-basierter Simulationen

---

---

Dissertation zur Erlangung des  
naturwissenschaftlichen Doktorgrades  
der Julius–Maximilians–Universität Würzburg

vorgelegt von

Manuel Fehler

aus  
Karlsruhe

Würzburg, 2010

Eingereicht am: 01.12.2010

bei der Fakultät für Mathematik und Informatik

1. Gutachter: Prof. Dr. Frank Puppe

2. Gutachter: Dr. habil. Franziska Klügl-Frohnmeier

Tag der mündlichen Prüfung: 08.07.2011

*Mein besonderer Dank gilt meinen Betreuern, meinem Doktorvater Prof. Dr. Frank Puppe und meiner Zweitgutachterin Dr. habil. Franziska Klügl-Frohnmeier, für ihre Unterstützung bei der Erstellung meiner Arbeit.*

*Für Isabelle und Dieter, meine lieben Eltern, und meine Liebste, Katja.*



# Inhaltsverzeichnis

<b>1</b>	<b>Motivation</b>	<b>9</b>
1.1	Abstract . . . . .	9
1.2	Das allgemeine Kalibrierungsproblem für ABS . . . . .	9
1.3	Existierende Kalibrierungsverfahren . . . . .	10
1.4	ABS-spezifische Probleme existierender Kalibrierungsverfahren . . . . .	11
1.5	Ziele der Arbeit . . . . .	14
1.6	Aufbau der Arbeit . . . . .	16
<b>I</b>	<b>Einführung</b>	<b>17</b>
<b>2</b>	<b>Agenten-basierte Simulation (ABS)</b>	<b>19</b>
2.1	Verteilte Künstliche Intelligenz . . . . .	19
2.2	Simulation . . . . .	21
2.3	Simulationsparadigmen . . . . .	28
2.4	Agenten-basierte Simulation (ABS) . . . . .	29
<b>3</b>	<b>Allgemeine Kalibrierung und Stand der Forschung</b>	<b>37</b>
3.1	Kalibrierung von Simulationen . . . . .	37
3.2	Stand der Forschung: Black Box Simulationsoptimierung . . . . .	53
3.3	Stand der Forschung: Automatischer Modellentwurf . . . . .	69
<b>4</b>	<b>ABS Kalibrierungsproblem und Stand der Forschung</b>	<b>71</b>
4.1	Kalibrierungsprobleme in ABS . . . . .	71
<b>5</b>	<b>ABS Kalibrierung: Stand der Forschung</b>	<b>85</b>
5.1	Stand der Forschung: Kalibrierung in ABS . . . . .	85
<b>6</b>	<b>Anforderungen an Verfahren zur ABS-Kalibrierung</b>	<b>97</b>
6.1	Allgemeine Anforderungen an Kalibrierungswerkzeuge . . . . .	97
6.2	Anforderungen an Verfahren zur ABS-Kalibrierung . . . . .	99
6.3	Ausblick auf die weitere Arbeit . . . . .	100
<b>II</b>	<b>Konzepte</b>	<b>101</b>
<b>7</b>	<b>Überblick: Makro-Mikro-Kalibrierungsverfahren und Fallstudie</b>	<b>103</b>

7.1	Makro-Mikro-Verfahren . . . . .	105
7.2	Robuste Kalibrierung im Makro-Mikro-Verfahren . . . . .	106
7.3	Heterogenitätssuche . . . . .	106
7.4	Analyseverfahren für das Kalibrierungsproblem . . . . .	107
7.5	Fallstudie: Agenten-basierte Einkaufssimulation . . . . .	108
<b>8</b>	<b>Problemanalyse</b>	<b>121</b>
8.1	Analyseverfahren für konkrete Kalibrierungsprobleme . . . . .	122
8.2	Beispielanalyse: Einkaufssimulation . . . . .	124
<b>9</b>	<b>Das Makro-Mikro-Verfahren mit robuster Kalibrierung</b>	<b>127</b>
9.1	Black-Box-Kalibrierung und die Makro-Mikro-Lücke . . . . .	127
9.2	Reverse-Engineering-Verfeinerung . . . . .	138
9.3	Robuste Kalibrierung auf der Mikro-Ebene . . . . .	143
9.4	Diskussion des Makro-Mikro-Verfahrens . . . . .	150
<b>10</b>	<b>Heterogenitätssuche</b>	<b>165</b>
10.1	Lokalisierung von Heterogenität während der Kalibrierung . . . . .	166
10.2	Kalibrierung mit Heterogenitätssuche . . . . .	168
10.3	Einbettung in das Makro-Mikro-Verfahren . . . . .	177
<b>III</b>	<b>Werkzeuge</b>	<b>179</b>
<b>11</b>	<b>Der DAVINCI-Kalibrierer: Technik</b>	<b>181</b>
11.1	SeSAM und verwendete Programmierbibliotheken . . . . .	182
11.2	Aufbau des DAVINCI-Kalibrierers . . . . .	184
11.3	Weitere Aspekte der technischen Umsetzung von DAVINCI . . . . .	197
11.4	Black-Box Suche in DAVINCI . . . . .	199
11.5	Zusammenfassung . . . . .	200
<b>12</b>	<b>Der DAVINCI-Kalibrierer: Interaktion</b>	<b>201</b>
12.1	Steuerung der Black-Box-Suchverfahren zur Laufzeit . . . . .	201
12.2	Laufzeitinteraktion mit dem DAVINCI-Kalibrierer . . . . .	203
<b>13</b>	<b>Der DAVINCI Kalibrierer: Vergleich</b>	<b>211</b>
13.1	Vergleichstabellen . . . . .	211
13.2	Ergebnis des Vergleichs . . . . .	211
<b>IV</b>	<b>Evaluation und Ausblick</b>	<b>215</b>
<b>14</b>	<b>Evaluation: Einleitung und Überblick</b>	<b>217</b>
14.1	Vorbemerkung zu verwendeten Suchverfahren und Validierungskriterien . . . . .	218
14.2	Überblick über die Evaluation und ihre Ergebnisse . . . . .	219

<b>15 Evaluation: Makro-Mikro-Verfahren für ABS</b>	<b>221</b>
15.1 Evaluationsziele . . . . .	221
15.2 Diskussion an künstlichem Evaluationsmodell . . . . .	222
15.3 Kurze Einleitung in die Fallstudie der Einkaufssimulation . . . . .	227
15.4 $KP_{GeoUmeaHom}$ : Problemstellung . . . . .	228
15.5 $KP_{GeoUmeaHom}$ : Lösung mit Black-Box Ansatz . . . . .	230
15.6 $KP_{GeoUmeaHom}$ : Lösung mit dem Makro-Mikro-Verfahren . . . . .	231
15.7 $KP_{GeoUmeaHom}$ : Evaluationsergebnis Makro-Mikro-Verfahren . . . . .	238
15.8 $KP_{GeoUmeaHet}$ : Problemstellung . . . . .	240
15.9 $KP_{GeoUmeaHet}$ : Lösung mit Black-Box-Ansatz . . . . .	241
15.10 $KP_{GeoUmeaHet}$ : Lösung mit Makro-Mikro-Verfahren . . . . .	241
15.11 Zusammenfassung der Evaluation des Makro-Mikro-Verfahrens . . . . .	248
<b>16 Evaluation: Heterogenitätssuche</b>	<b>251</b>
16.1 Evaluierungsziele . . . . .	251
16.2 Künstliches Evaluationsmodell ( $HP_{HetAbs}$ ) . . . . .	252
16.3 $HP_{HetAbs}$ : Black-Box-Verfahren . . . . .	255
16.4 $HP_{HetAbs}$ : Kalibrierung mit Heterogenitätssuche . . . . .	256
16.5 $HP_{HetAbs}$ : Ergebnis der Heterogenitätssuche . . . . .	257
16.6 Heterogenitätssuche in der Einkaufssimulation . . . . .	259
16.7 $HP_{EK3Gruppen}$ : Problembeschreibung . . . . .	259
16.8 $HP_{EK3Gruppen}$ : Kalibrierung mit Heterogenitätssuche . . . . .	261
16.9 $HP_{EK3Gruppen}$ : Kalibrierungsergebnis mit Heterogenitätssuche . . . . .	261
16.10 $HP_{EKDistGruppen}$ : Problembeschreibung . . . . .	262
16.11 $HP_{EKDistGruppen}$ : Heterogenitätssuche . . . . .	262
16.12 $HP_{EK3Gruppen}$ und $HP_{EKDistGruppen}$ : Evaluationsergebnisse . . . . .	265
<b>17 Evaluation: Robuste Kalibrierung</b>	<b>267</b>
17.1 Evaluierungsziele . . . . .	267
17.2 Problembeschreibung robuste Kalibrierung der Einkaufssimulation . . . . .	267
17.3 Lösung mit robuster Kalibrierung . . . . .	269
17.4 Kalibrierungsergebnisse . . . . .	271
17.5 Diskussion der Evaluierung . . . . .	275
<b>18 Zusammenfassung und Ausblick</b>	<b>277</b>
18.1 Anwendung des Makro-Mikro-Verfahrens auf weitere ABS . . . . .	278
18.2 Verzahnung von Kalibrierung und Datenanalyse . . . . .	278
18.3 Kalibrierung und interaktive Modelleingriffe . . . . .	280
18.4 Verteilte Simulation und Kalibrierung . . . . .	280
18.5 Modellierung entlang des Makro-Mikro-Verfahrens . . . . .	280
18.6 Kalibrierung und Modellausprägung mit Komponentenbibliotheken . . . . .	281
18.7 Übertragung auf die Kalibrierung und Optimierung realer Agenten . . . . .	281

<b>V</b>	<b>Anhang</b>	<b>283</b>
<b>A</b>	<b>Vorausgehende Fallstudien</b>	<b>285</b>
A.1	Krankenhaussimulation . . . . .	285
A.2	Simulation kooperativer Ampeln . . . . .	286
A.3	Projekt Evakuierungssimulation . . . . .	288
A.4	Simulation von Käferwanderungen in ländlichen Regionen . . . . .	288
A.5	Projekt Bienensimulation . . . . .	290
	<b>Abkürzungsverzeichnis</b>	<b>292</b>
	<b>Literaturverzeichnis</b>	<b>296</b>



# Kapitel 1

## Motivation

### 1.1 Abstract

In der vorliegenden Arbeit wird das Problem der Kalibrierung Agenten-basierter Simulationen (ABS) behandelt, also das Problem, die Parameterwerte eines Agenten-basierten Simulationsmodells so einzustellen, dass valides Simulationsverhalten erreicht wird. Das Kalibrierungsproblem für Simulationen an sich ist nicht neu und ist im Rahmen klassischer Simulationsparadigmen, wie z.B. der Makro-Simulation, fester Bestandteil der Forschung. Im Vergleich zu den dort betrachteten Kalibrierungsproblemen zeichnet sich das Kalibrierungsproblem für ABS jedoch durch eine Reihe zusätzlicher Herausforderungen aus, welche die direkte Anwendung existierender Kalibrierungsverfahren in begrenzter Zeit erschweren, bzw. nicht mehr sinnvoll zulassen. Die Lösung dieser Probleme steht im Zentrum dieser Dissertation.

Im weiteren Verlauf dieses Kapitels wird zunächst ein kurzer Überblick über das Kalibrierungsproblem für ABS gegeben und erläutert, welche besonderen Merkmale des ABS-Kalibrierungsproblems die Anwendung existierender Verfahren erschweren. Zum Abschluss der Einleitung wird ein Überblick über die in dieser Arbeit vorgeschlagenen Lösungsverfahren gegeben, mit deren Hilfe die beschriebenen Probleme gelöst werden können.

### 1.2 Das allgemeine Kalibrierungsproblem für ABS

Das in dieser Arbeit behandelte Problem ist die Kalibrierung Agenten-basierter Simulationen. Es besteht in der Aufgabe, geeignete Parameterwerte für ein Agenten-basiertes Simulationsmodell zu identifizieren, so dass das Gesamtverhalten der Simulation ausreichend valide ist. Parameterwerte dienen im Simulationsmodell als „Stellschrauben“ für das Agentenverhalten. Bei der Entwicklung eines Agenten-basierten Simulationsmodells müssen zunächst strukturelle Aspekte, z.B. eine Menge von Plänen des Agenten, festgelegt werden. Häufig sind solche Modellstrukturen parametrisiert, d.h. der Einfluss einer gewählten Modellstruktur auf das zugehörige Simulationsverhalten hängt von ihren zugehörigen Steuerungsparametern ab. Sollen sich die simulierten Agenten beispielsweise in einer bestimmten Weise bewegen, so muss zur genauen Einstellung des Agentenverhaltens eine geeignete Geschwindigkeit gewählt werden. Das Verhalten einer Agenten-basierten Simulation wird in dieser Arbeit als ausreichend valide betrachtet, wenn das Simulations-

verhalten der ABS dem Verhalten eines vorgegebenen simulierten Originalsystems auf allen Beobachtungsebenen ausreichend genau entspricht.

Das beschriebene Kalibrierungsproblem existiert grundsätzlich für alle Arten von Simulationen (siehe z.B. Fu [47]). Es gibt jedoch verschiedene spezifische Schwierigkeiten bei der Kalibrierung Agenten-basierter Simulationen. Im Vergleich zu klassischen Simulationsparadigmen, wie z.B. der Makrosimulation, zeichnen sich ABS häufig durch einen höheren Detaillierungsgrad aus, da die internen Strukturen des realen Systems weniger stark abstrahiert in das Modell abgebildet werden [76]. Agenten-basierte Simulationen (auch: Multiagentensimulationen (MASim)) sind ein mikroskopisches Simulationsparadigma. Bei der Entwicklung von ABS findet Modellierung des Originalsystems auf Ebene der einzelnen simulierten Einheiten statt (der so genannten Mikro-Ebene). In Multiagentenmodellen werden einzelne Individuen eines Originalsystems mit einer Reihe individueller Attribute, individuellem Verhalten und Zielen und einer gewissen Autonomie modelliert [76, 50]. Es wird versucht, die empirisch beobachteten aktiven Einheiten eines realen Systems direkt in modellierte Agenten des Simulationsmodells abzubilden (*Beobachtungseinheit = Modellierungseinheit*) [76]. Das aggregierte Gesamtverhalten der Simulation (das Verhalten auf der so genannten Makro-Ebene) ergibt sich aus den modellierten individuellen Verhaltensweisen und den daraus resultierenden Interaktionen der Agenten untereinander und mit ihrer Umwelt.

Die beschriebenen Eigenschaften haben Auswirkungen auf die Anwendbarkeit existierender Kalibrierungsverfahren auf ABS. Zur Analyse dieser Auswirkungen wird im Folgenden zunächst auf etablierte Kalibrierungsverfahren für allgemeine Simulationen eingegangen und dann deren Probleme bei der Anwendung auf Agenten-basierte Simulationen beschrieben.

### 1.3 Existierende Kalibrierungsverfahren

Wie bereits beschrieben ist das Kalibrierungsproblem an sich nicht neu, sondern praktisch für jedes Simulationsparadigma relevant, in dem bei der Entwicklung des Simulationsmodells auch Parameterwerte eingestellt werden müssen. Im Bereich klassischer Simulationsparadigmen, bei denen die Zusammenhänge eines Originalsystems aggregiert abgebildet werden (z.B. Makro-Simulationen), wurden bereits viele Verfahren zur Lösung des Kalibrierungsproblems vorgeschlagen (siehe z.B. [44, 5, 111]). Die zentrale Herangehensweise aller dieser Verfahren besteht darin, das Kalibrierungsproblem für Simulationen als allgemeines Optimierungsproblem aufzufassen und mit speziell angepassten Varianten allgemeiner Optimierungsverfahren zu lösen. Aufgrund der Herangehensweise spricht man daher nicht von Simulationskalibrierung, sondern allgemeiner von Simulationsoptimierung. Unter Simulationsoptimierung versteht man dabei genauer die Anpassung von Simulationsparametern, so dass der Wert eines bestimmten vom Nutzer vorgegebenen Performanzkriteriums verbessert bzw. optimiert wird [44]. Zur Anwendung auf die Simulationskalibrierung wählt man dabei z.B. das Performanzkriterium als Grad der Abweichung des Simulationsverhaltens vom Verhalten des Originalsystems.

Die etablierten Techniken zur Simulationsoptimierung und -kalibrierung führen die Ermitt-

lung der Parameterwerte als Suche im Raum aller möglichen Parametereinstellungen durch und behandeln dabei das Simulationsmodell als „Black Box“ (siehe Kapitel 3.2). Dies bedeutet, dass die tatsächlichen Modellstrukturen für die Parametersuche nicht veränderbar und daher für die Algorithmen der Suchverfahren nur insofern relevant sind, dass sie die Menge der vorhandenen Parameter bestimmen. Als Folge spielt die eigentliche Struktur eines Modells bei der Verwendung von „Black Box“ Suchverfahren nur eine untergeordnete Rolle. Es ist nicht wichtig, ob eine spezielle Struktur  $A$  oder  $B$  kalibriert wird. Wichtig ist die Menge und die Eigenschaften der Parameter, die mit der jeweiligen Struktur verbunden sind. Aus diesem Grund können Verfahren zur Simulationsoptimierung im Prinzip auf jedes Kalibrierungsproblem angewendet werden. Es gibt eine große Anzahl verschiedener „Black-Box“ Optimierungsstrategien, die im Rahmen dieser Arbeit beschrieben werden.

### 1.4 ABS-spezifische Probleme existierender Kalibrierungsverfahren

Bei der Anwendung von Black-Box-Optimierungsverfahren auf die Kalibrierung von ABS existieren jedoch besondere Schwierigkeiten. Diese resultieren aus der Möglichkeit, mit ABS ein Originalsystem sehr detailliert abbilden und im Modell sehr komplexe Zusammenhänge darstellen zu können und dabei sicher stellen zu müssen, dass das Simulationsverhalten auf allen Beobachtungsebenen valide ist: Ein Originalsystem wird in ABS nicht auf der Ebene des globalen Ein- und Ausgabeverhaltens modelliert (der so genannten Makro-Ebene [50]), sondern wie beschrieben auf der Ebene der einzelnen konkreten beobachteten aktiven Einheiten (Mikro-Ebene). Das Gesamtverhalten der Simulation ergibt sich dann aus den simulierten Wechselwirkungen zwischen den Agenten und ihrer Umwelt. In den folgenden Unterabschnitten werden die resultierenden ABS-Probleme im einzelnen betrachtet.

#### 1.4.1 Parametermenge und ABS-Laufzeit

Ein hoher Detaillierungsgrad und erhöhte Komplexität der Wirkungszusammenhänge zwischen den einzelnen simulierten Agenten in ABS kann dazu führen, dass die Menge zu kalibrierender Modellparameter und die Laufzeit einzelner Simulationsläufe deutlich größer sind als bei vergleichbaren Makromodellen [37]. Müssen z.B. für jeden einzelnen simulierten Agenten eigene Parameterwerte eingestellt werden, so steigt die Anzahl zu kalibrierender Parameter mit der Anzahl an Agenten im Modell. Eine Möglichkeit zur Reduktion der Menge an zu kalibrierenden Agentenparametern wäre es, die Agenten einfach in Gruppen mit gleichen Parameterwerten zusammen zu fassen. Im einfachsten Fall hätten dann z.B. alle Agenten die gleichen Parameterwerte (homogene Parameterwerte). Ein solches Vorgehen erfordert aber explizites Wissen welche Agenten gleiche und welche Agenten unterschiedliche Parameterwerte erhalten müssen. Dieses Wissen kann jedoch nicht vorausgesetzt werden.

Eine große Zahl zu kalibrierender Parameter führt zu einer kombinatorischen „Explosion“ des Parametersuchraumes. Wendet man das gleiche Optimierungsverfahren auf zwei Probleme - eines mit weniger und eines mit mehr zu kalibrierenden Parametern - an, so muss beim Problem mit mehr Parametern vom Simulationsoptimierungsverfahren eine größere Menge möglicher Parametereinstellungen nach der optimalen Einstellung durchsucht

werden. Betrachtet man zwei Simulationskalibrierungsprobleme, bei denen im einen Fall die Laufzeit einzelner Simulationsläufe länger ist als im anderen, dann können bei der Suche für das Problem mit längerer Simulationslaufzeit in beschränkter Zeit nur weniger unterschiedliche Parametereinstellungen betrachtet werden als im Problem mit kürzerer Simulationslaufzeit. Dies führt letztendlich dazu, dass für ABS mit vielen Agenten und langer Laufzeit die einfache Anwendung existierender Simulationsoptimierungsverfahren nicht mehr sinnvoll möglich ist.

Das Problem des Umgangs mit vielen Parametern und ihrer großen Wertebereiche und wenigen testbaren Parametereinstellungen existiert auch in anderen Simulationsparadigmen. Vorhandene Simulationsoptimierungsverfahren versuchen dieses Problem aber auf der Basis der beschriebenen „Black-Box“-Optimierungsverfahren und statistischer Analysen zu lösen und vernachlässigen dabei Möglichkeiten, die durch Ausnutzung von Wissen über die Modellstruktur potentiell vorhanden wären.

#### **1.4.2 Validierung auf verschiedenen Aggregierungsebenen des Simulationsverhaltens**

Bei der Anwendung von Verfahren zur Simulationsoptimierung in klassischen Simulationsparadigmen wird versucht, eine so genannte Zielfunktion zu formulieren, mit deren Hilfe der Validitätsgrad des globalen Ein-/Ausgabeverhaltens eines Simulationsmodells gemessen werden kann [5]. Bei der Optimierung wird dann eine Parametereinstellung gesucht, welche den Wert der Zielfunktion optimiert. Dieser Ansatz, die Modellkalibrierung mit Hilfe einer einzigen globalen Zielfunktion durchzuführen, ist bei der Kalibrierung Agentenbasierter Simulationen nur schwer umsetzbar. Der Grund hierfür ist, dass in ABS im Gegensatz zu klassischen Makro-Modellen sowohl das Verhalten auf der Mikro- als auch das Verhalten auf der Makro-Ebene betrachtet und validiert werden muss. Es genügt nicht valides globales Ein/Ausgabeverhalten herzustellen, wenn dieses nicht aus validem Verhalten der einzelnen Agenten entsteht. Bei Kalibrierung von ABS muss dabei mit dem folgenden Dilemma umgegangen werden:

1. Es soll valides Verhalten auf der Makro-Ebene des globalen Gesamtverhaltens, auf der Mikro-Ebene des Verhaltens einzelner Agenten und unter Umständen auf beliebigen Zwischenaggregierungsebenen (z.B. dem Verhalten von Agentengruppen) hergestellt werden.
2. Die zu kalibrierenden Agenten-Parameterwerte können während der Kalibrierung aber nur auf der Ebene des konkreten Agentenverhaltens angepasst werden. Das Verhalten auf der Makro-Ebene kann nur aus dem Zusammenwirken der Agenten in der Simulation erzeugt werden.
3. Der Zusammenhang zwischen globalem Gesamtverhalten und dem zugehörigen Verhalten auf der Mikro-Ebene ist weitgehend unbekannt (so genannte Makro-Mikro-Wissenslücke). Dies bedeutet, dass nur schwer, bzw. überhaupt nicht, abgeleitet werden kann, wie das Verhalten der einzelnen Agenten (und damit ihre Parameterwerte) geändert werden muss, wenn das Gesamtverhalten der ABS invalide ist.

Ein einfaches Beispiel für dieses Dilemma ist eine Simulation von Einkaufsverhalten einer Bevölkerung, in der jeder als Agent simulierte Haushalt aufgrund seiner individuellen

Präferenzen (die Parameter der Agenten) in einer Reihe von Geschäften einkaufen geht. Als Maß für die Validität des Gesamtverhaltens wird der erzielte Umsatz der Geschäfte betrachtet. Es ist relativ leicht zu messen, ob die simulierten Geschäftsumsätze denen eines Originalsystems entsprechen. Viel schwieriger ist es zu bestimmen, wie die Werte der Präferenzparameter der einzelnen Agenten eingestellt werden müssen, damit sowohl die Geschäftsumsätze korrekt reproduziert werden und gleichzeitig das Verhalten der einzelnen Agenten valide ist. Ein Simulationsmodell, das zwar die realen Geschäftsumsätze reproduzieren kann, aber bei dem die Agenten mehrere hundert Kilometer für einen Gang zum Supermarkt zurück legen, hat nur eine sehr begrenzte Aussagekraft und Glaubwürdigkeit.

Die Notwendigkeit der gleichzeitigen Kalibrierung auf verschiedenen Aggregierungsebenen des Simulationsverhalten ist spezifisch für Agenten-basierte Simulationen und wird in keinem der existierenden Kalibrierungsverfahren betrachtet, bzw. gelöst.

#### 1.4.3 Datenlage zur Modellentwicklung

Für jedes Simulationsmodell eines realen Systems ist es entscheidend, dass für den Entwurf und die Kalibrierung des Modells genügend Wissen und Daten über das reale System verfügbar sind. Sind nicht genügend Daten vorhanden, dann ist das Kalibrierungsproblem unterbestimmt. Als Folge können mehrere mögliche Lösungen des Kalibrierungsproblems gefunden werden, die zwar alle bezüglich der verfügbaren Daten valide erscheinen, aber für die nicht unterschieden werden kann, ob wirklich alle, nur eine davon oder keine tatsächlich das Verhalten des Originalsystems valide reproduzieren kann. Nur wenn Wissen und Daten in ausreichender Menge, ausreichender Detaillierung und ausreichender Korrektheit vorliegen, kann für ein Simulationsmodell nachgewiesen werden, dass es das Verhalten des Originalsystems ausreichend genau reproduziert.

Es existieren bereits viele Verfahren zum Umgang mit verfälschten, d.h. verrauschten, Daten während der Kalibrierung (siehe z.B. [55]). Wesentliche Herangehensweisen sind hierbei unter anderem die Datenaggregation (z.B. Durchschnittsbildung aus mehrfach erhobenen Daten zur Reduktion von Fehlern durch einzelne abweichende Werte) oder die Filterung einer Datenmenge (es werden mehr Daten als benötigt erhoben und dann fehlerhafte Daten entfernt). Beide Ansätze sind für Simulationsparadigmen, für die nur das Gesamtverhalten der Simulation relevant ist, gut einsetzbar, da hier tatsächlich nur aggregiertes Verhalten des simulierten Systems betrachtet wird.

In ABS wirkt sich das Problem fehlender Datenkorrektheit stärker aus als in Simulationen, in denen das Verhalten des Originalsystems auf aggregierterer Ebene nachgebildet wird. Der Grund hierfür ist, dass die Daten nicht nur auf der Ebene des Gesamtverhaltens, sondern auch auf der Ebene des konkreten Agentenverhaltens benötigt werden:

- Auf der Ebene des konkreten Agentenverhaltens ist es schwieriger Datenfehler durch Aggregation auszugleichen. Es kann auf dieser Ebene unter Umständen nicht mit „Durchschnittsagenten“ gearbeitet werden, sondern Daten müssen für jeden simulierten Agenten getrennt erhoben werden und korrekt sein. Theoretisch muss also Daten-Aggregation für jeden einzelnen Agenten getrennt durchgeführt werden. Dies führt im Vergleich zu z.B. einer entsprechenden Makro-Simulation zu einer größeren benötigten Datenmenge, die schwieriger zu erlangen ist und daher häufig nicht zur Verfügung steht.

- Es ist unter Umständen schwieriger, die auf der Mikro-Ebene benötigten Daten bezüglich der realen Agenten des Originalsystems zu erheben. Betrachtet man z.B. die Einkaufssimulation, kann bei reiner Betrachtung der Makro-Ebene z.B. mit Bevölkerungsstatistiken gearbeitet werden. Die Erhebung von Daten zu den konkreten Präferenzen der einzelnen Agenten kann jedoch sehr schwierig sein. Z.B. müssen detaillierte Umfragen unter den Einwohnern im Originalsystem durchgeführt werden, wobei man als Ergebnis nur subjektive und/oder qualitative Daten erhält. Eine individuelle Entsprechung der realen Menschen und der simulierten Agenten ist damit unmöglich.

Können für die ABS-Kalibrierung die Daten bezüglich des Verhaltens auf der Mikro-Ebene nicht in ausreichender Menge und Genauigkeit erhoben werden, dass Aggregierungstechniken sinnvoll eingesetzt werden können, erschwert dies das Kalibrierungsproblem. Die ABS-spezifischen Probleme beim Umgang mit potentiell nicht ausreichend korrekten Daten sind dann:

- Wie kann bei der Kalibrierung der Agentenparameter auf der Mikro-Ebene ermittelt werden, ob die Daten zur Kalibrierung einzelner simulierter Agenten so stark verfälscht sind, dass sie nicht sinnvoll zur Validierung/Kalibrierung eingesetzt werden können.
- Wenn es gelungen ist, entsprechende simulierte Agenten mit fehlerhaften Daten zu identifizieren: Wie kann mit dem Problem umgegangen werden? Ein einfaches Entfernen des betroffenen Agenten aus dem Gesamtmodell kann das resultierende Verhalten ebenso verzerren, wie die Kalibrierung des Modells auf Basis der fehlerhaften Daten.

Existierende Kalibrierungsverfahren bieten für die beiden genannten ABS-spezifischen Probleme bei der Kalibrierung bezüglich potentiell fehlerhafter Daten auf der Mikro-Ebene keine Lösungsmöglichkeiten.

## 1.5 Ziele der Arbeit

Das Ziel dieser Arbeit besteht darin, den Nutzer bei der Kalibrierung von ABS auf der Basis von unzureichenden, potentiell fehlerhaften Daten und Wissen zu unterstützen. Dabei sollen die drei genannten Probleme gelöst werden:

- Vereinfachung der Kalibrierung großer Agenten-Parameter Mengen auf der Mikro-Ebene in Agenten-basierten Simulationen durch Ausnutzung der spezifischen Struktur von ABS (nämlich dem Aufbau aus einer Menge von Agentenmodellen).
- Kalibrierung Agenten-basierter Simulationen, so dass auf allen relevanten Beobachtungsebenen valides Simulationsverhalten erzeugt wird (mindestens Mikro- und Makro-Ebene). Als erschwerende Randbedingung muss die Kalibrierung unter der Voraussetzung einer Makro-Mikro-Wissenslücke durchgeführt werden.
- Kalibrierung Agenten-basierter Simulationen auf der Mikro-Ebene unter der Voraussetzung, dass zur Kalibrierung einzelner Agentenmodelle nicht ausreichend und potentiell verfälschte Daten zur Verhaltensvalidierung zur Verfügung stehen.

Hierzu wird in dieser Arbeit das sogenannte Makro-Mikro-Verfahren zur Kalibrierung von Agenten-basierten Simulationen entwickelt. Das Verfahren besteht aus einem Basisverfahren, das im Verlauf der Arbeit um verschiedene Zusatzverfahren erweitert wird. Das Makro-Mikro-Verfahren und seine Erweiterungen sollen dazu dienen, die Modellkalibrierung trotz stark verrauschter Daten und eingeschränktem Wissen über die Wirkungszusammenhänge im Originalsystem geeignet zu ermöglichen und dabei den Kalibrierungsprozess zu beschleunigen:

- **Makro-Mikro-Kalibrierungsverfahren:** Das in dieser Arbeit entwickelte Makro-Mikro-Verfahren unterstützt den Nutzer durch eine kombinierte Kalibrierung auf der Mikro- und der Makro-Beobachtungsebene, die gegebenenfalls durch Zwischenebenen erweitert werden kann. Der Grundgedanke des Verfahrens besteht darin, das Kalibrierungsproblem in eines auf aggregierter Verhaltensebene und eines auf der Ebene des Mikro-Agentenverhaltens aufzuteilen. Auf der Makro-Ebene wird nach validen idealen aggregierten Verhaltensmodellen (*IVM*) der Agenten gesucht. Auf der Mikro-Ebene wird versucht die individuellen Modelle der Agenten auf Basis des erwünschten Gesamtverhaltens und der ermittelten *IVM* so zu kalibrieren, das insgesamt Simulationsverhalten entsteht, das sowohl auf Mikro- als auch auf Makro-Ebene valide ist. Zunächst wird ein Basis-Makro-Mikro-Verfahren vorgestellt, das dann durch ein Reverse-Engineering der *IVM* verfeinert wird. Das Reverse-Engineering adressiert das Problem, dass aufgrund von unvollständigem Wissen über das simulierte System möglicherweise der Zusammenhang zwischen validem Verhalten auf der Makro-Ebene und benötigten Parameterwerten der individuellen Agentenmodelle unbekannt ist.
- **Erweiterung 1: Robuste Kalibrierung:** Um den Umgang mit potentiell verrauschten Validierungskriterien (d.h. mit verrauschten Daten über ein Originalsystem, auf denen die Validierungskriterien der Simulation beruhen) und Modellteilen während der Kalibrierung von ABS zu ermöglichen, wird eine robuste Kalibrierungstechnik zur Anwendung im Makro-Mikro-Verfahren entwickelt. Die robuste Kalibrierung baut dabei auf die Ergebnisse des Reverse-Engineering auf. Sie dient dazu, durch verrauschte Daten verfälschte Validierungskriterien für Verhaltensweisen der simulierten Agenten oder entsprechende Modellteile zu identifizieren und ihre Verwendung bei der Kalibrierung des Modells auszuschließen, ohne dass die zugehörigen Agenten komplett aus dem Simulationsmodell entfernt werden müssen.
- **Erweiterung 2: Kalibrierung mit Heterogenitätssuche:** Als zweite Erweiterung des Makro-Mikro-Verfahrens wird ein Verfahren entwickelt, das das Problem des unklaren Detaillierungsgrades von ABS auf der Ebene der Parameterwerte adressiert. Prinzipell kann zwar jeder Agent unterschiedliche Parameterwerte verwenden, obwohl eine geringere Heterogenität zur Erzeugung validen Verhaltens ausreichend wäre. Die entwickelte Erweiterung versucht, während der Kalibrierung, eine geeignete Heterogenitätsausprägung für die Parameterwerte der Agenten zu ermitteln. Unter einer Heterogenitätsausprägung wird dabei eine Einteilung der simulierten Agenten in Gruppen mit jeweils gleichen Parameterwerten verstanden. Die Heterogenitätssuche dient dazu, einen Kompromiss zu finden zwischen der Notwendigkeit, sehr große Parametersuchräume durchsuchen zu müssen und gleichzeitig den Suchraum so klein wie möglich halten zu wollen.

- **Werkzeug zur Durchführung der Makro-Mikro-Kalibrierung:**

Die entwickelten Techniken werden (zusammen mit Techniken für Black-Box, White-Box und wissensintensive Kalibrierung mit Hilfe von Regeln) in ein Kalibrierungswerkzeug, den DAVINCI-Kalibrierer, integriert. Das Werkzeug wurde so entworfen, dass es dem Nutzer potentiell die kombinierte Anwendung der Verfahren ermöglicht, jedoch keine Technik zwingend erforderlich macht.

## 1.6 Aufbau der Arbeit

Um diese Ziele zu erreichen, wird in der weiteren Arbeit wie folgt vorgegangen:

- Im ersten Teil der Arbeit (Kapitel 2-6) wird zunächst in die Gebiete der Agenten-basierten Simulation und der Simulationskalibrierung eingeführt und die spezifischen Probleme bei der Kalibrierung Agenten-basierter Simulationen erarbeitet. Im Rahmen dieser Einführung wird der Stand der Technik im Bereich der Simulationskalibrierung sowohl für allgemeine Simulationen als auch speziell hinsichtlich Agenten-basierter Simulationen betrachtet. Ausgehend von dieser Bestandsaufnahme werden zum Abschluss dieses Teils Anforderungen abgeleitet, die ein spezielles Kalibrierungswerkzeug für ABS zur Lösung in identifizierten spezifischen Kalibrierungsprobleme erfüllen muss.
- Im zweiten Teil der Arbeit (Kapitel 7-10) werden die in dieser Arbeit entwickelten Lösungsverfahren zum Umgang mit den identifizierten Problemen bei der ABS-Kalibrierung theoretisch beschrieben. Zunächst wird ein Überblick über das Gesamtverfahren gegeben und dann die einzelnen Lösungsverfahren vorgestellt. Für jedes Verfahren wird jeweils zunächst das allgemeine Vorgehen beschrieben und dann das Verfahren anhand eines Beispiels verdeutlicht. Jeder Teilabschnitt enthält eine Diskussion über Vor- und Nachteile der Verfahren und setzt diese gegebenenfalls in Bezug zu ähnlichen Ansätzen.
- Im dritten Teil der Arbeit (Kapitel 11) wird das entwickelte Kalibrierungswerkzeug für Agenten-basierte Simulationen beschrieben, mit dem die im zweiten Teil dargelegten Lösungsverfahren umgesetzt werden können.
- Im vierten Teil der Arbeit (Kapitel 14-17) werden die entwickelten Lösungsverfahren evaluiert.
- Die Arbeit schließt mit einem Ausblick auf weitere Möglichkeiten zum Umgang mit Problemen bei der ABS-Kalibrierung.



**Teil I**

**Einführung**



## Kapitel 2

# Agenten-basierte Simulation (ABS)

In den folgenden Kapiteln werden die Probleme bei der Kalibrierung von Agenten-basierten Simulationen (auch: Multiagentensimulationen) näher beleuchtet. Um die Komplexität des Problems besser verstehen zu können, wird davor zunächst der Begriff der Multiagentensysteme (MAS) genauer geklärt. Als erstes wird die Frage „Was sind Agenten?“ behandelt, indem charakteristische Eigenschaften beschrieben werden. Als nächstes wird vorgestellt, was man unter Simulation versteht und welche Herangehensweisen es für Simulationsprobleme gibt. In diesem Rahmen wird Agenten-basierte Simulation als spezielles Simulationsparadigma eingeführt.

## 2.1 Verteilte Künstliche Intelligenz

In diesem Abschnitt werden die Begriffe *Agent* und *Multiagentensystem* geklärt. Multiagentensysteme bilden die Grundlage des Simulationsparadigmas *Agenten-basierte Simulation (ABS)*, für das in dieser Arbeit ein Kalibrierungsverfahren entwickelt wird.

### 2.1.1 Agenten

Unter einem Agenten  $A$  kann man eine Entität verstehen, die ihre Umwelt mittels **Sensoren** wahrnimmt und in irgendeiner Form auf diese Umwelt mit Hilfe von **Aktoren** einwirkt [98]. Eine wichtige Eigenschaft von Agenten ist ihre **Autonomie**. Agenten sind in der Lage, aufgrund ihrer Wahrnehmungen und ihrer Erfahrungen selbstständig Aktionen auszuwählen und in ihrer Umwelt durchzuführen.

#### 2.1.1.1 Charakteristische Agenteneigenschaften

Verschiedene Autoren fassen obige allgemeine Beschreibung genauer, indem sie charakteristische Merkmale von Agenten spezifizieren (siehe Ferber [39] und Klügl [76]):

- Agenten befinden sich in einer **Umwelt**  $U$  und sind in der Lage, diese zu einem gewissen Grad wahrzunehmen und in dieser zu handeln. Agenten haben häufig eine (eingeschränkte) **interne Repräsentation ihrer Umwelt**.
- Agenten sind **sozial**, interagieren also mit anderen [76]. Sie können direkt oder indirekt mit anderen kommunizieren [39].

- Eine wichtige Randbedingung, welche die Handlungsfähigkeit von Agenten einschränkt, ist ihre **Lokalität**. Lokalität bedeutet beispielsweise, dass ein Agent nur Wissen über seine unmittelbare Umgebung hat (individuelle Situation des Agenten) und dass er nur diese auch manipulieren kann. Benötigt der Agent Informationen über Systemteile außerhalb seiner Wahrnehmung, muss er versuchen, diese auf andere Weise (z.B. durch Kommunikation mit anderen Agenten) zu erhalten.
- Agenten werden oft als **rationale** Entscheider modelliert. Sie werden dabei von einer Menge von Neigungen angetrieben (z.B. individuelle Ziele), auf die sie hinarbeiten. Ist die Rationalität der Agentenentscheidungen durch bestimmte Randbedingungen beschränkt, spricht man von **bounded rationality**.
- Agenten sind zu einem gewissen Grad und im Rahmen ihres Designs **autonom**, d.h. sie bestimmen ihr Verhalten selbst.
- Agenten haben **eigene Ressourcen und Fähigkeiten**. Sie können diese Fähigkeiten auch anderen anbieten.

Aus den genannten Merkmalen resultieren verschiedene Randbedingungen an die Kalibrierung von ABS, auf die im weiteren Verlauf genauer eingegangen wird.

### 2.1.2 Multiagentensysteme

Bisher wurden nur einzelne Agenten betrachtet. Unter einem Multiagentensystem (MAS) versteht man eine Menge von Agenten in einer (meist räumlichen) Umwelt [39]. Neben den **Agenten** befindet sich eine Menge von **passiven Objekten** in der **Umwelt**. Die Agenten können untereinander und mit den passiven Objekten **interagieren**. Sie können die Objekte verändern, verbrauchen, zerstören oder neue erschaffen.

Weitere häufig geforderte Eigenschaften von MAS sind:

- **Beschränktheit einzelner Agenten:** Jeder einzelne Agent hat nur unvollständige Informationen über seine Umwelt und nur beschränkte Problemlösungsfähigkeiten. Zur Lösung komplexerer Probleme ist also Interaktion und Kooperation der Agenten nötig, um die individuellen Fähigkeiten der Agenten geeignet zu kombinieren.
- **Keine zentrale Kontrolle:** Es gibt keine zentrale Kontrolle des Agentenverhaltens. Dies bedeutet, dass es keine globale Einheit gibt, welche für alle Agenten bestimmt welche Aktionen sie zur Lösung eines Problems in welcher Weise und zu welchem Zeitpunkt durchführen müssen. Jeder Agent bestimmt selbstständig welche Aktionen er wann durchführt und mit welchen Interaktionspartnern er zusammenarbeitet.
- **Kooperativ oder kompetitiv:** Agenten können individuelle Ziele haben, die nicht notwendigerweise kompatibel mit den Zielen anderer Agenten sind und insbesondere nicht kompatibel mit dem globalen Systemziel sein müssen. Als Folge müssen Agenten um Ressourcenobjekte streiten und durch Verhandlungen versuchen, Konflikte zu lösen.
- **Abhängigkeit zwischen Aktionen:** Die Agenten handeln nicht komplett unabhängig voneinander. Die Aktionen der Agenten beeinflussen die übrigen Agenten direkt oder

indirekt. Dies führt dazu, dass die Aktionen eines Agenten in Abhängigkeit von den Aktionen der anderen Agenten im MAS stehen.

Das in dieser Arbeit entwickelte Kalibrierungsverfahren ist für simulierte Multiagentensysteme mit diesen Eigenschaften anwendbar.

## 2.2 Simulation

Das Ziel der Simulation besteht darin, Informationen über ein reales, geplantes oder imaginäres System zu erlangen. Dabei wird versucht, ein solches System geeignet in ein (Computer-)Modell zu übertragen, mit dem Experimente durchgeführt werden. Eine Definition des Begriffs Simulation kann von Fishwick [40] übernommen werden. Fishwick sieht drei zentrale Teilbereiche der Simulation:

*Die Computersimulation ist die Disziplin des Entwurfs eines Modells eines realen oder theoretischen Systems, der Ausführung des Modells auf einem Computer und der Analyse des Simulationsverhaltens.*

Es existieren verschiedene Simulationstypen und Modellierungsparadigmen. Ihr Nutzen und ihre Anwendbarkeit werden durch die Ziele der Simulationsexperimente und die Art und den Detaillierungsgrad der verfügbaren Daten über das zu simulierende System bestimmt.

### 2.2.1 System und Modell

Diese Arbeit befasst sich mit der Kalibrierung des Verhaltens von Simulationsmodellen. Ein (Simulations-)Modell ist eine abstrahierte Abbildung eines Systems. Für den Begriff des Systems gibt es viele mögliche Definitionen. Law und Kelton definieren ein System als eine Menge von Einheiten, die handeln und interagieren, um ein bestimmtes logisches Ende zu erreichen [80].

Charakteristische Eigenschaften von Systemen sind weiter:

- Ein System als Ganzes erfüllt eine gewisse Funktion, die wir als Beobachter in ihm erkennen. Systeme sind in dem Sinne nicht teilbar, als sie bestimmte Elemente und Relationen enthalten, deren Entfernung die Erfüllung des ursprünglichen **Systemzweckes** unmöglich machen würde [18].
- Systeme bestehen aus bestimmten **Teilelementen und Wirkungszusammenhängen** zwischen diesen. Bei der Untersuchung von Systemen sind besonders aus dem Wechselspiel der Elemente resultierende Raum-Zeit-Effekte von Interesse (siehe Fishwick [40] und Bossel [18]).
- Ein System hat **Grenzen** und kann sich in einer **Umwelt** befinden. Die Trennung zwischen verschiedenen Systemen und zwischen System und Umwelt hängt von der jeweiligen Definition der Systemgrenzen ab. Ein System kann sich mit anderen Systemen in einer Umwelt befinden und mit diesen interagieren. Es kann selbst wieder Teil eines größeren Systems sein. Über die Schnittstellen des Systems mit seiner Umwelt kann das System beobachtet und kontrolliert werden [83].

Ein **Modell ist ein System, das ein anderes System abbildet**. Ziel des Umgangs mit einem Modell, ist es dabei Informationen über das abgebildete System zu erlangen [83].

### 2.2.2 Ziele der Simulation

Es gibt zahlreiche Anwendungsgebiete für Simulation. Mögliche Einsatzszenarien für Simulation können Law und Kelton [80] oder Fishwick [40] entnommen werden. Cellier [27] unterscheidet zwischen den Simulationszielen der Vorhersage von Systemverhalten, der Erklärung von Zusammenhängen in einem System und der Kontrolle von Systemverhalten. In leichter Abwandlung zu Cellier wird im Folgenden auf vier wichtige Simulationsziele genauer eingegangen.

- **Simulationen zur Erklärung eines Systems:** Hierunter versteht man die Analyse von existierenden oder vergangenen realen Systemen, mit dem Ziel **Verständnis für die internen Zusammenhänge des Originalsystems** zu erhalten [27], das nicht direkt oder nur mit großem Aufwand (z.B. Zeit oder Geld) untersucht werden kann.
- **Simulationen zur Vorhersage:** In diesen Simulationen besteht das Ziel der Simulationsanalyse darin, aus einem bekannten Ist-Zustand des simulierten Systems, Aussagen über sein zukünftiges Verhalten zu generieren [27]. Ein Beispiel hierfür sind Simulationen zur Wettervorhersage [28].
- **Simulationen zum Entwurf von Systemen:** Das Ziel bei der Verwendung eines Simulationsmodells zum Entwurf eines Systems kann als Kombination der zuvor genannten Simulationsziele betrachtet werden. Bei dieser Anwendung wird versucht ein **geeignetes Modell zu entwerfen**, das ein erwünschtes Systemverhalten in der Simulation erzeugt. Versucht man ein zukünftiges System so zu entwerfen, dass es sich in bestimmten (Ausnahme-) Situationen in einer gewünschten Weise verhält, können mit Hilfe eines zugehörigen Planungsmodells vorab verschiedene mögliche Designentscheidungen hinsichtlich ihrer Brauchbarkeit analysiert werden. Dies geschieht, indem in der Simulation bewertet wird, wie sich das Systemverhalten bei unterschiedlichen Designentscheidungen in verschiedenen Szenarien ändert.
- **Simulationen zur Kontrolle von Systemen:** Sind die Wirkungszusammenhänge eines Systems bekannt und in ein Modell abgebildet und ist es möglich durch Simulation des Modells das Verhalten des Systems auf der Basis verschiedener Eingabewerte zu analysieren, kann Simulation auch zur Kontrolle eines Systems verwendet werden. In diesem Fall wird untersucht, welche Eingabewerte ein System jeweils erhalten muss, damit in verschiedenen möglichen Szenarien ein adäquates Verhalten erreicht werden kann [27].

Das in dieser Arbeit entwickelte Kalibrierungsverfahren ist bei jedem der beschriebenen Simulationsziele einsetzbar.

### 2.2.3 Phasen des Entwicklungsprozesses von Simulationen

Um besser zu verstehen, wo und warum Kalibrierung bei der Entwicklung von Simulationen eine Rolle spielt, wird im Folgenden ein kurzer Überblick über die Phasen des Simulationsentwicklungsprozesses gegeben. Der Überblick ist zunächst informell. Einzelne Bestandteile werden später formalisiert. Die entsprechenden Bezeichnungen werden in der informellen Beschreibung jeweils in Klammer angegeben.

In der Literatur finden sich verschiedene Methodologien zur Entwicklung von Simulationsmodellen und der Durchführung dazu gehöriger Experimente, sowohl für allgemeine Simulationsmodelle, als auch für ABS. Methodologien wurden unter anderem von Gilbert [50] und Law/Kelton [80] vorgestellt. Methodologien spezifisch für ABS finden sich z.B. bei Campos [24], Oechslein [87] und Drogoul [33].

Die verschiedenen Methodologien fokussieren auf unterschiedliche Teilbereiche des Entwicklungsprozesses. Ihnen können aber die wesentlichen Schritte im Entwicklungsprozess entnommen werden:

1. **Festlegung der Simulationsfragen:** Ausgangspunkt des Entwicklungsprozesses ist die Formulierung von Simulationsfragen, zu deren Beantwortung die durchzuführende Simulationsstudie geplant wird [80]. Die erzeugte Simulation muss in der Lage sein die Simulationsfragen zu beantworten.
2. **Konzeptmodell, Spezifikation und Validierung auf Konzeptebene:** In diesem Schritt wird die Struktur  $S$  des zu erstellenden Simulationsmodells  $M$  festgelegt. Basierend auf den formulierten Simulationsfragen wird ein Konzeptmodell, also eine Beschreibung des zu erstellenden Modells, entwickelt. Aus dem Konzeptmodell wird eine direkt implementierbare Spezifikationsbeschreibung für das Simulationsmodell abgeleitet [87]. Unter Umständen sind in der Spezifikation des Modells einige Teile des Modells noch nicht konkret festgelegt. Diese Teile werden durch die Parameter  $P$  des Modells beschrieben. Durch unterschiedliche Festlegung der Werte von  $P$  kann die Spezifikation in verschiedene konkrete Modelle  $M_i$  umgesetzt werden. Alle Modelle  $M_i$  zur Spezifikation, die sich nur in den Werten der  $P$  unterscheiden, werden in dieser Arbeit als zur gleichen Modellfamilie  $MF$  gehörend bezeichnet (genaueres und Definitionen siehe Abschnitt 2.2.4). Die entworfen Modellstrukturen im Konzeptmodell und der Spezifikation müssen auf konzeptioneller Ebene validiert werden. Dies geschieht entweder formal indem die entworfenen strukturellen Wirkungszusammenhänge auf Übereinstimmung mit bekannten Daten oder wissenschaftlichen Ergebnissen geprüft werden oder qualitativ durch Plausibilitätsbewertungen der entworfenen Strukturen. Nur auf Basis eines validen Modellkonzeptes kann später valides Simulationsverhalten erzeugt werden (mehr zum Thema Validierung folgt in Kapitel 3.1.2).
3. **Datenerhebung:** Im Idealfall werden auf Basis der Spezifikation die für die Umsetzung der Spezifikation in das Modell (konkrete Ausprägung von  $MF$  zu  $M$ ), benötigten empirischen Daten bestimmt und erhoben [80]. D.h. die Daten werden idealerweise erst dann erhoben, wenn aufgrund des entworfenen Modellkonzeptes bekannt ist, welche Daten benötigt werden. Werden zu Ermittlung des Modellkonzeptes selbst Daten benötigt, dann muss dieser Teil der Datenerhebung vor der Erstellung des Modellkonzeptes stattfinden. Die erhobenen Daten müssen das Originalsystem ausreichend genau charakterisieren, so dass die Werte der Modellparameter anhand dieser Daten so gewählt werden können, dass das Verhalten von  $M$  dem des Originalsystems mit ausreichender Genauigkeit entspricht.
4. **Implementation und Verifikation:** Im nächsten Schritt wird die entwickelte Modellspezifikation implementiert (d.h.  $MF$  wird erstellt) und die Modellstruktur verifiziert. Unter der **Verifikation eines Simulationsmodells** versteht man die Überprüfung, ob

die entwickelte Modellspezifikation und damit das Konzeptmodell korrekt implementiert wurde.

5. **Kalibrierung und Verhaltensvalidierung:** Während der **Kalibrierung** werden auf Basis der Implementation die Werte der Modellparameter so eingestellt, dass das Modellverhalten valide ist (Konkretisierung von  $MF$  zu  $M$ ). Unter der **Validierung des Simulationsverhaltens** versteht man die Überprüfung, ob das simulierte Verhalten des Modells das Verhalten des abzubildenden Systems bezüglich der Simulationsfragen ausreichend genau wieder gibt und somit geeignet ist, die aufgestellten Simulationsfragen zu beantworten. Die Validierung eines Simulationsmodells stellt häufig ein deutlich größeres Problem dar als dessen Verifikation, da es schwierig sein kann, ein geeignetes Validierungskriterium aufzustellen. Dies ist beispielsweise der Fall, wenn  $M$  das Originalsystem nur sehr abstrakt abbildet oder wenn die verfügbaren empirischen Daten das Originalsystem nicht ausreichend charakterisieren können (mehr dazu in Abschnitt 3.1).
6. **Experimentieren:** Das valide Simulationsmodell ist das Ergebnis des Entwicklungsprozesses. Mit ihm werden Simulationsexperimente durchgeführt, um die Simulationsfragen zu beantworten. Das Experimentieren mit einem Simulationsmodell dient dazu, mittels Simulation diejenigen Daten zu gewinnen, die es erlauben die Fragestellungen des Simulationsproblems zu beantworten. Simulationsexperimente werden durchgeführt, indem der Zustand eines Modellsystems mit seinen Änderungen über einen Zeitverlauf fortgeschrieben wird. Dies geschieht gegebenenfalls auch bei Variation der Eingabedaten der Simulation.

**Diese Arbeit befasst sich mit der Kalibrierung von ABS, d.h. das entwickelte Kalibrierungsverfahren kommt in obiger Beschreibung in der fünften Phase des Simulationsentwicklungsprozesses zum Einsatz.**

#### 2.2.4 Allgemeine Definitionen: Modell und Simulation

Im Folgenden werden einige bereits informell eingeführte Begriffe zur allgemeinen Simulation genauer bestimmt. Die Begriffsbestimmungen wurden so gewählt, dass sie geeignet auf die weiteren Abschnitte der Arbeit hinführen.



---

### Begriffsbestimmung 1: Modellfamilie

In dieser Arbeit besteht eine Modellfamilie  $MF$  aus der Modellstruktur  $S$ , einem Vektor von Modellvariablen  $V$  und einem Vektor von Modellparametern  $P$ . Die Bezeichnung Familie wurde in Anlehnung an den mathematischen Familienbegriff gewählt (z.B. Geradenfamilie). Eine Modellfamilie ist noch nicht simulierbar.

$$MF = (S, V, P)$$

- $\mathbf{V} = \langle V_1, \dots, V_l \rangle, l \in \mathbb{N}$ : Der Vektor der Modellvariablen. Dies sind alle Variablen eines Modells, die nicht zu den Eingabedaten des Modells gehören und deren Werte sich während eines Simulationslaufes verändern können. Sie bilden damit die Menge der Ausgabedaten des Modells.
    - $V_l = (\text{Bezeichner}, W(V_l))$ : Eine der Modellvariablen, beschrieben durch einen Bezeichner (symbolischer Name) und einen Wertebereich  $W(V_l)$ .
    - $W(V_l)$ : Der Wertebereich der Variablen  $V_l$ .
  - $\mathbf{P} = \langle P_1, \dots, P_n \rangle, n \in \mathbb{N}$ : Der Vektor der Modellparameter.
    - $P_i = (\text{Bezeichner}, W(P_i))$ : Einer der Modellparameter, beschrieben durch einen Bezeichner (symbolischer Name) und einen Wertebereich  $W(P_i)$ .
    - $W(P_i)$ : Der Wertebereich des Parameters  $P_i$ .
- 

Eingabewerte eines Modells werden in dieser Arbeit als Teil der Modellstruktur  $S$  betrachtet, da sie sich während eines Simulationslaufes eines zugehörigen Modells nicht ändern. Bei Änderung der Eingabewerte wird also eine geänderte Modellstruktur betrachtet und simuliert.

---

### Begriffsbestimmung 2: Kalibriertes Modell

In dieser Arbeit besteht ein kalibriertes Modell  $M$  (Kalibrierung siehe 3.1) aus einer Modellfamilie und einem Vektor zugehöriger Wertbelegungen  $PW$  für die Modellparameter. Ein kalibriertes Modell ist also eine Modellfamilie, für die eine konkrete Wertbelegung  $PW$  für die Modellparameter gegeben ist. In dieser Arbeit wird davon ausgegangen, dass ein kalibriertes Modell simulierbar ist.

$$M = (MF, PW) = (S, V, P, PW)$$

Zusätzlich sei der Einfachheit halber mit  $PW(M) = PW$  eine Funktion gegeben, die ein Modell auf seine Belegung mit Parameterwerten abbildet. Dabei beschreibt in dieser Arbeit  $X(Y)$  allgemein eine Funktion  $X$ , die auf ein Argument  $Y$  angewandt wurde.

---

Eine Modellfamilie  $MF$  umfasst also die Menge aller möglichen Simulationsmodelle  $M_i$  mit gleichen Modellstrukturen  $S$ , Parameter  $P$  und gleichen Modellvariablen  $V$ :

$$MF = \{M_i | M_i = (S, V, P, PW_i)\}, i \in \mathbb{N}$$

Im weiteren Verlauf wird für ein gegebenes Modell  $M$  durch Anfügen des Buchstaben  $F$  die zu  $M$  gehörige Modellfamilie  $MF$  bezeichnet. Also z.B. würde die zum Modell  $M_{\text{Beispiel}}$  gehörige Modellfamilie mit  $MF_{\text{Beispiel}}$  bezeichnet.

---

### Begriffsbestimmung 3: Modellzustand

Die Menge aller möglichen Zustände eines kalibrierten Modells  $M$  wird bezeichnet durch  $Z_M$ .

$$Z_M = W(V_1) \times \dots \times W(V_i) \times \dots \times W(V_l) \quad i, l \in \mathbb{N}$$

- $W(V_i)$ : Der Wertebereich der  $i$ -ten Modellvariablen.

Einen Zustand  $z$  eines Modells  $M$  erhält man durch Festlegung der Werte für die Modellvariablen  $V$  von  $M$ :  $z \in Z_M$ . Die Variablenwerte in  $z$ , charakterisieren den Modellzustand  $z$ .

---

**In dieser Arbeit werden Simulationsmodelle betrachtet, deren Simulationsverhalten auf der Basis diskreter Zeitschritte ("Ticks") voranschreitet.** Entsprechend ergibt sich die folgende Definition des Simulationsverhaltens.

---

### Begriffsbestimmung 4: Modellzustandstrajektorie

Unter einer Modellzustandstrajektorie  $SV_M$  eines Modells  $M$ , wird in dieser Arbeit eine Trajektorie, d.h. eine zeitlich geordnete Menge an Zuständen, verstanden (siehe z.B. [60]), die das Modell  $M$  während eines Simulationslaufes durchläuft. Ein deterministisches Modell durchläuft beim mehrfachen simulieren immer die gleiche Modellzustandstrajektorie. Ein Modell mit Zufallseinflüssen kann in unterschiedlichen Simulationsläufen unterschiedliche Modellzustandstrajektorien durchlaufen.

$$SV_M = (z_0, z_1, \dots, z_i, \dots, z_t), \quad i, t \in \mathbb{N} \text{ mit } z_i, z_t \in Z_M$$

Grundsätzlich ist die Länge  $t$  der Trajektorie nicht beschränkt. In dieser Arbeit werden jedoch nur terminierende Simulationsläufe betrachtet.  $z_i$  beschreibt den Zustand des Modells  $M$  zum  $i$ -ten Zeitschritt eines Simulationslaufes.  $SV$  beschreibt die Menge aller möglichen Modellzustandstrajektorien.

---

---

**Begriffsbestimmung 5: Simulation eines Modells**

Simulation wird als Funktion auf einem Modell verstanden, die zu einem Modell  $M$  die (im stochastischen Fall „eine“) Modellzustandstrajektorie  $SV_M$  des Modells über einen Zeitraum von  $t \in \mathbb{N}$  Zeitschritten generiert. Sei  $\mathcal{M}$  die Menge aller Modelle  $M$ .

$$Sim : \mathcal{M} \times \mathbb{N} \rightarrow SV$$

Für die Untersuchung des Simulationsverhaltens ist unter Umständen nur ein Teil der erzeugten Modellzustände relevant. Dies ist beispielweise der Fall, wenn die Simulation eine gewisse Einschwingphase hat. Irrelevant sind auch die möglichen Zustände, die nie erreicht werden.

---

Alternative Definitionsmöglichkeiten können z.B. bei Zeigler [123] oder Hofmann [60] entnommen werden.

**2.2.4.1 Beispiel: Modell, Modellfamilie, Struktur und Variablen**

Abbildung 2.1 zeigt ein sehr einfaches Beispiel für eine Modellfamilie. Die Modellfamilie bestehe aus einer Menge von quadratischen polynomiellen Modellen zur Beschreibung des zeitlichen Verlaufs von Datenreihen mit Hilfe einer Formel. Die Abbildung zeigt die allgemeine Form dieser Modellfamilie. Die Modellfamilie hat eine Modellvariable  $f(t)$ , drei Modellparameter  $a$ ,  $b$  und  $c$  und eine vorgegebene Modellstruktur beschrieben durch die mathematischen Operatoren  $=$ ,  $*$ ,  $+$  und  $^2$ . Die Ausprägung der Modellfamilie zu einem konkreten Modell erfolgt, indem die Werte der Modellparameter  $a$ ,  $b$  und  $c$  festgelegt werden. Durch Festlegung von  $t$  wird definiert der wievielte Zustand innerhalb der Modellzustandstrajektorie eines Simulationslaufes zu einem Modell der Modellfamilie betrachtet werden soll.

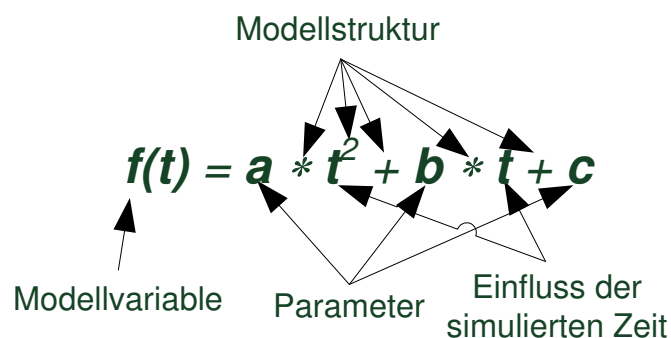


Abbildung 2.1: Struktur und Parameter eines Formelmodells.

Alle Modellbestandteile können entweder Teil der Modellstruktur oder der Modellparameter sein. Wäre die Modellstruktur im Beispiel nicht auf ein quadratisches Polynom festgelegt worden, wären auch die Werte der Exponenten variabel und damit Parameter. D.h. in einem Modell gibt es Elemente, die festgelegt sind, aber die Rolle von Parameter spielen könnten. Sind diese Werte z.B. durch vorhandene Daten festgelegt worden, dann werden sie in dieser Arbeit als Teil der Modellstruktur betrachtet.

In dieser Arbeit werden nur die während der Kalibrierung eines Modells (siehe Kapitel 3.1) veränderbaren Modellteile als Parameter bezeichnet.

### 2.2.5 Eigenschaften von Simulationsmodellen

Simulationsmodelle lassen sich über verschiedene Eigenschaften charakterisieren, welche während der Erstellung des Konzeptmodells festgelegt werden müssen. Ein bestimmender Faktor für diese Eigenschaften ist das für die Simulationsstudie gewählte Simulationsparadigma (siehe Abschnitt 2.3). Entsprechende Eigenschaften können von Law und Kelton [80] übernommen werden:

- **Statisch oder Dynamisch:** Ein statisches Modell ist die Repräsentation eines Referenzsystems zu einem bestimmten Zeitpunkt. Veränderungen des Zustandes eines Referenzsystems (siehe Definition 3) über die Zeit werden in dynamischen Modellen abgebildet. Dynamische Modelle sind schwieriger zu entwerfen, da unter Umständen viele Systemabhängigkeiten beachtet werden müssen, um den gewünschten Gesamtverlauf des Systemverhaltens über die Zeit zu erzeugen.
- **Deterministisch oder Stochastisch:** Enthält ein Simulationsmodell keine Zufallselemente nennt man es deterministisch, sonst stochastisch. Das Verhalten deterministischer Simulationsmodelle kann leichter untersucht werden als das stochastischer, da das Ergebnis jedes Simulationslaufes identisch ist. Im stochastischen Fall sind häufig mehrere Simulationsläufe notwendig, bevor das Verhalten des Modells sinnvoll beurteilt werden kann.
- **Kontinuierlich oder Diskret:** Simulationsmodelle können zeitlich diskret oder kontinuierlich sein. In diskreten Simulationsmodellen werden Zustandsänderungen der Simulation nur zu bestimmten Zeitpunkten betrachtet; in kontinuierlichen Modellen hingegen zu potentiell allen Zeitpunkten. Für kontinuierliche Simulationsmodelle werden meist Formelmodelle (Differential-Gleichungsmodelle) eingesetzt [80].

Ein zu modellierendes reales System ist häufig sehr komplex, d.h. ein zugehöriges Simulationsmodell muss dynamisch, stochastisch und kontinuierlich sein. Wird zur Erfüllung der Simulationsziele nicht die vollständige Komplexität des Originalsystems benötigt, so können diese Eigenschaften bei der Abbildung in ein Simulationsmodell vereinfacht werden. **In dieser Arbeit werden stochastische Simulationsmodelle mit diskreten Zustandsänderungen in mehreren Zeitschritten betrachtet.** Die simulierten Systeme können statisch oder dynamisch sein, wobei ein statisches System einfach ein dynamisches System mit nur einem Zustand in der Trajektorie ist.

## 2.3 Simulationsparadigmen

Es gibt verschiedene etablierte Arten von Simulationsparadigmen. Ihre Eignung für ein bestimmtes Simulationsproblem hängt von den konkreten Fragestellungen ab, die durch Simulation beantwortet werden sollen, dem dazu benötigten Aggregierungs- bzw. Abstraktionslevel für das Simulationsmodell und der Datenlage für die Modellerstellung. In dieser Arbeit wird die Kalibrierung von ABS adressiert. Im Folgenden wird das Paradigma der ABS als Mikroskopisches Simulationsparadigma im Vergleich zur Makrosimulationen eingeführt.

### 2.3.1 Makroskopische Simulation

Makroskopische Simulationsmodelle betrachten das simulierte System als Einheit, ohne die konkreten einzelnen Bestandteile des Systems näher zu betrachten. Der Zustand des Modells wird durch eine Menge von Zustandsvariablen beschrieben, deren Dynamik mit Hilfe von Formeln spezifiziert wird. Ziel des Makromodells ist es das Systemverhalten auf einer sehr aggregierten Beobachtungsebene (**Globale Beobachtungsebene**) mit Hilfe eines Formelmodells korrekt zu beschreiben. Durch Analyse des Formelmodells wird versucht die interessanten Simulationsfragen zu beantworten [80]. Meistens werden hierfür Differenzen- oder Differentialgleichungsmodelle eingesetzt. Die bekannteste Methodologie für die Makrosimulation in den Sozialwissenschaften ist *System Dynamics* [50].

### 2.3.2 Mikroskopische Simulation

Mikromodelle bilden das Gegenstück zu makroskopischen Modellen [76]. Während bei Makromodellen das Gesamtverhalten des simulierten Systems beschrieben wird, werden bei Mikromodellen die betrachteten Einheiten des Systems explizit voneinander getrennt und durch ihre individuellen Eigenschaften, Dynamik und Beziehungen untereinander beschrieben. Das Verhalten des Gesamtsystems wird dabei nicht, wie bei Makro-Modellen, explizit beschrieben, sondern ergibt sich aus dem Zusammenspiel der einzelnen Systemeinheiten. ABS ist eine Ausprägung des mikroskopischen Simulationsparadigmas.

Die beiden Extreme der Makro- und der Mikrosimulation können auch in einem Simulationsparadigma vereint werden. Bei der Multi-Level Simulation werden verschiedene Aggregierungsebenen des simulierten Systems gleichzeitig im Modell repräsentiert und simuliert. Beziehungen und Abhängigkeiten zwischen den verschiedenen Simulationsebenen werden explizit modelliert. Aggregiertere Simulationsebenen können hierbei weniger stark aggregierte beeinflussen und umgekehrt. Uhrmacher et al. [119] beschreiben z.B. eine Multi-Level-Simulation, in der Enzyme in einer Probenlösung simuliert werden.

## 2.4 Agenten-basierte Simulation (ABS)

Im letzten Abschnitt wurde ABS als Ausprägung des mikroskopischen Simulationsparadigmas eingeführt. Eine ABS ist die Simulation eines Agenten-basierten Modells. In einem Agenten-basierten Modell verwendet man eine Konzeptualisierung des Original- oder Referenzsystems als MAS. Dabei können die Agenten als aktive autonome Einheiten betrachtet werden, die in einer Umwelt situiert und persistent sind. Die Agenten können in dieser Umwelt eingeschränkte Wahrnehmung und nur lokale Interaktionsmöglichkeiten haben [79]. Aufgrund seines modellierten Verhaltens und seiner individuellen lokalen Wahrnehmung bestimmt jeder Agent seine nächsten Aktionen. Werden durch die simulierten Agenten konkrete Individuen eines Systems repräsentiert, etwa Fußgänger oder Ameisen, spricht man auch von individuenbasierten Modellen.

Nach Bonabeau [16] können die Vorteile von ABS gegenüber anderen Paradigmen in drei Aussagen zusammengefasst werden:

- ABS erlauben es emergente Phänomene abzubilden.
- ABS stellt eine natürliche Form zur Beschreibung von Systemen dar.

- Das ABS-Paradigma bietet dem Nutzer ein hohes Maß an Flexibilität beim Entwurf von Simulationsmodellen. Es ist möglich, Systeme auf einem sehr hohen Detaillierungsgrad abzubilden. Außerdem können mit ABS die anderen Simulationsparadigmen nachgebildet werden.

Der hohe mögliche Detaillierungsgrad und die Flexibilität beim Entwurf von ABS stellen einen Nutzer aber auch vor zusätzliche Herausforderungen. Ein Kernpunkt nach Bonabeau [16] ist, dass ABS zwar eine technisch einfache Möglichkeit zur Modellierung von Systemen bieten, aber der Entwurf einer ABS konzeptuell sehr schwierig sein kann. Dies ist insbesondere der Fall, wenn versucht wird interne Entscheidungsprozesse von Lebewesen realistisch abzubilden, über die nur sehr schwierig genaue Informationen ermittelt werden können. Ein Beispiel hierfür, auf das im weiteren Verlauf der Arbeit noch konkret eingegangen wird, sind menschliche Entscheidungen beim Einkaufen. Verschiedene Menschen können verschiedene Gründe haben, warum sie bestimmte Geschäfte besuchen und sich einen Artikel lieber in einem Geschäft *A* als in einem Geschäft *B* kaufen. Diese Gründe können den Menschen selbst bewusst oder unbewusst sein. Möchte man z.B. durch Befragungen von Einkäufern diese Gründe genauer ermitteln steht man vor dem Problem, dass man als Antworten möglicherweise ungenaue und unvollständige Informationen erhält, dies aber nur schwer überprüfen kann. Es kann daher schwierig sein aus den erhaltenen Antworten Rückschlüsse auf interne Entscheidungsprozesse von Menschen beim Einkaufen zu ziehen.

**Da der Entwurf detaillierter ABS besondere Herausforderungen an den Entwickler stellt, konzentrieren sich die in dieser Arbeit entwickelten Techniken auf das Paradigma der ABS.**

### 2.4.1 Einige Anwendungsgebiete von ABS

Wichtige Anwendungsgebiete für ABS liegen in den Gesellschaftswissenschaften [50], der Analyse biologischer Systeme [50], militärischen Simulationen [26], Verkehrssimulationen [1] und Wirtschaftssimulationen [2]. Die Unterschiedlichkeit der einzelnen Anwendungsgebiete zeigt, wie unterschiedlich die Anforderungen an das Agentenverhalten und damit an die internen Verhaltensstrukturen der Agenten sind. Beispiele für Verhaltensweisen, die als relativ gut verstanden betrachtet werden können, sind z.B. die von Ameisen. Bezüglich des Verhaltens von Ameisen existieren zumindest Hypothesen, dass sie nur auf bestimmte Stimuli, wie Pheromone reagieren. Dagegen stellt der Entwurf eines geeigneten Verhaltensmodells für Autofahrer und einkaufende Menschen eine wesentlich größere Herausforderung dar.

### 2.4.2 Definitionen: Agenten-basiertes Modell und Simulation

---

#### Begriffsbestimmung 6: Agenten-basiertes Modell

Ein Agenten-basiertes Modell (ABM) ist ein Modell. Ein ABM besteht aus einem Umweltmodell  $UM$  und der Menge der Agentenmodelle  $AM$ . Im weiteren Verlauf werden unter „Agenten“ die einzelnen simulierbaren Agentenmodelle verstanden.

$$ABM = (UM, AM)$$

- $UM = (S_{UM}, V_{UM}, P_{UM}, PW_{UM})$ : Das Umweltmodell.
  - $AM = \{IAM_1, \dots, IAM_a\} : a \in \mathbb{N}$ : Die Menge der  $a$  Agentenmodelle.
  - $IAM_i = (S_i, V_i, P_i, PW_i) : i \in \mathbb{N}$ : Das Modell eines einzelnen Agenten.
- 

Eine Familie Agenten-basierter Modelle wird analog zur Definition einer Modellfamilie charakterisiert. Sie besteht jeweils aus einer Modellfamilie  $UMF$  für das Umweltmodell  $UM$  und einer Modellfamilie  $AMF$ , in der die Modellfamilien  $IAMF$  zu den einzelnen Agentenmodellen zusammenfasst sind.

Ein wichtiger Bestandteil von ABM sind soziale Interaktionen der Agenten mit ihrer Umwelt und anderen Agenten [49]. Interaktionsmöglichkeiten können in einem ABM durch die Festlegung der Modellstrukturen von Umweltmodell  $S_{UM}$  und Agentenmodellen  $S_i$  definiert werden. Durch Kalibrierung der zugehörigen Parameterwerte können solche Interaktionsstrukturen des Modells geeignet eingestellt werden. In dieser Arbeit werden Interaktionsstrukturen als Teile der Agenten- und Umweltmodellstrukturen betrachtet. Sie sind daher implizit in Definition 6 enthalten.

### 2.4.3 Umweltmodell in ABS

Das Umweltmodell bildet die Grundlage jeder ABS. Es ist eine Abstraktion der Umwelt des Originalsystems, in der die simulierten Agenten „leben“ [78]. Es bildet den Rahmen für den Detailgrad und das Abstraktionslevel des gesamten Modells. Das Umweltmodell enthält Modelle aller relevanten passiven Elemente (keine Agenten) der Originalumwelt. Zusätzlich kann das Umweltmodell selbst als eine Art Welteinheit im Modell vertreten sein. Eine solche Welteinheit kann einen eigenen Zustand (repräsentiert durch Variablenwerte) und ein Modell für die dynamische Änderung dieses Zustandes besitzen. Ein Beispiel hierfür ist ein Umweltmodell, das verschiedene Jahreszeiten durchläuft und dabei die Temperaturveränderungen steuert. Weiter kann das Umweltmodell „Generatorprozesse“ für das Entstehen von Entitäten im Modell (auch Agenten) enthalten. Es gibt verschiedene Konzepte, um das dynamische Verhalten einer Umwelt, ihre Einflüsse auf das Agentenverhalten und den Einfluss der Agenten auf das Umweltverhalten zu beschreiben (siehe z.B. Helleboogh [57]). Durch das Umweltmodell können verschiedene Randbedingungen an das Verhalten

der simulierten Agenten gegeben sein, die das in dieser Arbeit behandelte Kalibrierungsproblem beeinflussen. Ein besonders wichtiger Aspekt ist, dass das Umweltmodell die **Lokalität** der Agenten bestimmen kann [49]. Beispielsweise kann durch ein räumliches Umweltmodell festgelegt sein, was die Agenten von ihrer aktuellen Position wahrnehmen können und mit welchen anderen Agenten sie interagieren können. Wurden im Umweltmodell die Interaktionsstrukturen des Modells festgelegt (z.B. ein Kommunikationsnetzwerk) so sind durch Festlegung des Umweltmodells Randbedingungen an das Verhalten der Agentenmodelle während eines Simulationslaufes vorgegeben.

#### 2.4.4 Simulierte Agenten

Die simulierten Agenten sind die zentralen Einheiten von ABS. Ein Agentenmodell bildet die individuellen autonomen Verhaltensweisen und Zustände der Agenten des Originalsystems ab. Die simulierten Agenten agieren im simulierten Umweltmodell. Die möglichen Eigenschaften simulierter Agenten wurden in Abschnitt 2.1.1 beschrieben. Im Folgenden wird auf besondere Herausforderungen des in dieser Arbeit adressierten Kalibrierungsproblems für ABS hingewiesen.

##### 2.4.4.1 Komplexe Modellierung: Agentenmodelle

Die Festlegung von Modellstrukturen für ein ABM kann ein großes Problem bei der Entwicklung von ABS darstellen. Der ABM-Entwurf wird abhängig von der Simulationsfragestellung auf einem recht hohen Detaillierungsgrad durchgeführt. Beispielsweise wird versucht, zur Simulation eines Bienenstaates ein ABM auf der Basis einzelner Bienenmodelle zu entwickeln [31]. Bei der Entwicklung detaillierter ABM besteht die Gefahr, dass keine ausreichenden Daten, Wissen, Theorien oder konkrete Hypothesen zur exakten Festlegung der Modellstrukturen verfügbar sind. Sind zumindest Theorien über die benötigten Modellstrukturen vorhanden, können diese im Modell umgesetzt werden. Unsicherheiten bei der Festlegung von Modellstrukturen können entstehen, wenn zwischen verschiedenen konkurrierenden Theorien ausgewählt werden muss. Ist das resultierende Simulationsverhalten des ABM nicht valide, kann es schwierig sein einzuschätzen, ob die umgesetzte Theorie ungeeignet war oder ob beispielsweise falsche Parameterwerte, ungenaue Ausgangsdaten oder fehlende bisher nicht modellierte Modellstrukturen das Verhalten negativ beeinflusst haben. Der ungünstigste Fall bei der Modellierung von Agentenverhalten ist gegeben, wenn der Entwurf ohne konkretes Vorwissen, Daten oder Theorien durchgeführt werden muss. Dann hat man meist kein Validitätskriterium, sondern muss auf abstrakter Ebene arbeiten. In diesem Fall kann es sehr schwer sein, aus beobachtetem unerwünschtem Simulationsverhalten Rückschlüsse über benötigte Modelländerungen zu ziehen. Das Problem der Festlegung geeigneter Modellstrukturen für ein ABM gehört nicht zum zentralen Fokus dieser Arbeit, der auf die Kalibrierung von Parameterwerten ausgerichtet ist. In Kapitel 11 wird kurz auf Möglichkeiten zum Umgang mit diesem Problem eingegangen.

##### 2.4.4.2 Komplexes Verhalten: Modelliertes Verhalten vs. beobachtetes Simulationsverhalten

Die Modellierung eines ABM erfolgt auf der Detaillierungsebene, auf der die zu simulierenden Einheiten des Originalsystems beobachtet wurden, der sogenannten **Mikro-Ebene**, d.h.



es gilt **Beobachtungseinheit = Modellierungseinheit**. Die Mikroebene ist die Ebene des lokalen Agentenverhaltens. Auf der Basis eines Simulationslaufes werden das Zusammenspiel der Agenten und das sich daraus ergebende Gesamtverhalten der Simulation analysiert. Abbildung 2.2 veranschaulicht die Entstehung des Simulationsverhaltens.

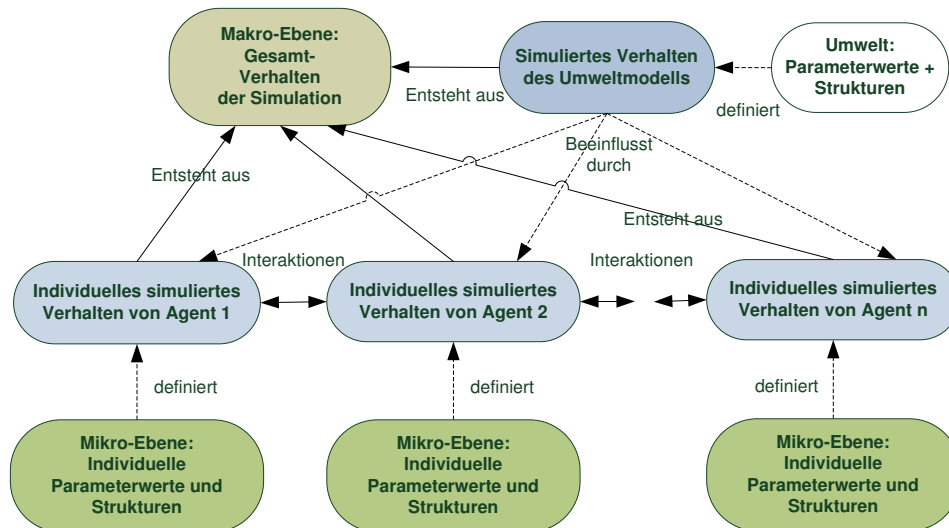


Abbildung 2.2: Entstehen von Makro- aus Mikroverhalten in ABS.

Die **Makro-Ebene** ist die Ebene des aggregierten Gesamtverhaltens. Sie wird im Allgemeinen nicht explizit im Modell repräsentiert. Die Agenten können so modelliert sein, dass sie während der Simulation **individuelle Ziele** verfolgen, die nicht notwendigerweise valides Gesamtverhalten der Simulation auf Makroebene produzieren. Trotzdem soll sich valides Gesamtverhalten aus dem Zusammenspiel der einzelnen Agenten ergeben.

### Mikroverhalten vs. Makroverhalten: Emergenz

Bei der Erstellung von ABS kann es sein, dass der konkrete Zusammenhang zwischen Makro- und Mikro-Ebene nicht oder nicht genau genug bekannt ist. Dies führt dazu, dass nicht direkt aus dem am Originalsystem beobachteten Gesamtverhalten auf die zu modellierenden Modellstrukturen geschlossen werden kann. Dies ist insbesondere bei der **Untersuchung emergenter Phänomene** der Fall, die einen wichtigen Anwendungsfall für ABS darstellt [61, 50]. Unter Emergenz versteht man das Zustandekommen eines Phänomens auf Makro-Beobachtungsebene, das durch das Zusammenwirken der einzelnen Systembestandteile möglich wird, ohne dass die einzelnen Systembestandteile für sich geplant auf das Verhaltensphänomen hinwirken. **Ein Beispiel für ein emergentes Phänomen** ist z.B. das Zusammenwirken menschlicher Gehirnzellen. Der Mensch ist zu komplexen Denkprozessen fähig, die aber auf dem Zusammenspiel vieler einzelner Neuronen beruhen, die jedes für sich nicht zu solchen Denkprozessen fähig sind. Der Hauptbezug zwischen ABS und Emergenz besteht darin, dass durch die Simulation emergente Phänomene untersucht werden sollen. Nach Darley [29] sind ABS für solche Untersuchungen das einzige sinnvolle Mittel.

### Validierung von ABS

Aus dem letzten Abschnitt geht hervor, dass die Untersuchung des Verhaltens einer ABS auf **mindestens zwei möglichen Beobachtungsebenen** durchgeführt werden kann: Der Mikro- und der Makroebene. Zur Laufzeit der Simulation wird auf der Mikro-Ebene beobachtet, welche konkreten individuellen Aktionen die Agenten durchführen, auf höherer Aggregationsebene, welche Interaktionen aus dem modellierten Agentenverhalten resultieren und auf der Makroebene welches Simulationsgesamtverhalten erzeugt wird. Entsprechend muss die **Validierung des Verhaltens einer ABS mindestens auf Makro- und Mikro-Ebene** erfolgen. Das Gesamtverhalten der Simulation muss valide sein und aus validem lokalem Verhalten der einzelnen Agenten entstehen. In Kapitel 4.1 wird der Aspekt der Validierung von ABS wieder aufgegriffen.

#### 2.4.4.3 Heterogenität der modellierten Agenten

Abhängig von den Fragestellungen, die durch Simulation geklärt werden sollen, ist es oft unklar, wie gleich die einzelnen Agenten entworfen werden müssen. Der Nutzer muss bei der Modellierung des individuellen Agentenverhaltens gegebenenfalls festlegen, welche Agenten sich gleich verhalten sollten und welche Agenten unterschiedliches Verhalten benötigen, um ein valides Gesamt-Simulationsverhalten zu erzeugen. Mögliche Festlegungsfaktoren können z.B. gleiche oder unterschiedliche individuelle Modellstrukturen oder Parameterwerte der Agenten sein. Ein System mit Agenten, die gleiche Strukturen und gleiche Parameterwerte haben, wird als **homogen**, ein System mit unterschiedlichen Agenten als **heterogen** bezeichnet. Heterogenität kann die Struktur der Agenten, ihre Parameterwerte oder z.B. das Verhalten der Agenten in der Simulation betreffen.

### Klasse von Agenten einer Modellfamilie

Unter einer Klasse  $AMK$  von Agentenmodellen werden alle Agentenmodelle  $IAM$  eines Modells  $ABM = (UM, AM)$  verstanden, die zu einer Modellfamilie gehören. Die Begriffe Agentenmodellfamilie und Agentenmodellklasse erscheinen ähnlich. Eine Agentenmodellfamilie  $IAMF_i$  umfasst alle möglichen Ausprägungen eines einzelnen Agentenmodells bezüglich seiner Parameterwerte. Eine Klasse von Agentenmodellen umfasst alle einzelnen Agentenmodelle von  $ABM$ , welche identische  $IAMF$  haben.

---

#### **Begriffsbestimmung 7: Klasse von Agentenmodellen**

*Eine Klasse  $AMK$  von Agentenmodellen wird definiert als Teilmenge aus allen Modellen eines  $ABM$ , in der alle Agenten die gleiche Agentenmodellfamilie haben.*

$$AMK \subseteq AM \text{ mit } \forall IAM_i, IAM_j \in AMK : IAMF_i = IAMF_j$$


---

### Gruppen von Agenten

Agenten, die zu einer Klasse gehören, haben gleiche Modellstrukturen. Die zugehörigen individuellen Agentenmodelle können sich aber immer noch in den jeweiligen Parameterwerten unterscheiden. Agentenmodelle einer Klasse mit homogenen Parameterwerten werden in dieser Arbeit als Gruppe von Agentenmodellen bezeichnet.

---

#### **Begriffsbestimmung 8: Gruppe von Agentenmodellen einer Klasse**

*Eine Gruppe AMG von Agentenmodellen wird definiert als Teilmenge aus allen Modellen einer Klasse von Agentenmodellen, in der alle Agenten die gleichen Parameterwerte haben.*

$$AMG \subseteq AMK, \text{ mit } \forall IAM_i, IAM_j \in AMG : PW(IAM_i) = PW(IAM_j), i, j \in \mathbb{N}, i \neq j$$

*Durch Gruppierung wird die Menge aller Agenten in einer Klasse von Agentenmodellen vollständig partitioniert.*

---

Diese Gruppdefinition bedeutet keine Festlegung der Parameterwerte auf konkrete Werte, sondern lediglich, dass bei einer Festlegung die Agentenmodelle  $IAM_i$  und  $IAM_j$  beide die gleichen Parameterwerte erhalten.

### Parameterheterogenität und Agentenverhalten

Während eines Simulationslaufes eines ABM kann sich das beobachtbare Verhalten der einzelnen simulierten Agenten aus verschiedenen Gründen voneinander unterscheiden. Neben offensichtlichen Gründen, wie unterschiedliche individuelle Strukturmodelle oder Parameterwerte, können auch simulierte Agenten, deren Modelle zu einer AMG gehören, unterschiedliches Verhalten aufweisen. Der Grund hierfür liegt in jeweils unterschiedlichen **individuellen Situationen**, in denen sich die einzelnen Agenten befinden. Unter der individuellen Situation eines Agenten werden die Zustände derjenigen anderen Modellteile verstanden, die aktuell für das Verhalten des Agenten relevant sind. Zur individuellen Situation eines Agenten gehört der Zustand eines Teils der simulierten Umwelt und die Zustände eines Teils der anderen simulierten Agenten. Die individuelle Situation eines simulierten Agenten ergibt sich daraus, dass die einzelnen Agenten meist nur einen begrenzten Interaktions- und Wahrnehmungsradius haben (vergleiche 2.1.1), innerhalb der sie ihre Umwelt und andere Agenten wahrnehmen und beeinflussen können (**Lokalität der Agenten**). Unterschiedliche individuelle Situationen (während eines Simulationslaufes) simulierter Agenten deren Modelle zu einer AMG gehören können die Wahl heterogener Parameterwerte für die zugehörigen IAM erforderlich machen.



## Kapitel 3

# Allgemeine Kalibrierung und Stand der Forschung

Dieses Kapitel gibt eine allgemeine Einführung in das Problem der Simulationskalibrierung. Das Kapitel ist in zwei Teile gegliedert. Im ersten Teil wird das Kalibrierungsproblem und zugehörige Lösungsansätze vorgestellt. Im zweiten Teil wird näher auf den aktuellen Stand der Forschung im Bereich der allgemeinen Simulationskalibrierung eingegangen. In Kapitel 4 wird dann ausführlich auf die Spezialisierung des Kalibrierungsproblems für Agentenbasierte Simulationen eingegangen.

### 3.1 Kalibrierung von Simulationen

In Kapitel 2.2.3 wurde Kalibrierung als fünfte Phase des Entwicklungsprozesses von Simulationen vorgestellt. Die Kalibrierung eines Simulationsmodells wird demnach durchgeführt nachdem das Modell implementiert und seine strukturelle Korrektheit festgestellt wurde. Bevor mit dem Modell experimentiert werden kann müssen noch die Parameterwerte des Modells so eingestellt werden, dass das Simulationsverhalten des Modells dem des Referenzsystems der Simulationsstudie entspricht.

In diesem Abschnitt wird das Problem der Kalibrierung von Simulationen allgemein eingeführt. Zunächst wird das Problem anhand eines Beispiels veranschaulicht, die Ziele des Kalibrierungsprozesses allgemein und dann das Kalibrierungsproblem formal beschrieben. Aufbauend auf dieser Einführung wird eine generelle Beschreibung des Vorgehens zur Lösung von Kalibrierungsproblemen gegeben und Möglichkeiten zur Automatisierung des Lösungsvorgehens aufgezeigt. Im dritten Teil wird auf den Zusammenhang zwischen Modellierung und Kalibrierung eingegangen.

#### 3.1.1 Einführendes Beispiele Einkaufssimulation

Zunächst soll ein Beispiel betrachtet werden. Das Beispiel demonstriert den Kalibrierungsbedarf anhand einer konkreten ABS. Es soll untersucht werden, welche Stadtteile einer gegebenen Stadt am geeignetsten für den Bau eines neuen Supermarktes sind und welche Größe und Sortimentseigenschaften dieser Supermarkt im Vergleich zu bereits existierenden Konkurrenzgeschäften haben sollte. Hierzu soll eine ABS entworfen werden in der

möglichst valide simuliert wird, wie die Einwohner der Stadt in den verschiedenen existierenden Geschäften der Stadt ihre Lebensmitteleinkäufe erledigen. Beim Experimentieren mit diesem Modell soll an unterschiedlichen Positionen des Stadtmodells ein neuer Supermarkt mit bestimmten Eigenschaften positioniert werden. Dann soll anhand von Simulationsläufen beobachtet werden, wie sich das Einkaufsverhalten der simulierten Einwohner ändert. Basierend auf diesen Beobachtungen soll nach einer optimalen Lage und Sortiment für den neuen Supermarkt gesucht werden.

Bevor der eigentliche Modellierungsprozess beginnt, werden Daten aus der realen Stadt erhoben. Hierzu werden für alle Haushalte der Stadt ihre Kaufkraft, räumliche Lage und Einkaufspräferenzen ermittelt. Weiter werden für die Geschäfte der Stadt die räumliche Lage, Sortimentseigenschaften und ihre tatsächlichen erzielten Umsätze festgestellt. Das Einkaufsverhalten der simulierten Einkaufsagenten soll nun so entworfen werden, dass das Modell während eines Simulationslaufes die erhobenen Daten zu Geschäftsumsätzen und Einkaufspräferenzen der Einwohner möglichst genau reproduzieren kann. Die Modellstruktur der Einkaufsagenten wird so entworfen, dass jeder Einkaufsagent einen Haushalt der Stadt repräsentiert. Einkaufsagenten können die Supermarktmodelle in einem (noch festzulegenden) Umkreis um die Position des zugehörigen Haushaltes wahrnehmen. Weiter werden sie so entworfen, dass sie eine Reihe von Attributen der Geschäfte, wie deren Preis, Sortiment oder die Qualität der Produkte, wahrnehmen können. Jeder Agent bewertet die wahrgenommenen Geschäfte indem er eine gewichtete Summe über seine Entfernung zu den Geschäften und ihre Sortimentseigenschaften bildet. Das Ergebnis der Summe ist die Präferenz des Agenten für das jeweilige Supermarktmodell. Basierend auf diesen Präferenzwerten gehen die Einkaufsagenten während eines Simulationslaufes in den Geschäften einkaufen. Durch Verteilung der Kaufkraft der Agenten auf die simulierten Geschäfte entstehen Umsätze der Supermärkte, die am Ende des Simulationslaufes mit den Umsätzen des Referenzsystems übereinstimmen sollen.

Die Frage, die nach der Festlegung der Modellstrukturen geklärt werden muss, ist, in welcher Weise die Einkaufsagenten die wahrgenommenen Geschäftsattribute zueinander gewichten müssen, damit der simulierte Geschäftsumsatz dem des Referenzsystems entspricht. Gleichzeitig müssen die während eines Simulationslaufes beobachtbaren Einkäufe der Agenten den ermittelten Einkaufspräferenzen der Haushalte der Referenzstadt entsprechen. Hierzu müssen die Parameterwerte der einzelnen Bewertungssumme der Einkaufsagenten eingestellt werden. Die Festlegung dieser Parameterwerte erfolgt während der Simulationskalibrierung.

### 3.1.2 Das Ziel der Kalibrierung: Validität des Simulationsverhaltens

Das Ziel der Simulationskalibrierung besteht darin, die Parameter eines Simulationsmodells so einzustellen, dass **ausreichend valides Modellverhalten** erzeugt wird, so dass mit dem resultierenden Modell die relevanten Simulationsfragen beantworten werden können.

#### 3.1.2.1 Definition: Modellvalidierung

Nach Sargent [103] wird die Modellvalidierung als

*Nachweis, dass ein Computermodell - im Rahmen seines möglichen Anwendungsbereiches - bezogen auf seinen Einsatzzweck, hinreichend genau ist*

definiert. Der Einsatzzweck des Modells leitet sich aus den zu klärenden Simulationsfragen (siehe Abschnitt 2.2.3) ab.

### 3.1.2.2 Arten von Validität und Validierung

In diesem Abschnitt wird auf verschiedene Möglichkeiten zur Validierung von Simulationsmodellen eingegangen und die im Rahmen dieser Arbeit relevanteste Form, die Verhaltensvalidität einer Simulation, vorgestellt.

#### Arten von Validität nach Zeigler

Eine klassische Einteilung in Validitätsarten von Simulationen kann von Zeigler [123] übernommen werden. Zeigler unterscheidet drei Arten von Validität:

- **Replikative Validität:** Eine Simulation mit replikativ validem Verhalten kann Daten, die im realen System gesammelt wurden, korrekt reproduzieren. Genauer werden diejenigen Daten über das reale System korrekt reproduziert, bezüglich derer das Simulationsmodell getestet, d.h. validiert, wurde.
- **Prediktive Validität:** Eine Simulation mit prediktiv validem Verhalten kann Daten über das reale System korrekt vorhersagen, die nicht zur Erstellung des Simulationsmodells (Testen und Validierung) verwendet wurden.
- **Strukturelle Validität:** Ein Simulationsmodell ist strukturell valide, wenn durch seine Simulation nicht nur am realen System ermittelte Daten reproduziert werden können, sondern wenn es die internen Strukturen des realen Systems, die zu dessen Verhalten führen, „korrekt“ wiedergibt.

Im Kern kann die Validität eines Simulationsmodells somit daran festgemacht werden, wie gut es das Verhalten eines realen Systems reproduzieren kann und wie genau es die für das Systemverhalten entscheidenden internen Strukturen des Systems nachbildet. Für einen bestimmten Anwendungsfall wird nicht immer jede Form von Validität benötigt. Die benötigten Validitätsformen hängen vom Anwendungszweck des Simulationsmodells und den Simulationsfragen ab.

#### Typen von Validität nach Sargent

Sargent [103] zeigt auf, dass Validierung in allen Phasen der Entwicklung eines Simulationsmodells (siehe Abschnitt 2.2.3) eine wichtige Rolle spielt. Er unterscheidet verschiedene relevante Teilprobleme bei der Simulationsvalidierung:

- **Validität des Modellkonzeptes:**

Bei der Validierung des Modellkonzeptes wird versucht nachzuweisen, dass das Konzept die dem Modell zugrunde liegenden Theorien und Annahmen korrekt widerspiegelt. Weiter wird versucht, nachzuweisen, dass die im Modell umgesetzten Strukturen, die interne Logik und die modellierten mathematischen und kausalen Zusammenhänge bezüglich der adressierten Simulationsfragen „sinnvoll“ gewählt wurden.

- **Verhaltensvalidität:**

Bei der Validierung des Verhaltens wird geprüft, ob das Simulationsverhalten des Modells im Rahmen seines Anwendungszwecks (d.h. zur Beantwortung der Simulationsfragen) hinreichend genau ist.

Für die in dieser Arbeit behandelte Kalibrierung von ABS ist besonders die Validität des Simulationsverhaltens relevant. **Bezüglich der Validität des Modellkonzeptes wird davon ausgegangen, dass der Nutzer diese beim Entwurf der Modellstrukturen nach bestem Wissen sicher stellt.**

### 3.1.2.3 Validierungskriterien

Die Bewertungsaspekte, auf deren Basis die Validierung des Simulationsverhaltens durchgeführt wird, werden in dieser Arbeit als **Validierungskriterien VK** bezeichnet. Das Verhalten einer Simulation wird als **ausreichend valide** angesehen, wenn das Simulationsverhalten alle *VK* erfüllt. Die genaue Art und Menge der *VK* ist problemspezifisch. Ein Validierungskriterium *VK* hängt einerseits vom jeweiligen Teil des Simulationsverhaltens, der validiert werden soll, und andererseits vom verfügbaren Nutzerwissen über das Verhalten des simulierten Originalsystems ab.

Teilprobleme bei der Aufstellung von *VK* sind:

- **Menge und Kombinierbarkeit der VK:** Die Validierung von Simulationsverhalten ist am einfachsten, wenn man das Verhalten eines gesamten Simulationslaufes anhand eines einzigen *VK* bewerten kann. Mit steigendem Detaillierungsgrad des Simulationsmodells wird es jedoch immer schwerer, alle Validitätsaspekte des Simulationsverhaltens in einem einzigen *VK* zu repräsentieren. *VK* können das Gesamtverhalten der Simulation oder das Verhalten einzelner Modellteile betreffen. Gibt es mehrere *VK* muss die Frage geklärt werden, wie diese *VK* miteinander kombiniert werden können. Genügt es einfach zu fordern, dass alle *VK* erfüllt sind oder sind manche *VK* wichtiger, während andere nur nebenrangig erfüllt sein sollten?
- **Formalisierbarkeit von VK:** Soll die Validierung automatisch durchgeführt werden, ist es wichtig, die jeweiligen *VK* automatisch berechenbar formalisieren zu können.
- **Messung von Validitätsgraden:** Möchte man die resultierende Validität des Simulationsverhaltens für verschiedene Parametereinstellungen bewerten, hilft eine Aussage *valide* oder *nicht valide* nicht. Es ist von Interesse zu wissen, wie nahe das Simulationsverhalten bereits validem Verhalten ist, auch wenn dieses noch nicht erreicht wurde. Unter bestimmten Gegebenheiten, die im Folgenden genauer erläutert werden, können auch **Grade von Validität** gemessen werden. Dies kann die Möglichkeit bieten, **graduelle Abstufungen der Validität des Simulationsverhaltens** zu messen und so die Validität unterschiedlicher Modelle zu vergleichen. Damit ist es möglich eine Aussage zu treffen wie viel valider das Verhalten eines Simulationslaufes im Vergleich zu einem anderen war.

In diesem Abschnitt werden zunächst verschiedene mögliche Techniken zur Formalisierung und Überprüfung von Validierungskriterien beschrieben und hinsichtlich der oben



beschriebenen Teilprobleme bewertet. Dann wird eine formale Definition für den Begriff *Validierungskriterium* gegeben.

### Automatische Bewertung von Validität mit Hilfe einer Zielfunktion

Für eine automatische Validierung ist es am besten, die Validität eines Simulationslaufes anhand einer so genannten **Zielfunktion**  $ZF$  zu messen. Mit Hilfe einer Zielfunktion der Simulation kann der Validitätsgrad des Simulationsverhaltens auf einen numerischen Wert abgebildet werden. Durch eine Zielfunktion wird es ermöglicht, automatisch **graduelle Abstufungen der Validität des Simulationsverhaltens** zu messen.

---

#### **Begriffsbestimmung 9: Zielfunktion**

Sei ein Modell  $M$  gegeben mit  $M = (S, V, P, PW)$ . Die generische Definition einer Zielfunktion  $ZF_M$  zur Bewertung der Verhaltensvalidität von  $Sim(M)$  bezüglich der Parametereinstellung  $PW$  lautet für den Fall einer deterministischen Simulation (siehe z.B. [86, 45, 10, 5]):

$$ZF_M : W(P_1) \times \dots \times W(P_n) \rightarrow \mathfrak{R}$$

---

$ZF_M$  ist eine Abbildung der Wertebelegung der Parameter nach  $\mathfrak{R}$ . Der Verlauf von  $ZF$  ist im Allgemeinen nicht bekannt und daher ein direktes Ausrechnen der Funktion für eine gegebene Parameterbelegung nicht möglich. Der Funktionswert kann nur durch die Simulation des Modells  $M$  und Bewertung der zugehörigen Verhaltensvalidität ermittelt werden. Wäre der Funktionsverlauf bekannt, könnte mit mathematischen Mitteln der Funktionsverlauf untersucht und direkt eine Parametereinstellung gewählt werden, die zu optimal validem Simulationsverhalten bezüglich  $ZF$  führen würde.

Im Falle einer **Simulation mit Zufallseinflüssen** kann durch einen einzelnen Simulationslauf nur ein durch den Zufallseinfluss im Simulationslauf bedingter Wert für  $ZF_M$  ermittelt werden. Ein solcher Wert auf Basis eines einzelnen Simulationslaufes hat nur wenig Aussagekraft, da in einem weiteren Simulationslauf ein anderer vom Zufall beeinflusster Wert für  $ZF_M$  ermittelt werden könnte. Um in diesem Fall durch Simulation einen allgemein gültigen Wert für  $ZF_M$  zu bestimmen, muss  $M$  mehrmals simuliert und jeweils der zugehörige  $ZF_M$  Wert ermittelt werden. Der statistische Erwartungswert für  $ZF_M$  über alle diese Einzelwerte kann dann als gültiger Wert für die Verhaltensvalidität der stochastischen Simulation verwendet werden.

#### **Formalisierung von Zielfunktionen**

Grundsätzlich ist der Modellierer bei der Spezifikation benötigter  $VK$ , d.h. in diesem Fall  $ZF$  frei. Häufig wird eine Zielfunktion zur Validierung jedoch so entworfen, dass

sie Abweichungen des Simulationsverhaltens vom am Originalsystem beobachteten Verhalten misst. Konkret bedeutet dies, dass relevante Zustände, welche ein Modell während eines Simulationslaufes durchlaufen hat, mit den am Originalsystem beobachteten Zuständen verglichen werden. Der Vergleich der Zustände erfolgt durch Vergleich der Werte der Modellvariablen  $V_i$  mit den entsprechenden Daten  $V'_i$  aus dem Originalsystem.

Im Folgenden werden zwei übliche Typen von Zielfunktionen vorgestellt, die auch in der Evaluationsstudie dieser Arbeit eingesetzt werden. Sei  $q_{PW}$  ein Vektor, in dem die  $n$  relevanten Werte  $V_i$  aus einem oder mehreren Zuständen, die ein Simulationsmodell mit Parametereinstellung  $PW$  während eines Simulationslaufes angenommen hat, gespeichert sind. Enthalte weiter der Vektor  $q'$  die entsprechenden Werte aus dem Originalsystem. Dann kann eine Zielfunktion  $ZF$  wie folgt aussehen:

$$ZF(PW) = \frac{1}{n} \sum_{j=0}^n |q_{PW_j} - q'_j|$$

Die Zielfunktion  $ZF(PW)$  misst die mittlere einfache Abweichung der simulierten Werte zu den Werten des Originalsystems. Verschiedene Varianten dieser Zielfunktion sind möglich.  $ZF(PW)$  ist besonders dann geeignet, wenn davon ausgegangen werden kann, dass die am Originalsystem ermittelten Referenzwerte absolut korrekt sind. Sind die Datenwerte  $q$ , die zur Validierung des Modells eingesetzt werden, jedoch verrauscht (= fehlerhaft), dann führt eine Kalibrierung auf der Basis von  $ZF(PW)$  zu falschen Werten für  $PW$ . Damit dies vermieden wird müssen während der Kalibrierungsphase die Parameter so eingestellt werden, dass sie unabhängig vom Datenrauschen zu validem Simulationsverhalten führen. Zum Umgang mit verrauschten Validierungsdaten existieren in der Literatur verschiedene Ansätze, die darauf beruhen, eine Annahme über die Art des Verrauschungsfehlers in den Daten zu machen. Auf Basis dieser Annahme wird eine Zielfunktion  $ZF$  entworfen, die unter der gegebenen Annahme dazu genutzt werden kann, die Werte der Parameter zu optimieren (so genannte **Maximum Likelihood Schätzer**). Der bekannteste Schätzer für die Abweichung einer Reihe von Modellausgaben von bestimmten Originaldaten ist die **mittlere quadratische Abweichung (MQA)**, bei der eine normalverteilte Verrauschung in den Messdaten angenommen wird:

$$MQA(PW) = \frac{1}{n} \sum_{j=0}^n (q_{PW_j} - v'_j)^2$$

Der Hauptgrund für die Verwendung der MQA ist, dass die Formel leicht aufzustellen und schnell zu berechnen ist.

### Automatische Bewertung von Validität durch Randbedingungen

Die Formulierung einer geeigneten Zielfunktion für valides Simulationsverhalten kann schwierig sein, wenn z.B. geeignete Daten fehlen. Eine andere Form, die Validität einer Simulation zu beurteilen, bei der es aber schwieriger ist Grade von Validität zu messen, ist die Formulierung von Randbedingungen an das Simulationsverhalten. In diesem Fall formalisiert man in booleschen Testfunktionen  $RB : W(P_1) \times \dots \times W(P_n) \rightarrow \{0, 1\}$ , welche Verhaltensweisen man während eines Simulationslaufes erwartet und welche nicht auftreten dürfen. Die Validität eines Simulationslaufes kann dadurch bestimmt werden, dass untersucht wird, welche Randbedingungen erfüllt sind und welche nicht. Es kann

zwischen **harten und weichen Randbedingungen** unterschieden werden. Ist eine **harte Randbedingung** verletzt, so wird das gesamte Simulationsverhalten als invalide eingestuft, unabhängig davon, ob andere Randbedingungen erfüllt sind oder nicht. **Weiche Randbedingungen** sind Randbedingungen, die erfüllt sein sollen. Im Gegensatz zu harten Randbedingungen wird nicht das gesamte Simulationsverhalten als invalide eingestuft, wenn eine weiche Randbedingung verletzt ist. D.h. ein Simulationsverhalten, bei dem - bei gleichen erfüllten harten Randbedingungen - mehr weiche Randbedingungen erfüllt sind als bei einem anderen hat einen höheren Validitätsgrad als das andere.

Als Beispiel für eine harte Randbedingung werde wieder der Vektor  $q_{PW}$  der relevanten Variablenwerte aus dem Beispiel der Zielfunktion betrachtet.  $q_{PW}$  enthalte die maximalen Einkaufsentfernungen  $Dist_j$  in Kilometern, welche die einzelnen simulierten Einkaufsagenten aus dem einführenden Beispiel während eines Simulationslaufes zurückgelegt haben. Es sei nun eine harte Randbedingung gegeben, dass keiner der Einkaufsagenten in einer Entfernung größer als 10 Kilometer eingekauft hat. Dies kann wie folgt formalisiert werden:

$$RB_{Dist}(PW) = \begin{cases} 0, & \text{falls } \exists j \text{ mit } q_{PW_j} = Dist_j > 10 \\ 1, & \text{Sonst} \end{cases}$$

### Bewertung von Validität als Plausibilität

In Fällen, in denen es weder möglich ist eine geeignete Zielfunktion anzugeben, noch ausreichende Randbedingungen an das Simulationsverhalten zu formulieren, können automatische Verfahren zur Bewertung der Validität des Simulationsverhaltens nicht eingesetzt werden. Zum Umgang mit derartigen Problemfällen wurden verschiedene andere Validierungstechniken, wie die Visualisierung des Modellverhaltens und dessen Bewertung durch Experten, Tests des Modellverhaltens unter Extrembedingungen oder Analyse von degeneriertem Modellverhalten entwickelt (für einen Überblick siehe z.B. Sargent [103]). Die meisten dieser Techniken benötigen nicht-automatische Validitätseinschätzungen eines Menschen. Der Modellierer oder Fachexperte muss die Plausibilität des Simulationsverhaltens bewerten (engl. „Face Validity“, siehe z.B. Sargent [103] oder Balci [11]). Zur Validierung mit Plausibilität ist es daher entscheidend, dass der Modellierer das Simulationsverhalten sinnvoll analysieren kann. Im für diese Arbeit entwickelten Kalibrierungsverfahren spielen nur formalisierbare Validierungskriterien eine Rolle. Aus diesem Grund wird die Validierung durch Plausibilität im Weiteren, d.h. insbesondere in den folgenden Definitionen, nicht mehr betrachtet.

### Definition: Validität von Simulationsverhalten

Basierend auf den verschiedenen Formen zur Evaluierung von Validität, kann nun der Begriff *Validierungskriterium* definiert werden.

---

### **Begriffsbestimmung 10: Validierungskriterien für Simulationsverhalten**

Sei  $VK = \{VK_1, \dots, VK_n\}$  eine Menge von  $n$  Validierungskriterien bezüglich des Simulationsverhaltens eines Modells  $M = (S, V, P, PW)$ .

Ein Validitätskriterium  $VK_i \in VK$  mit  $i \in \{1, \dots, n\}$  wird definiert durch:

$$VK_i = (TM_i, VT_i)$$

- $TM_i = (S_i, V_i = V|_{S_i}, P_i = P|_{S_i}, PW_i = PW|_{S_i})$ : Teilmodell von  $M$ , dessen Validität mittels  $VT_i$  bewertet wird. Dabei ist  $P|_{S_i}$  die Menge aller Parameter, die für  $S_i$  relevant sind („eingeschränkt auf  $S_i$ “).  $V|_{S_i}$  und  $PW|_{S_i}$  werden analog festgelegt.
- $VT_i \in ZF_{TM_i} \cup RB_{TM_i}$  mit  $ZF_{TM_i}$  als der Menge der Zielfunktionen bezüglich  $SV_{TM_i}$  und  $RB_{TM_i}$  als der Menge der Randbedingungen bezüglich  $SV_{TM_i}$ .
- $W(VK_i)$  sei definiert als der mögliche Wertebereich des Validierungskriteriums  $VK_i$ . Für  $ZF$  ist der Wertebereich  $\mathfrak{R}$  und für  $RB$  ist der Wertebereich boolsch.

Es wird nicht gefordert, dass die einzelnen  $TM_i$  unabhängig von den übrigen  $TM_i$  simulierbar sind. Sie müssen nur getrennt evaluierbar sein.

---

Die **Gesamt-Validitätsbewertung** des Simulationsverhaltens eines Modells  $M$  durch alle Kriterien  $VK$  wird einfach durch  $VK(M)$ , die Bewertung durch ein einzelnes Kriterium  $VK_i$  durch  $VK_i(M)$  beschrieben. Es ist klar, dass  $VK_i(M) \in \mathfrak{R}$  falls  $VK_i$  eine Zielfunktion und  $VK_i(M) \in \{Wahr, Falsch\}$  falls  $VK_i$  eine Randbedingung ist.

### Kombinierbarkeit von Validitätskriterien

Existieren mehrere Validitätskriterien zur Bewertung der Validität verschiedener Teilaspekte des Simulationsverhaltens, so werden Techniken benötigt, diese Funktionen geeignet zu kombinieren.

Für Zielfunktionen werden üblicherweise werden zwei Techniken eingesetzt, nämlich Verkettung gewichteter Teilzielfunktionen [45] und die Übertragung von Verfahren zur Mehrzieloptimierung auf die Simulationsoptimierung [10]. Jede dieser beiden Herangehensweisen hat Vor- und Nachteile, die hauptsächlich aus dem Problem einer geeigneten Gewichtung der einzelnen Teilziele zueinander herrühren. Für das in dieser Arbeit entwickelte Kalibrierungsverfahren haben die beiden Ansätze keine unmittelbare Relevanz, können aber durchaus im Rahmen des entwickelten Ansatzes eingesetzt werden.

Für Randbedingungen ist es möglich, aus mehreren Randbedingungen eine Zielfunktion zu konstruieren, z.B. indem man die Anzahl erfüllter Randbedingungen misst und bei Verletzung harter Randbedingungen die Anzahl erfüllter Randbedingungen immer auf Null setzt. Gegebenenfalls können die weichen Randbedingungen aufgrund ihrer Wichtigkeit für den Modellierer oder Fachexperten zueinander gewichtet werden. Es ist jedoch nicht trivial, eine geeignete Gewichtung zu spezifizieren. Da weiche Randbedingungen so durch Zielfunktionen abgebildet werden können, werden im weiteren Verlauf der Arbeit nur noch Zielfunktionen und harte Randbedingungen betrachtet.

Das Vorhandensein verschiedener Validierungskriterien wird besonders relevant, wenn versucht wird, das Kalibrierungsproblem mit White-Box-Zerlegung zu vereinfachen (siehe Abschnitt 3.1.5.3). In diesem Fall werden getrennte Kalibrierungen bezüglich der einzelnen

Zielfunktionen durchgeführt.

Theoretisch ist ein Simulationsverhalten valide, wenn alle  $ZF$  optimale Werte annehmen, alle  $RB$  erfüllt sind und das Simulationsverhalten dem Modellierer oder Fachexperten plausibel scheint. Praktisch wird versucht, **ausreichend valides Simulationsverhalten** zu erzeugen. Was genau unter „ausreichend valide“ verstanden wird, ist dabei problem-spezifisch. Beispielsweise kann der Nutzer Schwellwerte für die  $ZF$ -Werte angeben, ab denen das Simulationsverhalten bezüglich  $ZF$  als erfüllt und damit akzeptabel betrachtet wird.

In der weiteren Arbeit wird das Simulationsverhalten eines Modells für eine Parameter-einstellung  $PW_1$  als „valider“ erachtet als für eine Parametereinstellung  $PW_2$ , wenn:

- Für  $PW_1$  alle vorhandenen Randbedingungen erfüllt sind und für  $PW_2$  nicht.

Oder

- Für  $PW_1$  und  $PW_2$  alle vorhandenen Randbedingungen erfüllt sind und  $PW_1$  bezüglich einer Zielfunktion  $ZF$  besser bewertet wird als  $PW_2$ .

### 3.1.3 Das Problem geeigneter Strukturwahl

In Abschnitt 3.1.2 wurde darauf hingewiesen, dass in dieser Arbeit davon ausgegangen wird, dass der Nutzer die Validität des Modellkonzeptes sicher stellt. Modellierungsprobleme und sonstige strukturelle Unsicherheiten bei der Erstellung von ABS stehen nicht im Fokus dieser Arbeit. In diesem Abschnitt wird daher nur kurz auf Möglichkeiten zur automatischen Behandlung struktureller Unsicherheiten bei der Modellerstellung eingegangen. Für detaillierte und komplexe Simulationsmodelle in Einsatzgebieten für die noch keine etablierten Modellstrukturen existieren, wie z.B. ABS, ist es nicht selbstverständlich, dass die zur Erzeugung validen Gesamtverhaltens benötigten Modellstrukturen eindeutig vorgegeben, d.h. bekannt, sind. Es kann durchaus möglich sein, dass für bestimmte (Teil-)Strukturen zu einem (Teil)-Modellkonzept eine **Menge von konkurrierenden strukturellen Umsetzungen** existiert. Jede dieser Umsetzungen erscheint auf Konzeptebene valide. Aus den strukturellen Umsetzungen muss nun diejenige ausgewählt werden, die zur Erzeugung validen Simulationsverhaltens am besten geeignet ist. Das Problem der geeigneten Strukturauswahl wird in Entwurfsmethodologien für Simulationsmodelle nicht explizit behandelt, sondern als iterativer Prozess aus Konzeptionalisierung, Implementation, Kalibrierung, Validierung des Simulationsverhaltens, Änderung des Konzeptes etc. behandelt. Eine Ausnahme kann Müller [82] entnommen werden. Er beschreibt eine Methodologie zur Modellierung komplexer Systeme mittels Differentialgleichungen, die explizit das Vorhandensein möglicher Modellvarianten behandelt. Hierbei wird zur Lösung das *Bewerten und Auswählen* Verfahren [72, 73] verwendet.

Den Vorgang der Anpassung struktureller Aspekte eines Modells durch die Auswahl geeigneter Design-Optionen, kann man als **Konfigurationsproblem des Modells** ansehen [94, 95]. Bei diesem Ansatz wird es nötig, für alternative Modellstrukturen zusätzliche Parameter einzuführen und die Modellstrukturen über diese Parameter zu konfigurieren. Beispielsweise kann bei der Erstellung einer Fabrikhallensimulation unklar sein, welcher Typ von

Förderband den Produktionsdurchsatz der Fabrik maximieren kann. Man würde nun verschiedene (Teil-)modelle von Förderbändern erstellen. Über einen Parameter im Gesamtmodell kann konfiguriert werden, welches Förderband während eines Simulationslaufes verwendet werden soll.

Im weiteren Text werden solche für die Modellstruktur verantwortlichen Parameter explizit als **Konfigurationsparameter** bezeichnet, um sie von den übrigen nicht Struktur ändernden Parametern abzugrenzen. Konfigurationsparameter unterscheiden sich von anderen Parametern hauptsächlich dadurch, dass auf ihrem Wertebereich meistens keine Ordnung gegeben ist, d.h. dass ihr Wertebereich nicht einfach in  $\mathfrak{R}$  abgebildet werden kann. Dies führt dazu, dass viele existierende Kalibrierungsverfahren (siehe Kapitel 3.2) nicht einfach auf die Kalibrierung von Konfigurationsparametern angewendet werden können. In Kapitel 3.3 wird kurz auf entsprechende Arbeiten verwiesen.

### 3.1.4 Formalisierung des Kalibrierungsproblems

Simulationskalibrierung, d.h. die Kalibrierung von Simulationen, wird benötigt, wenn die zur Erzeugung validen Simulationsverhaltens benötigten Parameterwerte während der Modelerstellung nicht bekannt sind. Geeignete Parameterwerte können entweder durch zufälliges Ausprobieren von Wertekombinationen oder durch intelligente Vorgehensweisen ermittelt werden. Um intelligente Entscheidungen über die Auswahl bestimmter Parameterwerte treffen zu können, wird Wissen über den Zusammenhang zwischen bestimmten Wertekombinationen für die Modellparameter und dem resultierenden Simulationsverhalten benötigt. Ist solches Wissen nicht vorhanden, besteht das **generische Vorgehen zur Simulationskalibrierung** darin, die Auswirkungen bestimmter Parameterwerte auf das Verhalten der Simulation zu untersuchen und daraus Rückschlüsse über benötigte Änderungen an den Parameterwerten zu ziehen.

Bei der Simulationskalibrierung wird für eine Modellfamilie  $MF$  (siehe Definition 1 in Kapitel 2.2) nach Werten für die zugehörigen Parameter  $P$  gesucht, so dass das Simulationsverhalten des resultierenden Modells  $M$  dem des Originalsystems ausreichend genau entspricht. Die Bewertung der Verhaltensvalidität erfolgt mittels einer Reihe vom Modellierer oder Fachexperten spezifizierter  $VK$ .

---

#### Begriffsbestimmung 11: Kalibrierungsproblem

*Ein Kalibrierungsproblem  $KP$  ist durch eine zu kalibrierende Modellfamilie  $MF$  und eine Menge von Validierungskriterien  $VK$  gegeben.*

$$KP = (MF, VK)$$


---

Die Lösung eines Kalibrierungsproblems wird definiert durch:

---

#### Begriffsbestimmung 12: Lösung eines Kalibrierungsproblems

*Die Lösung eines Kalibrierungsproblems  $KP = (MF, VK)$  ist ein Modell  $M$  zu  $MF$ , dessen Simulationsverhalten ausreichend valide, bezüglich  $VK$  ist.*

---

Im Idealfall gibt es zu einem  $KP$  nur ein Lösungsmodell. Im Allgemeinen muss  $M$  jedoch nicht eindeutig bestimmbar sein.

---

**Begriffsbestimmung 13: Kalibrierung**

*Kalibrierung ist der Lösungsprozess eines Kalibrierungsproblems.*

---

### 3.1.4.1 Generisches Vorgehen bei der Kalibrierung von Simulationen

Es ist klar, dass es für Modellfamilien  $MF$  unter Umständen nicht möglich ist, in begrenzter Zeit alle zugehörigen  $M$  hinsichtlich ihrer Verhaltensvalidität zu bewerten. Gründe hierfür können in einer langen Laufzeit von Simulationsläufen zugehöriger Modelle  $M$  und in einer großen Menge zu kalibrierender Parameter liegen. Im Folgenden werden verschiedene mögliche generische Herangehensweisen zur Lösung des Kalibrierungsproblems vorgestellt und in den nächsten Abschnitten konkretisiert.

#### Naiver Ansatz

Eine naive Möglichkeit zur Identifikation geeigneter Parameterwerte für  $MF$  besteht darin, dass der Modellierer oder Fachexperte Parameterwerte für  $MF$  rät oder zufällig auswählt. Ein entsprechendes Vorgehen ist jedoch wenig erfolgversprechend, wenn es sehr viele mögliche Kombinationen von Parameterwerten gibt und die Auswirkungen von Änderungen an bereits getesteten Parametereinstellungen auf das Simulationsverhalten nicht vollständig vorhersehbar sind.

#### Systematischer Ansatz

In einem systematischeren Ansatz wird das Kalibrierungsproblem als **Suchproblem** nach einer geeigneten Parametereinstellung aufgefasst. Die Suche findet im Raum aller möglichen Parametereinstellungen  $PW$  statt, dem so genannten **Parametersuchraum**. Im Parametersuchraum werden die möglichen Parametereinstellungen mit der aus ihnen resultierenden Validität des Simulationsverhaltens in Beziehung gesetzt.

Die **Struktur des Parametersuchraum**, d.h. der Zusammenhang zwischen Verhaltensvalidität und Parametereinstellung, ist zu Beginn der Kalibrierung unbekannt. Die einzige Möglichkeit den Zusammenhang zwischen einer Parametereinstellung  $PW$  und der zugehörigen Verhaltensvalidität zu ermitteln, besteht darin, das Modell mit konkreten  $PW$  zu simulieren und das resultierende Simulationsverhalten hinsichtlich seiner Validität zu bewerten. Aus diesem Grund ist das Kalibrierungsproblem ein **inverses Problem**. Bei einem **direkten Problem**, schließt man ausgehend von einer bekannten Ursache auf deren Auswirkungen. Im Gegensatz dazu schließt man bei einem inversen Problem durch Beobachtung des Simulationsverhaltens auf den Zusammenhang zwischen Parametereinstellungen und resultierendem Verhalten. Da es während der Kalibrierung meist nicht möglich ist, in begrenzter Zeit alle möglichen Parametereinstellung hinsichtlich der aus ihnen resultierenden

Validität des Simulationsverhaltens zu untersuchen, kann der Zusammenhang zwischen den  $PW$  und  $VK$  nicht vollständig ermittelt werden.

Das generische Vorgehen zur Simulationskalibrierung einer  $MF$  besteht darin, nach und nach verschiedene  $M$  bezüglich der  $VK$  zu bewerten. Ist das Verhalten von  $M$  nicht ausreichend valide, wird versucht, aus dem nun bekannten Wissen über den Zusammenhang zwischen den Parameterwerten in  $M$  und der Bewertung bezüglich  $VK$  Rückschlüsse auf die Struktur des Suchraums zu ziehen und so eine geeignete nächste zu testende Parameter-einstellung zu wählen. Es ist klar, dass je mehr Parametereinstellungen bezüglich der  $VK$  ausgewertet wurden, desto genauer die Struktur des Suchraums ermittelt werden kann.

Das generische Vorgehen zur Simulationskalibrierung kann wie folgt zusammengefasst werden.

**1. Initialisierung der Parameterwerte:**

Die Simulationskalibrierung kann entweder von vorgegeben oder beliebigen z.B. zufälligen Parametereinstellungen beginnen. Initiale Parametereinstellungen können vorgegeben werden, wenn der Modellierer oder Fachexperte, aufgrund von Erfahrung, Intuition, Theorien oder Hypothesen bereits eine ungefähre Vorstellung einer geeigneten Parameterbelegung hat.

**2. Auswahl der nächsten zu testenden Parametereinstellung  $PW$ :**

Die bereits getesteten  $PW$  werden genutzt, um Rückschlüsse über die Struktur des Suchraums zu ziehen. Auf Basis dieser Schlussfolgerungen wird eine neue noch nicht getestete  $PW_{neu}$  gewählt, von der erwartet wird, dass sie „besseres“ Simulationsverhalten erzeugt. Genaueres wird in den Kapiteln 3.1.5.1 und 3.2 beschrieben.

**3. Bewertung der Parametereinstellung:**

Als nächstes wird ein Simulationslauf für das zu  $PW_{neu}$  gehörige  $M_{neu}$  durchgeführt und das resultierende Simulationsverhalten auf seine Validität hinsichtlich der  $VK$  bewertet.

**4. Ende der Kalibrierung oder Iteration:**

Die Kalibrierung ist beendet, wenn die Validität von  $M_{neu}$  mit  $PW_{neu}$  hinsichtlich  $VK$  „ausreichend“ hoch ist.

Ist dies nicht der Fall, wird zu Schritt 2 zurückgekehrt.

Mit zunehmender Anzahl an Iterationsschritten kann aus den bereits getesteten  $PW$  ein immer genauerer Verlauf der Suchraumstruktur approximiert werden, so dass es immer wahrscheinlicher wird, dass die nächste gewählte Parametereinstellung  $PW_{neu}$  tatsächlich valideres Verhalten als  $PW$  erzeugen kann und schließlich eine Parameter-einstellung gefunden wird, die zu ausreichend validem Verhalten führt.

Obige Beschreibung ist bewusst generisch gehalten. Es gibt viele unterschiedliche Techniken zur Simulationskalibrierung, bei denen auf unterschiedliche Weise direkt oder indirekt Modelle der Suchraumstruktur oder auch nur von Teilen der Struktur erstellt werden. Kapitel 3.2 gibt einen Überblick über diese Techniken.

### 3.1.5 Automatisierung des Kalibrierungsprozesses

Die Kalibrierung eines Simulationsmodells „von Hand“ vorzunehmen, kann ein sehr mühseliger und langwieriger Prozess sein, da sie vom Nutzer viel Aufwand für die Ana-



lyse einzelner Simulationsläufe und daraus resultierend für die Herleitung einer neuen Parametereinstellung erfordert. Weiter können zwischen den einzelnen Einstellungsschritten lange Wartezeiten aufgrund der Dauer einzelner Simulationsläufe liegen können. Es ist also aus verschiedenen Gründen attraktiv den Kalibrierungsprozess zu automatisieren:

- **Effizienz:**

Der offensichtlichste Vorteil besteht natürlich darin, dass die Kalibrierung potentiell unbeaufsichtigt erfolgen kann und dem Modellierer somit langwierige stupide Arbeit erspart bleibt.

- **Automatische Objektivierbarkeit:**

Es kann schwierig sein, „von Hand“ die Verhaltensvalidität einer bestimmten Parametereinstellung im Vergleich zu allen bisher getesteten Einstellungen zu beurteilen und so Rückschlüsse über die Struktur und den Verlauf des Parametersuchraums zu ziehen. Ein automatisches Verfahren, das formalisierte *VK* verwendet und systematisch explizit oder implizit ein Modell (von Teilen) des Parametersuchraums aufbaut, kann entsprechende Beurteilungen viel exakter und dadurch weniger fehleranfällig treffen.

- **Systematische Änderung der Einstellungen:**

Die Änderung von Parametereinstellungen kann, wie der Validitätsvergleich, schwierig sein. Kann der Modellierer oder Fachexperte seine Strategien zur Anpassung der *PW* auf Grundlage des Wissens über die bisher getesteten Parametereinstellungen formalisieren, so kann nicht nur der Vergleich getesteter *M*, sondern auch die systematische Auswahl des nächsten zu testenden *M* automatisiert werden.

In dieser Arbeit werden zwei Kernvorgehensweisen zur automatischen Kalibrierung von Simulationsmodellen betrachtet. Möglich sind auch **hybride Ansätze** bei denen versucht wird Teile aus beiden Kernvorgehensweisen miteinander zu verbinden:

- **Black-Box Vorgehensweisen:** Diese Vorgehensweisen umfassen den Großteil der existierenden Kalibrierungsverfahren für Simulationen. Die Verfahren sind so entworfen, dass es irrelevant ist was für eine Art von Simulation kalibriert wird. Es werden lediglich die zu kalibrierenden Parameterwerte in Bezug auf messbare *VK* (meist *ZF*) betrachtet. Es ist nicht relevant was für eine Art von Modell oder Simulationslauf zur Bewertung der Parameterwerte berechnet wird.

- **Wissensbasierte Vorgehensweisen:** Unter wissensbasierten Vorgehensweisen werden in dieser Arbeit Vorgehensweisen zur Kalibrierung verstanden, die explizit auf eine ganz bestimmte Modellfamilie abgestimmt sind. Es wird zur Kalibrierung Hintergrundwissen über das modellierte Originalsystem eingesetzt. Das Ziel bei der Verwendung von explizitem Wissen über die Wirkungszusammenhänge im Originalsystem besteht darin, die Kalibrierung für diese konkrete Modellfamilie effizienter durchführen zu können, als dies mit einer Black-Box-Vorgehensweise möglich wäre. Der Nachteil solcher spezialisierten Verfahren besteht darin, dass sie zur Kalibrierung anderer Modellfamilien vollkommen unbrauchbar sein können.

Als **White-Box-Zerlegungsansätze** für die Kalibrierung werden in dieser Arbeit generische Techniken zur Zerlegung von Simulationsmodellen in Teilmodelle bezeichnet, die dann

getrennt kalibriert werden. Die Zerlegungstechniken sind generisch, jedoch ist explizites Wissen über das modellierte System notwendig, um geeignete Zerlegungen anwenden zu können. Je nach Zerlegung kann pro Teilmodell entschieden werden, ob eine Black-Box oder wissensbasierte Vorgehensweise zur Kalibrierung geeigneter ist.

In dieser Arbeit wird ein neues **Makro-Mikro-Kalibrierungsverfahren** entwickelt. Das Makro-Mikro-Kalibrierungsverfahren ist eine hybride Vorgehensweise, die explizit zur Kalibrierung von ABS ausgelegt ist. Das Verfahren ist generisch auf ABS anwendbar und nützt spezifisches Wissen über die Struktur der zu kalibrierenden ABS aus.

### 3.1.5.1 Black-Box Ansatz zur Simulationskalibrierung

Der am meisten verbreitete existierende Ansatz zur Kalibrierung von Simulationsmodellen ist der sogenannte Black-Box Ansatz. Der Ansatz besteht darin, das Kalibrierungsproblem für Simulationen als globales Optimierungsproblem aufzufassen und dieses mit allgemeinen, also nicht Simulations-spezifischen, Optimierungsverfahren zu lösen. Es gibt viele verschiedene Varianten des Black-Box Kalibrierungsansatzes mit unterschiedlichen Strategien zur Auswahl der nächsten  $PW$  auf der Grundlage der bisher getesteten Parameter-einstellungen. In Abschnitt 3.2 wird ausführlich auf verschiedene Vertreter von Black Box Ansätzen eingegangen.

In Black-Box Ansätzen wird die  $MF$  als „Black-Box“ betrachtet. Es ist für diese Form der Kalibrierung irrelevant, wie genau das zu kalibrierende Simulationsmodell intern aufgebaut ist. Relevant ist nur, welche  $PW$  in  $MF$  eingesetzt werden und welche Validitätsbewertung bezüglich der  $VK$  durch Simulation des zugehörigen  $M$  erreicht wurde. Allen Varianten des Black-Box Ansatzes ist gemeinsam, dass valides Simulationsverhalten mit Hilfe einer Zielfunktion  $z$  beschrieben werden muss, um den Grad der Validität des Simulationsverhaltens für ein bestimmtes  $PW$  berechnen zu können (siehe Abschnitt 3.1.2.3).

Mathematisch ist die Optimierung die Minimierung oder Maximierung des Ausgabewertes einer Funktion, so dass vorhandene Randbedingungen über den Eingabeparametern der Funktion erfüllt werden [86]. Je nach Problem können entweder Maxima (z.B. maximale Performanz des Multiagentensystems) oder Minima (z.B. minimale Abweichung des Modellverhaltens von empirischen Daten) gesucht sein. Es ist leicht ersichtlich, dass jedes Maximierungsproblem als Minimierungsproblem formuliert werden kann, indem man die Zielfunktion des Problems negiert. Daher spricht man im Allgemeinen nur von Optimierungsproblemen.

### 3.1.5.2 Wissensbasierter Ansatz zur Simulationskalibrierung

In wissensbasierten Kalibrierungsansätzen wird versucht vorhandenes Wissen über die internen Zusammenhänge und Wechselwirkungen des Originalsystems und zugehörigen Simulationsmodells in ein problemspezifisches Suchverfahren für ein konkretes Kalibrierungsproblem umzusetzen. Dies unterscheidet wissensbasierte Ansätze stark von reinen Black-Box Strategien, bei denen allgemeine Suchstrategien unabhängig vom konkreten Originalsystem und Modell eingesetzt werden. Die Umsetzung von vorhandenem Wissen in ein problemspezifisches Suchverfahren erfolgt im wissensbasierten Ansatz durch die Formulierung von Kalibrierungsregeln und/oder der Durchsetzung möglichst

vieler Randbedingungen an das Simulationsverhalten. Kalibrierungsregeln erhalten als Eingabe die aktuelle Parametereinstellung  $PW$  und daraus resultierende Validitätsbewertung des Simulationsverhaltens bezüglich der  $VK$ . Aus diesen Eingabedaten generieren die Kalibrierungsregeln Vorschläge für Änderungen an den Parameterwerten, d.h. eine neue Parametereinstellung  $PW_{neu}$ , deren zugehöriges Simulationsverhalten wieder getestet wird usw. Beispiele für wissensbasierte Kalibrierungsansätze finden sich z.B. in Suchverfahren, die ein Expertensystem verwenden [65, 120], aber auch in Lernverfahren [98, 116, 90]. Ein auf ein konkretes Kalibrierungsproblem hin entworfenes Lernverfahren ist ein spezialisierter Parameteroptimierungsalgorithmus, bei dem einerseits Randbedingungen in Form einer Zielfunktion und andererseits ein vorgegebenes Parameteränderungsverfahren angewendet werden. Lernverfahren werden in dieser Arbeit daher als wissensbasierte Ansätze betrachtet. Das Ziel beim Einsatz wissensbasierter Ansätze besteht darin, für ein bestimmtes Problem schneller eine geeignete Parametereinstellung zu finden, als mit Black-Box-Suchmethoden. Die Verwendung wissensbasierter Ansätze kann den Kalibrierungsprozess im Vergleich zu Black-Box-Verfahren deutlich beschleunigen. Es kann jedoch sehr schwierig sein ein geeignetes wissensbasiertes Kalibrierungsverfahren zu entwickeln. Wurde nicht genug oder falsches Hintergrundwissen integriert oder wird das Hintergrundwissen ungeeignet im Verfahren eingesetzt, so besteht die Gefahr, dass überhaupt keine geeignete Lösung für das Kalibrierungsproblem gefunden werden kann oder das Verfahren weniger effizient wird als ein reiner Black-Box-Kalibrierungsansatz.

Bei einem wissensbasierten Ansatz zur Simulationskalibrierung sind zusätzlich zum Kalibrierungsproblem  $KP$  (siehe Begriffsbestimmung 11) eine explizit vom Modellierer/Fachexperten mit Systemwissen definierte Menge von Änderungsfunktionen  $\ddot{A}R$  bezüglich der zu kalibrierenden Modellparameter gegeben. Der Unterschied zu Black-Box-Ansätzen ist dabei, dass die Änderungsregeln explizit problemspezifisch vorgegeben werden. Es werden keine generischen, problem-unabhängigen Änderungsregeln eingesetzt:

$$KP^{WB} = (KP, \ddot{A}R)$$

Durch das hochgestellte  $^{WB}$  wird das Lösungsverfahren angegeben als dessen Eingabe das Kalibrierungsproblem formuliert wird. Die Änderungsfunktionen  $\ddot{A}R$  berechnen aus einer gegebenen Parameterbelegung  $PW$  und der zugehörigen Validitätsbewertung des Simulationsverhaltens bezüglich  $VK$  eine geänderte Parameterbelegung  $PW'$ , die als nächstes getestet wird.

$$\ddot{A}R_{wiss} : W(VK_1) \times \dots \times W(VK_m) \times W(P_1) \times \dots \times W(P_n) \rightarrow W(P_1) \times \dots \times W(P_n), \\ n, m \in \mathbb{N}$$

Dabei ist  $W(VK_i)$  der Wertebereich des  $i$ -ten Validierungskriteriums. Die  $\ddot{A}R$  können als Eingabewerte beliebige Teilmengen der verwendeten  $VK$  haben. In den  $\ddot{A}R$  hat der Nutzer explizites Wissen über Zusammenhänge zwischen  $VK$ -Werten und benötigten Änderungen an den Parameterwerten zur Steigerung der Validität des Simulationsverhaltens formalisiert.

### 3.1.5.3 Generische White-Box Zerlegung zur Vereinfachung von Simulationskalibrierung

White-Box Zerlegung von Kalibrierungsproblemen stellt eine Möglichkeit zur Vereinfachung komplexer Black-Box oder wissensbasierter Kalibrierungsprobleme dar. Die White-Box Zerlegung ist an sich kein eigenes Kalibrierungsverfahren, sondern dient der Zerlegung des Kalibrierungsproblems auf der Meta-Ebene. Bei der White-Box Zerlegung wird versucht, Wissen über den internen Aufbau des Simulationsmodells und des  $KP$  zu nutzen, um  $KP$  geeignet in einfacher zu lösende unabhängige Teilprobleme  $TKP$  zu zerlegen. Das Kalibrierungsproblem  $KP$  wird dann mittels *Divide&Conquer Strategien* gelöst, d.h. die Teilprobleme werden nach Möglichkeit getrennt kalibriert und die Kalibrierungsergebnisse zur Gesamtlösung zusammengesetzt. Die Hoffnung besteht darin, so schneller zu einem Kalibrierungsergebnis für das Gesamtproblem zu kommen.

---

#### Begriffsbestimmung 14: Zerlegung eines Kalibrierungsproblems

Die Zerlegung eines  $KP = (MF, VK)$  mit  $MF = (S, V, P)$  in  $n \in \mathbb{N}$  Teilkalibrierungsprobleme  $TKP_i$  ( $i \in [1; n]$ ) wird in dieser Arbeit definiert durch

$$KP = (\{TKP_1, TKP_2, \dots, TKP_n\}, \prec)$$

- $\prec$ : Partielle Ordnung auf der Menge der Kalibrierungsteilprobleme.

Sei  $MF|_{TKP_i}$  die Einschränkung von  $MF$  auf die für  $TKP_i$  relevanten Teile. Analoges sei für  $VK$ ,  $P$  und  $V$  definiert. Für zwei beliebige Teilprobleme  $TKP_i$  und  $TKP_j$  von  $KP$  mit  $TKP_i \prec TKP_j$  gilt  $TKP_i = (MF|_{TKP_i}, VK|_{TKP_i})$  und  $TKP_j = (MF|_{TKP_j}, VK|_{TKP_j})$  mit  $MF|_{TKP_i} = (S|_{TKP_i}, V|_{TKP_i}, P|_{TKP_i})$  und  $MF|_{TKP_j} = (S|_{TKP_j}, V|_{TKP_j}, P|_{TKP_j})$  wird gefordert, dass  $P|_{TKP_i} \neq P|_{TKP_j}$ .

---

Durch  $\prec$  wird bestimmt, in welcher Reihenfolge die einzelnen Teilprobleme gelöst werden sollen. Für zwei Teilprobleme  $TKP_i$  und  $TKP_j$  mit  $TKP_i \prec TKP_j$  gilt, dass zur Kalibrierung von  $TKP_i$  nicht das Kalibrierungsergebnis der Kalibrierung von  $TKP_j$  benötigt wird. Die Kalibrierung von  $TKP_j$  muss nach der von  $TKP_i$  durchgeführt werden. Die Relation  $\prec$  ist **nicht symmetrisch**, d.h. auch wenn  $TKP_i$  unabhängig von  $TKP_j$  kalibrierbar ist, bedeutet dies nicht notwendigerweise, dass auch  $TKP_j$  unabhängig von  $TKP_i$  kalibrierbar ist! Ein Kalibrierungsproblem  $KP = (MF, VK)$  ist in zwei Teilprobleme  $TKP_1 = (MF|_{TKP_1}, VK|_{TKP_1})$  und  $TKP_2 = (MF|_{TKP_2}, VK|_{TKP_2})$  zerlegbar, wenn gilt:

- $P|_{TKP_1}$  und  $P|_{TKP_2}$  kalibrieren unterschiedliche Parameter.
- $VK|_{TKP_1}$  charakterisiert die Validität von  $TKP_1$  bezüglich des Zwecks des Gesamtmodells „ausreichend genau“.
- $VK|_{TKP_2}$  charakterisiert die Validität von  $TKP_2$  bezüglich des Zwecks des Gesamtmodells „ausreichend genau“.
- Die resultierenden Parametereinstellungen  $PW|_{TKP_1}$  und  $PW|_{TKP_2}$  sind eine Lösung von  $KP$ .

Parameterwerte, die erst in  $TKP_2$  kalibriert werden, dürfen zur Lösung von  $TKP_1$  keine Rolle spielen. Wohl aber dürfen die kalibrierten Parameterwerte von  $TKP_1$  als Teil der Modellstruktur von  $TKP_2$  bei der Kalibrierung von  $TKP_2$  benötigt werden.

Sind  $TKP_1$  und  $TKP_2$  sogar vollständig unabhängig voneinander kalibrierbar wird dies durch die Relation  $||$  beschrieben. Gilt  $TKP_1 || TKP_2$  so können  $TKP_1$  und  $TKP_2$  potentiell parallel kalibriert werden. Dies bietet zusätzlich die Möglichkeit, die Kalibrierung dadurch zu beschleunigen, dass die  $TKP$  gleichzeitig kalibriert werden.

Die Anwendbarkeit von White-Box Zerlegungen zur Vereinfachung des Kalibrierungsproblems hängt vom jeweils zu lösenden Kalibrierungsproblem ab. Entscheidend ist, in wie weit der Nutzer unabhängig kalibrierbare Teilmodelle identifizieren und ob er geeignete Zerlegungen dafür spezifizieren kann. Mögliche allgemeine Ansätze für White-Box Zerlegungstechniken können Frantz und Panayiotou entnommen werden [41, 91]. White-Box Zerlegungsansätze speziell für Agenten-basierte Simulationen wurden von mir in [37] veröffentlicht. Wie die Zerlegbarkeit hängt auch die Integrierbarkeit von  $TKP$  zu einer Gesamtlösung stark von den jeweiligen Kalibrierungsproblemen ab. Im Rahmen dieser Arbeit wird unter der **Integration eines  $TKP_i$  in ein  $TKP_j$**  die Übertragung der in  $TKP_i$  ermittelten Parameterbelegungen in  $TKP_j$  verstanden. Wie diese Parameterbelegungen in  $TKP_j$  konkret verwendet werden, ist problemspezifisch.

#### 3.1.5.4 Kombitierbarkeit verschiedener Ansätze

Black-Box- und Wissensbasierte Kalibrierung sind nicht strikt voneinander getrennt, sondern können kombiniert eingesetzt werden. Auch White-Box-Zerlegungen sind nicht einen der beiden Ansätze beschränkt. Durch White-Box-Zerlegungen kann zwar versucht werden, ein Kalibrierungsproblem zu vereinfachen, konkrete Parametereinstellungen können mit der Zerlegung jedoch nicht festgestellt werden. Hierfür werden z.B. Black-Box oder wissensbasierte Techniken benötigt. Eine Möglichkeit zur gemeinsamen Nutzung von Black-Box und wissensbasierten Verfahren, die im für diese Arbeit entwickelten Werkzeug umgesetzt wurde, besteht darin, die Verfahren in zwei Stufen zu kombinieren. Zunächst wird ein Teil der Parametereinstellungen durch Black-Box Suche geändert. Dann werden die übrigen Modellparameter dazu mit Hilfe der Änderungsregeln eingestellt. Die Gesamtparametereinstellung wird dann bezüglich der resultierenden Validität bewertet.

## 3.2 Stand der Forschung: Black Box Simulationsoptimierung

In diesem Teilkapitel werden in Literatur und Industrie etablierte Lösungsverfahren für die automatische Black Box Kalibrierung von Simulationsmodellen vorgestellt. In Abschnitt 3.2.1 werden zunächst Charakterisierungsmöglichkeiten für Black Box Optimierungsverfahren aufgrund der verwendeten Suchtechniken und „Optimierungsfähigkeiten“ vorgestellt. In den weiteren Abschnitten wird ein Überblick über die Vielzahl existierender Optimierungsverfahren gegeben. Das Kapitel endet mit einer kurzen allgemeinen Diskussion des Black Box Optimierungsansatzes zur Simulationskalibrierung.

### 3.2.1 Black Box Optimierungstechniken

Die Black Box Simulationsoptimierung verwendet zur Ermittlung geeigneter Modellparameterwerte etablierte, meist automatische, Optimierungs- und Suchverfahren für

kombinatorische Suchprobleme. Die Simulationsoptimierung ist seit Jahrzehnten ein aktuelles und intensiv bearbeitetes Forschungsgebiet (siehe z.B. [10, 46, 47, 3, 15, 5]). Es gibt verschiedene Überbegriffe für die dabei verwendeten Methoden, wie „Optimierung für Simulation“ [45], „Optimierung mittels Simulation“ [43], jedoch hat sich hauptsächlich der Begriff „**Simulationsoptimierung**“ durchgesetzt. Im folgenden Kapitel wird ein Überblick über verschiedene existierende Optimierungsverfahren gegeben. Die Verfahren werden aufgrund ihrer Eigenschaften charakterisiert und ihr Kernalgorithmus beschrieben. Im weiteren Verlauf werden die Begriffe *Lösung* und *Suchraum* synonym für eine *Kandidaten-Parametereinstellung* des Simulationsoptimierungsproblems verwendet.

Automatische Suchverfahren lassen sich, je nach verwendeter Suchtechnik und Lösungsfähigkeit in verschiedene Kategorien einteilen. Im Folgenden werden zunächst zwei zentrale Charakterisierungen von Lösungen vorgestellt. Dann werden zwei grundlegende Einteilungen von Suchverfahren beschrieben: Die bezüglich lokaler oder globaler Lösungsfähigkeit und die in verwendete globale und lokale Suchtechniken.

### 3.2.1.1 Charakterisierung von Lösungen

Während der Durchführung einer Simulationsoptimierung ist jede betrachtete Kandidatenparametereinstellung potentiell die Gesamtlösung für das Optimierungsproblem. Um dies genauer zu untersuchen, muss die Parametereinstellung im Vergleich zu den anderen potentiellen Lösungen bewertet werden.

- **Lokales Optimum:** Eine Lösung ist ein lokales Optimum wenn sie im Vergleich zu (bis zu einem gewissen Grad) ähnlichen Lösungen optimal ist. Es kann also in der Nachbarschaft der Ergebnislösung keine Lösung gefunden werden, die optimaler bezüglich des Suchproblems ist.

Genauer für ein Minimierungsproblem: Eine Lösung  $x^*$  ist ein **lokales Minimum** der Funktion  $f$ , wenn es eine Nachbarschaft  $NB$  von  $x^*$  gibt, so dass für alle Lösungen  $x \in NB$  gilt:  $f(x^*) \leq f(x)$ . Für ein lokales Minimum gilt **nicht notwendigerweise**, dass für alle möglichen Lösungen  $x$  gilt:  $f(x^*) \leq f(x)$ .

- **Globales Optimum:** Die bezüglich des Suchproblems beste Lösung bezeichnet man als das globale Optimum des Suchproblems.

Genauer für ein Minimierungsproblem: Eine Lösung  $x^*$  ist ein **globales Minimum** der Funktion  $f$ , wenn für alle möglichen Lösungen  $x$  gilt:  $f(x^*) \leq f(x)$ .

### 3.2.1.2 Globale und lokale Lösungsfähigkeiten

Die grundlegendste Einteilung von Optimierungsverfahren ist die bezüglich ihrer lokalen oder globalen Lösungsfähigkeit [86, 114]. Bei dieser Einteilung wird auf die Fähigkeiten der Optimierungsverfahren, die insgesamt beste Lösung identifizieren zu können, fokussiert.

- **Globale Optimierungsverfahren** sind Suchverfahren, die in der Lage sind, aus der Menge der potentiellen Lösungen die bezüglich des Suchproblems insgesamt beste zu identifizieren (globale Lösungseigenschaft). Wird allerdings ein heuristisches Suchverfahren mit globaler Lösungsfähigkeit verwendet (siehe Abschnitt 3.2.4), hängt die

tatsächliche globale Lösungsfähigkeit von der Güte der jeweiligen Heuristik ab. Entscheidend ist, dass auch diese heuristischen Verfahren korrekt konfiguriert potentiell in der Lage sind, das globale Optimum zu finden.

- **Lokale Optimierungsverfahren** sind Suchverfahren, welche die globale Lösungsfähigkeit nicht aufweisen. Stattdessen können solche Verfahren nur garantieren, dass die von Ihnen gefundene Lösung ein lokales Optimum des Suchraums darstellt.

### 3.2.1.3 Globale und lokale Suchtechniken

Eine weitere Einteilung von Optimierungsverfahren ist die gemäß ihrer verwendeten Suchtechnik [114]. Bei dieser Einteilung wird auf die Suchstrategie der Verfahren fokussiert.

- Verfahren mit **globaler Suchtechnik** arbeiten auf einer (Teil)-Menge der möglichen Lösungen. Die Idee besteht darin, dass es, wenn man ein globales Optimum identifizieren möchte, Sinn ergibt mit einer ganzen Menge von Lösungskandidaten zu beginnen, die über den gesamten Suchraum verteilt sind.

Globale Suchverfahren haben auch globale Lösungsfähigkeiten. Beispiele für Verfahren mit globaler Suchtechnik sind evolutionäre Algorithmen (siehe Abschnitt 3.2.4.2) oder Intervallschachtelungsverfahren (siehe Abschnitt 3.2.5.1).

- Verfahren mit **lokaler Suchtechnik** „tasten“ sich ausgehend von einer bestimmten Lösung und Informationen über sehr ähnliche Einstellungen immer von einer Einstellung zur nächsten voran. Dabei wird meistens versucht sich bei einem Übergang von einer Lösung zur nächsten zu verbessern.

Aus der Verwendung einer lokalen Suchtechnik folgt nicht notwendigerweise, dass das jeweilige Suchverfahren auch nur eine lokale Lösungseigenschaft besitzt. Ein Beispiel für ein Optimierungsverfahren mit lokaler Suchtechnik aber globaler Lösungseigenschaft ist das Simulated Annealing-Verfahren (siehe Abschnitt 3.2.4.1). Beispiele für Verfahren mit lokaler Suchtechnik und lokaler Lösungseigenschaft sind reine Gradienten-basierte Verfahren, bei denen nicht vorgesehen ist, dass sie sich in einem Suchschritt auch verschlechtern können (siehe Abschnitt 3.2.2.1).

In der Praxis sind die Übergänge zwischen globalen und lokalen Suchtechniken relativ fließend:

- **Iterativer Ansatz:** Ein Verfahren mit lokaler Suchtechnik kann leicht in eines mit globaler Technik verwandelt werden, indem mehrere lokale Suchen von verschiedenen Startpunkten aus durchgeführt werden. Umgekehrt verwenden einige globale Optimierungsverfahren auch lokale Suchtechniken, um aus einer Lösung der aktuellen Kandidatenmenge eine benachbarte Lösung zu generieren (z.B. entspricht die Anwendung des Mutationsoperators bei evolutionären Algorithmen einer lokalen Suchtechnik, siehe Abschnitt 3.2.4.2).
- **Multiresolutionsansatz:** Eine weitere Methode, um mittels lokaler Techniken global suchen zu können, ist der Einsatz so genannter Multiresolutions-Techniken. Hierbei wird nicht das Simulationsmodell, sondern der Parametersuchraum in verschiedenen Auflösungsstufen betrachtet [32, 81]. Auf diese Weise wird versucht lokale Optima in niedrigen Auflösungsstufen einfach zu überspringen.

- **Clusteringansatz:** Dieser Ansatz stellt eine Verfeinerung des iterativen Ansatzes dar. Ziel ist es zu verhindern, dass bei iterativen Neustarts mit lokalen Suchtechniken immer wieder das gleiche Minimum ermittelt wird. Beim Clusteringansatz wird die Annahme gemacht, dass es nur maximal  $k$  lokale Minima gibt. Parametereinstellungen werden aufgrund der Ähnlichkeit ihrer Parameterwerte zu Clustern zusammengefasst. Dabei wird in jedem Cluster immer nur ausgehend vom besten Repräsentanten des Clusters bezüglich der Zielfunktion mittels lokaler Suchtechniken gesucht [113].

In diesem Teilabschnitt wurden allgemeine Eigenschaften von Black Box Optimierungsverfahren vorgestellt. Im weiteren Verlauf werden nun verschiedene konkrete existierende Optimierungsverfahren vorgestellt.

Die Verfahren werden basierend auf ihrer Grundstrategie in die folgenden Klassen eingeteilt:

- Numerische Optimierungsmethoden
- Stochastische Approximationsverfahren
- Heuristische Suchverfahren
- Modellbasierte Suchverfahren
- Statistische Rasterexperimente
- Response Surface Methodologie
- Hybride Suchverfahren

### 3.2.2 Numerische Optimierungsmethoden

Numerische Optimierungsmethoden verwenden lokale Suchtechniken. Zu einem gegebenen Punkt  $X$  im Parametersuchraum wird versucht, möglichst genau die Struktur des lokalen Suchraumverlaufs um  $X$  zu ermitteln. Auf Basis des ermittelten Verlaufswissens wird dann versucht geeignete Änderungen an  $X$  vorzunehmen, so dass die resultierende Lösung bezüglich der verwendeten Zielfunktion als besser bewertet wird. Die Struktur des Suchraums um  $X$  wird durch den so genannten Gradienten beschrieben. Der Gradient von  $X$  beschreibt die Richtung der Änderung des Zielfunktionswertes von  $X$  zu seinen Nachbarlösungen und den Grad der Änderung.

#### 3.2.2.1 Gradientenbasierte Optimierungsmethoden:

Gradientenbasierte Optimierungsmethoden verwenden eine numerische lokale Suchtechnik [86]. Diese berechnet zunächst den Gradienten für eine gegebene Kandidatenlösung im Suchraum und wählt dann die nächste Kandidatenlösung so, dass sie gemäß dem berechneten Gradienten zur größtmöglichen Verbesserung führt (auch bekannt als Hill-Climbing Suche).

Um den Gradienten exakt berechnen zu können, ist die Kenntnis der ersten Ableitung des Simulationsverhaltens erforderlich, die im Allgemeinen nicht verfügbar ist. Daher wird häufig ein einfaches direktes Schätzverfahren zur Berechnung des Gradienten  $g$  an der Stelle  $x$  eingesetzt [86]:



$$g(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

Ist  $x$  ein Parametervektor, so wird zu jedem betrachteten Eintrag des Vektors mit obiger Formel der Gradient berechnet. Praktisch hat die Wahl der Schrittweite  $h$  entscheidenden Einfluss auf die Genauigkeit der Gradientenberechnung. Ist  $h$  zu groß, beschreibt der ermittelte Gradient möglicherweise nicht den tatsächlichen lokalen Suchraumverlauf um  $x$ . Ist  $h$  zu klein, kann unter Umständen kein Unterschied in den Zielfunktionswerten von  $x$  und  $x+h$  gemessen werden. In diesem Fall ist  $f(x)$  nicht sensitiv gegenüber kleinen Veränderungen an  $x$ . Bei der so genannten **Vorwärtsschätzung des Gradienten** kann eine ausreichend exakte Annäherung des Gradienten für  $n$  Parameter mit  $n+1$  Funktionsauswertungen (= Simulationsläufe) erreicht werden. Nach der Berechnung des Gradienten wird dann versucht entlang der vom Gradienten vorgegebenen Richtung eine bessere Lösung zu finden.

Die Methoden zur Lösung dieser Richtungssuche lassen sich in zwei Klassen teilen (siehe z.B. Nocedal [86]):

- **Suche entlang einer Linie (engl. „Line Search“):** Bei diesen Methoden wird versucht entlang der durch den Gradienten vorgegebenen Richtung eine Lösung zu finden, die besser ist, als die aktuelle Lösung. Eine besonders wichtige Suchrichtung für Linien-Suchmethoden ist die *Newton-Richtung* [86]. Die Berechnung der Newton-Richtung ist sehr komplex (Berechnungsaufwand  $O(n^3)$  und Speicheraufwand  $O(n^2)$ ). Der Grund hierfür ist, dass die Hessematrix der Zielfunktion berechnet werden muss, was die Berechnung sowohl der ersten, als auch der zweiten Ableitung der Zielfunktion notwendig macht.
- **Suche in der so genannten Trust Region:** Bei diesen Methoden wird aus den gesammelten Informationen über  $f$  eine Modellfunktion  $m$  aufgebaut, die sich nahe der aktuellen Lösung  $x_k$  ähnlich zu  $f$  verhält [86]. Kann in dieser Trust Region keine bessere Lösung gefunden werden (ist also  $x_k$  optimal oder  $m_k$  zu schlecht geschätzt), wird die Trust Region verkleinert. Das Problem bei diesem Vorgehen ist, dass auch hier die Hessematrix der ursprünglichen zu optimierenden Funktion benötigt wird. Daher ist dieses Verfahren für Simulationsmodelle, bei denen diese nicht oder nur sehr aufwendig berechnet werden kann, nicht sinnvoll verwendbar.

### Gradienten-Basierte Suchverfahren in der Simulationsoptimierung:

Im Allgemeinen existieren für komplexere Simulationsmodelle und insbesondere für ABS keine Darstellungen, die es erlauben direkt die erste oder zweite Ableitung der Zielfunktion zu berechnen [86]. Daher werden häufig so genannte **Quasi-Newton-Verfahren** eingesetzt, welche die benötigten Ableitungen und insbesondere die Hessematrix schätzen. Das beliebteste Quasi-Newton Verfahren für die numerische Optimierung ist die **BFGS-Methode**, benannt nach ihren Entwicklern Broyden, Fletcher, Goldfarb und Shanno [86]. BFGS verwendet die Suchtechnik entlang einer Linie. Kern des Verfahrens zur Berechnung einer geeigneten Schrittweite entlang der Gradientenlinie sind die **(starken) Wolfe Bedingungen**

[86].

Es existieren Erweiterungen des Grundalgorithmus, welche die Integration von Ungleichheitsrandbedingungen und eine Beschränkung des benötigten Speicherplatzes ermöglichen [21]. Weitere Quasi-Newton Techniken, auf die an dieser Stelle aber nicht mehr genauer eingegangen wird, sind DFP, SR-1 und die Algorithmen der Broyden Klasse [86].

### 3.2.2.2 Das Simplex-Verfahren

Das Simplex-Verfahren dient zur Optimierung linearer Probleme, also von Problemen mit linearer Zielfunktion und linearen Randbedingungen [86, 51]. Ein Simplex ist ein  $n$ -dimensionaler mathematischer Körper, ein so genanntes Polytop. Ein  $n$ -Simplex besteht aus  $n + 1$  Punkten. Dabei erweitert jeder Punkt den Simplex in eine andere Dimension. Ein  $n$ -Simplex lässt sich aus einem  $(n - 1)$ -Simplex konstruieren, indem ein Punkt hinzugenommen wird, der in einer neuen Dimension liegt. Mit diesem Punkt werden dann alle bereits vorhandenen Punkte verbunden.

In diesem Abschnitt wird der Algorithmus des Simplex-Verfahrens nach Gilli [51] beschrieben (siehe Algorithmus 1). Eine tiefer gehende Beschreibung des Verfahrens findet sich bei Nocedal [86]. Ausgangspunkt des Verfahrens ist ein Parameteroptimierungsproblem in  $n$  Parametern mit einer linearen Zielfunktion  $f$ , die minimiert werden soll. Der Algorithmus arbeitet, indem er aus einem aktuellen Simplex im Parametersuchraum ein direkt angrenzendes Simplex konstruiert, das näher am Problemoptimum liegt. Dabei ist jede Ecke eines Simplex eine Kandidatenparametereinstellung.

---

#### Algorithm 1: Simplex Optimierung

---

**Input:** Parametervektor  $x$  und lineare Zielfunktion  $f$

**Output:** Optimale Parametereinstellung  $x^*$

Konstruiere initialen Simplex mit Parametereinstellungsecken  $x_k$ . Die  $x_k$  sind mögliche Parameterbelegungen für  $x$ ;

**foreach** Ecke  $x_k$  **do**

  | Berechne  $f(x_k)$ ;

**end**

**while** Abbruchkriterium nicht erfüllt **do**

  | Identifiziere am schlechtesten bewertete Simplexecke  $x_{schlecht}$ ;

  | Konstruiere  $x_{neu}$  durch den Simplexmittelpunkt gegenüber  $x_{schlecht}$ ;

**if**  $f(x_{neu})$  besser als  $f(x_{schlecht})$  **then**

    | Suche weitere Verbesserung entlang Linie  $x_{neu}$  und  $x_{schlecht}$ ;

**end**

**else**

    | Schrumpfe Simplex;

**end**

**end**

Gib beste bekannte Ecke  $x^*$  zurück;

---

Das Verfahren wird fortgesetzt, bis ein Abbruchkriterium erfüllt ist, also z.B. das Simplex eine bestimmte Größe unterschreitet, also die Ecken des Simplex sehr nahe beieinander liegen.

Es gibt noch mehrere weitere numerische Optimierungsverfahren für spezielle Optimierungsprobleme, wie die konjugierten Gradienten Methode (CG-Verfahren), das Gaus-Newton Verfahren und die Levenberg-Marquardt Methode ([86, 55]).

### 3.2.2.3 Diskussion: Gradienten-Basierte Suchverfahren

Ein entscheidender Vorteil der gradientenbasierten Suchverfahren ist, dass sie sich im Suchraum immer in die Richtung der größten Verbesserung bewegen. Als lokale Suchtechniken ist jedoch zumindest für die Grundalgorithmen nicht garantiert, dass sie ein globales Optimum finden und nicht auf einem lokalen Optimum oder einem Sattelpunkt (also einem Punkt mit Gradientem = 0) „hängen“ bleiben. Das automatische Umgehen mit derartigen Problemen ist ein aktives Forschungsgebiet [13]. Weiter ist es bei vielen zu optimierenden Parameterwerten bereits sehr aufwendig, an einer einzigen Stelle im Suchraum den Gradienten zu berechnen. Der Grund hierfür ist, wie zuvor beschrieben, dass bei  $n$  zu optimierenden Parameterwerten zur Schätzung des Gradienten mindestens  $n + 1$  einzelne Simulationsläufe durchgeführt werden müssen. Um einen groben Überblick über den gesamten Parametersuchraum zu erhalten, kann es daher effizienter sein, einfach zufällige Parametereinstellungen auszuwerten.

### 3.2.3 Stochastische Approximationsverfahren (SA)

Wie in der Diskussion zu numerischen Optimierungsverfahren beschrieben, können viele einzelne Simulationsläufe nötig sein, um den Gradienten an einer Stelle des Suchraumes mit ausreichender Genauigkeit zu schätzen. Dies ist insbesondere der Fall, wenn die Parameterwerte stochastischer Simulationsmodelle optimiert werden sollen. Stochastische Approximationsverfahren versuchen die Funktionsweise der gradientenbasierten Suchverfahren bei deterministischen Optimierungsproblemen geeignet auf den stochastischen Fall zu übertragen (siehe z.B. Fu [47]).

Übliche Verfahren hierfür sind [47]:

- **Finite Difference Stochastic Approximation (FDSA):** Hierunter versteht man die direkte Übertragung der Techniken aus Abschnitt 3.2.2.1 auf den stochastischen Fall. Der Gradient  $g$  an einer Stelle  $x$  wird wie im deterministischen Fall direkt aus den stochastisch verrauschten Werten berechnet. Zusätzlich wird eine Skalierungssequenz von Werten  $a_k > 0$  für die Schrittweiten des Suchverfahrens definiert. Der Vektor der nächsten zu testenden Parametereinstellung zu einem Zeitpunkt  $k + 1$  berechnet sich dann nach:

$$x_{k+1} = x_k - a_k * g(x_k)$$

- **Simultaneous Perturbation Stochastic Approximation (SPSA):** SPSA ist eine Variante von FDSA, die lediglich zwei einzelne Simulationsläufe zur Schätzung eines Gradienten verwendet [47]. Dies ist insbesondere bei großen Parametersuchräumen von Vorteil, bei denen viele einzelne Simulationsauswertungen zur Berechnung eines einzelnen Gradienten benötigt werden. Die Berechnung des Gradienten in SPSA erfolgt dabei durch zwei gegenüberliegende Nachbarlösungen zur aktuellen Parametereinstellung. Die Nachbarlösungen werden durch zufällige Änderung der aktuellen Parametereinstellung generiert [107, 108].

### 3.2.3.1 Diskussion: Stochastische Approximationsverfahren

Stochastische Approximationsverfahren verwenden eine lokale Suchtechnik und berechnen, soweit möglich, eine Gradientennäherung zur Kalibrierung stochastischer Simulationen. Die Suchverfahren selbst enthalten keine Zufallselemente. Der Erfolg der Verfahren hängt dabei einerseits von der Stärke der Verrauschung und von der Wahl der Skalierungssequenz ab. Die Verfahren sind beweisbar konvergent. Es ist jedoch nicht garantiert, dass sie die optimale Parametereinstellung finden. Sie benötigen daher als Ausgangspunkt der Suche bereits eine „vielversprechende“ Parametereinstellung.

### 3.2.4 Heuristische Suchverfahren

Heuristische Suchverfahren verwenden Heuristiken, mittels derer die eigentlichen Suchverfahren gesteuert werden. Auf diese Weise soll es ermöglicht werden, lokalen Optima wieder zu entkommen (Simulated Annealing), bekannte Kandidatenlösungen zu besseren neuen zu kombinieren (Evolutionäre Algorithmen) oder mit Hilfe eines aus den bisher bekannten Lösungen bestehenden Filtersystems eine neue Kandidatenlösung auszuwählen (Tabu Suche).

Alle heuristischen Suchverfahren, die in diesem Abschnitt beschrieben werden, haben die globale Lösungseigenschaft. Diese Eigenschaft hängt jedoch stark von der Güte der jeweils gewählten Suchheuristik ab. Sind die Verfahren falsch konfiguriert, wird unter Umständen nur ein lokales Optimum gefunden.

#### 3.2.4.1 Simulated Annealing (SAN)

Simulated Annealing [74] ist eine lokale Suchtechnik, deren Haupteigenschaft darin besteht, dass das Suchverfahren lokalen Optima unter gewissen Umständen wieder entkommen kann. Kirkpatrick entwickelte das Verfahren in Anlehnung an das Auskühlen von Metall (simuliertes Ausglühen). Korrekt konfiguriert kann SAN als globales Optimierungsverfahren eingesetzt werden.

Die Grundidee besteht darin, ausgehend von einem Startpunkt eine lokale Suche durchzuführen, bei der zu einem aktuellen Suchpunkt nach irgendeinem Kriterium ausgewählte (z.B. zufällig) benachbarte Kandidaten getestet werden. Hat eine getestete Nachbarlösung einen besseren Wert bezüglich der Zielfunktion, so wird sie als Startpunkt für die weitere Suche verwendet. Ist sie schlechter, wird der Übergang nur mit einer bestimmten Wahrscheinlichkeit gemacht. Diese Wahrscheinlichkeit hängt einerseits vom Grad der Verschlechterung beim Übergang zur Nachbarlösung ab und andererseits davon, wie weit das Suchverfahren bereits fortgeschritten ist.

Entscheidend für die Performanz von SAN ist die Wahl der Wahrscheinlichkeitsfunktion zur Akzeptanz schlechterer Lösungen. Der Standardalgorithmus nutzt ein als **Metropolis Bedingung** bekanntes Annahmekriterium [62], das im Folgenden für ein **Minimierungsproblem** beschrieben wird:

$$p_{akzept}(T, x, x') = \begin{cases} 1 & f(x') \leq f(x) \\ \exp\left(\frac{f(x) - f(x')}{T}\right) & \text{sonst} \end{cases}$$

- **T**: ein **Temperaturparameter**. Dieser wird zu Beginn des Algorithmus festgelegt und ist dann entweder fest oder wird nach einer Abkühlungsvorschrift (z.B. Reduktion auf

90% seines vorherigen Wertes alle  $n$  getesteten Lösungen) verringert.

- $x$ : die aktuelle Suchraumlösung, von der aus neue Kandidatenlösungen erkundet werden und  $x'$  die aktuell getestete Nachbarlösung.
- $f$ : die Zielfunktion.

Wird ein **konstanter Temperaturwertes**  $T$  verwendet, kann direkt zu Beginn des Suchverfahrens festgelegt werden, ab welchem Verschlechterungsgrad bezüglich der Zielfunktion eine Kandidatenlösung relativ sicher angenommen wird. Führt man zusätzlich eine **Abkühlungsvorschrift** ein, so wird das Annahmekriterium mit fortschreitender Suchzeit verschärft. Zu Beginn werden also Verschlechterungen mit größerer Wahrscheinlichkeit angenommen, als gegen Ende der Suche. Auf diese Weise kann Simulated Annealing so eingestellt werden, dass es sich am Anfang wie eine (globale) Zufallssuche verhält und gegen Ende, wenn davon ausgegangen wird, dass sich das Suchverfahren bereits in der Nähe der optimalen Lösung befindet, wie ein Hill-Climbing Verfahren, also einem Verfahren, das sich immer nur in die Richtung einer Verbesserung bewegt.

Da das Annahmekriterium eine wichtige Rolle spielt, gibt es zahlreiche Erweiterungen für den vorgestellten Grundalgorithmus, wie *Boltzmann Annealing*, *Simulated Quenching*, *Fast Annealing* oder *Very Fast Simulated Annealing*. Ingber [64] gibt hierzu einen vergleichenden Überblick und stellt selbst ein Adaptive Simulated Annealing vor, das mit etwa 100 Algorithmusparametern umfassend konfigurierbar ist.

### 3.2.4.2 Evolutionäre Algorithmen

Evolutionäre Algorithmen sind globale Suchtechniken. Sie zeichnen sich dadurch aus, dass sie **populationsbasiert** sind, d.h. sie arbeiten mit einer Menge an Lösungen und bewegen sich nicht, wie z.B. Simulated Annealing von einer Kandidatenlösung zur nächsten.

Dies ermöglicht eine Kombination von Suchtechniken, die ihre Motivation in der Evolutionstheorie haben [53]. Im Folgenden werden eine Reihe zentraler Begriffen und Techniken für evolutionäre Algorithmen vorgestellt:

- **Fitness:** Unter der Fitness einer Lösung  $x$  wird ihre Bewertung durch eine vorgegebene Zielfunktion  $f(x)$  verstanden. Eine Lösung hat eine höhere Fitness als eine andere, wenn ihr Zielfunktionswert "optimaler" ist als der der anderen.
- **Mutation:** Der Mutationsoperator erlaubt die Abänderung einer bestehenden Kandidatenlösung. Die Mutation ist im Prinzip das gleiche, wie ein einzelner Schritt einer lokalen Suchtechnik von einer Lösung zu einer zufälligen Nachbarlösung.
- **Crossover:** Der Crossoveroperator erlaubt es, zwei oder mehrere Parametereinstellungen aus der Lösungspopulation zu neuen Lösungen zu kombinieren. Die Hoffnung besteht darin, dass Teillösungen aus einzelnen für sich suboptimalen Kandidateneinstellungen zu einer besseren Lösung kombiniert werden können. Durch den Crossoveroperator werden große Sprünge im Parametersuchraum ermöglicht. Enthalten etwa zwei Parametereinstellungen komplementäre Teillösungen können diese durch den Crossoveroperator direkt kombiniert werden, anstatt die einzelnen Lösungen lokal zu verfeinern.

- **Selektion:** Durch die Anwendung des Mutations- und des Crossoveroperators werden neue Kandidatenlösungen aus der bisherigen Population generiert. Um möglichst effektiv suchen zu können, wird versucht, die verwendete Lösungspopulation so zu gestalten, dass sie einerseits möglichst gute Lösungskandidaten (hohe Fitness) und andererseits möglichst unterschiedliche Lösungen (unterschiedliche Parameterwerte) enthält. Der Selektionsoperator dient dazu, aus der bisherigen Population und den neu generierten Kandidatenlösungen die Population für den weiteren Verlauf des Algorithmus auszuwählen. Dies geschieht üblicherweise auf der Basis der Fitnesswerte der Kandidatenlösungen und gegebenenfalls weiterer Kriterien wie z.B. die Unterschiedlichkeit der Lösungen.

Es gibt verschieden Typen von evolutionären Algorithmen. Zwei bekannte Klassen sind:

- **Genetische Algorithmen:** Genetische Algorithmen wurden von John Holland entwickelt [53]. Bei genetischen Algorithmen werden die Parameterwerte einer Kandidatenlösung in einem Bitstring kodiert. Die evolutionären Operatoren arbeiten dann auf den kodierten Kandidatenlösungen und nicht auf den tatsächlichen Parametern.
- **Evolutionäre Strategien:** Evolutionäre Strategien wurden von Ingo Rechenberg [97] entwickelt. Im Unterschied zu den genetischen Algorithmen arbeiten evolutionäre Strategien direkt auf den zu optimierenden Parameterwerten.

Hoffmeister und Bäck [59] haben diese alternativen Ansätze miteinander verglichen und kommen zu dem Ergebnis, dass Evolutionäre Strategien lokale Suchrauminformationen stärker ausbeuten als genetische Algorithmen, was dazu führt, dass genetische Algorithmen eine deutlich geringere Konvergenzrate haben als evolutionäre Strategien. Umgekehrt decken genetische Algorithmen besser den gesamten Suchraum ab.

### Tabu-Suche:

Die Tabu-Suche ist, wie das Simulated Annealing, eine **lokale Suchtechnik**, verwendet aber eine andere Technik, um lokalen Optima zu entkommen [62]. Bei der Tabu-Suche werden eine **Filterfunktion** und ein **Speicher für bereits besuchte Lösungen** verwendet. Dabei wird auf Basis bereits besuchter Kandidatenlösungen die Liste möglicher Kandidatenlösungen für den nächsten Schritt des Algorithmus gefiltert. Die einfachste und etablierte Form der Tabu-Suche ist eine einfache iterative Verbesserungssuche mit einem Speicher der zuletzt besuchten Lösungen [62]:

Die Grundstrategie der Suche besteht darin, von einer aktuellen Kandidatenlösung immer zur besten (also nicht notwendigerweise besseren) Nachbarlösung zu gehen. Verlässt der Suchalgorithmus dabei ein lokales Optimum, würde er eigentlich im nächsten Schritt sofort zu diesem zurückkehren. Um dies zu vermeiden, werden die zuletzt besuchten Lösungskandidaten zum erneuten Besuch verboten. Als Folge kann das „Hängenbleiben“ auf Plateaus des Suchraumes verhindert werden. Auf diese Weise kann z.B. vermieden werden, mehrfach gleiche Suchraumteile zu durchlaufen oder z.B. auf Plateaus des Suchraums hängen zu bleiben. Die beste bisher besuchte Lösung wird zusätzlich als Ergebnis gespeichert.

### 3.2.4.3 Zusammenfassung: Heuristische Suchverfahren

Heuristische Suchverfahren unterscheiden sich von numerischen Optimierungsverfahren dadurch, dass sie keine explizite Gradientenberechnung durchführen. Stattdessen hängt die Performanz des Suchverfahrens stark von der Güte der gewählten Heuristik in Bezug auf das Kalibrierungsproblem ab, da diese sowohl die Laufzeit als auch die (globale) Lösungsfähigkeit der Optimierung beeinflusst. So benötigt Simulated Annealing geeignete Temperaturparameter und evolutionäre Algorithmen geeignete Lösungskodierung und Anfangspopulation.

Genetische Algorithmen haben gegenüber anderen Verfahren den Vorteil, dass sie keine metrische Struktur des Suchraumes voraussetzen, zusätzlich benötigen diese Algorithmen nicht notwendigerweise exakte Informationen wie viel besser/schlechter eine Parametereinstellung gegenüber einer anderen ist. Es genügt grundsätzlich zu wissen, **dass** eine Einstellung besser oder schlechter ist als eine andere.

Alle genannten Verfahren lassen sich miteinander kombinieren. So kann Simulated Annealing ein Tabu-Gedächtnis haben oder ein evolutionärer Algorithmus Simulated Annealing zur lokalen Suche während des Mutationsschrittes verwenden.

### 3.2.5 Modellbasierte Suchverfahren

Modellbasierte Suchverfahren bauen aus allen bisher ausgewerteten Kandidatenlösungen ein **Modell des Suchraumes** auf und entscheiden aufgrund dieses Modells, welche Kandidatenlösungen als nächste getestet werden sollten. Im Gegensatz zu gradientenbasierten Suchverfahren wird hier nicht iterativ versucht, sehr exakte Modelle der Suchraumstruktur für lokale Teilbereiche zu ermitteln und sich so im Suchraum „voranzutasten“. Stattdessen wird versucht, ein Modell des gesamten Suchraums, bzw. großer Suchraumbereiche, zu erzeugen und anhand dieses Modells zu entscheiden, welche Suchraumbereiche mit höherer Wahrscheinlichkeit optimale Lösungen enthalten als andere. Die Suche wird dann auf solche Bereiche konzentriert. Praktisch entspricht das von modellbasierten Suchverfahren ermittelte Suchraummodell einer Wahrscheinlichkeitsfunktion zur Auswahl weiterer Kandidatenlösungen, die vielversprechende Suchraumbereiche „bevorzugt“.

#### 3.2.5.1 Intervallschachtelungsmethoden

Intervallschachtelungsmethoden (engl: nested partitioning) verwenden ein Suchraummodell, das aus zwei Intervallen hinsichtlich des Wertebereichs der zu optimierenden Parameter besteht. Ein (kleineres) Intervall  $I_{Innen}$ , das den Wertebereich repräsentiert, in dem aktuell die optimalsten Parametereinstellungen „vermutet“ werden und ein (größeres) Intervall  $I_{Aussen}$ , das den restlichen Suchraum repräsentiert. Die Idee der Intervallschachtelungsmethoden besteht darin, den Suchaufwand hauptsächlich auf  $I_{Innen}$  zu konzentrieren, ohne  $I_{Aussen}$  vollständig zu vernachlässigen (da unbekannt ist, ob sich die optimalste Lösung tatsächlich in  $I_{Innen}$  und nicht doch in  $I_{Aussen}$  befindet).

Dabei wird wie folgt vorgegangen [88, 89, 93]:

1. Zunächst werden uniform  $k$  Zufallskandidatenlösungen aus dem gesamten Suchraum ausgewählt.

2. Dann wird der Suchraum nach einem vorgebbaren Schema in eine Menge von Teilintervallen (hinsichtlich des Wertebereichs der Parametereinstellungen) zerlegt.

Im einfachsten Fall sind dies  $n$  möglichst gleichförmige Intervalle mit jeweils einer der Kandidatenlösungen im Zentrum. Man wählt das Intervall der aktuell besten Lösung. Dieses Intervall bildet nun  $I_{Innen}$ . Die übrigen Intervalle werden zu einem einzigen Restintervall  $I_{Aussen}$  verschmolzen.

3. Nun wird wieder bei Schritt 1 begonnen, mit dem Unterschied, dass aus  $I_{Innen}$  und dem (im Allgemeinen größeren) Restintervall  $I_{Aussen}$  jeweils gleich viele Lösungskandidaten zufällig ausgewählt werden.
  - Wird bei diesem Prozess eine bessere Lösung in  $I_{Innen}$  gefunden, wird dieses Intervall wie in 2) aufgespalten und das neue Restintervall mit  $I_{Aussen}$  verschmolzen. Es wird also in das Intervall der aktuell besten Lösung „hineingezoomt“.
  - Wird dagegen eine bessere Lösung in  $I_{Aussen}$  gefunden, so wird aus  $I_{Innen}$  wieder „herausgezoomt“, d.h. es wird wieder eine Intervallschachtelungsstufe zurückgegangen und das Intervall um die neue beste Lösung wird weiter unterteilt, wie in 3).
4. Der Algorithmus endet, wenn die Intervallgröße des Intervalls um die beste Lösung „klein genug“ ist, oder nur noch ein Element enthält.

Das **Intervallschachtelungs-Verfahren** lässt sich leicht mit anderen Verfahren zur lokalen Suche kombinieren. So verwendet Kim [71] zur lokalen Suche genetische Algorithmen und Pichitlamken [93] eine Kombination aus Tabu Suche und Hill-Climbing, um lokale Verbesserungen innerhalb eines Intervalls zu erreichen. Eine weitere Variante des Intervallschachtelungsverfahrens wird von Sambridge [101, 100] beschrieben. Im **Neighbourhood Algorithm** werden Voronoi Zellen eingesetzt, um Teilsuchräume für lokale Suchtechniken zu identifizieren.

### 3.2.5.2 Schwarm Intelligenz Methoden:

Suchmethoden der Swarm Intelligence [17], wie z.B. **Ant Colony Optimization** [62] versuchen, Problemlösungsstrategien sozialer Insekten auf Optimierungsprobleme zu übertragen. Die Grundidee besteht darin, eine Menge lokaler Problemlöser („Ameisen“) zu verwenden, die (mittels Pheromonspuren) miteinander kooperieren. Gestartet wird mit  $k$  Ameisen, jede bearbeitet eine eigene Kandidatenlösung und führt auf dieser eine eigene lokale Suche durch. Die Art der Änderung ihrer aktuellen Kandidatenlösung, die eine Ameise durchführt, wird mittels einer Wahrscheinlichkeitsfunktion bestimmt, die aus der Qualität der benachbarten Lösungen berechnet wird (Pheromonspuren).

Ein weiterer bekannter Algorithmus aus dieser Familie der Swarm Intelligence Algorithmen ist der so genannte **Partikelschwarm-Optimierungsalgorithmus**. In diesem Algorithmus werden die Kandidatenlösungen des Suchproblems als Schwarm von Lösungen aufgefasst [70]:

1. Wähle eine Menge gleich verteilter Kandidatenlösungen. Jede Lösung erhält eine zufällige Geschwindigkeit (Schrittweite für lokale Suche) und eine Flugrichtung (Änderungsvektor für lokale Suche).



2. In jeder Iteration „bewegt“ sich eine Kandidatenlösung mit ihrer individuellen Geschwindigkeit in ihre individuelle Flugrichtung und es werden die zugehörigen neuen Kandidatenlösungen ausgewertet. Dann modifiziert jede Kandidatenlösung ihre Flugrichtung so, dass sie einerseits in Richtung der aktuell insgesamt besten bekannten Lösung und andererseits in die Richtung ihrer individuellen bisherigen besten Lösung „fliegt“. Beide Flugrichtungen und Geschwindigkeiten können einfach linear zu einem neuen Flugvektor kombiniert werden.

### 3.2.5.3 Diskussion: Modellbasierte Suchverfahren

Gemeinsam ist allen Swarm Intelligence Algorithmen, dass sie mit einer Population von Lösungen arbeiten, die sich im Verlauf des Algorithmus immer näher aufeinander zu bewegen. Dies führt dazu, dass im Bereich viel versprechender Lösungskandidaten immer mehr Suchraumpunkte betrachtet werden. Swarm Intelligence Algorithmen haben die gleichen problematischen Aspekte wie andere populationsbasierte Verfahren (z.B. evolutionäre Algorithmen). Wird die Anfangspopulation ungeeignet gewählt, ist unsicher, ob und nach wie vielen Iterationen die Suche das globale Optimum finden kann.

## 3.2.6 Statistische Rasterexperimente

In den bisher beschriebenen Verfahren wurde versucht lokale (z.B. Gradienten-basierte Verfahren) oder globale (z.B. modellbasierte) (Teil)-Modelle des Parametersuchraums aufzubauen und anhand dieser nach optimalen Parametereinstellungen zu suchen. In diesem Abschnitt wird beschrieben, wie Techniken zur Sensitivitätsanalyse dazu eingesetzt werden können, systematisch ein Modell des Verlaufs des gesamten Parametersuchraums aufzubauen, anhand dessen dann versucht werden kann, eine (ausreichend) optimale Parametereinstellung zu identifizieren.

Der Hauptunterschied zwischen den bisher beschriebenen Methoden und den Verfahren dieses Abschnitts besteht darin, dass hier nicht versucht wird, die Auswertung von Parametereinstellungen auf möglichst vielversprechende Bereiche zu konzentrieren. Stattdessen wird versucht ein Modell des gesamten Parametersuchraums zu ermitteln, das alle Suchraumteile in etwa gleich gut beschreiben kann.

### 3.2.6.1 Ansätze zur Suchraumrasterung

Die offensichtliche Schwierigkeit bei der Ermittlung eines Suchraummodells durch Sensitivitätsanalysen besteht darin, dass nicht alle möglichen Parametereinstellungen zur Ermittlung des Suchraummodells getestet werden können. Da die möglichen Parametereinstellungsauswertungen nicht auf bestimmte Bereiche konzentriert werden (wie bei den bisher beschriebenen Verfahren), stellt sich die Frage, welche Einstellungen ausgewertet werden sollten, um ein möglichst genaues Modell des Suchraumverlaufs zu erhalten.

Zur Lösung dieses Problems wurden verschiedene Experimentierstrategien entwickelt, die den Suchraum möglicher Parametereinstellungen nach verschiedenen Kriterien rastern und nur die Parametereinstellungen an den Rasterpunkten testen [84, 102].

Eine solche Sensitivitätsanalyse auf Basis statistischer Rasterexperimente ist eine sinnvolle strategische Maßnahme, um ein ungefähres Bild des Parametersuchraumes zu ermitteln. Um ein genaueres Bild des Parametersuchraumes zu ermitteln, müssen jedoch unter

Umständen sehr viele Simulationsläufe durchgeführt werden, was besonders für Modelle mit langer Laufzeit problematisch ist. Als Folge kann für komplexere ABS mit Rasterexperimenten in begrenzter Zeit nur ein grobes Bild des Parametersuchraumes erstellt werden. Dies reicht zwar zur Identifikation optimaler Parametereinstellungen wahrscheinlich nicht aus, jedoch kann es erlauben, einen „interessanten“ Startpunkt für ein weiteres Optimierungsverfahren mit lokaler Suchtechnik zu identifizieren.

### 3.2.6.2 Anwendung: Data Farming

Vom amerikanischen Militär wird der Ansatz des Data Farming propagiert [19, 12, 42]. Beim Data Farming wird argumentiert, dass ausreichend Berechnungskraft zur Verfügung steht, um für komplexe Simulationsmodelle viele Millionen einzelne Simulationsläufe mit verschiedenen Parametereinstellungen durchführen zu können. Auf diese Weise wird ein detailliertes Modell des Zusammenhangs von Parameterwerten und dem Modellverhalten ermittelt. Dieses detaillierte Modell des Parametersuchraums kann dann dazu genutzt werden, um z.B. in Simulationen von Gefechten Parameterwerte zu ermitteln, die optimale Verhaltensstrategien für die simulierten Truppen erzeugen.

Als Teilschritte nennt Brandstein [19]:

- **Düngen:** Militärischen Experten wird vermittelt, in welchen Bereichen Wissen über Systemzusammenhänge benötigt wird.
- **Das Feld bestellen:** Experten beschreiben, welche Aspekte für die zu simulierende Situation wichtig sind.
- **Anpflanzen:** Erstellung eines Simulationsmodells nach Expertenangaben.
- **Ernten:** Sammeln von vieler Ausgabedaten des Modells und Analyse der Daten mit statistischen Methoden.

Data Farming ist den anderen Ansätzen zur Simulationsoptimierung relativ stark entgegengesetzt. Hier werden umfangreiche Analysetechniken zur Optimierung von Simulationsverhalten eingesetzt, während bei den anderen Verfahren versucht wird, mit möglichst wenig Aufwand eine möglichst optimale Parametereinstellung zu finden. Data Farming ist damit eine Brute-Force Methode zur Analyse des Zusammenhangs zwischen Parameterwerten und dem zugehörigen Verhalten eines Simulationsmodells. Die Ergebnisse der Sensitivitätsanalysen werden zur Ermittlung optimaler Verhaltensweisen simulierter Einheiten, d.h. zur Optimierung von Parameterwerten, verwendet. Die einzelnen Teilschritte des Vorgehens sind übliche Techniken zur Erstellung und Verhaltensanalyse von Simulationsmodellen. Im Verlauf der Analyse/Optimierung werden enorme Mengen von Parametereinstellungen systematisch durchgetestet. Existieren Zufallseinflüsse auf das Simulationsverhalten, wird jeder Simulationslauf möglichst oft wiederholt.

### 3.2.7 Die Response Surface Methodologie (RSM)

In diesem Abschnitt wird die sogenannte *Response Surface Methodologie* beschrieben. Die RSM verwendet die in Abschnitt 3.2.6 beschriebenen statistischen Analyseverfahren, fokussiert aber deutlich stärker auf die Optimierung von Parameterwerten als z.B. das Data Farming. Die RSM ist aus Techniken zum statistischen Entwurf von Experimenten entstanden (siehe

Abschnitt 3.2.6 und [84]). Die Idee besteht darin, ein möglichst einfach zu optimierendes Metamodell des Zusammenhangs zwischen den Parametern der Simulation und den Werten der Zielfunktion zu generieren. Dieses Metamodell wird dann an Stelle der Originalsimulation mit Hilfe von deterministischen Verfahren optimiert. Es wurde bereits beschrieben, dass dies für große Suchräume nicht einfach durchführbar ist. Aus diesem Grund wird die RSM iterativ in Teilbereichen des Parametersuchraumes durchgeführt. Statistische Experimente im Allgemeinen und die RSM im Besonderen finden breite Anwendung im Bereich der Optimierung und Analyse von Prozessen und Simulationsmodellen [109, 84, 4, 85]. Als Metamodelle werden meistens entweder einfache polynomielle Funktionen ersten oder zweiten Grades [84], oder neuronale Netze (siehe z.B. [109] ) verwendet. Das Vorgehen ist hierbei wie folgt [84]:

1. Zunächst werden **statistische Experimente** auf dem gesamten Suchraum oder einem Teil davon durchgeführt und die jeweils mit der Zielfunktion bewertet. Hierfür werden verschiedene statistische Rasterdesigns eingesetzt (siehe Abschnitt 3.2.6).
2. Im nächsten Schritt wird aus den verfügbaren Simulationsdaten das **Metamodell** geschätzt. Verwendet man ein polynomielles Modell, so wird dieses mittels Regression berechnet. Im Falle eines neuronalen Netzes wird dieses entsprechend gelernt.
3. Im nächsten Schritt wird für das Metamodell der **Gradient** an der entsprechenden Stelle im Parametersuchraum berechnet. Mit Hilfe dieses Gradienten wird dann versucht eine bessere Parametereinstellung für das Originalsimulationsmodell zu finden. Dies wird fortgesetzt, bis keine Verbesserung mehr erzielt werden kann. Der Grund hierfür kann entweder sein, dass das Optimum gefunden wurde, dass das Metamodell für den neuen Bereich des Suchraums ungeeignet ist oder dass eine verfeinerte Form des Metamodells benötigt wird. Im zweiten Fall muss wieder bei Schritt 1 begonnen werden. Im dritten Fall muss zu einem feineren Modell gewechselt werden, etwa zu einem Polynom höheren Grades.

Der Vorteil der RSM gegenüber Gradienten-basierten Methoden besteht darin, dass sie es erlaubt, die Menge an zur Schätzung des Gradienten verwendeten Simulationsauswertungen zu dimensionieren. Dieser Vorteil ist gleichzeitig ein möglicher Nachteil. Werden zu wenige Parametereinstellungen zur Ermittlung des Metamodells ausgewertet, gibt es keine Garantie, dass das Metamodell den Verlauf des Suchraums genau genug approximiert [47], um eine Optimierung der Parameterwerte auf der Basis des Metamodells zu erlauben.

### 3.2.8 Hybride Suchverfahren: Streuungssuche

Für viele der hier vorgestellten Suchverfahren existieren zahlreiche Varianten und Erweiterungen und Kombinationen, auf die in den jeweiligen Abschnitten hingewiesen wurde. Auch zahlreiche Kombinationen der verschiedenen Suchverfahrenstypen sind bekannt. So werden insbesondere Kombinationen aus lokalen und globalen Suchtechniken eingesetzt, um die jeweiligen Stärken auszunutzen (siehe z.B. die Kombination von evolutionären Verfahren und Simulated Annealing in Abschnitt 3.2.4 oder die Kopplung des Intervallschachtelungsansatzes mit lokaler Suche in Abschnitt 3.2.5.1).

Ein etabliertes hybrides Suchverfahren, das Elemente von Tabu-Suche, modellbasierten Suchverfahren und evolutionären Algorithmen kombiniert, ist die Streuungs-Suche (engl. Scatter Search) [52, 47].

Die Streuungssuche besteht aus den folgenden Kernschritten [52]:

1. Auswahl einer **initialen Referenzpopulation** von Lösungen: Hier wird versucht eine möglichst unterschiedliche Menge von Startlösungen zu wählen
2. Erzeugung neuer Lösungen durch **Linearkombinationen von Lösungen** aus der Referenzpopulation (= gewichtete Summen von Parameterwerten bereits getesteter Lösungen) und Verbesserung dieser neuen Lösungen. Die Kombinationen werden so gebildet, dass sie sowohl innerhalb als auch außerhalb der von der Referenzpopulation aufgespannten konvexen Regionen liegen. Zusätzlich werden diese neuen Lösungen gegebenenfalls abgeändert. Dies bedeutet, dass Lösungen, die außerhalb des Wertebereiches liegen geeignet modifiziert werden.
3. Im nächsten Schritt werden die neuen Lösungen mittels **heuristischer Verfahren** weiter optimiert. Die Optimierung zielt dabei einerseits darauf, die Lösungen bezüglich ihres Zielfunktionswertes zu verbessern und andererseits für möglichst viel Diversität innerhalb der Lösungen zu sorgen.
4. Schließlich wird aus den optimierten neuen Lösungen eine **„geeignete“ Teilmenge ausgewählt**, und zur Referenzpopulation hinzugefügt.

Die Scatter-Suche verwendet Crossover Reproduktion ähnlich anderen Evolutionären Algorithmen, lokale Verbesserungstechniken, und Heuristiken, in der Art von Tabu-Suche, um den Suchaufwand auf viel versprechende Suchraumbereiche zu fokussieren. Sie kommt jedoch mit deutlich kleineren Populationsgrößen als z.B. evolutionäre Algorithmen aus [52].

### 3.2.9 No-Free-Lunch Theorem für Optimierungsprobleme

Die No-Free-Lunch Theoreme für Optimierungsprobleme befassen sich mit dem Problem der universellen Anwendbarkeit von Black Box Optimierungsverfahren [122]. Ihre Kernaussage besteht darin, dass alle Suchalgorithmen gleich gut oder schlecht sind, wenn man sie über alle möglichen mathematischen Suchprobleme vergleicht. Es gibt also kein Suchverfahren, das im Allgemeinen besser ist als eine Zufallssuche. Insbesondere warnen Wolpert und Macready [122] davor, Suchverfahren nur anhand weniger Beispielprobleme miteinander zu vergleichen. Die Schlussfolgerung dieser Aussage ist zum einen, dass es gewinnbringend ist, für ein spezielles Optimierungsproblem ein spezielles Optimierungsverfahren zu suchen, anstatt immer Verfahren wie genetische Algorithmen einzusetzen. Zum anderen folgt, dass ein Werkzeug zur Simulationskalibrierung, verschiedene Kalibrierungsverfahren bereitstellen sollte, wenn es nicht nur zur Kalibrierung von Modellen aus einer speziellen Domäne (z.B. der Fabriksimulation) eingesetzt werden können soll.

### 3.2.10 Zusammenfassung: Black Box Simulationskalibrierung

Der entscheidende und nicht zu unterschätzende Vorteil automatischer Black Box Optimierungsverfahren besteht darin, dass sie das Simulationsmodell als „Black Box“ behandeln, die tatsächliche Modellstruktur also irrelevant ist. Allgemeine Optimierungsverfahren werden seit vielen Jahren erforscht und erfolgreich zur Kalibrierung von Simulationsmodellen eingesetzt.

Im Kapitel wurde eine Vielzahl von Black Box Suchverfahren mit unterschiedlichen Stärken und Techniken dargelegt. Unterschiede liegen in der Geschwindigkeit der Konvergenz auf ein lokales Optimum oder der Abdeckungsgüte des gesamten Suchraumes. Die Performanz vieler solcher Optimierungsverfahren hängt zum einem von der Struktur des Suchraums des Kalibrierungsproblems und zum anderen von den gewählten Einstellungsparametern des Optimierungsverfahrens ab, deren sinnvolle Einstellung ein tiefgehendes Verständnis für die Funktionsweise der einzelnen Verfahren erfordert und daher für „Optimierungs-Laien“ nur schwer möglich ist. Welches Optimierungsverfahren für ein Simulationsmodell tatsächlich eingesetzt werden sollte, kann für neu entwickelte Modelle häufig nur schwer eingeschätzt werden. Ist überhaupt nichts über den Parametersuchraum bekannt, bieten sich globale Suchtechniken an, ist eine Schätzung initialer Parameterwerte verfügbar, können lokale Techniken effizienter sein.

### 3.3 Stand der Forschung: Automatischer Modellentwurf

In Kapitel 3.1.3 wurde darauf hingewiesen, dass auch Probleme bei der Festlegung der Modellstrukturen parametrisiert werden können (Konfigurationsprobleme). In der Literatur finden sich verschiedene Ansätze, Modelle automatisch zu erzeugen und ihre internen Strukturelemente geeignet zu wählen. Strukturelle Modellvarianten werden dabei meist durch qualitative Modellparameter, den Konfigurationsparametern des Modells, repräsentiert [10], aber auch andere Ansätze sind möglich. Das Problem bei der Verwendung qualitativer Modellparameter besteht darin, dass viele existierende Black Box Optimierungsverfahren mit lokalen Suchtechniken (siehe Kapitel 3.1.5.1), wie Gradientenmethoden oder Simulated Annealing, nicht einfach verwendet werden können, da sie zur lokalen „Untersuchung“ des Suchraumverlaufs eine Ordnung auf den Werten der Parameter benötigen (z.B. Wert A ist größer/kleiner als Wert B).

In diesem Abschnitt wird auf existierende Methoden zum automatischen Umgang mit solchen Modellierungsproblemen eingegangen. Die einzelnen Verfahren werden nur kurz genannt und auf die entsprechende Literatur verwiesen, da die Einstellung solcher Konfigurationsparameter nicht im Fokus dieser Arbeit liegt. Je nach Ansatz werden dabei unterschiedliche Aspekte automatisiert und die Anwendung der Ansätze erfordert unterschiedliche Mengen an Wissen und Vorbereitung des Modellierers:

- Vollständiges und zufälliges Austesten von Strukturkonfigurationen: Hierbei wird der Prozess des Testens und der Bewertung der Kandidatenmodelle hinsichtlich ihrer Verhaltensvalidität automatisiert. Gegebenenfalls wird auch noch die zufällige Auswahl zu testender Kandidatenmodelle automatisiert [10]. Der Nachteil bei einer **zufälligen Auswahl** von Modellstrukturen besteht darin, dass bei vielen möglichen Kandidatenstrukturen entweder die tatsächlich gesuchte nicht in der Zufallsmenge dabei ist oder dass sehr viele Kandidatenstrukturen getestet werden müssen, um eine geeignete zu finden.
- **„Bewerten und Auswählen“ Techniken:** Bei diesem Ansatz (engl: **Ranking and Selection R&S**) wird neben dem Prozess des Testens und Bewertens insbesondere der Auswahlprozess für zu testende Kandidatenmodelle automatisiert. Dabei werden nicht einfach nur alle Kandidatenmodelle oder eine zufällige Teilmenge bewertet,

sondern es wird versucht, automatisch auf der Basis bereits bewerteter Kandidatenmodelle die nächsten zu testenden Modelle auszuwählen [72, 73, 112].

- **Modellentwurf mit genetischen Algorithmen:** Bei diesem Ansatz wird neben dem Prozess des Testens und Bewertens von Modellstrukturen insbesondere auch der Prozess der Modifikation bereits getesteter Modellstrukturen zu neuen Kandidatenmodellen automatisiert [9, 92, 94, 95]. Dieses Vorgehen hat die üblichen Vor- und Nachteile genetischer Algorithmen, da es für dieses Kalibrierungsverfahren keine Ordnung auf den zu kalibrierenden Parameterwerten benötigt wird.
- **Modellentwurf mit Expertensystem:** Bei diesem Ansatz wird ein Expertensystem zur Modifikation von Modellstrukturen auf der Basis des Modellverhaltens in der Simulation entwickelt. Neben der automatisierten Durchführung des Testens und Bewertens von Modellstrukturen sollen mit Hilfe des Expertensystems auch automatische Strukturänderungen bei invalidem Modellverhalten durchgeführt werden [65, 120]. Der Nachteil dieser Verfahren besteht darin, dass zu ihrer Anwendung umfassendes Expertenwissen über die Wirkungszusammenhänge in einer bestimmten Domäne gegeben sein muss. Dies ist insbesondere bei der Entwicklung von ABS häufig nicht der Fall, z.B. wenn es bei Modellentwicklung und Simulation explizit darum geht zu erforschen, wie sich globales Systemverhalten aus dem Verhalten auf lokaler Ebene erzeugen lässt.

## Kapitel 4

# ABS Kalibrierungsproblem und Stand der Forschung

In Kapitel 3 wurde das Kalibrierungsproblem und mögliche Lösungsansätze für allgemeine Simulationen vorgestellt. Dieses Kapitel beschäftigt sich mit spezifischen Problemen bei der Kalibrierung von ABS.

### 4.1 Kalibrierungsprobleme in ABS

Im ersten Teil dieses Abschnitts werden die in Abschnitt 3.1 eingeführten Konzepte auf ABS übertragen. Im zweiten Teil werden spezifische Problemaspekte bei der Kalibrierung von ABS vorgestellt, die in dieser Arbeit behandelt werden.

#### 4.1.1 Kalibrierungsproblem für Agenten-basierte Simulationen

Aufgrund seiner generischen Definition ist es klar, dass das allgemeine Kalibrierungsproblem aus Definition 11 auch die Kalibrierung von ABS abdeckt. In dieser Arbeit wird ein etwas verfeinertes Kalibrierungsproblem für ABS betrachtet, wobei die Verfeinerung durch Präzisierung der  $MF$  und  $VK$  Bestandteile von  $KP_{ABS}$  vorgenommen wird. Ein Kalibrierungsproblem für ABS wird beschrieben durch:

$$KP_{ABS} = (MF, VK)$$

mit

- $MF = (UMF, AMF = \{IAMF_1, \dots, IAMF_n\})$
- $VK = \{VK_1, \dots, VK_v, IVK_1, \dots, IVK_w\}$

Die Modellfamilie  $MF$  des Gesamtmodells setzt sich zusammen aus der **Modellfamilie UMF zur Beschreibung des Umweltmodells UM** und der Familie aller Agentenmodelle  $AMF$ .  $AMF$  besteht aus den **Modellfamilien  $\{IAMF_1, \dots, IAMF_n\}$  zur Beschreibung der  $n \in \mathbb{N}$  simulierten Agentenmodelle (Individuelle AgentenModellFamilien)**. Die Menge der Validierungskriterien wird unterteilt in die Menge an  $v \in \mathbb{N}$  Validierungskriterien

$\{VK_1, \dots, VK_v\}$  bezüglich „kombinierten“ Verhaltens (z.B. Gesamtverhalten der Simulation oder Gruppenverhalten simulierter Agenten) und die Menge an  $w \in \mathbb{N}$  Validierungskriterien  $\{IVK_1, \dots, IVK_w\}$  (Individuelle Validierungskriterien)<sup>1</sup> bezüglich des Mikro-Ebenen-Verhaltens einzelner Agentenmodelle  $IAM_i$  zu den gegebenen Familien  $IAMF_i \in \{IAMF_1, \dots, IAMF_n\}$ . Die Zuordnung von  $IVK$  zur  $IAMF$  ist wie bei anderen  $VK$  durch die Festlegung des zugehörigen Teilmodells in  $IVK$  gegeben.

Zur Validierung von ABS können natürlich auch andere Techniken wie z.B. Plausibilitätseinschätzungen durch Fachexperten benötigt werden (siehe Klügl [77] für eine umfassende Validierungsmethodologie für ABS).

In den folgenden Abschnitten wird zunächst beschrieben, wie die in Abschnitt 3.1 eingeführten existierenden Kalibrierungsansätze und Techniken auf  $KP_{ABS}$  übertragen werden können. Im Anschluss werden verschiedene spezifische Kalibrierungsprobleme für ABS beschrieben, welche die Anwendung existierender Techniken erschweren, bzw. Möglichkeiten zum Einsatz effizienterer Kalibrierungsansätze bieten.

#### 4.1.2 Übertragung existierender Lösungsansätze auf $KP_{ABS}$

Im Folgenden wird zunächst der in Abschnitt 3.1.5.1 beschriebene Black Ansatz auf die Kalibrierung von ABS übertragen. Weiter wird beschrieben, wie versucht werden kann  $KP_{ABS}$  durch White-Box-Zerlegung (siehe Abschnitt 3.1.5.3) zu vereinfachen.

##### 4.1.2.1 Reiner Black Box Kalibrierungsansatz für ABS

Bei einem reinen Black Box Ansatz zur Lösung eines Kalibrierungsproblems wird die gesamte Simulation als unzerlegte Black Box betrachtet. Das ABS-Kalibrierungsproblem für eine reine Black Box Kalibrierung  $KP_{ABS}^{BB}$  wird also genauso behandelt wie jedes andere generische  $KP$ , d.h. ohne Ausnutzung spezieller Eigenschaften des Problems oder des zugehörigen Simulationsmodells. Individuelle Agentenmodellfamilien  $IAMF_i$  werden nicht explizit betrachtet. Die Validierungskriterien  $VK$  werden nicht bezüglich des Verhaltens einzelner Agentenmodelle, sondern lediglich bezüglich des Gesamtverhaltens der Simulation betrachtet. Alle Teilkalibrierungskriterien von  $KP_{ABS}$  werden zu einem globalen Validierungskriterium  $GVK$  zusammengefasst. Das ursprüngliche  $KP_{ABS}$  wurde mit dem Validierungsausgangspunkt „Das Verhalten konkreter einzelner Agentenmodelle ist nicht ausreichend valide“ formuliert. Dies wird in  $KP_{ABS}^{BB}$  wieder zum aggregierten Validierungsprinzip „Das Gesamtverhalten ist nicht ausreichend valide“ zusammengefasst.  $GVK$  bezeichnet die Menge aller Validierungskriterien, die sich auf das Gesamtmodell beziehen. Nach dem gleichen Prinzip beschreibt  $GZF$  die Menge aller Zielfunktionen, die sich auf das Gesamtmodell beziehen.

Betrachtet man die Kalibrierung einer ABS als allgemeines Black Box Kalibrierungsproblem können zwei Problemaspekte identifiziert werden, die in ABS das Kalibrierungsproblem deutlich erschweren können:

<sup>1</sup>Im weiteren Verlauf der Arbeit werden noch die Abkürzungen  $IVM$  („ideale Verhaltensmodelle“) und  $IZF$  („ideale Zielfunktionen“) für Teil-Modelle eines Mikro-Modells eingeführt. Diese sind nicht mit den hier genannten  $IVK$  zu verwechseln, die zur Bewertung individuellen Verhaltens von Agenten dienen.



- **Größe des Parametersuchraums:** Müssen viele  $IAMF$  und  $UMF$  kalibriert werden, kann dies zu sehr großen Parametersuchräumen führen.
- **Dauer einzelner Simulationsläufe:** Werden sehr viele Agentenmodelle  $IAM$  simuliert und/oder habe diese komplexe Entscheidungsmechanismen, kann die Dauer eines einzelnen Simulationslaufes signifikant zunehmen. Als Folge wird es schwierig, in begrenzter Zeit für eine Black Box Kalibrierung eine ausreichende Menge an Parameter-einstellungen testen zu können.

#### 4.1.2.2 White-Box-Zerlegung für ABS

Mit White-Box-Zerlegungen wird versucht, ein Kalibrierungsproblem durch die Betrachtung abstrahierter Probleme oder durch seine Zerlegung in einfacher zu lösende Teilprobleme zu reduzieren (siehe Definition 14). Es ist klar, dass die Zerlegbarkeit von  $KP_{ABS}$  stark von der Modellstruktur und dem verfügbaren Wissen über die  $VK$  des Modells abhängt, also dem Wissen über valides Verhalten einzelner Modellteile. Nur dieses Wissen erlaubt es, einzelne Modellteile zu validieren und potentiell Teilmodelle isoliert zu simulieren.

Betrachtet man die Definition eines Agenten-basierten Modells (siehe Definition 6) kann leicht ein natürlicher Ansatzpunkt zur Zerlegung von  $KP_{ABS}$  identifiziert werden: Einzelne Agenten und das Umweltmodell. Existieren z.B.  $IVK$  für die einzelnen Agenten, durch welche ihr Verhalten unabhängig voneinander validiert werden kann, könnte versucht werden,  $KP_{ABS}$  in einzelne  $TKP_{IAMF}$  für jeden Agenten oder zumindest in  $TKP_{AMF}$  bezüglich Teilgruppen der simulierten Agenten zu zerlegen (siehe Begriffsbestimmung 14):

$$KP_{ABS} = (\{TKP_{UMF}, TKP_{AMF_1}, \dots, TKP_{AMF_l}\}, \prec), l \in \mathbb{N}$$

- **$TKP_{UMF}$ :** Das Teilkalibrierungsproblem für Umweltmodellfamilie.  $TKP_{UMF}$  kann gegebenenfalls weiter zerlegt werden.
- **$TKP_{AMF_i}$ :** Ein Teilkalibrierungsproblem auf der Menge der Agentenmodellfamilien. Dabei ist  $AMF_i \subseteq \{IAMF_1, \dots, IAMF_n\}$  eine Teilmenge aller  $n \in \mathbb{N}$  zu kalibrierenden Agentenfamilien.

Die Zerlegbarkeit hängt entscheidend von den verfügbaren  $VK$  und den Interaktionen und Beziehungen zwischen den Agenten ab. Eine Zerlegung von  $KP_{ABS}$  in Teilkalibrierungsprobleme bezüglich einzelner Agentenmodellfamilien ist nur dann möglich, wenn für diese isolierte  $IVK$  verfügbar sind.

Es existieren verschiedene Ansatzpunkte zur Zerlegung und Abstraktion speziell für Agenten-basierte Simulationsmodelle, z.B. die Aggregation von Agentengruppen und ihres Verhaltens oder die getrennte Kalibrierung von Teilverhaltensweisen der Agenten anstelle der Kalibrierung ihres Gesamtverhaltens. Im Rahmen dieser Arbeit wird hierauf nicht ausführlich eingegangen. Entsprechende Ansatzpunkte wurden von mir jedoch in [37] veröffentlicht.

#### 4.1.2.3 Wissensbasierter Ansatz zur Kalibrierung von ABS

In Abschnitt 3.1.5.2 wurde die Möglichkeit eines wissensintensiven Kalibrierungsvorgehens mit Änderungsregeln für die Werte der zu kalibrierenden Parameter beschrieben. Als Beispiele wurden der Einsatz von Expertensystemen und von Lernverfahren genannt, die

eine Problem-spezifische Adaption von Parameterwerten umsetzen können. Wie in Abschnitt 3.1.5.2 beschrieben, ist ein Lernverfahren eigentlich ein spezialisierter Parameter-optimierungsalgorithmus und setzt somit ein **wissensbasiertes Kalibrierungsverfahren** um. Lernverfahren finden häufig Einsatz bei der Kalibrierung von ABS, bzw. der Optimierung des Verhaltens von MAS (wobei hier auch besonders die Möglichkeit einer dynamischen Adaption an sich ändernde Bedingungen eine Rolle für ABS spielt). Relevante Lernverfahren werden in Abschnitt 5.1.5 vorgestellt. Bei der Verwendung eines wissensbasierten Ansatzes wird explizit mit Hilfe von Änderungsregeln beschrieben, wie die individuellen Agenten ihr Verhalten an eine bestimmte Situation anpassen sollen. Dabei werden z.B. explizite Gradienten-basierte Änderungsregeln für die individuellen Agenten aufgestellt, mit der sie sich entlang ihrer individuellen Performanzfunktion (hier  $IVK$ ) adaptieren (siehe hierzu auch den Abschnitt 5.1.5 zu Lernverfahren).

**In dieser Arbeit wird der Begriff „wissensbasierter Ansatz“ verwendet, um allgemein zu beschreiben, dass der Nutzer explizites Kalibrierungswissen für die Agenten in Form von individuellen Änderungsregeln formalisieren kann.** Während der Kalibrierung werden die individuellen Parameterwerte der Agenten anhand dieser Regeln angepasst. Beispielsweise würde während der Kalibrierung zunächst ein Simulationslauf durchgeführt und die Validität des Verhaltens einzelner Agenten anhand individueller Validitätskriterien  $IVK$  bewertet. Auf der Basis dieser  $IVK$  können nach Ende des Laufs für jeden einzelnen Agenten Änderungsregeln angewendet werden, um dessen Parameterwerte zu adaptieren. Dann wird ein erneuter Simulationslauf auf der Basis der adaptierten Werte durchgeführt usw. Das Verfahren endet nach einer bestimmten Anzahl an Iterationen oder wenn die Parameterwerte der Agenten konvergieren.

Ein ABS-Kalibrierungsproblem, bei dem nur die Parameter einer Menge von Modellfamilien einzelner Agentenmodelle mit Änderungsregeln kalibriert werden sollen und das Umweltmodell bereits kalibriert ist (UMF enthält nur ein einziges Modell), hat z.B. die Form:

$$KP_{ABS}^{WB} = (((\{UM\}, \{IAMF_1, \dots, IAMF_n\}), VK), \ddot{A}R), n \in \mathbb{N}$$

Dabei wird die Kalibrierung mit Hilfe einer Menge von Änderungsfunktionen  $\ddot{A}R$  bezüglich der Parameterwerte einzelner Agentenfamilien durchgeführt. Für eine konkrete Agentenmodellfamilie  $IAMF_i = (S_i, V_i, P_i) \in \{IAMF_1, \dots, IAMF_n\}$  ( $n \in \mathbb{N}$ ) hätten zugehörige Änderungsregeln auf der Basis eines individuellen Validierungskriteriums  $IVK_{IAM_i}$  bezüglich  $IAM_i$  die Form:

$$\ddot{A}R : W(IVK_{IAM_{i_1}}) \times \dots \times W(IVK_{i_m}) \times W(P_{i_1}) \times \dots \times W(P_{i_n}) \rightarrow \\ W(P_{IAM_{i_1}}) \times \dots \times W(P_{IAM_{i_n}}), n, m \in \mathbb{N}$$

Wie in Abschnitt 4.1.2.3 beschrieben, erfordert die Anwendung wissensbasierter Kalibrierungsverfahren ausreichendes Wissen über die Wirkungszusammenhänge des modellierten Systems, was nicht notwendigerweise gegeben ist. Gerade in Simulationen emergenter Phänomene ist es oft zentrales Ziel der Simulation diese Wirkungszusammenhänge erst herauszufinden.

### 4.1.3 Spezifische Probleme bei der Kalibrierung von ABS

Spezifische Kalibrierungsaspekte von  $KP_{ABS}$  resultieren daraus, dass ABS auf der Mikroebene entworfen werden, d.h. es gilt **Beobachtungseinheit = Modellierungseinheit**. Die

Menge *VK* kann Validierungskriterien auf unterschiedlichen Beobachtungsebenen enthalten. Auf aggregierter Ebene können dies *VK* bezüglich des Gesamtverhaltens der Simulation sein. *VK* können aber auch bezüglich des Verhaltens eines einzelnen Agenten in der Simulation spezifiziert werden. Je nach Verfügbarkeit von Validierungs- und Modellierungswissen erlaubt dies unterschiedliche Herangehensweisen an  $KP_{ABS}$ .

Als Beispiel für spezifische Probleme bei der Kalibrierung von ABS kann ein Szenario bei der Entwicklung von ABS in den Gesellschafts- und Wirtschaftswissenschaften betrachtet werden. Ein besonderer Anwendungsfall von ABS in den Gesellschafts- und Wirtschaftswissenschaften besteht in der Abbildung und Analyse von Theorien über kognitive Prozesse von Individuen der realen Welt. Ist man bei der Entwicklung solcher Modelle in der Situation, dass noch keine etablierten Theorien zur Kognition verfügbar sind, so sind bei der Erstellung solcher ABS valide Strukturen unbekannt. Als Folge kann praktisch nicht bestimmt werden, welche Daten zur Beschreibung validen Verhaltens eines realen Systems erhoben und mit welcher Methodik die Datenerhebung durchgeführt werden muss. Untersuchungen, ob sich bestimmte globale Systemverhaltensweisen aus modellierten Interaktionen auf lokaler Ebene erzeugen lassen, werden sehr schwierig, wenn der tatsächliche Wirkungszusammenhang zwischen lokaler und globaler Ebene unbekannt ist [49]. Auch Doran weist auf die enormen Probleme bei der Durchführung der ABS-Kalibrierung bezüglich Aggregierungsgrad, Komplexität der kognitiven Strukturen und die große Menge an zu kalibrierenden Eingabeparametern hin. [30].

#### 4.1.3.1 Kalibrierung auf mehreren Beobachtungsebenen

Wie bei der Validierung von ABS (siehe Abschnitt 2.4.4.2), muss es auch bei der Kalibrierung von ABS notwendig sein, diese auf mehreren Beobachtungsebenen, (z.B. der Makro- und der Mikroebene) durchzuführen.

##### Kalibrierung auf der Mikro-Ebene

Auf der Mikro-Ebene wird versucht das Verhalten der simulierten Agenten so zu kalibrieren, dass die individuelle Verhaltensweise jedes einzelnen Agenten für sich betrachtet valide ist. Hierfür muss der Nutzer individuelle Validierungskriterien *IVK* für jeden simulierten Agenten spezifizieren. Anhand dieser *IVK* kann das Verhalten jedes einzelnen Agenten validiert werden. Valides individuelles Agentenverhalten auf der Mikro-Ebene garantiert aber noch nicht, dass das Gesamtverhalten des Simulationsmodells, das aus dem Verhalten der einzelnen Agenten und ihren Interaktionen im simulierten Umweltmodell entsteht, dem des Originalsystems entspricht. Stattdessen muss das Verhalten der Agenten so kalibriert werden, dass aus den Interaktionen der Agenten valides Gesamtverhalten entsteht.

##### Kalibrierung auf der Makro-Ebene

Beobachtet und validiert man eine ABS nur auf der Ebene des Gesamtverhaltens (= Makro-Ebene) können die konkreten Agentenaktionen auf der Mikroebene möglicherweise invalide sein. Dies ist z.B. der Fall, wenn verschiedene Verhaltensweisen auf der Mikro-Ebene das gleiche Verhalten auf der Makro-Ebene produzieren, aber nicht alle dieser

Verhaltensweisen auf der Mikro-Ebene valide sind. Entsprechend besteht bei der Kalibrierung einer ABS rein auf der Basis eines Gesamtvalidierungskriteriums *GVK* die Gefahr, dass auf der Mikro-Ebene beliebiges Verhalten eingestellt wird.

### Beispiel für Kalibrierung auf Mikro- und Makro-Ebene

Im einführenden Beispiel der Einkaufssimulation aus Kapitel 3.1 muss das Simulationsverhalten auf Makroebene so kalibriert werden, dass die aus statistischen Daten bekannten Geschäftsumsätze optimal reproduziert werden. In diesem Fall könnte *GVK* z.B. die Summe der Abweichungen aller simulierter Geschäftsumsätze von den Originaldaten messen. Auf der Mikroebene soll das Verhalten der einzelnen Einkaufsagenten so eingestellt werden, dass glaubwürdiges Einkaufsverhalten entsteht. Je nach Wohnort, Alter, Geschlecht, etc. könnte für jeden Einkaufsagenten ein *IVK* entworfen werden, anhand dessen beurteilt wird, ob das Verhalten des Agenten valide ist.

Aus Makrosicht wäre es am einfachsten, das Verhalten der Agenten so einzustellen, dass sie in jedem Geschäft gerade so viel Geld ausgeben, dass dessen Originalumsatz reproduziert wird. Werden jedoch die *IVK* ignoriert kann ein solches Vorgehen leicht zu invalidem Verhalten auf Mikroebene führen. Beispielsweise könnten die Agenten in viel zu großen Entfernungen zu ihrem Wohnort einkaufen gehen, was jedoch auf Makro-Ebene irrelevant ist, solange die Geschäftsumsätze reproduziert werden. Umgekehrt können die einzelnen Agentenmodelle auf Mikroebene basierend auf Umfragen über menschliches Einkaufsverhalten plausibel entworfen worden sein, ohne dass aus validem Mikroverhalten zwingend valides Makroverhalten folgen muss.

### Problematik bei der Kalibrierung auf mehreren Beobachtungsebenen

Die vorherigen Beschreibungen und das Beispiel sollen verdeutlichen, dass eine Kalibrierung bezüglich mehrerer Beobachtungsebenen nur dann einfach durchgeführt werden kann, wenn die *IVK* und die *GVK* der jeweiligen Ebenen zueinander passen. Aufgrund des potentiell hohen Detailgrades von ABS und ihrem Einsatz zur Untersuchung emergenter Phänomene (siehe Abschnitt 2.4.4.2) kann es jedoch schwierig sein, den **konkreten Zusammenhang zwischen validem Verhalten auf Mikro- und Makroebene** zu spezifizieren. Stellt man invalides Gesamtverhalten der Simulation fest, ist es schwierig, zu ermitteln, welche individuellen Agentenverhaltensweisen in welcher Weise geändert werden müssen, um das Gesamtverhalten zu verbessern. Abbildung 4.1 veranschaulicht das Problem. Das individuelle Verhalten der einzelnen Agenten während eines Simulationslaufes kann zwar beobachtet werden, es ist aber unklar welche einzelnen Agentenverhaltensweisen das invaliden Gesamtverhalten der Simulation beeinflussen (dünne abwärts gerichtete Pfeile) und daher welche individuellen Agentenparameter und Umweltparameter geändert werden müssen (dicke abwärts gerichtete Pfeile).

Es müssen daher die Wissenslücken über den Zusammenhang zwischen Validierungskriterien auf verschiedenen Beobachtungsebenen überbrückt werden. Dies bedeutet z.B., dass ausgehend von invalidem Verhalten auf aggregierten Beobachtungsebenen, Agentenmodelle auf der Mikro-Ebene identifiziert werden müssen, deren Verhalten für valides Gesamtverhaltens problematisch ist. Die Parameterwerte dieser Agenten müssen auf der Mikro-Ebene geeignet abgeändert werden, um valideres Gesamtverhalten zu erreichen.

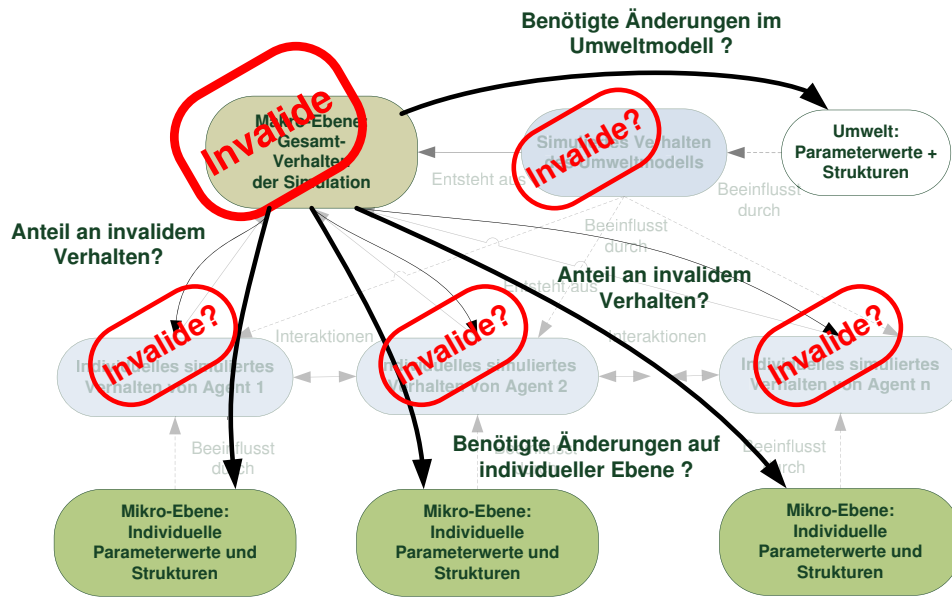


Abbildung 4.1: Das Problem der Identifikation relevanter Modelländerungen in der Kalibrierung von ABS.

Gleichzeitig muss aber sichergestellt bleiben, dass das individuelle Verhalten auf der Mikro-Ebene weiterhin für sich valide bleibt.

Das Vorhandensein dieser Makro-Mikro-Lücke ist ein inhärentes Problem von ABS, da in dieser Form der Simulation das Gesamtverhalten nicht konkret im Modell beschrieben, sondern aus dem Mikroverhalten „generiert“ wird. Es existiert kein explizites Wissen, wie ein valides Mikro-Verhalten aussehen muss, um ein valides Makroverhalten zu erzeugen. Dies wird zusätzlich erschwert durch chaotische Parameterstrukturen bei denen kleine Wertänderungen große Auswirkungen haben können, insbesondere wenn die Annahme homogener Parameterwerte gemacht wird. Weitere Probleme sind nicht-lineare Wirkungszusammenhänge, Zufallseinflüsse und einen beliebigen Detaillierungsgrad der Mikro-Ebene (siehe z.B. Klügl [79]). Dass dieses Problem typisch für ABS ist, kann man auch sehr gut einer Untersuchung von Heath, Hill und Ciarall [56] entnehmen. Heath, Hill und Ciarall erhoben charakteristische Merkmale über den praktischen Einsatz von ABS. Dabei wurden 279 Veröffentlichungen aus den Jahren 1998 bis 2008 betrachtet. Ein klares Ergebnis der Untersuchung ist, dass für annähernd 100% der untersuchten ABS galt: Entweder es war nur wenig über das modellierte Originalsystem bekannt und das Ziel der Simulationsuntersuchung bestand hauptsächlich darin zu prüfen, ob ein Modellkonzept/Theorie in der Lage ist ein beobachtetes Gesamtverhalten zu erzeugen. Konnte dieser Schritt abgeschlossen werden, wurde das Simulationsverhalten untersucht, um zu verstehen welche charakteristischen Wirkungszusammenhänge existieren und wie durch sie das Gesamtverhalten aus dem Mikroverhalten entstehen kann. In keinem Fall wurde eine ABS zu einem Originalsystem mit initial gut verstandenen Wirkungszusammenhängen entwickelt, bei der klar war, wie das Makro-Verhalten konkret aus dem Mikro-Verhalten generiert wird.

#### 4.1.3.2 Heterogenität in den Parameterwerten der Agenten

Allgemein gilt, dass je größer der Parametersuchraum während der Kalibrierung ist, desto aufwändiger wird es, eine Parametereinstellung zu finden, die valides Simulationsverhalten erzeugen kann. In ABS hängt die Menge der zu kalibrierenden Parameterwerte im Extremfall direkt von der Menge an simulierten Agenten ab. Müssen beispielsweise für jeden einzelnen Agenten unabhängig von den anderen Agenten individuelle Parameterwerte eingestellt werden, so multipliziert sich die Anzahl zu kalibrierender Parameter, im Vergleich zur Verwendung homogener Parameterwerte für alle Agenten einer Klasse, mit der Anzahl simulierter Agenten. Hängt die Anzahl an Parametern eines Agenten von der Anzahl der übrigen Agenten ab (z.B. spezifische Interaktionsparameter für jeden anderen Agenten), steigt die Zahl zu kalibrierender Parameterwerte quadratisch, usw.

#### Black-Box-Kalibrierung heterogener Agentenparameter

Für große Agentenzahlen kann die Kalibrierung heterogener Agentenparameter mit einer Suchraumkomplexität verbunden sein, mit der existierende Black Box Ansätze (siehe Abschnitt 3.2) in begrenzter Zeit nicht mehr sinnvoll umgehen können. Zum einen ist es in größeren Suchräumen wahrscheinlicher, dass in begrenzter Zeit die benötigte Parametereinstellung nicht gefunden werden kann. Zum anderen ist es in einem großen Parametersuchraum wahrscheinlicher unterschiedliche Parametereinstellungen zu finden, die zu ähnlichem globalem Systemverhalten führen (sogenannte multiple lokale oder globale Optima des Suchraumes). Muss die Kalibrierung vor Ende des jeweiligen Suchalgorithmus abgebrochen werden, kann es schwierig sein zu bestimmen:

- Ob die aktuell „beste“ Parametereinstellung die gesuchte ist.
- Ob es mehrere Parametereinstellungen gibt, die zu ähnlich validem Simulationsverhalten führen.

#### Black-Box-Kalibrierung homogener Agentenparameter

Kann man die Annahme treffen, dass für alle Agenten einer Klasse homogene, d.h. die gleichen, Parameterwerte verwendet werden können, ermöglicht dies die drastische Vereinfachung des Suchraums im Vergleich zur Verwendung unterschiedlicher Parameterwerte für jeden einzelnen Agenten. In einem reduzierten Suchraum ist die Wahrscheinlichkeit, dass ein verwendeter Black Box Ansatz in begrenzter Zeit eine global „beste“ Parametereinstellung finden kann, erhöht. Ist die gemachte Annahme aber falsch, kann eine solche Festlegung auf homogene Parameterwerte das Kalibrierungsproblem unlösbar machen: Egal welche Parametereinstellung gewählt wird, das resultierende Simulationsverhalten kann immer invalide bleiben. Dies ist beispielsweise der Fall, wenn die falsche Annahme getroffen wird, dass alle Agenten identische Verhaltensweisen haben sollten, aber aufgrund unterschiedlicher räumlicher Gegebenheiten für die einzelnen simulierten Agenten einer Klasse unterschiedliche Verhaltensweisen (= Parameterwerte) benötigt werden, um valides Gesamtverhalten der Simulation zu erzeugen.

### Problematik einer geeigneten Heterogenitätswahl für die Parameterwerte der Agenten

Die tatsächlich benötigte Menge und Art an Heterogenität in den Parameterwerten der Agenten muss sich nicht auf eines der beiden Extreme vollständiger Homogenität oder Heterogenität beschränken. Möglicherweise werden für manche Agentengruppen homogene und für andere heterogene Parameterwerte benötigt. In diesem Fall genügt es weniger Parameterwerte zu kalibrieren, als bei vollständig heterogenen Agentenparametern, es müssen aber mehr Parameterwerte kalibriert werden, als bei der Verwendung vollständig homogener Parameterwerte. Ist tatsächlich benötigte Heterogenität in den Parameterwerten bekannt, kann das zu lösende Kalibrierungsproblem gerade so gewählt werden, dass der Parametersuchraum eine minimale Größe bezüglich der Erreichung von validem Verhalten hat. Ist die benötigte Heterogenität in den Parameterwerten nicht bekannt, muss theoretisch mit voller Heterogenität in den Parameterwerten kalibriert werden, da hierdurch das Kalibrierungsproblem auf jeden Fall gelöst werden kann. Dies kann wie beschrieben zu einer Explosion des Suchraumes führen.

### Heterogenitätssuche als Teil des Kalibrierungsproblems

Eine Alternative zur Kalibrierung vollständig heterogener Parameterwerte bei unbekannter benötigter Heterogenität besteht darin, die **Menge und Art an benötigter Heterogenität in den Parameterwerten selbst als Teil des Kalibrierungsproblems aufzufassen**. In diesem Fall sind die Größe des Parametersuchraumes und seine Dimensionen zunächst unbekannt und müssen während der Kalibrierung erst ermittelt werden. Die Hoffnung dabei besteht darin, dass ein solches Kalibrierungsproblem mit Heterogenitätssuche immer noch einfacher zu lösen ist, als das Kalibrierungsproblem mit vollständiger Heterogenität in den Parameterwerten.

### Beispielproblem

Im Einkaufsbeispiel kann potentiell jeder Einkaufsagent ein eigenes Einkaufsverhalten benötigen. Bei 100.000 Haushaltsagenten des simulierten Gebietes und 3 Parametern pro Agent müsste bei völlig heterogenen Parameterwerten während der Kalibrierung ein Suchraum aus  $100.000^3 = 10^{15}$  Parametern durchforstet werden. Verwendet man homogene Parameterwerte für alle Agenten müssen insgesamt nur 3 Parameterwerte eingestellt werden. Eine gute Lösung könnte eine Unterteilung der Agenten bezüglich bestimmter geographischer Regionen oder demographischer Kategorien sein. Werden  $n$  Regionen/Kategorien verwendet müssen  $n^3$  Parameter kalibriert werden. Bei diesem Vorgehen stellt sich das Problem in welcher Weise die Zuordnung der Agenten zu entsprechenden Gruppen mit gleichen Parameterwerten durchgeführt werden soll.

#### 4.1.3.3 Kalibrierung offener ABS

Die Strukturen von ABS können variabel sein und sich dynamisch während eines Simulationslaufes ändern [118]. In offenen ABS können während eines Simulationslaufes Agenten zur Simulation hinzukommen oder diese verlassen. Beispielsweise können Agenten „geboren“, „erzeugt“ werden oder die simulierte Umwelt „betreten“. Umgekehrt können

Agenten auch „sterben“ oder, falls z.B. in einer biologischen Simulation das Wachstum einer Spezies nachgestellt wird, eine andere Entwicklungsstufe einnehmen. Theoretisch kann die dynamische Variabilität der Modellstrukturen durch eine Modellübergangsfunktion formalisiert werden (siehe Uhrmacher [118]). Die Modellübergangsfunktion bestimmt für eine Modellstruktur und den Zustand des Modells während der Simulation, welche nächste Modellstruktur auf eine vorherige folgt. Bezogen auf die Kalibrierung von Modellparametern bedeutet die Offenheit von ABS, dass sich die Menge zu kalibrierender Parameterwerte erweitern kann (z.B. sind zu Beginn noch nicht alle zu kalibrierenden Agenten Teil des Modells) oder für Agenten können während eines Simulationslaufes unterschiedliche Parameterwerte relevant werden. Werden stochastische Simulationsmodelle simuliert, kann es zusätzlich sein, dass die Menge zu kalibrierender Parameterwerte zwischen zwei Simulationsläufen unterschiedlich ist, z.B. wenn aufgrund von Zufallseinflüssen beim ersten Simulationslauf andere Agenten Teil der Simulation sind, als im Zweiten.

Kann eine entsprechende Modellübergangsfunktion formalisiert werden, so können alle möglichen Modellstrukturen und damit alle möglichen zu kalibrierenden Parameterwerte explizit ermittelt werden. Der maximale Parametersuchraum ist somit zu Beginn der Kalibrierung bekannt. Es kann für den Modellierer oder Fachexperten jedoch problematisch sein, eine solche Modellübergangsfunktion zu formalisieren. Dies ist z.B. der Fall, wenn sehr komplexe Interaktionen oder sehr viele stochastische Einflüsse mit vielen Wechselwirkungen die strukturelle Variabilität des Modells beeinflussen. In solchen Fällen ist es einem Modellierer oder Fachexperten zum Startzeitpunkt einer ABS nicht notwendigerweise bekannt, welche Agenten während eines Simulationslaufes Teil der Simulation sind und damit für welche Agenten Parameterwerte eingestellt werden müssen, bzw. ob in den folgenden Simulationsläufen der Kalibrierung unter Umständen andere, bzw. zusätzliche Parameterwerte kalibriert werden müssen. Effektiv bedeutet dies, dass zu Beginn der Kalibrierung die Größe des Parametersuchraumes nicht bekannt ist und der Parametersuchraum im Verlauf der Kalibrierung anwachsen kann. Simulierte Agenten, die während eines Simulationslaufes das System verlassen verkleinern den Parametersuchraum hingegen nicht. Erst am Ende eines Simulationslaufes kann bewertet werden, ob das Gesamtverhalten der Simulation valide war und ob die Parameterwerte der Agenten angepasst werden müssen. Es ist dabei egal wann welcher Agent Teil des simulierten Systems war.

#### 4.1.3.4 Umgang mit verrauschten Daten während der Kalibrierung von ABS

Empirische Daten bilden eine wichtige Grundlage jeder Form von Simulationsmodellen realer Systeme. Je detaillierter das System abgebildet werden soll, desto schwieriger wird es, Daten in ausreichender Menge und Qualität erheben zu können, um das System exakt abzubilden. Man unterscheidet **quantitative und qualitative Daten**. Quantitative Daten können das Originalsystem mit Hilfe eines Zahlenwertes widerspiegeln. Qualitative Daten beschreiben Wissen über Systemaspekte, die nicht exakt gemessen werden können. Oft sind die verfügbaren Daten fehlerhaft, unvollständig, inkonsistent oder bezüglich einer konkreten Fragestellung nicht vollständig passend. Solche Daten werden als „verrauscht“ bezeichnet. Wird Verrauschung bei der Validierung nicht erkannt, ignoriert oder kann sie nicht kompensiert werden, besteht die Gefahr, dass kein sinnvolles Systemverhalten erzeugt werden kann. Wird nun nach Modellierung und Kalibrierung auf anderem Wege invalides Modellverhalten festgestellt, kann es schwierig sein festzustellen, ob die Ursache invaliden Modellverhaltens in den verrauschten Daten, ungeeignet eingestellten Steuerungsparametern oder



der ungeeignet entworfenen Modellstruktur gesucht werden muss.

### Ursachen und Probleme von Datenrauschen

Die Verrauschung von Daten zu einem Simulationsproblems hängt mit der Art ihrer Erhebung zusammen. Je nach Bedarf und Möglichkeiten können unterschiedliche Techniken zur Erhebung von Daten über ein MAS der realen Welt zum Einsatz kommen [14]. Dabei können unterschiedliche Arten von Verrauschung auftreten. Werden bei der Datenerhebung falsche Messmethodiken angewendet oder ungeeignete Stichproben erhoben, dann können Verrauschungsprobleme auftreten. Solche Probleme sind z.B. Inkonsistenzen in den erhobenen Daten, zur Erstellung eines konkreten Modells unpassende Daten (z.B. zu aggregierte oder zu spezifische Daten), zu subjektiv erhobene Daten oder einfach das Fehlen relevanter Datenwerte (für einen ausführlichen Überblick siehe [14]). Es kann schwierig sein bei der Modellerstellung die Verrauschung verfügbarer Daten einzuschätzen, wenn das Modell auf der Grundlage von nicht selbst erhobenen Daten erstellt werden muss und die Erhebungsmethodik der Daten nicht ausreichend genau dokumentiert ist.

### Robuste Kalibrierung für ABS mit verrauschten Daten

Ein robustes Kalibrierungsverfahren wird benötigt, wenn der Modellierer keine direkte Kontrolle über die Menge und Qualität der Daten hat. Speziell im Bereich der Erstellung von ABS stellt sich das Problem, dass Daten auf dem Detaillierungsniveau der Mikro-Ebene nur schwierig erfasst, aufbereitet und getestet werden können. Es ist auf dem ABS-Detaillierungsniveau kaum möglich Verrauschung in verfügbaren Daten über das individuelle Agentenverhalten durch Mittelwertbildung über verschiedene am Originalsystem erhobene Datensätze zu verringern. Der Grund hierfür ist, dass es schwierig ist die hierfür benötigte große Datenmenge auf so hoher Detaillierungsebene zu erheben. Das Problem stark verrauschter Datenwerte ist besonders kritisch, wenn diese Daten zur Festlegung individueller Validierungskriterien  $IVK$  zu einzelnen Agentenmodellen auf der Mikro-Ebene verwendet werden. In diesem Fall ist es schwierig, Datenrauschen bereits vor der Kalibrierung zu erkennen, da unklar ist, welches individuelle Agentenverhalten benötigt wird, um ein bestimmtes Gesamtverhalten zu erzeugen (siehe Makro-Mikro-Kalibrierungsproblem). Unter einem Kalibrierungsproblem für ABS, das mit einem robusten Kalibrierungsverfahren gelöst werden muss, wird daher in dieser Arbeit ein Problem verstanden, bei dem nicht nur bestimmte Parameterwerte eingestellt werden müssen, sondern bei dem während der Kalibrierung individuelle Validierungskriterien auf der Mikro-Ebene identifiziert und geeignet ausgefiltert werden müssen, so dass das resultierende Simulationsverhalten nicht durch deren Einfluss verzerrt wird.

Ein solches ABS-Kalibrierungsproblem für ein robustes Kalibrierungsverfahren wird als  $KP_{ABS}^{Robust}$  bezeichnet. Das Problem unterscheidet sich vom allgemeinen  $KP_{ABS}$  dadurch, dass zusätzlich Wissen benötigt wird, welche  $IVK$  als korrekt und welche als potentiell verrauscht eingeschätzt werden. Die Menge  $IVK = \{IVK_1, \dots, IVK_w\}$  lasse sich einteilen in  $IVK = SIVK \cup UIVK$  mit  $SIVK \cap UIVK = \emptyset$ :

- $SIVK$ : Teilmenge der als korrekt eingestuften  $IVK$ .

- $UIVK$ : Teilmenge der  $IVK$ , deren Korrektheit als unsicher eingestuft wird.

Die mögliche Folge eines Lösungsversuchs von  $KP_{Robust}$  mit „normaler“ Kalibrierung wäre, dass nur „scheinbare Validität“ erzeugt würde. D.h. dass das Verhalten der Agenten nach der Kalibrierung zwar valide bezüglich  $IVK$ , jedoch das Gesamtverhalten invalide bezüglich der Gesamtmenge  $\{VK_1, \dots, VK_v, IVK_1, \dots, IVK_w\}$  ist. Das invalide Gesamtverhalten folgt dabei nicht aus unvollständiger Kalibrierung der Agenten, sondern aus den fehlerhaften  $IVK$ .

### Existierende Verfahren zur robusten Kalibrierung

Eine gängige und erfolgreiche Herangehensweise zum Umgang mit gleichmäßiger Verrauschung in den Kalibrierungsdaten ist die Verwendung eines *Maximum Likelihood Schätzers* [55]. Der Fall gleichmäßigen Datenrauschens steht daher nicht im direkten Fokus dieser Arbeit. Stattdessen wird der Fall adressiert, dass einzelne Validierungskriterien besonders stark und potentiell sehr unterschiedlich verrauscht sind.

Das Grundprinzip der hier vorgestellten robusten Kalibrierung besteht darin fehlerhafte Datenelemente, die für die Kalibrierung eines Modellverhaltens verwendet werden, zu „erkennen“. Die Kalibrierung wird so durchgeführt, dass das entstehende Modellverhalten durch die fehlerhaften Datenelemente möglichst wenig verzerrt wird. Existiert nicht nur gleichmäßiges Datenrauschen, sondern auch einzelne starke Verrauschungsausreißer in den Datenwerten, so kann das Kalibrierungsergebnis trotz Einsatz eines Maximum-Likelihood Schätzers verzerrt werden, wenn diese Ausreißer nicht vor der Kalibrierung entfernt werden. Im Bereich der automatischen Bildverarbeitung wurde für ein analoges Problem das RANSAC-Verfahren (RANDOM SAMPLING CONSENSUS) entwickelt [55]. Es dient dazu, z.B. Transformationsmatrizen oder Abbildungsmatrizen aus Bilddaten zu schätzen. Soll beispielsweise aus Korrespondenzpunkten zweier Aufnahmen einer Szene eine Abbildungsmatrix zwischen diesen Aufnahmen geschätzt werden, wird versucht, mit Hilfe des RANSAC-Verfahrens fehlerhafte Korrespondenzpunkte auszufiltern.

Das Verfahren wird im Folgenden genauer beschrieben, weil es eine wichtige Grundlage für eine in der Arbeit entwickelten Kalibrierungstechniken zur robusten Kalibrierung von ABS darstellt.

### RANSAC

In diesem Abschnitt wird das RANSAC-Verfahren aus der automatischen Bildverarbeitung nach Hartley [55] beschrieben. Die Idee des RANSAC-Verfahrens besteht darin, zur Kalibrierung der Parameter einer zu bestimmenden Transformationsmatrix zwischen zwei Bildern nicht alle verfügbaren Datenpunkte, sondern die kleinste benötigte Teilmenge der verfügbaren Daten (sogenannte minimale Teilmenge) zu verwenden, aus der theoretisch noch eine korrekte Transformationsmatrix ermittelt werden kann. Die Hoffnung dabei ist, dass in dieser minimalen Teilmenge nur korrekte Datenpunkte enthalten sind und daher die Matrix korrekt ermittelt werden kann. Da die Menge der fehlerhaften Datenpunkte unbekannt ist, kann eine geeignete minimale Teilmenge nicht einfach bestimmt werden. **Zur Lösung des Problems werden in mehreren Iterationen immer wieder zufällig Datenpunkte gewählt, auf Basis dieser Kandidatenteilmenge das Modell der Transformationsmatrix kalibriert und dieses Modell validiert.**

Die Kalibrierung wird in zwei Stufen durchgeführt [55]:

1. Während der Unterkalibrierung bezüglich der gewählten Teilmenge wird die Validität des Modells durch die Abweichung der berechneten Datenpunkte von den Originaldatenpunkten bewertet (z.B. mittels MLS).
2. Nach dem Ende eines Unterkalibrierungslaufs bezüglich einer Teilmenge wird die Qualität des Modells bezüglich aller verfügbaren Datenpunkte (also nicht nur der ausgewählten) bewertet. Dabei ist nicht die konkrete Abweichung von den Datenpunkten relevant, sondern die Bewertung erfolgt über die Anzahl an Datenpunkten, die „gut genug“ reproduziert werden konnten. Hierzu wird vom Nutzer ein Schwellwert für die maximal erlaubte Abweichung zwischen einem errechneten Datenpunkt und seinem Korrespondenzpunkt in den Originaldaten angegeben. Liegt die Abweichung innerhalb des Schwellwertes, gilt der Originalpunkt als korrekt reproduziert, sonst nicht. Die Bewertung eines Modells bezüglich aller Datenpunkte wird über die Anzahl an Datenpunkten, welche vom Matrixmodell korrekt abgebildet werden, durchgeführt. Es wird also dasjenige aus einer minimalen Teilmenge berechnete Modell gesucht, welches am besten zu allen Daten passt.

Nach dem Ende der Kalibrierungsiterationen, d.h. wenn eine geeignete minimale Teilmenge identifiziert wurde, wird noch eine Verfeinerungskalibrierung des validesten bekannten Modells durchgeführt. Als Validierungsdaten werden alle Datenpunkte eingesetzt, welche zuvor als zu diesem Modell „passend“ identifiziert worden waren.

Es existieren noch weitere robuste Kalibrierungsverfahren, wie z.B. das LMS-Verfahren (Least Median of Squares) mit unterschiedlichen Stärken und Schwächen. Der Vorteil des RANSAC-Verfahrens besteht darin, dass es auch bei mehr als 50% verrauschter Datenpunkte angewendet werden kann.

#### 4.1.4 Zusammenfassung: Kalibrierungsprobleme in ABS

In diesem Abschnitt wurden verschiedene Problemaspekte bei der Kalibrierung von Agenten-basierten Simulationen beschrieben. Die Kernprobleme, aus denen sich alle übrigen Probleme ableiten sind:

- Sehr große Parametersuchräume, wenn heterogene Parameterwerte für individuelle Agenten gesucht werden müssen.
- Aufwändige Validierung des Simulationsverhaltens während der Kalibrierung, da auf verschiedenen Beobachtungsebenen bis hin zur individuellen Ebene validiert werden muss.
- Umfangreiches benötigtes Wissen über das Originalsystem zur Erstellung einer validen ABS. Ist entsprechendes Wissen nicht, nur lückenhaft oder nur verrauscht vorhanden, resultieren daraus verschiedene Probleme bei Modellierung und Kalibrierung.
- Ein wichtiger Problempunkt ergibt sich aus dem Problem die Validierung auf Makro- und Mikro-Ebene durchführen zu müssen. Die Auswirkung der Validierungskriterien bezüglich des Mikro-Verhaltens auf das Gesamtverhalten zu untersuchen ist relativ

einfach: Die individuellen Agenten werden ausschließlich bezüglich ihrer individuellen Validierungskriterien (*IVK*) kalibriert und das in der Simulation resultierende Gesamtverhalten untersucht. Der umgekehrte Weg ist schwieriger, da die Kalibrierung einer ABS auf der Basis von Validierungskriterien der Makro-Ebene zu potentiell beliebigem Verhalten auf der Mikro-Ebene führen kann. Es wird daher eine Möglichkeit benötigt, die Auswirkung der Validierungskriterien auf der Makro-Ebene auf valides Verhalten auf der Mikro-Ebene geeignet untersuchen zu können (= Überbrückung der Wissenslücke über den Zusammenhang zwischen validem Makro- und Mikroverhalten). In Zusammenhang mit Datenrauschen in den *IVK* wird weiter eine Möglichkeit benötigt, einzelne *IVK* auf der Basis der Validierungskriterien auf der Makro-Ebene und der übrigen *IVK* hinterfragen zu können.

Einige Kalibrierungsprobleme mit den beschriebenen Problemen können theoretisch auch mit Black Box Optimierungsverfahren gelöst werden. Sind die Berechnungskapazitäten jedoch begrenzt werden Techniken benötigt, mit denen entsprechende Kalibrierungsprobleme effizienter gelöst werden können.

## Kapitel 5

# ABS Kalibrierung: Stand der Forschung

Nachdem im letzten Kapitel Besonderheiten des ABS-Kalibrierungsproblems vorgestellt wurden, wird in diesem Kapitel auf den Stand der Forschung im Bereich der ABS-Kalibrierung eingegangen.

### 5.1 Stand der Forschung: Kalibrierung in ABS

In diesem Abschnitt werden Techniken und Methodologien untersucht, die speziell zur Kalibrierung von ABS eingesetzt oder vorgeschlagen wurden. Es wird gezeigt, dass nur sehr wenige Ansätze spezifische Problemeigenschaften von ABS behandeln. Im Verlauf des Abschnitts wird zum einen aufgezeigt, wie bekannte Verfahren der Simulationsoptimierung auf Agenten-basierte Simulationen übertragen werden und zum anderen untersucht, welche spezielleren Verfahren zur Lösung von Kalibrierungsproblemen in bestimmten Fachbereichen angewendet werden. Die in den folgenden Abschnitten beschriebenen Verfahren sind jeweils nach zugrunde liegendem Kalibrierungskonzept oder Anwendungsbereich zusammengefasst:

- Übertragung existierender Simulationsoptimierungsmethoden
- Uminterpretation populationsbasierter Optimierungsverfahren
- Methodologien in Gesellschafts- und Wirtschaftswissenschaften
- Methodologie in Agenten-basierten Verkehrsmodellen
- Agentenlernen

#### 5.1.1 ABS Kalibrierung mit Black-Box Ansätzen

In diesem Abschnitt wird auf Methodologien zur Kalibrierung Agenten-basierter Modelle eingegangen, die auf Black-Box Ansätzen beruhen. Eine etablierte allgemeine Vorgehensweise existiert nicht, jedoch haben verschiedene Forscher versucht Simulationsoptimierungsmethoden geeignet auf ABS anzuwenden.

### 5.1.1.1 Kalibrierung im SADDE Framework

Carles Sierra [106] verwendet in seinem Framework SADDE (Social Agents Driven By Equations) einen genetischen Algorithmus zur Kalibrierung der Steuerungsparameter der simulierten Agenten. Im Framework wird ein Multiagentensystem entlang vier Entwurfschritten erzeugt. Der finale Schritt erzeugt ein Multiagentensystem, dessen Parameter automatisch mit einem genetischen Suchverfahren eingestellt werden.

Als Beispiel führt Sierra die Kalibrierung eines elektronischen Marktes an. Der Markt besteht aus 30 Produzenten und 60 Konsumenten. Jeder Konsument hat drei, jeder Produzent fünf zu optimierende Parameter. Da Sierra eine homogene Belegung für die Agentenparameter ermittelt, werden vom genetischen Algorithmus lediglich 8 Parameterwerte optimiert. Der problematischere Fall heterogener Parameterwerte (hier wären es 330 zu kalibrierende Parameterwerte) wird von Sierra nicht betrachtet.

### 5.1.1.2 Kalibrierungsframework von Calves und Hutzler

Calvez und Hutzler [22] beschreiben ein Black-Box Framework zur Einstellung von Parameterwerten in ABS, das wie der Ansatz von Sierra auf einem genetischen Algorithmus basiert. Das Framework besteht aus den Schritten (1) Bestimmung des Ziels der Simulationsstudie, (2) Entwicklung des Simulationsmodells, (3) Auswahl der zu kalibrierenden Modellparameter, (4) Auswahl der Zielfunktion, (5) Untersuchung des Einflusses von Zufallselementen auf das Simulationsverhalten, (6) Anpassung des Kalibrierungsverfahrens auf Basis des ermittelten Zufallseinflusses, (7) Untersuchung der Laufzeitkomplexität, (8) Festlegung der Populationsgröße und (9) Globale Optimierung. Dabei werden in Schritt (8) Änderungen an der verwendeten Populationsgröße für den genetischen Algorithmus vorgenommen und entschieden wie viele Auswertungsläufe von Parametereinstellungen gemacht werden sollen. Gegebenenfalls werden diese Analysen alle  $X$  Generationen wiederholt.

Calvez und Hutzler verwenden das Framework, um eine ABS von Ameisen auf Basis zweier nicht näher spezifizierter Zielfunktionen zu kalibrieren. Im Evaluationsmodell werden lediglich zwei Parameterwerte homogen für alle simulierten Agenten eingestellt.

In [23] beschreiben Calvez und Hutzler, die Anwendung eines Black-Box-Intervallschachtelungsverfahrens (siehe Abschnitt 3.2.5.1) auf ABS. Die Kalibrierung erfolgt dabei weiterhin als Black-Box-Verfahren.

### 5.1.1.3 Globale Optimierungsheuristik für ABS nach Gilli und Winkler

Gilli und Winkler [51] stellen eine globale Optimierungsheuristik für Agenten-basierte Modelle vor. Sie verwenden ihr Verfahren, um Agenten-basierte Modelle von Finanzmärkten zu kalibrieren. Gilli und Winkler verwenden eine Kombination aus Simplex Suche (siehe Abschnitt 3.2.2.2) und dem Schwellenakzeptanz-Verfahren<sup>1</sup>. Die Kombination funktioniert so, dass bei der Ermittlung einer neuen Simplexecke diese auch dann akzeptiert wird, wenn sie das Ergebnis etwas verschlechtert (aber eben nicht mehr als ein gegebener Schwellwert).

---

<sup>1</sup> Das Schwellenakzeptanzverfahren ist eine deterministische Variante des Simulated Annealing Algorithmus (siehe Abschnitt 3.2.4.1). Solange eine Verschlechterung während der Suche eine gewisse feste Schwelle nicht überschreitet, wird der entsprechende Wert trotzdem akzeptiert.

Gilli und Winkler nutzen ihren Optimierungsansatz, um ein Agenten-basiertes Modell mit 100 Agenten zu kalibrieren. Bei der Kalibrierung werden aber lediglich drei Modellparameter homogen für alle Agenten eingestellt.

#### 5.1.1.4 Black-Box Optimierung in militärischen ABS

Im militärischen Bereich werden ABS zum Training von Personal und zur Hilfe bei der strategischen Entscheidungsfindung eingesetzt. Dabei werden Strategien und Taktiken gegen virtuelle Feinde getestet, um die Struktur der Streitkräfte und deren benötigte Ausrüstung geeignet zu wählen. Das Ziel solcher Untersuchungen besteht darin, zum einen die schlagkräftigsten Waffensysteme und Kombinationen von Waffensystemen zu finden [26], und zum anderen Auswirkungen verschiedener Angriffsstrategien zu analysieren [63].

- **Identifikation effektiver Kampfstrategien:**

Bei diesem Anwendungsszenario für ABS wird Modellkalibrierung, genauer Optimierung, dazu eingesetzt, um Steuerungsparameter für modellierte Kampftaktiken so einzustellen, dass ein bestimmtes Einsatzziel gegen virtuelle Gegner erreicht wird. Als Optimierungskriterium sollen z.B. möglichst viele eigene Truppen das Hauptquartier der Gegner bei möglichst wenigen eigenen Verlusten erreichen. Zur Ermittlung der sechs homogenen Parameterwerte seiner ABS verwendet Ilachinski einen genetischen Black-Box Algorithmus [63].

- **Analyse von Systemeinflüssen in Kriegsszenarien:**

Bei diesem Anwendungsszenario für ABS wird versucht, ein reales Kriegsszenario so in ein Modell abzubilden, dass dessen Verhalten in der Simulation dem des Originalsystems entspricht. Mit der Simulation sollen dann, die entscheidenden Einflussfaktoren auf den Kampfverlauf identifiziert werden. Dies geschieht mittels umfangreicher statistischer Black-Box Sensitivitätsanalysen (siehe Abschnitte 3.2.6 zu Rasterverfahren und 3.2.6.2 zum Data-Farming).

#### 5.1.1.5 Diskussion der Verfahren

Allen in diesem Abschnitt beschriebenen Kalibrierungsansätzen ist gemeinsam, dass die Tatsache, dass eine ABS kalibriert wird, eigentlich keine Rolle spielt. Die Ansätze behandeln die ABS als Black Box. Sierra verwendet zwar einen Multi-Level-Ansatz zur Beschreibung des Ziel-MAS, die eigentliche Kalibrierung findet aber nur auf einer Modellstufe statt. Keines der Verfahren behandelt spezifische Probleme Agenten-basierter Simulationen (siehe Kapitel 4.1) oder nutzt Eigenschaften von ABS aus, um die Berechnungskomplexität während der Kalibrierung zu reduzieren. Die Anwendung der vorgeschlagenen Kalibrierungsverfahren wird (bis auf das Beispiel der Simulation realer Kriegsszenarien) nur anhand von „Spielzeug-Beispielen“ demonstriert. In den ersten vier beschriebenen Ansätzen werden zwar ABS behandelt, die in den Demonstrationsbeispielen gelösten Kalibrierungsprobleme sind jedoch nicht besonders schwer zu lösen. Die Menge zu kalibrierender Parameter ist nicht abhängig von der Agentenzahl und die Modellstrukturen sind vorgegeben. Es wird jeweils nur eine geringe Anzahl homogener Agentenparameter gesucht, z.B. acht bei Sierra, zwei bei Calvez und drei bei Gilli. Weiter ist die Dauer eines einzelnen Simulationslaufes jeweils recht gering. Sierra macht keine näheren Angaben zur Laufzeit. Calvez beziffert sie mit 15 Sekunden und Gilli schreibt, dass über 1,5 Millionen Modellauswertungen pro

Sekunde durchgeführt wurden. Das letzte beschriebene Anwendungsgebiet (Simulation realer Kriegsszenarien) behandelt detaillierte und berechnungsintensive ABS. Zur Ermittlung des Zusammenhangs zwischen Modellparametern und Simulationsverhalten wird jedoch die Strategie verfolgt, Brute-Force möglichst viele Parametereinstellungen zu bewerten (siehe Abschnitt 3.2.6.2).

## 5.1.2 Uminterpretation populationsbasierter Optimierungsverfahren

Populationsbasierte Optimierungsverfahren (z.B. genetische Algorithmen) verwenden Populationen von Parametereinstellungen, um die beste oder geeignetste Einstellung zu identifizieren. Manche Wissenschaftler verfolgen den Ansatz Populations-basierte Ansätze für Suchprobleme so umzuinterpretieren, dass diese zur Kalibrierung und Optimierung von Agentensystemen eingesetzt werden können.

### 5.1.2.1 Agenten in genetischen Algorithmen

Drogoul beschreibt ein Kalibrierungsverfahren, das er als „Agenten-unterstützte Kalibrierung“ bezeichnet [33]. Drogoul adressiert das Problem der Kalibrierung großer „realistischer“ Populationen von Agenten in Fällen, in denen Daten zur Beschreibung kleiner Teilgruppen der Agenten zur Verfügung stehen. Als Beispiel nennt er die Anwendung von Said [99] in dem die Parameter von virtuellen Einkaufsagenten mit Hilfe eines genetischen Algorithmus kalibriert werden. In dieser Anwendung werden die Parameter einzelner Teilgruppen von Einkaufsagenten so evolviert, dass bestimmte Marktanteile für Marken und bestimmte Verhaltensweisen der Agenten erreicht werden. Der genetische Algorithmus ist so modifiziert, dass jeder Einkaufsagent Teil der evolvierenden Population ist. Das adressierte Kalibrierungsproblem besteht darin die Parameter der Bevölkerung so einzustellen, dass valides Bevölkerungsverhalten auf globaler Beobachtungsebene erzeugt wird.

Said kalibriert die Parameterwerte einer Stadtbevölkerung mit 40.000 Einwohneragenten, für die 100 unterschiedliche Verhaltensprofile und globale Durchschnittswerte zur Verhaltensbeschreibung zur Verfügung stehen. Konkret bedeutet dies, dass nicht für jeden der 40.000 Einwohneragenten eigene Parameterwerte kalibriert werden. Stattdessen werden die Agenten in 100 Teilgruppen mit jeweils homogenen Parameterwerten innerhalb der Gruppen eingeteilt. Die Kalibrierung wird bei Said für jede der Teilgruppen mit Hilfe von normaler Black-Box Optimierung getrennt und unabhängig voneinander durchgeführt (Genetischer Algorithmus bei dem Said die Parametereinstellungen der Population als Agenten bezeichnet). Abhängigkeiten zwischen dem Verhalten einzelner Teilgruppen, die das Gesamtverhalten der Bevölkerung beeinflussen, werden beim Ansatz von Said nicht beachtet. Weiter ist unklar, warum genau 100 Agentengruppen verwendet werden. Ist die Gruppenszahl lediglich aufgrund der Verfügbarkeit der 100 Verhaltensprofile gewählt, besteht die Gefahr, dass die Einteilung in keinem Zusammenhang mit dem Gesamtverhalten des simulierten Systems steht. Unter Umständen werden andere oder zusätzliche Verhaltensprofile und damit Gruppierungen benötigt, um das Verhalten des Originalsystems valide reproduzieren zu können.



### 5.1.2.2 Agenten in Schwarm-Optimierungsmethoden nach Pugh und Martinoli

Pugh und Martinoli [96] setzen das Schwarm-Optimierungsverfahren (siehe Abschnitt 3.2.5.2) explizit in einem simulierten Agentensystem um. Sie kalibrieren die Parameter eines Multiagentensystems aus simulierten Khepera Robotern auf Basis zweier verschiedener Zielfunktionen. Das Verhalten der simulierten Roboter wird durch ein neuronales Netz mit 22 einstellbaren Gewichten (= Parameter) bestimmt. Pugh verwendet Zielfunktionen, die explizit die Anpassung von Agenten an ihre Nachbarschaft erfordern.

- **Zielfunktion 1: Möglichst großer Abstand der Agenten**

Bei der Verwendung dieser Zielfunktion, versuchen die Roboter möglichst viel Abstand voneinander zu halten. Durch die Zielfunktion ist es wahrscheinlich, dass ein Agent eine gute Bewertung erhält, wenn seine Nachbarn eine gute Bewertung erhalten. Von der Bewertung des „besten“ Agenten kann allerdings nicht unbedingt auf eine generell gute Parametereinstellung geschlossen werden, da sich die Bewertung eines Agenten aus seinen eigenen Parametern und denen seiner Nachbarn ergibt.

- **Zielfunktion 2: Möglichst große Nähe zu anderen Agenten:**

Wie oben hängt die Bewertung eines Agenten wieder von seiner eigenen Parametereinstellung und der seiner Nachbarn ab.

Pugh und Martinoli setzen Partikel-Schwarm Optimierungsmethoden ein, um die simulierten Roboter bezüglich ihrer individuellen Zielfunktionen zu optimieren und untersuchen die Performanz des Optimierungsverfahrens im Vergleich zum Einsatz genetischer Algorithmen. In der Umsetzung der Partikelschwarmoptimierung repräsentiert jeder simulierte Roboter einen Lösungskandidaten des Optimierungsverfahrens. Das umgesetzte Verfahren entspricht dabei nicht ganz dem in Abschnitt 3.2.5.2 beschriebenen Standardalgorithmus. Im Standardalgorithmus werden die Lösungen der Lösungspopulation in jedem Iterationsschritt auf der Basis der am besten bewerteten Lösung angepasst. Übertragen auf die simulierten Roboter würde dies einer globalen Wahrnehmung für die einzelnen Roboteragenten und Adaption der individuellen Parameterwerte bezüglich des insgesamt am besten bewerteten Roboters entsprechen. Pugh und Martinoli verwenden hingegen eine eingeschränkte lokale Wahrnehmung für die Roboteragenten, d.h. die Agenten können ihre Parameterwerte nur aufgrund der wahrgenommenen Nachbaragenten adaptieren. Pugh und Martinoli kommen zum Schluss, dass die Optimierung der Parameterwerte mit Partikelschwarmoptimierungsmethoden, der mit genetischen Algorithmen überlegen ist. Der beschriebene Vorteil der Partikelschwarmoptimierung gegenüber dem genetischen Algorithmus scheint jedoch eher dadurch entstanden zu sein, dass die gewählten Zielfunktionen und die lokale Adaption der Agenten dieses Optimierungsverfahren bevorteilt haben.

Bei beiden Zielfunktionen ist es wichtig, dass ein Agent sein Verhalten auf das seiner unmittelbaren Umgebung abstimmt. Damit ist der Ansatz ein Beispiel dafür, dass in Umwelten mit heterogener Agentenwahrnehmung lokale Lern- und Optimierungsverfahren zur individuellen Parametereinstellung einen Vorteil gegenüber globalen Optimierungsverfahren haben können. Trotzdem kann bei Verwendung der Partikelschwarmoptimierung mit den oben beschriebenen Anpassungen von Pugh und Martinoli nicht ausgeschlossen werden, dass die Parametersuche nur ein lokales Optimum der Zielfunktion findet.

### 5.1.2.3 Impliziter Einsatz evolutionärer Optimierungsmethoden in ABS

In einigen ABS werden zwar keine explizite Kalibrierung oder Optimierung von ABS durchgeführt, es werden aber Optimierungsansätze als expliziter Teil der Modellstruktur verwendet. Dabei werden meist evolutionäre Optimierungsansätze umgesetzt, um beispielsweise Entwicklungsprozesse in einer realen Population nachzubilden oder das simulierte Verhalten durch den modellierten Entwicklungsprozess so zu optimieren, dass ein bestimmtes Entwicklungsergebnis erzielt wird. Entsprechende Modelle sind für diese Arbeit nicht relevant und werden daher nicht genauer behandelt. Ein Überblick findet sich bei Hare [54].

### 5.1.2.4 Zusammenfassung

In den beschriebenen Ansätzen werden populationsbasierte Optimierungsverfahren so interpretiert, dass die einzelnen simulierten Agenten selbst die Kandidatenlösungen der Population sind.

Beim Ansatz von Said wird letztendlich jedoch nur ein normaler genetischer Algorithmus ausgeführt. Die Parameter der einzelnen Bevölkerungsteile werden zwar getrennt voneinander kalibriert, es werden also tatsächlich nicht heterogene Parameterwerte für 40.000 Einwohner einer Stadt kalibriert. Stattdessen werden getrennt homogene Parameterwerte für vorgegebene Teilgruppen ermittelt. Es bleibt unklar, warum genau diese Teilgruppen gewählt wurden. Es wird nicht geklärt, wie oder ob überhaupt, die unterschiedlichen Ergebnisse zusammengesetzt werden (können), da eventuell relevante Wechselwirkungen zwischen den verschiedenen Bevölkerungsteilen nicht beachtet werden.

Pugh und Martinollis Verfahren ist ein lokales Adaptionverfahren und zumindest theoretisch zur Suche nach heterogenen Agentenparametern geeignet. Tatsächlich wurde in den beschriebenen Fällen aber nach global optimalen Strategien gesucht. Es wird nicht geklärt, wie gut dieses Verfahren ist, wenn bei deren Kalibrierung die Gefahr besteht, dass in einzelnen lokalen Bereichen zu wenige Agenten (= Kandidatenlösungen) vorhanden sind, um mit evolutionären Verfahren eine optimale lokale Lösung ermitteln zu können.

## 5.1.3 Methoden aus Gesellschafts- und Wirtschaftswissenschaften

In diesem Abschnitt werden Kalibrierungsmethoden betrachtet, die für ABS in den Gesellschaftswissenschaften [49] und den theoretischen Wirtschaftswissenschaften [34] entwickelt wurden. In diesen Anwendungsdomänen werden ABS genutzt, um gesellschaftliche Prozesse und dynamische Veränderungen in Gesellschaftsstrukturen zu untersuchen.

### 5.1.3.1 Lösungsansätze aus den Gesellschafts- und Wirtschaftswissenschaften

In diesem Abschnitt wird kurz auf verschiedene Kalibrierungsansätze für ABS eingegangen, die im Forschungsbereich der theoretischen Wirtschaftswissenschaften vorgeschlagen wurden (siehe Fagiolo [34] für einen Überblick). Betrachtet man die Literatur zu Kalibrierung gesellschaftswissenschaftlicher ABS und der theoretischen Wirtschaftswissenschaften, fällt auf, dass das Hauptproblem bei der Validierung und Kalibrierung darin besteht, dass nicht ausreichend empirische Daten verfügbar sind. Als Folge wurden sehr viele abstrakte rein theoretische Modelle entworfen. Erst in jüngerer Vergangenheit wird hinsichtlich der Notwendigkeit der Einbettung in empirische Daten argumentiert [115, 50].

In den Gesellschafts- und Wirtschaftswissenschaften werden verschiedene Lösungsansätze vorgeschlagen. Obwohl diese Ansätze dem aktuellen Stand der Forschung in den theoretischen Wirtschaftswissenschaften entstammen (siehe Brenner und Fagiolo [20, 34]), werden eigentlich Problemlösungen präsentiert, die in anderen Einsatzgebieten für Simulation schon längst zum etablierten Vorgehen gehören (entsprechende Verfahren wurden in Abschnitt 3.2 vorgestellt). Aus diesem Grund wird an dieser Stelle nicht genauer auf die Verfahren eingegangen, sondern es wird nur kurz auf die Verfahren und ihre dahinter liegenden Vorgehensweisen hingewiesen.

Der **indirekte Kalibrierungsansatz** (siehe Fagiolo [34]) beschreibt den Entwurf von Modellstrukturen aufgrund verfügbarer Literaturdaten. In diesem Ansatz zur Erstellung eines Simulationsmodells wird keine explizite Form von Kalibrierung auf der Basis von Validierungskriterien beschrieben. Empirische Daten über das erwünschte globale Verhalten werden lediglich genutzt, um den Wertebereich der Modellparameter so weit wie möglich einzuschränken.

Der **Werker-Brenner-Ansatz** [20] verwendet Zufallsexperimente (Monte-Carlo-Simulation), um die Validität von Parametereinstellungen bezüglich der Reproduktionsfähigkeit empirischer Daten zu ermitteln. Die beiden letzten Schritte der beschriebenen Prozedur sind nichts anderes als die Bewertung einer Parametereinstellung mit Hilfe einer Zielfunktion. Auch für diesen Ansatz gilt, dass die vorgeschlagenen Methoden existierenden Ansätzen entsprechen. Der Einsatz zufälliger Parametereinstellungen ist zudem im Vergleich zu anderen Kalibrierungsmethoden (siehe Abschnitt 3.2) ineffizient.

Der **geschichtsfreundliche Ansatz** [34] schließlich beschreibt lediglich, dass die Validitätsbewertung für die Kalibrierung eines Modells nicht nur aufgrund eines statischen Verhaltens sondern auch durch die Analyse einer zeitlichen Entwicklung erfolgen kann.

Zusammenfassend kann gesagt werden, dass in den theoretischen Wirtschaftswissenschaften das Kalibrierungsproblem für ABS zwar durchaus erkannt wurde, aber die vorgeschlagenen Lösungen jedoch bestehenden Black-Box-Verfahren entsprechen und bereits effizientere Verfahren entwickelt wurden.

#### 5.1.4 Kalibrierungsstrategie von Chu mit mehreren Beobachtungsebenen

Chu et al [25] beschreiben eine konkrete Kalibrierungsstrategie für einen speziellen Typ von Agenten-basierten Verkehrssimulationen, die mit dem PARAMICS<sup>2</sup> Framework für Verkehrssimulationen entwickelt wurden. Chu et al setzen hier für ihren speziellen Anwendungsfall einen White Box Ansatz zur Kalibrierung ihrer ABS auf mehreren Beobachtungsebenen um. Konkret betrachtet sie drei Beobachtungsebenen: 1) Planung von Fahrten durch die simulierten Agenten mit Auswahl eines Start/Zielortes. 2) Auswahl konkreter Routen für die einzelnen Fahrten durch die Agenten. 3) Verhalten eines simulierten Agenten während einer konkreten Fahrt. Auf Basis dieser Beobachtungsebenen zerlegen sie das Kalibrierungsproblem in vier Teilbereiche:

- **Kalibrierung des Fahrverhaltensmodells der Agenten:**

- Im ersten Schritt werden die lokalen Parameter des Beschleunigungs- und Überholverhaltens der Agenten auf der Mikroebene kalibriert.

---

<sup>2</sup><http://paramics-online.com/> (November 2010)

- **Kalibrierung des Routenwahlmodells:** In nächsten Schritt werden Parameter für die Verrauschung des von den Agenten wahrgenommenen Verkehrsaufkommens und für den Kenntnisgrad des Routennetzes bei den Agenten eingestellt.
- **Schätzung von Start-Ziel-Matrizen für die Agenten:**  
In diesem Schritt werden unabhängig von den bisherigen Kalibrierungsschritten Startort-Zielort-Matrizen für die simulierten Agenten kalibriert.
- **Globale Verfeinerungskalibrierung:** In diesem Schritt werden, ausgehend von den bisher ermittelten Parameterwerten, noch einmal alle Parameter im Gesamtmodell kalibriert.

Chu strukturiert das Kalibrierungsproblem des Simulationsmodells auf Basis der klassischen Schritte eines Verkehrsmodells (konkret: Planung, Routenwahl und konkretes Verhalten). Die Kalibrierung wird nicht für alle Schritte gleichzeitig gemacht, sondern für die einzelnen Schritte getrennt. Chu et al zerlegen das Gesamtkalibrierungsproblem mit **White Box Kalibrierungsstrategien** in Teilprobleme, die getrennt kalibriert werden. Die bereits ermittelten Parameterwerte eines Teilproblems bilden jeweils Randbedingungen für die weitere Kalibrierung.

Versucht man das Verfahrens von Chu et al für ihren spezifischen Anwendungsfall zu verallgemeinern können die folgenden Hürden identifiziert werden:

1. Das Kalibrierungsverfahren wurde nur für einen speziellen Typ von Verkehrs-ABS entworfen. Es gibt keine Informationen wie das Verfahren auf andere Verkehrs-ABS oder allgemeine ABS übertragen werden kann.
2. Das Kalibrierungsverfahren geht davon aus, dass das Gesamtproblem vollständig in drei sequentiell kalibrierbare Teilprobleme zerlegbar ist, bei denen jeweils bereits kalibrierte Teile Randbedingungen für die weitere Kalibrierung bilden. Ist das resultierende Gesamtverhalten nicht valide, so ist der einzige anwendbare Prozessschritt die Kalibrierung aller Parameterwerte gleichzeitig in der Gesamtmodellfamilie. Eine Betrachtung der Abhängigkeiten zwischen verschiedenen Beobachtungsebenen gibt es nicht.
3. Auf allen Beobachtungsebenen besteht Unklarheit, ob für alle Agenten jeweils individuelle Pläne und Parameterwerte kalibriert werden müssen oder ob (auf allen oder Teilen der Beobachtungsebenen) die vereinfachende Annahme homogener Agenten getroffen werden kann. Es gibt keinen Prozessschritt zur Klärung dieses Problems und es werden auch keine Hinweise gegeben, wie das Problem effizient gelöst werden könnte.
4. Es ist im Kalibrierungsvorgehen kein Prozessschritt vorgesehen, der die Kalibrierung robust gegenüber der Verrauschung von Referenzdaten über das Originalsystem macht.
5. Insgesamt bleibt unklar ob durch die Kalibrierung einzelner Module Vorteile für die Kalibrierung des gesamten Modells im Vergleich zu einer direkten Kalibrierung aller Parameter gleichzeitig gewonnen werden könnten.

### 5.1.5 Agentenlernen

In diesem Abschnitt wird beschrieben, wie Lernverfahren zur Kalibrierung, bzw. Optimierung, von Agentenverhalten eingesetzt werden kann. Die Grundidee hinter Lernmechanismen für Agenten besteht darin, die aktuellen Wahrnehmungen eines Agenten nicht nur zur Ermittlung dessen unmittelbare Reaktionen zu verwenden, sondern auch zur Verbesserung des zukünftigen Verhaltens des Agenten in gleichen oder ähnlichen Situationen [98]. Hierzu wird versucht einen Zusammenhang zwischen der Agentenwahrnehmung, gewählten Agentenaktionen und einem individuellen Performanzkriterium für den Agenten herzustellen.

Soll Agentenlernen nicht nur dazu genutzt werden, ein bestimmtes individuelles Agentenverhalten, sondern auch das Gesamtverhalten des zugehörigen simulierten Systems zu kalibrieren, muss ein Performanzmaß gefunden werden, auf dessen Basis individuelle Parameterwerte eingestellt werden können, die dazu führen, dass einerseits ein valides lokales Agentenverhalten erzeugt wird und andererseits valides Modellverhalten auf Makro-Ebene.

#### 5.1.5.1 Lernverfahren

In vielen komplexen Domänen, in denen ein System lernen soll, sich optimal zu verhalten, ist das sogenannte **Verstärkungslernen (Reinforcement Learning)** eine etablierte Möglichkeit die Parametereinstellungen des Systems geeignet vorzunehmen [98]. Es gibt verschiedene Formen von Verstärkungslernen. Beim **passiven Verstärkungslernen** haben die Agenten eine vorgegebene Verhaltensvorschrift. Sie versuchen, innerhalb dieser Vorschrift die Nützlichkeit der einzelnen Zustände zu lernen, in denen sie sich befinden können. Beim **aktiven Lernen** müssen die Agenten zusätzlich auch noch eine geeignete Verhaltensvorschrift lernen.

Bekannte Verstärkungslernverfahren sind [90]:

- **Temporal Difference Learning:** Ein Agent lernt eine Nützlichkeitsfunktion auf den möglichen Zuständen, in denen er sich befinden kann. Er wählt seine Aktionen so, dass er in möglichst nützliche Zustände gelangt [110].
- **Q-Lernen:** Agenten, die mittels Q-Lernen Erfahrung sammeln, lernen die Nützlichkeit bestimmter Aktionen, wenn sie sich in bestimmten Zuständen befinden [68].

#### Mögliche Probleme beim Agentenlernen

Beim **Lernen in Multiagentensystemen** muss mit zwei Problemen umgegangen werden.

- **Gegenseitige Beeinflussung beim individuellen Lernen:**

Die Agenten lernen meist gleichzeitig ihr individuelles Verhalten. Änderungen am Verhalten eines Agenten können dabei die Lernsituationen der übrigen Agenten verändern [90]. Ein einmal gelernter individueller Zusammenhang aus Wahrnehmung und Aktion kann ungültig werden, wenn sich die Situation eines Agenten aufgrund des Lernprozesses der übrigen Agenten verändert.

Ein einzelner Agent kann sich nicht notwendigerweise darauf „verlassen“, dass seine Umwelt sich im nächsten Lernschritt noch wie zuvor verhält.

- **Lernen von globalen Zielen** Ein zweites Problem kann entstehen, wenn die Agenten des simulierten Multiagentensystem individuelle Verhaltensweisen lernen sollen, die zu einem erwünschten (z.B. validen) Gesamtverhalten der Simulation führen [90]. In diesem Fall muss ein Lernverfahren so entworfen werden, dass die **individuellen Validierungskriterien** der einzelnen Agenten und das **globale Validierungskriterium auf der Makro-Ebene** in der Nützlichkeit für die Agenten ausgedrückt werden können.

Zur Lösung solcher Lernprobleme, bei denen individuelles Verhalten und das Verhalten auf der Makro-Ebene in Einklang gebracht werden müssen, werden verschiedene kombinierte Nützlichkeitsfunktionen und Lerntechniken erforscht [90]. Der Ansatz von Tumer [116] wird im nächsten Abschnitt näher beschrieben.

### 5.1.5.2 Agentenlernen in Collectives

In diesem Abschnitt wird ein Lernverfahren beschrieben, mit dem versucht wird auf der Mikro-Ebene individuelle Verhaltensweisen zu lernen, die zu einem erwünschten Gesamtverhalten führen. Tumer und Wolpert [116] beschreiben Entwicklungsansätze für so genannte **Collectives**. Unter einer Collective verstehen sie ein Multiagentensystem, bei dem einerseits jeder Agent eine individuelle Nützlichkeitsfunktion (übertragen auf die Kalibrierung ein Validierungskriterium) hat und andererseits eine globale Weltnützlichkeit (Validierungskriterium auf der Makro-Ebene) existiert. Tumer und Wolpert adressieren das Problem, eine private, d.h. individuelle, Nützlichkeitsfunktion für den Lernprozess zu wählen, die in geeigneter Form globale und individuelle Nützlichkeiten kombiniert, so dass die globale Nützlichkeit maximiert wird, wenn die Agenten nur aufgrund ihrer privaten Nützlichkeiten agieren. Übertragen auf die Kalibrierung von ABS werden also individuelle Validierungskriterien gesucht, die valide individuelle Verhaltensweisen beschreiben, die zu validem globalen Gesamtverhalten führen. Als Ergebnis ihrer Untersuchungen favorisieren Tumer und Wolpert [121, 117] die Verwendung einer sogenannten „Wonderful Life Utility“, insbesondere wenn sehr viele Agenten verwendet werden. Diese individuelle Nützlichkeitsfunktion misst wie gut ein einzelner Agent zum Ziel-Gesamtverhaltens beiträgt. Dies geschieht dadurch, dass die globale Nützlichkeit, die durch die kombinierten Aktionen aller Agenten erreicht wurde, mit derjenigen verglichen wird, die erreicht worden wäre, hätte es den betroffenen Agenten nicht im System gegeben.

### 5.1.5.3 Diskussion: Agentenlernen

Die Beschreibung in diesem Abschnitt hat einerseits den Nutzen lokaler Lernverfahren zur individuellen Parametereinstellung und andererseits die möglichen Probleme gezeigt. Wie in Abschnitt 3.1.5.2 beschrieben, ist ein Lernverfahren eigentlich ein spezialisierter Parameteroptimierungsalgorithmus und setzt somit ein **wissensbasiertes Kalibrierungsverfahren** um. Wie bei allen wissensbasierten Kalibrierungsverfahren ist die **Art des gewählten Lernverfahrens** für den Erfolg der Parameterkalibrierung entscheidend. Wie in Abschnitt 4.1.2.3 beschrieben, erfordert dies ausreichendes Wissen über die Wirkungszusammenhänge des modellierten Systems, was nicht notwendigerweise gegeben ist.

### 5.1.6 Zusammenfassung: Kalibrierung in ABS

In diesem Abschnitt wurden verschiedene für ABS vorgeschlagene Kalibrierungs- und Optimierungsverfahren vorgestellt und diskutiert. Kalibrierung Agenten-basierter Modelle ist Bestandteil aktueller Diskussion und Forschung. Es fällt auf, dass zur Kalibrierung von ABS kein einheitliches Vorgehen existiert. Hierfür können folgende mögliche Gründe angeführt werden:

- Auf vielen Gebieten existieren noch **keine etablierten Theorien für spezifische Fragestellungen**, d.h. Theorien, warum bestimmte reale Systeme in einer bestimmten Weise funktionieren. Damit ist unklar wie das Modell letztendlich aussehen muss und welche Parameter überhaupt relevant sind. Es gibt auch keine naturkonstanten Werte die für ein Modell von vornherein festgelegt wären.
- Aufgrund des hohen Detaillevels Agenten-basierter Simulationen stehen **keine ausreichenden Daten und Wissen zur Validierung auf unterschiedlichen Beobachtungsebenen** und damit zur Kalibrierung zur Verfügung.

Die meisten Verfahren zur Kalibrierung von ABS behandeln keine spezifischen Eigenschaften von ABS, sondern transferieren bestehende Black-Box Ansätze. Es gibt aber auch Ansätze die explizit versuchen das Kalibrierungsproblem zu zerlegen (White Box) oder mittels wissensbasierter Kalibrierung individuelle Parameterwerte für die Agenten zu ermitteln. Zur **Ermittlung individueller Agentenparameter** wird entweder Agentenlernen eingesetzt oder es werden populationsbasierte Optimierungsverfahren uminterpretiert. Keines der beschriebenen Parametersuchverfahren adressiert das Problem, wie in begrenzter Zeit heterogene Parameterwerte realistisch großer Suchräume eingestellt werden können.

### 5.1.7 Abgrenzung vom Stand der Forschung

In diesem Abschnitt wird das in dieser Arbeit entwickelte Makro-Mikro-Verfahren zur Kalibrierung von ABS zu den beschriebenen existierenden Arbeiten abgegrenzt. Das entwickelte Makro-Mikro-Verfahren ist ein Kalibrierungsverfahren, das auf die Kalibrierung von ABS unabhängig des speziellen ABS-Typs angewendet werden kann.

Die Grundidee des Makro-Mikro-Verfahrens besteht darin ein ABS-Kalibrierungsproblem auf unterschiedlichen Beobachtungsebenen in jeweils zwei verknüpfte Teilprobleme aufzuspalten. Auf der Makro-Ebene wird während der Kalibrierung nach Validierungskriterien idealen aggregierten Verhaltens für die simulierten Agentenmodelle gesucht, deren Erfüllung zu validem Gesamtverhalten der Simulation auf der Makro-Ebene führt. Auf der Mikro-Ebene werden die Agentenmodelle so kalibriert, dass sie die aktuellen Kandidaten der idealen Validierungskriterien von der Makro-Ebene erfüllen. Die Suche wechselt dabei iterativ zwischen den beiden Ebenen und nimmt jeweils Anpassungen an den idealen Validierungskriterien oder den Parameterwerten vor.

Von den beschriebenen existierenden Kalibrierungsverfahren für ABS grenzt sich das Makro-Mikro-Verfahren durch folgende Punkte ab:

1. Das Makro-Mikro-Verfahren enthält einen Analyseschritt, um das ABS-Kalibrierungsproblem hinsichtlich seiner Teilprobleme zu untersuchen und Möglichkeiten zur An-

wendung von White-Box-Zerlegungstechniken zu identifizieren. In existierenden Verfahren zur ABS-Kalibrierung wird das Modell entweder nur als Ganzes kalibriert oder es wird direkt eine spezifische Zerlegung präsentiert. Ein allgemeines Vorgehen zur Problemanalyse und Zerlegung für die ABS-Kalibrierung gibt es bisher nicht.

2. Im Makro-Mikro-Verfahren wird der Umgang mit der Makro-Mikro-Wissenslücke und die Kalibrierung auf verschiedenen Beobachtungsebenen adressiert, was bisher von keinem Verfahren getan wurde.
3. Im Makro-Mikro-Verfahren wird das Problem der Kalibrierung vieler heterogener Agentenparameter adressiert. Bisherige Verfahren bieten hierfür keine Lösung und kalibrieren daher meist homogene Parameterwerte.
4. Im Makro-Mikro-Verfahren wird eine Lösung für das Problem der Kalibrierung von Agentenmodellen auf der Mikro-Ebene auf der Basis verrauschter Daten vorgeschlagen. Dieses Problem wurde bisher in keinem existierenden Verfahren zur ABS-Kalibrierung adressiert.
5. Viele der existierenden Verfahren zur ABS-Kalibrierung übertragen einfach direkt einzelne existierende Verfahren zur Parametersuche (z.B. genetische Algorithmen) auf ABS. Im Vergleich zu diesen Ansätzen ist beim Makro-Mikro-Verfahren die konkrete Parametersuche nur ein Teilschritt des Gesamtprozesses. Das Makro-Mikro-Verfahren enthält auch Prozessschritte, die auf einer Meta-Ebene angesiedelt sind.



## Kapitel 6

# Anforderungen an Verfahren zur ABS-Kalibrierung

In diesem Kapitel werden Anforderungen für Kalibrierungsverfahren und -werkzeuge für ABS aufgestellt. Zunächst werden aus der Literatur bekannte Anforderungen an Kalibrierungswerkzeuge vorgestellt. Diese Anforderungen für die Kalibrierung allgemeiner Simulationsmodelle werden um zusätzliche Anforderungen bezüglich ABS erweitert.

Für jede Anforderung wird im Anschluss an die Auflistung bereits hier kurz angegeben ob und wenn ja durch welchen Teil dieser Arbeit sie behandelt wird.

### 6.1 Allgemeine Anforderungen an Kalibrierungswerkzeuge

In den Jahren 2000 und 2001 wurden führende Wissenschaftler und Mitarbeiter von Simulationsfirmen, wie z.B. Fred Glover, Michael Fu und Royce Bowden und andere, dazu befragt, welche kritischen Punkte sie bei der Simulationskalibrierung selbst und bei der Entwicklung zugehöriger Werkzeuge sehen [46, 15]: Fu weist darauf hin, dass die meisten Kalibrierungsalgorithmen nicht beweisbar konvergent sind und in ihnen die Tatsache, dass ein Simulationsmodell kalibriert wird eher sekundär ist. Das Modell wird als Black-Box behandelt. Andradottier beschreibt die Notwendigkeit spezielle Kalibrierungsverfahren für spezielle Simulationsprobleme zu entwickeln; Ho, dass die Struktur der meisten Kalibrierungsprobleme sehr schlecht verstanden ist, was die Entwicklung geeigneter Kalibrierungsalgorithmen erschwert. Bowden fasst viele Aspekte zusammen und spezifiziert Anforderungen an sechs Teilbereiche der Simulationskalibrierung:

1. **Schnittstellen** zwischen Benutzer und Kalibrierungswerkzeug und Werkzeug und Simulationsmodell: Zum einen sollen Kalibrierungswerkzeuge für den Nutzer leicht zu verwenden sein und andererseits mächtig genug, um umfassende Änderungen am Simulationsmodell vornehmen zu können
2. **Problemformulierung:** Die Auswahl von geeigneten Validierungskriterien und Kalibrierungsrandbedingungen ist entscheidend für den Erfolg der Kalibrierung. Aufgrund des breiten Anwendungsspektrums für Simulationsmodelle benötigt der Nutzer hier weitgehende Freiheit.

3. **Kalibrierungsalgorithmen:** Um eine automatische Modellkalibrierung durchführen zu können, werden effiziente Kalibrierungsalgorithmen zur Einstellung der Parametereinstellungen benötigt.
4. **Klassifikation:** Sind die Struktur des Kalibrierungsproblems und geeignete Kalibrierungsverfahren bekannt, werden Analyse- und Klassifikationstechniken benötigt, um so zu diesem Kalibrierungsproblem eine geeignete Kalibrierungsmethode zu wählen.
5. **Kalibrierungsstrategien und -taktiken:** Um vorhandene Computerressourcen möglichst effizient einsetzen zu können, werden je nach Art des Optimierungsproblems unterschiedliche Strategien zur Einteilung der für die Kalibrierung zur Verfügung stehenden Simulationsläufe benötigt.
6. **Intelligenz:** In das Kalibrierungsprogramm müssen automatische Auswahlmechanismen integriert werden, mit denen zu einem klassifizierten Kalibrierungsproblem ein geeignetes Lösungsverfahren gewählt wird, d.h. um möglichst geeignete Strategien und Taktiken zu wählen.

Zusammenfassend werden also Kalibrierungsverfahren gesucht, die mit möglichst wenig Berechnungsaufwand eine möglichst geeignete Parametereinstellung finden. Dies soll erreicht werden, indem die Verfahren mit möglichst viel Wissen über ein bestimmtes Simulationsproblem ausgestattet werden. Für die Kalibrierung von ABS wird ein Werkzeug benötigt, das es erlaubt, mit den charakteristischen Probleme bei der Kalibrierung solcher Modelle umzugehen.

In dieser Arbeit werden die beschriebenen Anforderungen wie folgt adressiert. Zur Erfüllung von Punkt 1 wurde das Kalibrierungswerkzeug DAVINCI entwickelt, das in Kapitel 11 vorgestellt wird. Zur Erfüllung von Anforderung 2 bietet das Makro-Mikro-Verfahren Hilfestellungen zur Identifikation geeigneter Validierungskriterien sowohl auf Makro- als auch auf Mikro-Ebene oder anderen Beobachtungsebenen. Diese können in DAVINCI modelliert und zur Kalibrierung eingesetzt werden. Die dritte Anforderung wird im Makro-Mikro-Verfahren bei der Kalibrierung individueller Agentenparameter behandelt. Durch die verwendeten Techniken kann der Aufwand zur Kalibrierung individueller Agentenparameter im Vergleich zu reinen Black-Box-Kalibrierungsansätzen reduziert werden (siehe Kapitel 9.1.3, 10 und den Evaluationsteil 15). Die 4. Anforderung wird dadurch behandelt, dass das Makro-Mikro-Verfahren als Bestandteil ein Analyseverfahren für das Kalibrierungsproblem beinhaltet, anhand dessen die Struktur des Problems untersucht und benötigte Kalibrierungstechniken ermittelt werden (siehe Kapitel 8.1). Das Makro-Mikro-Verfahren setzt sich weiter aus verschiedenen Techniken zusammen, die nach Bedarf einzeln oder kombiniert eingesetzt und durch weitere Kalibrierungstechniken erweitert werden können (siehe DAVINCI-Werkzeug in Kapitel 11). Hierdurch wird die 5. Anforderung erfüllt. Die 6. und letzte Anforderung wird nicht behandelt, da in dieser Arbeit keine automatischen Auswahlmechanismen für spezifische Kalibrierungstechniken behandelt werden. Der Grund hierfür ist, dass noch sehr viel Forschungsarbeit im Bereich der Kategorisierung von ABS geleistet werden muss, bevor verstanden werden kann was in den Modellen genau geschieht und wie diese geeignet kategorisiert werden können. Dies kann im Rahmen dieser Arbeit nicht geleistet werden.

## 6.2 Anforderungen an Verfahren zur ABS-Kalibrierung

Aus den zuvor beschriebenen Problemen bei der Kalibrierung von ABS lassen sich folgende Anforderungen an ein Kalibrierungsvorgehen und -werkzeug herleiten.

### 6.2.1 Agenten-spezifische Anforderungen

1. **Kalibrierung auf verschiedenen Beobachtungsebenen:** Das Kalibrierungsverfahren muss so ausgelegt sein, dass es die Kalibrierung von ABS auf mehreren Beobachtungsebenen integriert.

Ein Verfahren muss:

- es erlauben die unterschiedlichen Ebenen für sich zu kalibrieren
  - es erlauben die Ergebnisse einzelner Ebenen auf andere zu übertragen
  - den Nutzer dabei unterstützen Wissenslücken über den Wirkungszusammenhang zwischen unterschiedlichen Beobachtungsebenen zu ermitteln.
2. **Kalibrierung potentiell heterogener Agentenparameter:** Ein Kalibrierungsverfahren muss dem Nutzer bei der Kalibrierung heterogener Parameterwerte für die simulierten Agenten unterstützen. Ein geeigneter Umgang mit möglicher Heterogenität in den Parameterwerten der Agenten ist sowohl aus Effizienzgründen, als auch um es dem Nutzer zu ermöglichen, explizite Heterogenitätsrandbedingungen zu formulieren, relevant. D.h. neben den Alternativen homogener und heterogener Parameterwerte muss das Verfahren dem Nutzer die Möglichkeit geben unterschiedliche Heterogenitätseinstellungen für Parameterwerte zu spezifizieren oder im Idealfall automatisch ermitteln zu lassen. Die automatische Ermittlung von Heterogenitätseinstellungen ist besonders relevant, wenn der Nutzer nicht a priori geeignete homogene Gruppen von Agenten festlegen kann.
  3. **Umgang mit verrauschten Daten und Wissen bezüglich individuellen Agentenverhaltens:** Das Kalibrierungsverfahren muss den Nutzer beim Umgang mit verrauschten oder unsicheren Daten und Wissen unterstützen, die individuelles Agentenverhalten beschreiben. Problematische Daten müssen identifiziert und gegebenenfalls geeignet modifiziert, bzw. entfernt werden.
  4. **Kombinierte Lösung von Modellierungs- und Kalibrierungsproblemen:** Ein Kalibrierungsverfahren für ABS muss den Modellierer beim Umgang mit Problemen bei der Auswahl von Modellstrukturen im Rahmen der Kalibrierung unterstützen. Entscheidende Schritte sind die Identifikation möglicher Modellausprägungen, die Möglichkeit die zugehörigen Modellierungsunsicherheiten so formalisieren zu können, dass diese im Kalibrierungsverfahren behandelt werden können, die automatische Kalibrierung der resultierenden Kandidatenmodelle und die Identifikation des/der validesten Gesamtmodelle. Je nach Art und Anzahl möglicher Modellausprägungen kann es potentiell nötig sein, sehr viele Kandidatenmodelle in begrenzter Zeit zu kalibrieren und zu evaluieren. Daher werden automatische Techniken zum sinnvollen Einsatz der zur Verfügung stehenden Berechnungsressourcen (Rechenleistung des Computers) benötigt.

- 5. Kombinierbarkeit unterschiedlicher Kalibrierungsansätze:** In den vorherigen Kapiteln wurde dargelegt, welche unterschiedlichen Kalibrierungsansätze existieren und dass für spezifische Kalibrierungsprobleme spezifische Kalibrierungstechniken erforderlich sein können. Um dem Nutzer möglichst viel Flexibilität beim Einsatz unterschiedlicher Techniken zu bieten, soll ihm ein Kalibrierungsverfahren die Möglichkeit bieten, unterschiedliche Ansätze miteinander zu kombinieren, um Stärken der einzelnen Techniken zur effizienten automatischen Kalibrierung unterschiedlicher Probleme nutzen zu können. Beispielsweise soll es möglich sein, gleichzeitig Modellierungsprobleme zu lösen und die Parameterwerte, z.B. durch kombinierten Einsatz von Black Box Optimierung(en) und wissensbasierter Suche, zu kalibrieren.

In dieser Arbeit werden die beschriebenen Agenten-spezifische Anforderungen wie folgt adressiert. Anforderung 1 wird durch das in dieser Arbeit entwickelten Makro-Mikro-Kalibrierungsverfahren behandelt (siehe Kapitel 9.1.3). Anforderung 2 kann durch die in dieser Arbeit entwickelte Heterogenitätssuche während der Parameterkalibrierung gelöst werden (siehe Kapitel 9.1.3). Die 3. Anforderung wird durch die in dieser Arbeit entwickelte robuste Kalibrierung im Rahmen des Makro-Mikro-Verfahrens abgedeckt (siehe Kapitel 9.3). Die automatische Konfiguration von Modellstrukturen aus Anforderung 4 steht nicht im Fokus dieser Arbeit und wird daher im entwickelten Kalibrierungsvorgehen nicht explizit behandelt. Das für diese Arbeit entwickelte Kalibrierungswerkzeug DAVINCI stellt jedoch automatische Techniken zur gleichzeitigen Strukturverfeinerung und Parameterkalibrierung bereit. Die 5. Anforderung wird dadurch erfüllt, dass das Makro-Mikro-Verfahren sich aus verschiedenen Techniken zusammensetzt, die nach Bedarf einzeln oder kombiniert eingesetzt und durch weitere Kalibrierungstechniken erweitert werden können. Die einzelnen Techniken zur Erfüllung der beschriebenen Anforderungen wurden im DAVINCI-Kalibrierungswerkzeug umgesetzt (siehe Kapitel 11).

### 6.3 Ausblick auf die weitere Arbeit

In diesem Abschnitt wurden, basierend auf den beschriebenen Problemen bei der Entwicklung von ABS, Anforderungen an Kalibrierungsverfahren für ABS entwickelt. Auf Grundlage der aufgestellten Anforderungen wird im weiteren Verlauf der Arbeit ein Makro-Mikro-Kalibrierungsverfahren entwickelt. Das Verfahren wird mit existierenden Black-Box-Verfahren verglichen. Ein expliziter Vergleich mit White-Box- und Wissens-basierten Techniken erfolgt nicht, da das Makro-Mikro-Verfahren selbst entsprechende Techniken nutzt. Es wird stattdessen an den jeweiligen Stellen auf die Nutzung der White-Box-Techniken hingewiesen. Weiter wird das Makro-Mikro-Verfahren um zusätzliche Kalibrierungstechniken erweitert, die den Nutzer bei der Lösung der Agenten-spezifischen Kalibrierungsprobleme unterstützen sollen. Die entwickelten Techniken werden schließlich im Kalibrierungswerkzeug DAVINCI für ABS umgesetzt und evaluiert. DAVINCI unterstützt die Kalibrierung auf der Basis formalisierbarer Validierungskriterien.

**Teil II**

**Konzepte**



## Kapitel 7

# Überblick: Makro-Mikro-Kalibrierungsverfahren und Fallstudie

In dieser Arbeit wird ein Kalibrierungsverfahren mit der Bezeichnung Makro-Mikro-Kalibrierungsverfahren entwickelt. Der Ausgangspunkt des Verfahrens ist die Black-Box-Kalibrierung einer ABS, die nicht zum erwünschten Kalibrierungsergebnis geführt hat, weil keine eindeutige Lösung gefunden werden konnte oder das resultierende Simulationsverhalten nicht den Validitätsgrad hat, der eigentlich erreicht werden sollte.

Im Kern des Makro-Mikro-Verfahrens steht die Ausnutzung der Mehrebeneneigenschaft von ABS zur Verbesserung des Kalibrierungsergebnisses. Das Verfahren wird zunächst in einer Basis-Variante beschrieben, in der durch Analyse des Modells Teilmodelle und zugehörige zusätzliche Validierungskriterien identifiziert werden. Diese zusätzlichen Validierungskriterien werden dann in die Kalibrierung mit einbezogen, was vor der Einführung der zugehörigen Betrachtungsebene nicht möglich war.

Der problematische Teil an obigem Vorgehen ist die Festlegung entsprechender Validierungskriterien. Diese können oft nicht direkt spezifiziert werden. Um mit diesem Problem umgehen zu können, wird das Makro-Mikro-Verfahren um die Reverse-Engineering-Verfeinerung erweitert. Die Reverse-Engineering-Verfeinerung kann eingesetzt werden, wenn der Modellierer eine Idee für die Struktur der Validierungskriterien hat. In diesem Fall werden eine Art Regressionsmodelle der identifizierten Teilmodelle auf aggregierter Ebene ermittelt. Diese Regressionsmodelle werden zur Kalibrierung des tatsächlichen Modells verwendet.

Die durch das Reverse-Engineering ermittelten Regressionsmodelle werden zusätzlich in einer Erweiterung des Makro-Mikro-Verfahrens zur robusten Kalibrierung eingesetzt. Hierbei werden als fehlerhaft identifizierte Teilmodelle durch ihre zugehörigen Regressionsmodelle ersetzt.

Als zusätzliche Erweiterung des Makro-Mikro-Verfahrens wird die Heterogenitätssuche vorgestellt. Die Heterogenitätssuche löst das Problem wie heterogen die Parameterwerte für die simulierten Agenten bei der Kalibrierung gewählt werden müssen, indem die minimal notwendige Heterogenitätsausprägung bestimmt und auf ihrer Basis die optimale Menge an Parametern kalibriert wird.

Abbildung 7.1 zeigt einen Überblick über die möglichen Anwendungsvarianten des Makro-Mikro-Verfahrens und des zur Umsetzung des Ansatzes entwickelten Kalibrierungswerkzeuges DAVINCI. Als Anwendungsvariante wird dabei der Einsatz des Basis-Makro-Mikro-Verfahrens allein oder in Kombination mit einer erweiternden Technik verstanden. Im weiteren Fokus dieser Arbeit stehen nur die Agenten-spezifischen Teile des entwickelten Gesamtverfahrens. Die übrigen Teile (in Abbildung 7.1 transparent/schwächer gedruckt dargestellt) wurden zwar im Werkzeug DAVINCI als Hilfstechniken umgesetzt, werden aber im weiteren Verlauf nicht genauer beschrieben. Abbildung 7.2 veranschaulicht den allgemeinen Prozess der Durchführung des Makro-Mikro-Verfahrens und dem Einsatz der Erweiterungstechniken während des Verfahrens.

In ersten Teil dieses Kapitels wird ein kurzer Gesamtüberblick über das Makro-Mikro-Verfahren gegeben, bevor es ausführlich beschrieben wird. Im zweiten Teil wird eine zentrale Fallstudie dieser Arbeit eingeführt anhand der die entwickelten Techniken demonstriert und evaluiert werden.

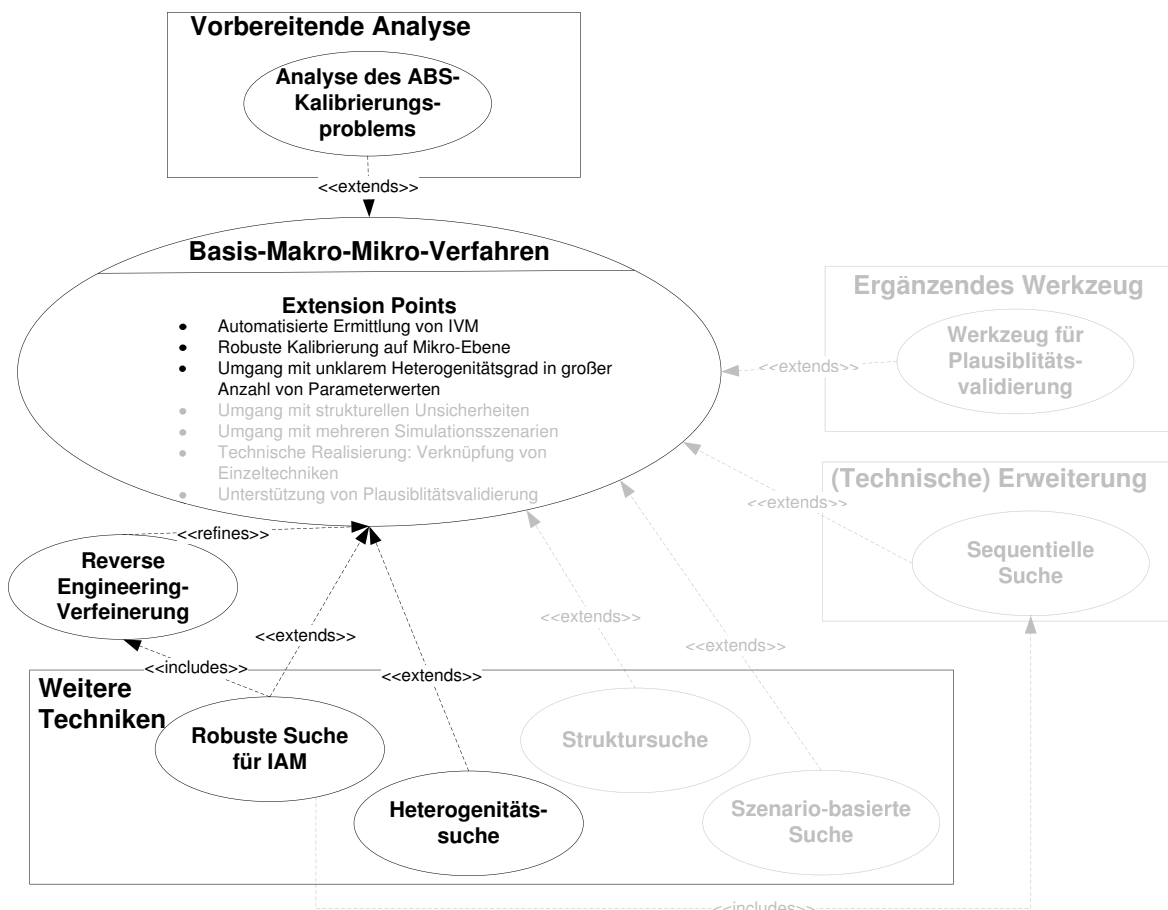


Abbildung 7.1: Überblick: Anwendungsvarianten des entwickelten Makro-Mikro-Verfahrens.



## 7.1 Makro-Mikro-Verfahren

### 7.1.1 Basis-Makro-Mikro-Verfahren

Beim Basis-Makro-Mikro-Verfahren wird ein ABS-Kalibrierungsproblem in zwei verknüpfte Kalibrierungsprobleme aufgespalten: Eines auf der Makro- und eines auf der Mikro-Ebene. Während der Kalibrierung auf der Makro-Ebene werden ausgehend von den vorgegebenen *VK* für das Gesamtverhalten, passende Modelle idealen Verhaltens für Teilmodelle des Gesamtmodells *IVM* (*ideale Verhaltensmodelle*)<sup>1</sup> bestimmt. Die *IVM* können, je nachdem was für eine Art von Teilmodell durch sie beschrieben wird, Teile der Umwelt und Agenten, Gruppen von Agenten oder auch nur einzelne Agenten beschreiben. Die Modelle sind nicht Teil des eigentlichen Simulationsmodells, sondern beschreiben das Verhalten, das aus den konkreten Aktionen und Interaktionen der simulierten Agenten auf der Mikro-Ebene untereinander und mit ihrer Umwelt entstehen muss, damit sich insgesamt valides Gesamtverhalten bezüglich der *VK* ergibt. In den *IVM* wird das Verhalten des zugehörigen Teilmodells abstrakter beschrieben, als es auf der Mikro-Ebene des tatsächlichen Modells der Fall ist. Im nächsten Schritt dienen die *IVM* dazu, Validierungskriterien für die eigentlichen Teilmodelle zu ermitteln, mit denen dann die Kalibrierung auf der Mikro-Ebene durchgeführt werden kann. Sind die *IVM* bestimmt, wird das Verhalten auf der Mikro-Ebene dadurch validiert, dass das tatsächliche simulierte Verhalten des Teilmodells mit dem Verhalten des *IVM* verglichen wird. Im Idealfall (jeder simulierte Agent bildet ein eigenes Teilmodell) wird jedes Agentenmodell aufgrund seiner spezifischen Zielfunktion kalibriert.

Die Ermittlung der *IVM* ist nicht trivial. Im optimalen Fall können die *IVM* für relevante Teilmodelle des Gesamtmodells vom Fachexperten vorgegeben und dann bei der Kalibrierung eingesetzt werden. Im schlechtesten Fall, wenn nichts über valides Verhalten der Teilmodelle bekannt ist, können die *IVM* nur „qualifiziert geraten“ werden. Dies ist z.B. der Fall, wenn das Makro-Mikro-Verfahren zur Optimierung eines Systems eingesetzt werden soll und nicht bekannt ist, durch welche Teilverhaltensweisen der Agenten optimales Gesamtverhalten erreicht werden kann oder wenn unklar ist, aus welchem Mikroverhalten ein beobachtetes Makro-Phänomen entsteht.

Eine Möglichkeit das Problem der *IVM*-Ermittlung zu entschärfen besteht, wenn der Modellierer zumindest eine Idee zur Beschreibung der Struktur der *IVM* hat. In diesem Fall kann die Ermittlung der *IVM* im Rahmen dieser vorgegebenen Strukturen als Teil des Kalibrierungsproblems mit betrachtet werden. Hierfür wird das Basis-Makro-Mikro-Verfahren durch ein Reverse-Engineering zur Ermittlung von Regressionsmodellen für die *IVM* verfeinert.

### 7.1.2 Reverse-Engineering von *IVM*

Wie beschrieben ist die Ermittlung der idealen Verhaltensmodelle *IVM* nicht trivial. Oft können die *IVM* nicht direkt spezifiziert werden, da unklar ist, wie genau das Verhalten eines Teilmodells aussehen muss, damit valides Gesamtverhalten entstehen kann.

Die grundlegende Idee zur Lösung dieses Problems in der vorliegenden Arbeit besteht darin, keine kompletten *IVM* zu spezifizieren, sondern nur deren Struktur *IVMF* vorzugeben.

---

<sup>1</sup>Nicht zu verwechseln mit den zuvor eingeführten *IVK*. *IVK* sind vorgegebene Validierungskriterien zur Bewertung individuellen Verhaltens auf der Mikro-Ebene. *IVM* sind Modelle des idealen Verhaltens von Teilmodellen.

Die Modellstrukturen werden dann als Regressionsmodelle verfeinert, um so die benötigten *IVM* zu ermitteln. Vom Modellierer wird daher lediglich gefordert, dass er für die *IVM* eine zugehörige Modellstruktur spezifizieren kann. Dies ist leichter, als zu beschreiben wie dieses Verhalten konkret aussehen muss.

Auf Grundlage der *IVMF*-Strukturen werden dann durch den Reverse-Engineering-Ansatz die zugehörigen *IVM* als eine Art von Regressionsmodellen idealen Teilmodellverhaltens ermittelt.

## 7.2 Robuste Kalibrierung im Makro-Mikro-Verfahren

Die Entwicklung der *IVM* hat einen weiteren positiven Nebeneffekt. Die *IVM* repräsentieren das für das Gesamtverhalten der Simulation relevante Verhalten eines Teilmodells. Dies bedeutet, dass das Teilmodell durch sein zugehöriges *IVM* ersetzt werden kann und trotzdem das gleiche Gesamtverhalten entsteht. D.h. würde man statt der originalen Teilmodelle auf Mikro-Ebene nur ihre aggregierten Pendant-*IVMs* simulieren muss daraus valides Gesamtverhalten entstehen. Da die *IVM* ermittelt werden bevor die zugehörigen Teilmodelle auf der Mikro-Ebene kalibriert werden, können sie benutzt werden, um als fehlerhaft identifizierte Teile des Modells auf der Mikro-Ebene durch ihre validen *IVM*-Pendants zu ersetzen. Hierdurch wird die Validität des Gesamtverhaltens auf Makro-Ebene erhöht, da dieses nicht mehr durch fehlerhafte Teilmodelle verzerrt wird. Ein Nachteil dieses Vorgehens ist, dass durch Ersetzung von Mikro-Modellteilen durch idealisierte Teile der Erklärungswert des Modells reduziert wird, weil nicht mehr das gesamte Makroverhalten aus dem zugrundeliegenden Mikroverhalten generiert wird. Bewusst angewendet überwiegt aber der Vorteil einer erhöhten Verhaltensvalidität auf Makro-Ebene, wenn der Erklärungswert nicht zu sehr beschränkt wird.

Der entwickelte Ansatz zu robusten Kalibrierung versucht, solche „Problemteile“ zu identifizieren und ihren Einfluss auf das Simulationsverhalten durch den der zugehörigen ermittelten *IVM* zu ersetzen.

## 7.3 Heterogenitätssuche

Ein zentraler Punkt, der im Basis-Makro-Mikro-Verfahren nicht betrachtet wird ist, dass in ABS die erhoffte Validität trotz grundsätzlich korrekter Modellstrukturen für die Agenten auch oft deswegen nicht erreicht werden kann, weil die Annahme homogener Parameterwerte für die Agenten nicht zutreffend war. Eine solche Annahme wird z.B. deshalb getroffen, um die Menge zu kalibrierender Parameterwerte zu reduzieren, wenn die Kalibrierung vollständig heterogener Parameterwerte für die Agenten keine praktikable Möglichkeit bei der Kalibrierung darstellt.

Zur Lösung dieses Problems wird die Heterogenitätssuche vorgestellt. Die Heterogenitätssuche kombiniert Kalibrierung mit Clusteringverfahren und identifiziert Gruppen von Agenten, welche homogene Parameterwerte haben sollten. Sie bestimmt die minimal notwendige Heterogenität für die Kalibrierung und kalibriert so die optimale Menge an Parameterwerten. Weiter kann die Heterogenitätssuche auch zur Analyse des Gesamtmodells eingesetzt werden.

## 7.4 Analyseverfahren für das Kalibrierungsproblem

Vor der Ermittlung der *IVM* für das Makro-Mikro-Verfahren ist die Identifikation geeigneter Teilmodelle ein wichtiges Problem. Die Wahl der Teilmodelle entscheidet, ob White-Box-Zerlegungstechniken für das Kalibrierungsproblem, ob Reverse-Engineering und zur Kalibrierung welcher Teile Heterogenitätssuche eingesetzt werden kann. Zur Identifikation relevanter Teilmodelle und der verschiedenen Aspekte des Gesamtkalibrierungsproblems wurde ein Analyseverfahren für ABS-Kalibrierungsprobleme zur Vorbereitung des Makro-Mikro-Verfahrens entwickelt.

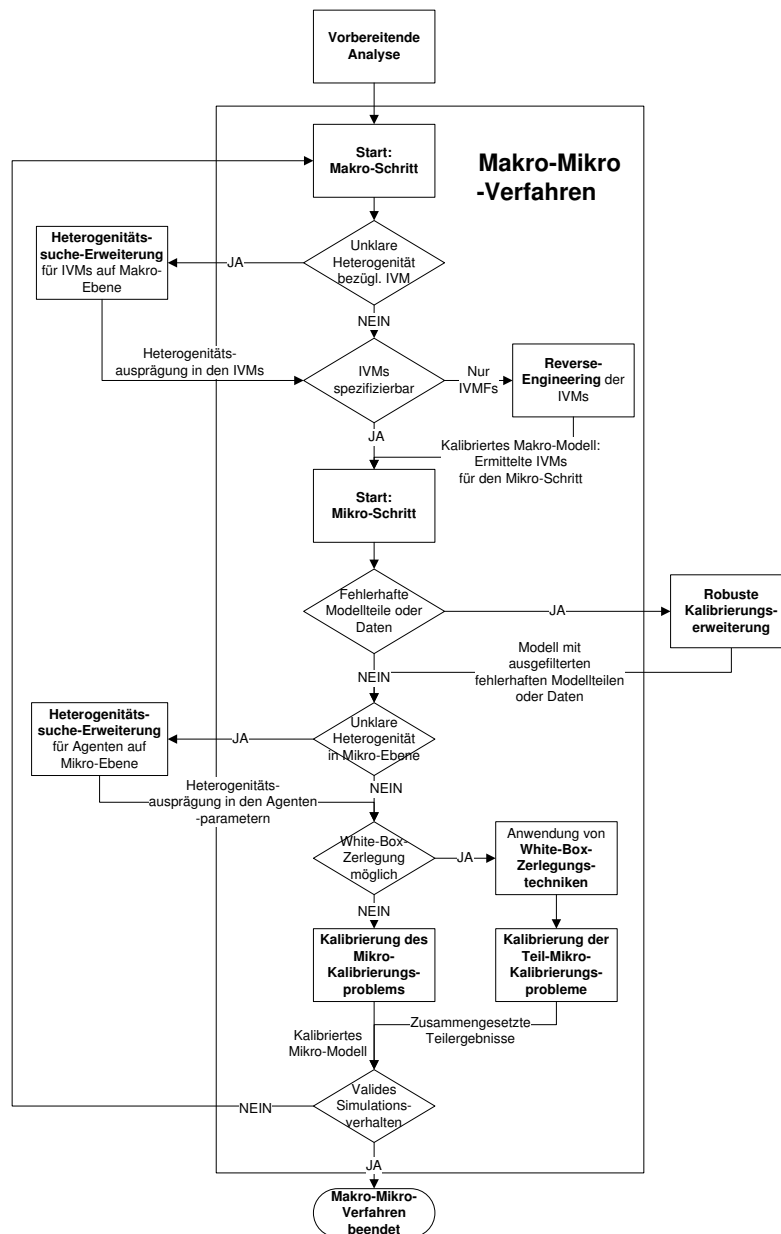


Abbildung 7.2: Überblick: Anwendung der verschiedenen Techniken im Makro-Mikro-Verfahren.

## 7.5 Fallstudie: Agenten-basierte Einkaufssimulation

In diesem Abschnitt wird die Fallstudie einer ABS zu Einkaufsentscheidungen beim Lebensmitteleinkauf beschrieben. Die Fallstudie wurde zusammen mit Projektpartnern vom Geographischen Institut der Universität Würzburg (Tilman Schenk, Jürgen Rauh und Günther Löffler) entwickelt [105]. Das Ziel der Fallstudie war es, ein Agenten-basiertes Simulationsmodell menschlicher Entscheidungen beim Lebensmitteleinkauf in der Region um die nordschwedische Stadt Umea zu entwickeln. Dabei sollten verschiedene Simulationsfragen beantwortet werden:

- **Verhaltensklärung:** Die wichtigste zu beantwortende Frage war, wie die Einkaufsentscheidungen der simulierten Bewohner modelliert werden sollten. Es sollte geklärt werden, aus welchen validen individuellen Einkaufsentscheidungen der Bewohner Umsätze der simulierten Geschäfte generiert werden können, die denen der echten Geschäfte entsprechen. Da bei der Modellierung der Einkaufsentscheidungen verschiedene mögliche Theorien in Frage kamen, waren unterschiedliche Modellausprägungen möglich.
- **Analyse alternativer Szenarien:** Wurde ein entsprechendes Simulationsmodell mit validen Einkaufsentscheidungen erstellt, sollten auf der Basis dieses Modells alternative Szenarien für Geschäftskonstellationen untersucht werden. Auf diese Weise sollten attraktive Standorte und relevante Eigenschaften für neue Supermärkte identifiziert werden. Umgekehrt sollte untersucht werden, welche Auswirkungen ein neuer Supermarkt auf die Umsätze bereits vorhandener Märkte haben würde, z.B. ob bestehende kleinere Geschäfte durch einen neuen Supermarkt „in den Ruin getrieben“ werden.

Im Rest des Abschnitt wird das Konzept des Simulationsmodells, verfügbares Wissen und Daten und die daraus resultierenden Modellierungs- und Kalibrierungsprobleme beschrieben. Die Beschreibung der Fallstudie ist auf die für diese Arbeit relevanten Aspekte begrenzt. Eine ausführliche Beschreibung des entwickelten Modells und der Ergebnisse der Simulationsstudie kann [105] entnommen werden.

### 7.5.1 Modellkonzept

In diesem Abschnitt wird auf den Aufbau des Modells eingegangen. Zunächst werden die einzelnen Komponenten des Modells beschrieben und dann die zu modellierenden Verhaltensweisen.

#### 7.5.1.1 Zeitfortschritt

Die Simulation ist Tick-basiert, d.h. die simulierte Zeit schreitet in gleichmäßigen 1-Jahresschritten fort.

#### 7.5.1.2 Modellbestandteile

Das Modell besteht im Wesentlichen aus drei Bestandteilen. Dem Umweltmodell, das die Region um die nordschwedische Stadt Umeå (Abbildung 7.3 zeigt eine Karte der simulierten Region) repräsentiert, den simulierten Geschäften des Lebensmittel-Einzelhandels und

den simulierten Bewohnern der Region, die ihren Lebensmitteleinkauf in den jeweiligen Geschäften durchführen.

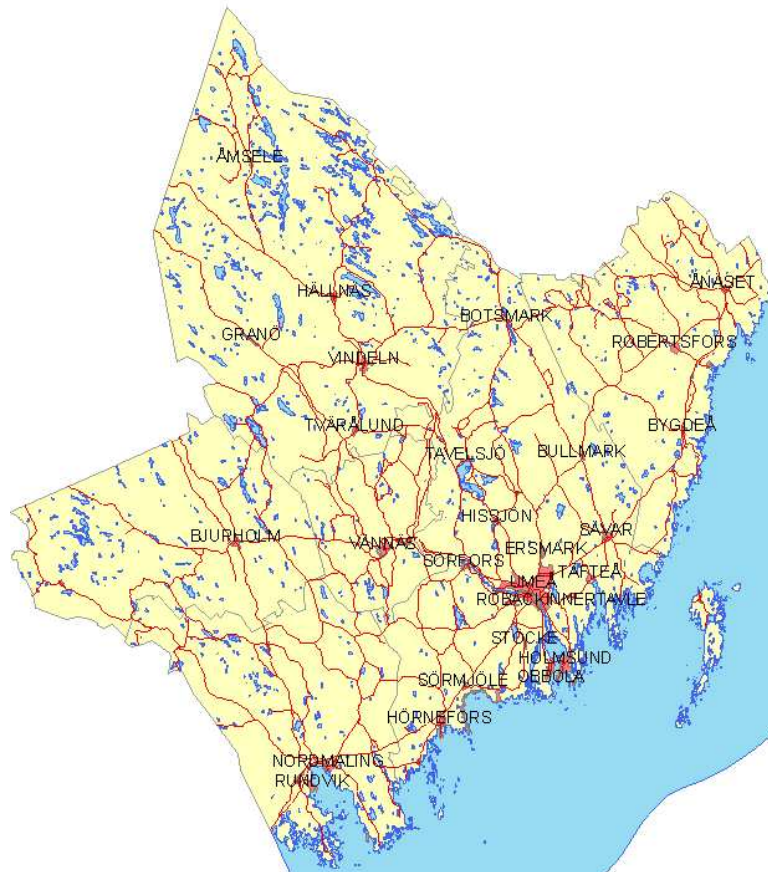


Abbildung 7.3: Karte der simulierten Region in Nordschweden.

- **Umweltmodell:**

Das Umweltmodell dient dazu, die räumlichen Gegebenheiten des simulierten Gebietes wiederzugeben. Für jedes Geschäft und jeden simulierten Einkäufer sind die genauen Standorte, respektive Wohn- und Arbeitsorte aus schwedischen Bevölkerungsstatistiken bekannt. Das Umweltmodell repräsentiert eine  $200 \text{ km}^2$  große Region. Den Kern der Region bildet die größte Stadt der Region, Umeå. In der Stadt befinden sich die meisten und größten Lebensmittelgeschäfte der ganzen Region. Das übrige simulierte Gebiet besteht lediglich aus mehreren kleinen Dörfern und wenigen mittleren Städten, die aber im Vergleich zu Umeå unbedeutend sind.

- **Modelle der Geschäfte:**

Im Modell werden ausschließlich Geschäfte des Lebensmitteleinzelhandels repräsentiert. Für jedes Geschäft ist dessen Jahresumsatz bekannt. Zusätzlich wurden für die Geschäfte eine Reihe von charakteristischen Merkmalen erhoben, anhand derer die einzelnen Geschäfte voneinander unterschieden werden können:

- **Preisniveau:** Das durchschnittliche Preisniveau des Geschäfts, z.B. für Discoun-

ter oder Delikatessengeschäfte.

- **Sortimentsgröße:** Die Angebotsvielfalt des Geschäftes.
  - **Qualität der Produkte:** Gemessen anhand der Qualität der angebotenen Obst- und Gemüsesorten.
  - **Qualität der Beratung:** Die Kompetenz des Personals bei Beratungsgesprächen.
  - **Atmosphäre im Geschäft:** Die allgemeine Atmosphäre im Geschäft. War es hell und sauber oder z.B. schmutzig?
  - **Lage des Geschäftes:** Die Lage des Geschäftes in Bezug auf andere Geschäfte. Agglomerationen von Geschäften können einzelne Geschäfte attraktiver machen, da mehrere Einkäufe auf einmal erledigt werden können.
  - **Standort des Geschäftes:** Die konkrete Position des Geschäftes. Über die Position kann jeder Einkäufer seine Entfernung zwischen Wohnort und betroffenem Geschäft ermitteln. Es wird vermutet, dass näher liegende Geschäfte für Einkäufer attraktiver sind.
- **Agentenmodelle der Einkäufer:**  
Die Bewohner der Region werden als Einkaufsagenten im Modell abgebildet. Jeder Einkaufsagent hat einen Wohnort in der simulierten Region von dem Einkaufstouren begonnen werden können. Der Ort wird entsprechend verfügbarer Daten über das Originalsystem festgelegt. Ein Einkaufsagent verfügt über ein Budget für den Lebensmitteleinkauf, das er in den simulierten Geschäften für Lebensmittel ausgibt. Jeder Einkaufsagent wird durch eine Reihe spezifischer sozio-ökonomischer Merkmale (Alter, Geschlecht, Familienstand, ...) charakterisiert. Über die sozio-ökonomischen Merkmale wird auch die Höhe des jeweiligen Einkaufsbudgets bestimmt. Abhängig vom Familienstand ist das Einkaufsbudget jeweils bezüglich einer Familie von Einkaufsagenten mit gleichem Wohnort oder bezüglich eines einzelnen (Single-)Agenten gegeben, der in diesem Fall seine eigene Familie bildet.

### 7.5.1.3 Verhaltenskonzept der Einkaufsagenten

Die Einkaufsagenten sind die einzigen Simulationsbestandteile mit Verhalten. Dieses Verhalten basiert auf den Schritten: „Bewerten“, „Auswählen“, „Geld Verteilen“. Das Einkaufsverhalten der Agenten wird zum einen durch die Wahrnehmung der charakteristischen Geschäftsmerkmale und zum anderen durch die individuellen sozio-ökonomischen Merkmale der Agenten bestimmt. Jeder Agent nimmt eine (Teil-)Menge der simulierten Geschäfte wahr und bewertet deren Attraktivität für sich. Einflussfaktoren für diese Bewertung können die charakteristischen Geschäftsmerkmale und individuelle Präferenzen der Einkaufsagenten sein. Letztere wurden mittels Regressionsanalysen aus Umfragedaten auf Basis der individuellen sozio-ökonomischen Merkmale ermittelt.

Für die Simulation wird die Annahme gemacht, dass die Lebensmitteleinkäufe in jährlichen Runden stattfinden. Aus diesem Grund ist das Einkaufsverhalten der Agenten wie folgt modelliert: Aus den Geschäftsmerkmalen aller von einem bestimmten Einkaufsagenten wahrgenommenen Geschäfte ergibt sich für diesen Agenten ein persönlicher Attraktivitätswert für das jeweilige Geschäft. Der Einkaufsagent gibt sein Einkaufsbudget dann proportional zu den so ermittelten Attraktivitätswerten in den jeweiligen Geschäften aus.

#### 7.5.1.4 Abhängigkeiten zwischen den Agenten

Es bestehen keine direkten, aber indirekte Abhängigkeiten zwischen den Verhaltensweisen der einzelnen Einkaufsagenten, die durch die unterschiedlichen Wohnorte bedingt sind. Macht ein Geschäft durch die Einkäufe von Agenten mit Wohnorten nahe dem Standort des Geschäftes bereits seinen Zielumsatz, so „müssen“ weiter entfernt wohnende Agenten dieses Geschäft als weniger attraktiv empfinden, da es sonst zu viel Umsatz machen würde. Umgekehrt könnte ein Agent in einer „abgelegenen“ Gegend ein Geschäft besonders attraktiv finden „müssen“, da nicht genügend andere Einkaufsagenten in dieser Gegend einkaufen gehen, um den aus den Daten bekannten Zielumsatz des Geschäftes zu erzeugen.

#### 7.5.2 Ablauf einer Simulation

Das entworfene Simulationsmodell wird in einzelnen zeitbasierten Simulationsschritten ausgeführt. Innerhalb eines Simulationsschrittes wird aus jeder Familie zufällig jeweils ein Einkaufsagent ausgewählt, der mit dem Familienbudget einkaufen geht. Die gewählten Agenten bewerten die Geschäfte auf der Basis ihrer individuellen Wahrnehmungen und geben entsprechend das Budget aus. Dieser Prozess aus zufälliger Agentenauswahl und Einkaufen wird solange iteriert, bis die durchschnittlichen Umsätze der Geschäfte konvergieren. Dies ist notwendig, da nicht bekannt ist, welche Einkaufsagenten einer Familie wie oft für die Familie einkaufen gehen. Das Verhaltensmodell der Agenten ist während eines Simulationslaufes statisch. Ein bestimmter Einkaufsagent kauft also immer auf die gleiche Weise ein und adaptiert sein Verhalten nicht. Das Einkaufsverhalten der Agenten kann somit nur durch Änderung ihrer individuellen Parameterwerte von außen angepasst werden.

#### 7.5.3 Verfügbare Daten und Wissen über das Originalsystem

Da Verfügbarkeit und Qualität von Daten entscheidenden Einfluss auf die Schwierigkeit des Modellierungs- und Kalibrierungsproblems haben, wird in diesem Abschnitt genauer auf die zur Modellentwicklung zur Verfügung gestandenen Daten eingegangen.

##### 7.5.3.1 Daten zu den simulierten Geschäften

Zu den simulierten Geschäften stehen Daten für verschiedene charakteristische Geschäftsmerkmale zur Verfügung.

Die Qualität der Daten ist potentiell stark unterschiedlich:

- **Positionsdaten:**

Die Positionen der Geschäfte sind bekannt und können als exakt betrachtet werden. Es waren jedoch keine Straßennetze verfügbar, was die Aussagekraft von Distanzen zwischen Agenten und Geschäften stark einschränkte.

- **Statistische Daten:**

Jedes simulierte Geschäft wird durch eine Reihe individueller Daten beschrieben. Die individuellen Umsätze der simulierten Geschäfte entstammen offiziellen amtlichen Statistiken und werden daher als korrekt betrachtet. Sortimentsgröße und das Preisniveau werden ebenfalls offiziellen Statistiken entnommen. Die entsprechenden Daten müssen trotzdem als potentiell ungenau betrachtet werden, da sie nur diskretisiert (jeweils 5 Stufen) zur Verfügung stehen. Aufgrund der Diskretisierung besteht die

Gefahr, dass auf ihrer Basis keine ausreichend genaue Unterscheidung zwischen den Geschäften getroffen werden kann. Zusätzlich ist fraglich, ob die realen Einwohner der Region diese Geschäftsmerkmale tatsächlich entsprechend der objektiven Daten wahrnehmen.

- **Subjektiv erhobene Daten:**

Die Daten zu den übrigen charakteristischen Geschäftsmerkmalen sind subjektiv von den Domänenexperten so systematisch wie möglich ermittelt. Zur Ermittlung der Qualität wurde z.B. ein Referenz-Warenkorb aus Obst und Gemüse verwendet. Die übrigen Merkmale wurden durch ähnliche Sub-Merkmale erhoben.

Da die Datenerhebung nur subjektiv geschah, kann keine beliebig genaue Unterscheidung zwischen den einzelnen Geschäften getroffen werden. Entsprechend werden die übrigen charakteristischen Merkmale in lediglich drei Stufen unterschieden.

Die subjektive Erhebungsmethodik macht es schwierig, die Eignung der Daten zur Verwendung bei Modellierung und Validierung einzuschätzen.

### 7.5.3.2 Daten zu simulierten Einkaufsagenten

Zur Beschreibung der simulierten Einkaufsagenten stehen amtliche Daten zu sozio-ökonomischen Merkmalen der Einwohner der Region auf Haushaltsebene zur Verfügung. Bezüglich des zu diesen Merkmalen gehörenden Verhaltens sind jedoch nur wenige Daten verfügbar, deren Verwendbarkeit unklar ist:

- **Positionsdaten:**

Die Positionen der Wohn- und Arbeitsorte der Agenten sind bekannt.

- **Amtliche statistische Daten:**

Die Einkaufsbudgets der Agenten sind mit Hilfe amtlicher Statistiken ermittelt und werden daher als korrekt angesehen. Weiter stehen individuelle Daten zu den sozio-ökonomischen Merkmalen der Agenten zur Verfügung.

- **Umfragedaten:**

In Umfragen wurde geklärt, welche Geschäftsmerkmale bestimmte Einkäufergruppen besonders wichtig finden.

Da die Umfrage vor dem Entwurf des ersten Konzeptmodells durchgeführt wurde, wurde eine Fragestellung verwendet, die für später zu erstellende Simulationsmodell ungeschickt war. Dennoch wurden mit Hilfe von Regressionsanalysen aus den sozio-ökonomischen Merkmalen der Einwohner und Umfragedaten individuelle Präferenzen für die Einkaufsagenten ermittelt.

Insgesamt ist für die ermittelten Präferenzdaten unklar, ob sie die echten Präferenzen der Einkäufer korrekt widerspiegeln oder nicht zu verwenden sind. Im Projekt stellte sich heraus, dass die ermittelten individuellen Präferenzen das Simulationsverhalten nicht verbessern konnten. Effektiv stellten sie nur eine Verrauschung des Agentenverhaltens dar. Da die Präferenzdaten für die Entwicklung und Kalibrierung des Simulationsmodells problematisch waren, werden diese Daten im weiteren Verlauf dieser Arbeit nicht verwendet und im Evaluationsteil nicht diskutiert.



### 7.5.4 Validierungsaspekte

Die Validität des Simulationsverhaltens muss auf zwei Beobachtungsebenen sichergestellt werden, der Ebene der Gesamtverhaltens und der Ebene des individuellen Agentenverhaltens.

#### 7.5.4.1 Validierung auf der Makroebene

Das Gesamtverhalten des Modells wird daran validiert, wie gut die simulierten Umsätze der Geschäfte die Umsätze der Geschäfte im Originalsystem reproduzieren können. Das individuelle Verhalten der einzelnen Einkaufsagenten spielt hier keine Rolle.

Als Validierungskriterium wurde im Projekt auf Wunsch der Domänenexperten die folgende Zielfunktion  $ZF_{unquadrat}$ , die im weiteren Verlauf als das „**unquadratische Gütemaß**“ bezeichnet wird, verwendet:

$$ZF_{unquadrat} = 1 - \frac{\sum_i^{\#Geschäfte} |SimUmsatz_i - OrigUmsatz_i|}{\sum_i^{\#Geschäfte} OrigUmsatz_i}$$

wobei:

- $SimUmsatz_i$  der simulierte Umsatz von des i-ten Geschäftes.
- $OrigUmsatz_i$  der echte Umsatz des i-ten Geschäftes.

Die Validierung des Modells wird auch in dieser Arbeit auf der Basis von  $ZF_{unquadrat}$  durchgeführt. Aufgrund der Struktur von  $ZF_{unquadrat}$  gehen Geschäfte mit größerem Umsatz wesentlich stärker in das Gütemaß ein, als Geschäfte mit kleinen Umsätzen (der Unterschied in der Höhe der einzelnen Umsätze beträgt aufgrund der Heterogenität der Geschäfte mehr als Faktor 100). Als Folge kann die Verwendung dieses Gütemaßes während der Kalibrierung Probleme bereiten. Dies ist z.B. der Fall, wenn zur Erzeugung validen Modellverhaltens Geschäfte mit geringeren Umsätzen eine größere Rolle spielen müssten, d.h. die Kalibrierung stärker auf Basis der Umsätze der kleinen Geschäfte durchgeführt werden müsste.

Im Evaluationsteil der Arbeit (siehe Kapitel 14), wird an einer Stelle an der die Untersuchungsergebnisse nahe legen, dass bestimmte Effekte während der Kalibrierung auf die besondere Struktur von  $ZF_{unquadrat}$  zurückzuführen sind, eine alternative Zielfunktion  $ZF_{echt}$  eingesetzt. Diese wird im weiteren Verlauf als „**echtes unquadratisches Gütemaß**“ bezeichnet.

$$ZF_{echt} = 1 - \sum_i^{\#Geschäfte} \frac{|SimUmsatz_i - OrigUmsatz_i|}{OrigUmsatz_i}$$

In  $ZF_{echt}$  gehen die relativen Verschätzungen aller Geschäfte in gleichem Maß in die Validitätsbewertung ein.

Sowohl  $ZF_{unquadrat}$  als auch  $ZF_{echt}$  haben ihr Optimum bei 1, d.h. das Simulationsverhalten wird als valide bewertet, wenn die Funktionen den Wert 1 annehmen.

Die Struktur der Parametersuchräume von  $ZF_{unquadrat}$  und  $ZF_{echt}$  ist jedoch jeweils unterschiedlich. Dies bedeutet, dass eine Parametereinstellung A, die bezüglich  $ZF_{unquadrat}$  besser bewertet wird, als eine Einstellung B, für  $ZF_{echt}$  durchaus schlechter bewertet werden kann als B. Auch andere Zielfunktionen wären für eine Kalibrierung möglich.

#### 7.5.4.2 Validierung auf der Mikroebene

Als Validierungskriterium auf der Mikroebene wird von den Domänenexperten gefordert, dass die einzelnen simulierten Einkaufsagenten ihr Einkaufsbudget plausibel auf die einzelnen Geschäfte verteilen. Wichtige Aspekte für die Validierung sind dabei, in welchen Geschäften die Agenten die größten Anteile ihres Budgets ausgegeben. Im Projekt war es den Experten nicht möglich formale Validierungskriterien auf der Mikro-Ebene zu spezifizieren. Es war ihnen zwar möglich durch Betrachtung der individuellen Verteilung des Einkaufsbudgets einzelner Agenten der Simulation zu bewerten, wie plausibel diese erschienen. Da jedoch keine konkreten Daten über die individuellen Einkaufsspuren der Einwohner im echten System verfügbar waren, konnte diese Plausibilitätseinschätzung nicht formalisiert werden. Eine Alternative zur Validierung anhand konkreter Daten wäre die Formulierung von Randbedingungen gewesen. Beispielsweise hätten maximale Längen für valide Einkaufstouren spezifiziert werden können. Leider war auch die Formulierung konkreter Tourlängen oder ähnliches problematisch, da aufgrund der regionalen Unterschiede (Stadt, Umland, kleine Stadt, Dorf, Land . . .) und unterschiedlicher sozio-ökonomischer Merkmale nicht davon ausgegangen werden konnte, dass für alle Agenten die gleichen Randbedingungen an das Einkaufsverhalten gültig waren. Die Validierung auf der Mikroebene musste daher durch **Plausibilitätsanalysen** „von Hand“ durchgeführt werden.

Zur Plausibilitätsanalyse wurden während eines Simulationslaufes die Einkaufswege der simulierten Agenten aufgezeichnet, d.h. es wurde visualisiert in welchen Geschäften die Agenten wie große Anteile ihres Einkaufsbudgets ausgegeben haben. Die in der Simulation ermittelten Einkaufswege wurden dann für Stichproblem einzelner Agenten von den Experten hinsichtlich ihrer Plausibilität im Vergleich zu bekanntem realen Einkaufsverhalten bewertet. Das Verhalten wurde als valide erachtet, wenn bei den Stichproblem kein Agent gefunden werden konnte, dessen Einkaufswege als unrealistisch erachtet wurden.

#### 7.5.5 Modellierungsprobleme

Bezüglich des Einkaufsverhaltens der Agenten sind mit den Einkaufsbudgets und Geschäftsumsätzen gute Daten zur Validierung auf der Makro-Ebene verfügbar. Zur Modellierung und Validierung des Mikroverhaltens der Agenten sind hingegen kaum konkrete Daten vorhanden. Als Folge müssen viele Modellierungsentscheidungen auf Annahmen basieren. Sind bei der Realisierung einzelner Modellaspekte unterschiedliche Annahmen möglich, führt dies zu unterschiedlichen möglichen Modellausprägungen, aus denen eine geeignete ausgewählt werden muss.

Bei der Modellierung stellen sich die folgenden Probleme:

- Wie muss der interne Wahrnehmungs- und Entscheidungsprozess der Agenten modelliert werden?
- Darauf aufbauend stellt sich die Frage, aufgrund welcher charakteristischen Geschäftsmerkmale die Agenten ihre Einkaufsentscheidung treffen müssen. Möglicherweise ist nur eine Teilmenge relevant oder es fehlen entscheidende Geschäftsmerkmale zur Erstellung des Modells?
- Sollen alle Agenten das gleiche Verhalten bekommen oder wird Heterogenität in den Eigenschaften der Agenten benötigt?

Im Folgenden werden die wesentlichen Problemaspekte der Fallstudie beschrieben.

### 7.5.5.1 Allgemeine Realisierung der Bewertung von Geschäftsmerkmalen

Das grundlegende interne Entscheidungsmodell der Agenten wurde auf Basis von „Bewerten“, „Auswählen“, „Geld Verteilen“ festgelegt [104]. Hierzu muss für die Agenten zunächst ein Modell zur individuellen Bewertung der einzelnen Geschäfte festgelegt werden. Anhand dieses Bewertungsmodells sollen die Agenten beurteilen, welche Geschäfte sie attraktiv finden und welche nicht.

Als dieses interne Bewertungsmodell für die Agenten wird eine Verrechnungsfunktion  $BewFkt$  gewählt. Mit  $BewFkt$  bilden die Agenten ihre Wahrnehmung der charakteristischen Merkmale der einzelnen Geschäfte auf einen spezifischen Attraktivitätswert für das jeweilige Geschäft ab. Die genaue Festlegung der Struktur  $BewFkt$  ist nicht trivial. Für den Entwurf von  $BewFkt$  wurden im Simulationsprojekt verschiedene Alternativen untersucht gegeneinander verglichen [104]. Für die Illustration von Kalibrierungsmethoden in dieser Arbeit wird mit der folgenden Grundstruktur von  $BewFkt$  gearbeitet.

#### Verrechnungsfunktion $BewFkt$

Eine mögliche Grundstruktur von  $BewFkt$  ist:

$$BewFkt(Geschäft_i, Agent_k) = \sum_{j=1}^{\#charakt. Merk.} f_{Wahrn(k,j)}(Geschäft_i), \\ i, k, j \in \mathbb{N}, i \in [1; \#Geschäfte], k \in [1; \#Agenten]$$

In diesem Fall wird von einem Agenten  $k$  die Attraktivitätsbewertung zu einem  $Geschäft_i$  berechnet, indem Agent  $k$  einzelne charakteristische Merkmale  $j$  von  $Geschäft_i$  bewertet. Die Bewertung für jedes Merkmal erfolgt jeweils mit einer eigenen Teilbewertungsfunktionen  $f_{Wahrn(k,j)}$ . Die Einzelwerte der Teilbewertungen werden aufsummiert. Das Problem beim Entwurf von  $BewFkt$  besteht darin, eine geeignete Ausprägung von  $BewFkt$  zu finden, d.h. zu entscheiden, welche individuellen Geschäftsmerkmale in die Bewertung der Agenten einfließen müssen.

$BewFkt$  ist nur ein mögliches Beispiel für eine entsprechende Bewertungsfunktion. Die einzelnen  $f_{Wahrn(k,j)}$  könnten z.B. auch multipliziert oder dividiert werden.

Im Folgenden wird auf Modellierungsmöglichkeiten für die einzelnen  $f_{Wahrn(k,j)}$  eingegangen. Dabei wird zwischen der Wahrnehmungsfunktion für die Entfernung zwischen den Geschäften und den Wohnorten der einzelnen Agenten und Nicht-Distanz-basierte Wahrnehmungsfunktionen für die übrigen charakteristischen Geschäftsmerkmale unterschieden.

#### Nicht-Distanz-basierte Wahrnehmungsfunktionen

In diesem Abschnitt werden die  $f_{Wahrn(k,j)}$  behandelt, die nicht zur Wahrnehmung der Entfernung zu den einzelnen Geschäften dienen (= Wahrnehmungsfunktionen für nicht-Distanz Merkmale der Geschäfte). Wie für  $BewFkt$  kann auch die Struktur für die  $f_{Wahrn(k,j)}$  nicht von vorne herein festgelegt werden. Zur Auswahl stehen z.B. lineare, logarithmische, exponentielle oder logistische Varianten für die  $f_{Wahrn(k,j)}$ . Im Projekt wurden diese verschiedenen Alternativen getestet.

Im Evaluationsteil dieser Arbeit werden für jedes Merkmal der Geschäfte, die nicht die Entfernung zu den Wohnorten der Einkaufsagenten betreffen, logistische Teilbewertungsfunktionen  $f_{Wahrn(k,j)}$  mit jeweils drei Parametern  $P_{j_1}$ ,  $P_{j_2}$  und  $P_{j_3} \in \mathbb{R}$  für jeden Agenten  $j$  verwendet:

$$f_{Wahrn(k,j)} = \frac{10^{P_{j_1}}}{1 + 10^{P_{j_2}} * e^{(-P_{j_3} * k)}}$$

Diese Auswahl wurde getroffen, da mittels der logistischen Funktion auch lineare, exponentielle und logarithmische Verläufe beschrieben werden können [104].

### Distanz Wahrnehmungsfunktion

Die Entfernung zwischen ihrem Wohnort und den einzelnen Geschäften durch die Agenten ist ein besonders wichtiger Wahrnehmungsaspekt. Während die Stufenwerte der Nicht-Distanz Geschäftsmerkmale durch die Agenten homogen wahrgenommen werden können, muss die Wahrnehmung des Standortes eines Geschäftes für jeden Agenten heterogen sein. Jeder Agent hat einen eigenen Wohnort, von dem aus er die Distanz zu den jeweiligen Geschäften bestimmt.

Echte Kartendaten mit realen Straßenverläufen etc. standen nicht zur Verfügung, so dass auch der Entwurf der Distanzwahrnehmung nicht trivial war. Für die Modellierung der Distanzwahrnehmung wurden im Projekt verschiedene Realisierungsmöglichkeiten (z.B. ebenfalls logistische oder exponentielle Funktionen) getestet. Die vielversprechendsten Realisierungsmöglichkeiten sind:

- **Lineare Distanzwahrnehmung:** In diesem Fall wird von den Agenten die Luftlinien-Distanz zu den einzelnen Geschäften wahrgenommen.
- **Rang-basierte Distanzwahrnehmung:** In diesem Fall nehmen die Agenten nicht die absolute Distanz zu den Geschäften wahr. Stattdessen „sortiert“ jeder Agent die von ihm wahrgenommenen Geschäfte nach seiner Nähe zu diesen und ordnete ihnen entlang der Sortierung einen Rang zu. Die Geschäfte wurden dann als „nächstes Geschäft“, „zweitnächstes Geschäft“, ... etc. wahrgenommen. Geschäfte mit gleicher absoluter Distanz zu einem Agenten haben für den Agenten den gleichen Rang. Der Effekt der Rang-basierten im Vergleich zur linearen Wahrnehmung liegt darin, dass die Agenten Geschäfte unabhängig von ihrer absoluten Distanz miteinander vergleichen. Ein Agent auf dem Land, dessen nächster Supermarkt 20 km entfernt liegt, gibt diesem Rang 1, ebenso wie ein Agent in der Stadt, dessen nächster Supermarkt nur 1 km entfernt ist. Diese Variante der Distanzwahrnehmung wird auch im Evaluationsteil der Arbeit verwendet.

#### 7.5.5.2 Auswahl relevanter Geschäftsmerkmale

Da es zur Erklärung des Einkaufsverhaltens wichtig ist, die für das Verhalten relevanten Merkmale zu identifizieren, müssen verschiedene Modellausprägungen mit unterschiedlichen Kombinationen wahrgenommener Merkmale untersucht werden. Nur so können die Auswirkungen verschiedener Wahrnehmungskonstellationen untersucht werden.

### 7.5.5.3 Modellierung individueller Agenten

In diesem Abschnitt werden Unsicherheiten bei der Modellierung individueller Agenten dargelegt. Zur Erstellung des Modells standen Daten für 130.000 Einkaufsagenten zur Verfügung. Im Modell selbst wurde eine Teilmenge von 1000 Einkaufsagenten simuliert. Eine Entscheidung, ob für die Agenten homogene Parameterwerte gewählt werden können, konnte von den Experten vorab nicht getroffen werden. Die Domänenexperten vermuteten zwar, dass es möglich wäre für bestimmte Gruppen von Agenten homogene Parameterwerte zu suchen, allerdings war es ihnen nicht möglich geeignete Kriterien für konkrete Einteilungen der Agenten vorzugeben.

Die Teilbewertungsfunktionen  $f_{Wahrn(k,j)}$  bilden für jeden Agenten  $j$  die Wahrnehmung charakteristischer Geschäftsmerkmale ab.

Zur Beschreibung individuellen Agentenverhaltens stehen die folgenden Daten zur Verfügung:

- Die aus ihren sozio-ökonomischen Merkmalen bestimmte Höhe ihres Einkaufsbudgets für den Lebensmitteleinkauf.
- Ihren Ausgangsort für ihre individuellen Einkaufstouren.

Im weiteren Verlauf dieser Arbeit und im Modell, das im Evaluationsteil dieser Arbeit verwendet wird, wird versucht auf der Basis dieser beiden Eigenschaften eine Individualisierung der Agenten zu erzeugen (z.B. hinsichtlich der individuellen Parameterwerte der Agenten).

## 7.5.6 Probleme bei der Kalibrierung

In diesem Abschnitt wird auf die konkreten Parameterkalibrierungsprobleme eingegangen, die für bestimmte Modellausprägungen gelöst werden müssen. Die für das Modell zu kalibrierenden Parameter sind jeweils die Steuerungsparameter der einzelnen Wahrnehmungsfunktionen  $f_{Wahrn_j}$ .

### 7.5.6.1 Größe und Struktur des Parametersuchraums

Sollen die Parameterwerte aller Agenten kalibriert werden, kann dies zu sehr großen Parametersuchräumen führen. Die genaue Anzahl zu kalibrierender Parameterwerte ergibt sich aus der konkreten Anzahl simulierter Agenten mit heterogenen Parameterwerten und der Anzahl an relevanten Geschäftsmerkmalen, aufgrund derer die Agenten ihre Geschäftsbewertung ermitteln (benötigte Parameter in den Bewertungsfunktionen).

Ein mögliches Kalibrierungsproblem für einen Black Box Ansatz mit 1000 individuellen Einkaufsagenten, festgelegter Modellstruktur für die Umwelt (= feste räumliche Verteilung und Eigenschaften von Geschäften und Einkäufern) und heterogenen Agentenparameterwerten hätte beispielsweise die Form:

$$KP_{EK} = ((UMF, AMF), VK) = ((\{UM\}, \{IAMF_1, \dots, IAMF_{1000}\}), ZF_{unquadrat})$$

Jede der zu kalibrierenden Einkaufsagentenfamilien  $IAMF_i$  würde durch ein Tupel  $(S_i, V_i, < P_{i_{11}}, P_{i_{12}}, \dots, P_{i_{73}} >)$  beschrieben. In obiger Formel müssen für 1000 Familien einzelner Agentenmodelle ( $IAMF$ ) für jeweils sieben Wahrnehmungsfunktionen von Geschäftsmerkmalen jeweils drei Parameterwerte für die zugehörigen logistischen Teilbewertungsfunktionen eingestellt werden. Dabei steht z.B.  $P_{1000_{61}}$  für den ersten Parameter der sechsten Teilbewertungsfunktion der tausendsten zu kalibrierenden Agentenmodellfamilie. Es müssen folglich bei 1000 simulierten Agenten 21.000 und bei 100.000 Agenten entsprechend 2.100.000 Parameterwerte eingestellt werden.

Werden homogene Parameterwerte kalibriert, d.h. alle 1000 Agenten erhalten die gleichen Parameterwerte, sind es nur insgesamt 21 Parameter. Angenommen jeder Parameterwert könne lediglich 10 verschiedene Werte annehmen, so bestünde der Parametersuchraum für homogene Werte bereits aus  $10^{21}$  und für 100.000 Agenten aus  $100.000^{21}$  Werten. Die durchschnittliche Dauer eines einzelnen Simulationslaufes beträgt für das Modell dieser Fallstudie etwa 1 Minute. Dennoch ist es nur schwer möglich, das Modell in begrenzter Zeit durch einfaches Austesten aller möglichen Parameterkombinationen zu kalibrieren. Auch die Anwendung von reinen Black-Box-Kalibrierungsansätzen ist bei einer solch großen Zahl heterogener Parameterwerte nur wenig vielversprechend.

#### 7.5.6.2 Makro-Mikro-Lücke

Als Alternative zum Einsatz von Black-Box-Suchverfahren könnte man versuchen, Regelbasierte Kalibrierungsverfahren anzuwenden. Es war den Domänenexperten jedoch nicht möglich aus invalidem Verhalten auf der Makroebene die benötigten Änderungen an den individuellen Verhaltensweisen der einzelnen Agenten herzuleiten. Genauer gesagt war der Zusammenhang zwischen den Umsätzen der Geschäfte (Makro-Verhalten) und den Parameterwerten der individuellen Wahrnehmungsfunktionen der Agenten (Mikro-Verhalten) unklar. Daher konnten keine geeigneten Lernregeln für die Kalibrierung der Agenten auf Mikroebene spezifiziert werden.

#### 7.5.6.3 Verrauschte und fehlende Daten

Wie in Abschnitt 7.5.3 beschrieben wurde, sind Teile der Daten, welche die Grundlage für die Modellierung und Kalibrierung des Modells bilden, verrauscht. Die meisten Daten, wie Sortimentsgröße und Preisniveau stammen zwar aus Statistiken, aber es ist unklar, ob die Einkäufer diese Merkmale tatsächlich wie in der Statistik oder wie von den Experten erhoben wahrnehmen. Das Problem ist in dieser Fallstudie besonders kritisch, da das simulierte Agentenverhalten während der Kalibrierung so eingestellt werden soll, dass die Wahrnehmung der Geschäftsmerkmale durch die Agenten zur Reproduktion der Geschäftsumsätze aus dem Originalsystem führt. Werden die Geschäftsmerkmale durch verrauschte Daten beschrieben, so dass kein Zusammenhang zwischen den Merkmalen und den Geschäftsumsätzen besteht, können die Verhaltensweisen der Agenten nicht eingestellt werden. Ein zusätzliches Problem besteht darin, dass problematische Geschäftsmerkmale nicht einfach aus dem Modell entfernt werden können, da ihre (korrekte) Wahrnehmung durch die Agenten unter Umständen entscheidenden Einfluss auf die Validität des Gesamtverhaltens hat.

### 7.5.7 Zusammenfassung

Die beschriebene Fallstudie einer Agenten-basierten Einkaufssimulation wird in dieser Arbeit sowohl zur Demonstration einzelner Techniken, als auch zur Evaluierung der entwickelten Verfahren eingesetzt. Die Beschreibung der Fallstudie zeigt, dass die Entwicklung des Modells von einer großen Zahl an unsicheren Faktoren, sowohl auf Modellierungs- als auch auf Kalibrierungsseite bestimmt wird. Die beschriebenen Probleme sind kein Spezialfall dieser Fallstudie, sondern typisch für die Entwicklung von Simulationsmodellen, in denen Originalsysteme mit großer Komplexität auf einem hohen Detaillierungsgrad abgebildet werden sollen.

Sieht man von dem Problem unklarer Daten ab und nimmt man an, dass die Struktur des Modells festgelegt sei, ist das eigentliche Kalibrierungsproblem für heterogene Agentenparameter immer noch so komplex, dass es in begrenzter Zeit mit Black Box Methoden nicht sinnvoll gelöst werden kann. Im weiteren Verlauf dieser Arbeit werden verschiedene Techniken aufgezeigt, die dazu dienen mit den Modellierungsproblemen, den verrauschten Daten und dem Problem, einen geeigneten Kompromiss zwischen vollkommen homogenen und heterogenen Parameterwerten zu finden, umzugehen.





## Kapitel 8

# Problemanalyse

Im Einführungsteil dieser Arbeit wurde darauf hingewiesen, dass mit ABS sehr unterschiedliche Domänen behandelt werden können. Auch innerhalb einer Domäne können ABS sehr unterschiedliche Strukturen und Funktionsweisen haben. Bei einem konkreten zu erstellenden Simulationsmodell können verschiedene für die jeweilige Modellerstellung spezifische Modellierungs- und Kalibrierungsprobleme auftreten. Die in dieser Arbeit entwickelten Kalibrierungsverfahren adressieren diese Spezialprobleme. Die Lösungsverfahren können jedoch erst dann sinnvoll in einer konkreten Modellentwicklung eingesetzt werden, wenn die spezifischen Kalibrierungs- und Modellausprägungsprobleme dieses Problems identifiziert wurden.

Zur Ermittlung solcher Probleme wird in dieser Arbeit ein Analyseverfahren vorgeschlagen. Das Verfahren soll den Modellierer dabei unterstützen, während der Erstellung des Modellkonzeptes:

- Wissenslücken über den Zusammenhang zwischen Verhalten auf verschiedenen Beobachtungsebenen zu erkennen.
- Probleme bei der Festlegung von Heterogenität in den simulierten Agenten zu identifizieren.
- Modellausprägungsprobleme für (Teil)strukturen des Modells zu erfassen und zu beschreiben.
- Abhängigkeiten zwischen der Kalibrierung von Teilmodellen und damit Möglichkeiten für White-Box-Zerlegungen konkreter ABS-Kalibrierungsprobleme zu erkennen.

Im weiteren Verlauf dieses Kapitels wird das in dieser Arbeit entwickelte graphische Analyseverfahren für die Analyse konkreter ABS-Kalibrierungs- und Modellierungsprobleme beschrieben. Im Anschluss wird das Analyseverfahren anhand der im letzten Kapitel vorgestellten Fallstudie demonstriert. Das Analyseverfahren und die anhand der Fallstudie identifizierten Kalibrierungsprobleme bilden die Ausgangspunkte für das im weiteren Verlauf der Arbeit beschriebene Makro-Mikro-Kalibrierungsverfahren und die zugehörigen Ergänzungstechniken.

## 8.1 Analyseverfahren für konkrete Kalibrierungsprobleme

Das Ziel des Analyseschrittes im Rahmen der Modellerstellung besteht darin, auf der Ebene des Modellkonzeptes Probleme für die Kalibrierung und Validierung zu identifizieren, so dass diese Probleme formalisiert, im Modellkonzept berücksichtigt und durch spezielle Kalibrierungsverfahren behandelt werden können.

Bei der Durchführung der Analyse wird ein Graph erstellt, in dem ausgehend von validem Gesamtverhalten des Modells (abhängig vom jeweiligen Ziel der Simulationsstudie) die Ableitung der folgenden Punkte unterstützt wird:

- Relevante (Teil-)Verhaltensaspekte, aus denen das Gesamtverhalten der Simulation entsteht: Nur wenn diese validen Teilverhaltensweisen während eines Simulationslaufes erzeugt werden, kann daraus valides Gesamtverhalten entstehen.
- Relevante Beobachtungsebenen zur Analyse und Validierung des Simulationsverhaltens: Es wird geklärt auf welchen Beobachtungsebenen das Simulationsverhalten des Modells untersucht und validiert werden soll.
- Relevante Strukturteile für bestimmte (Teil-)verhaltensweisen: Damit das Gesamtkalibrierungsproblem mit White-Box-Techniken in Teilprobleme zerlegt werden kann, muss untersucht werden, welche Teilverhaltensweisen durch welche Teilmodelle auf Grundlage welcher festzulegenden Parameterwerte erzeugt werden.
- Unsicherheiten über den Wirkungszusammenhang zwischen den relevanten Beobachtungsebenen: Durch Analyse des Wirkungszusammenhangs unterschiedlicher Beobachtungsebenen wird die Makro-Mikro-Wissenslücke konkretisiert. Ist klar welche konkreten Wirkungszusammenhänge unbekannt sind können diese explizit während der Kalibrierung behandelt werden.
- Unsicherheiten bei der Festlegung von Heterogenität in den Werten der Agentenparameter: Es wird geklärt, für welche Agentenmodelle einer Modellfamilie bereits bekannt ist, ob sie gleiche oder unterschiedliche Parameterwerte benötigen und für welche dieser Punkt nicht geklärt ist. Gibt es Agentenmodelle bei denen ungeklärt ist, ob sie homogene oder heterogene Parameterwerte erhalten sollten kann die in dieser Arbeit entwickelte Heterogenitätssuche eingesetzt werden.
- Unsicherheiten bei der Realisierung bestimmter Teilstrukturen: Es wird geklärt, ob es Teile des Modells gibt, für die noch nicht klar ist wie genau die Modellstruktur entworfen werden sollte. Werden solche unklaren Teilstrukturen identifiziert, wird festgestellt, welche alternativen Strukturrealisierungen in Frage kommen. Sind die Alternativen bekannt und im Modell implementiert, so kann deren Vergleich automatisiert in den Kalibrierungsprozess integriert werden.
- Abhängigkeiten zwischen den Festlegungen (Modellierung und/oder Kalibrierung) bestimmter Modellteile bestehen: Eine Entscheidung bezüglich der Festlegung einer Teilstruktur *A* kann erst getroffen werden, wenn eine Entscheidung bezüglich der Festlegung einer Teilstruktur *B* getroffen wurde. Eine Teilverhaltensweise *C* kann nur auf der Basis einer bereits kalibrierten Teilverhaltensweise *D* durch Kalibrierung der

zugehörigen Teilmodellfamilie erzeugt werden. Auch dieser Analyseschritt dient dazu den Einsatz von White-Box-Techniken zu ermöglichen. Es genügt nicht nur Teilverhaltensweisen zu identifizieren. Zusätzlich muss auch geklärt werden, welche zugehörigen Teilkalibrierungsprobleme in welcher Reihenfolge gelöst werden können, bzw. müssen.

### 8.1.1 Graphische Elemente

Bei der Analyse eines konkreten Kalibrierungs- und Modellausprägungsproblems wird dann wie folgt vorgegangen:

1. Identifikation relevanter Beobachtungsebenen für Analyse und Kalibrierung.
2. Identifikation relevanter (Teil)verhaltensweisen der Simulation und Einordnung in die Beobachtungsebenen.
3. Identifikation von Strukturelementen, die zur Erzeugung der (Teil)verhaltensweisen benötigt werden.
4. Identifikation von Modellausprägungsunsicherheiten für die Strukturen.
5. Identifikation von Heterogenitätsunsicherheiten.
6. Identifikation von Abhängigkeiten zwischen (Teil)verhaltensweisen und Strukturfestlegungen.

Die zur Erstellung der graphischen Analyse verwendbaren Elemente werden in Abbildung 8.1 dargestellt. Element (a) dient zur Beschreibung bestimmter Verhaltensaspekte, die während eines Simulationslaufes erzeugt werden sollen. Mit Element (b) werden strukturelle Modellteile beschrieben, die geeignet entworfen werden müssen. Element (c) beschreibt problematische Heterogenitätsaspekte bei der Erstellung des Modells, also Unsicherheit welche Agentenmodelle homogene und welche heterogene Parameterwerte erhalten sollten. Element (d) dient zur expliziten Spezifikation möglicher alternativer Strukturvarianten zu einem durch (b) beschriebenen Strukturelement (die Alternativen werden durch verknüpfte Boxen untereinander dargestellt). Die Auflistung alternativer Strukturvarianten entspricht einer XOR-Beziehung zwischen den Varianten, d.h. es kann nur genau eine der Varianten im Modell umgesetzt werden. Mit (e) werden zu einem Element (b) mögliche Teilstrukturen beschrieben, die potentiell (e) in Kombination realisieren sollen (die kombinierbaren Elemente werden durch verknüpfte Boxen nebeneinander dargestellt). Die Auflistung kombinierbarer Strukturelemente entspricht einer OR-Beziehung zwischen den Elementen, d.h. es kann eine beliebige Teilmenge der Varianten im Modell umgesetzt werden. Eine explizite AND-Beziehung wird in der Analyse nicht verwendet. Teilstrukturen, die nur kombiniert im Modell vorkommen können werden bei der Analyse als zusätzliche Alternative zusammengefasst. Das Pfeilelement (f) dient dazu Abhängigkeiten zwischen der Festlegung bestimmter Strukturelemente und der Durchführung von Teilkalibrierungen zu deklarieren. Neben der Beschreibung von Modellabhängigkeiten mittels des Elements (f), können mit Hilfe zusätzlicher Gruppierungsboxen (g) Beobachtungsebenen, die für Analyse, Validierung und Kalibrierung relevant sind, gekennzeichnet werden.

Die Funktionsweise der graphischen Analyse wird im Folgenden anhand einer Beispielanalyse für das im letzten Kapitel eingeführte Einkaufsmodell verdeutlicht.

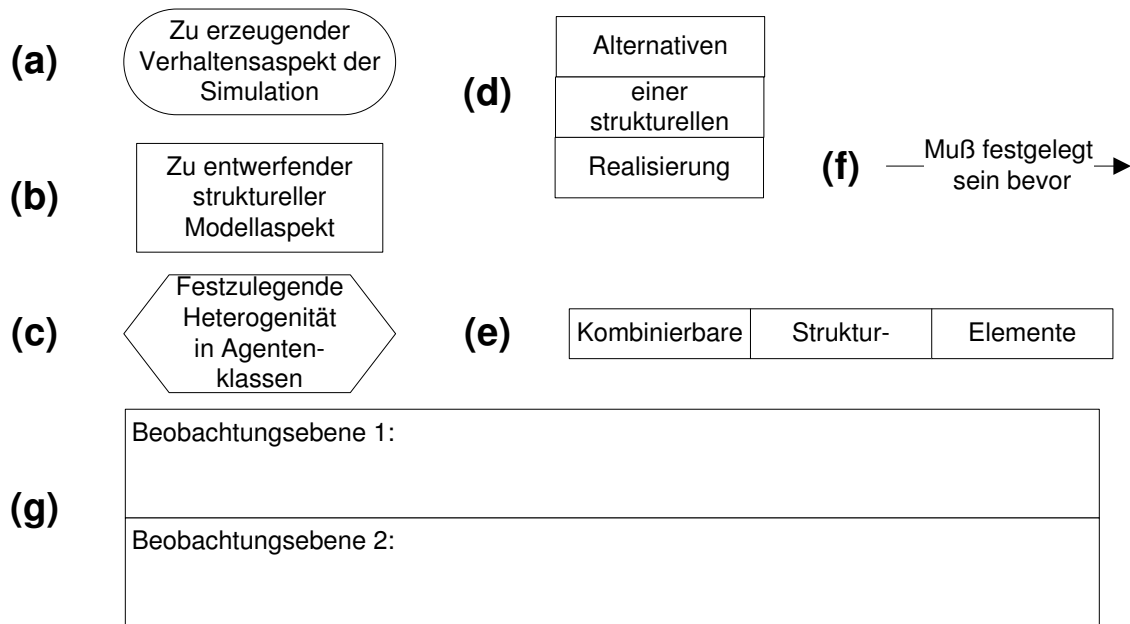


Abbildung 8.1: Zur graphischen Analyse des Modellierungs- und Kalibrierungsproblems verwendete graphische Elemente.

## 8.2 Beispielanalyse: Einkaufssimulation

Die Ergebnisse der Analyse werden in Abbildung 8.2 dargestellt.

**Beobachtungsebenen:** Der erste Analyseschritt besteht darin, zu klären, welche Beobachtungsebenen für die Analyse und Kalibrierung des Simulationsverhaltens relevant sind. Die Mikro-Ebene (in Abbildung Ebene 3) ist gegeben durch die Ebene auf der das konkrete Agentenverhalten mit Wahrnehmung, Bewertung und Verrechnung der Geschäftsmerkmale modelliert werden soll. Auf der Makro-Ebene (Ebene 1) wird das resultierende Gesamtverhalten der Simulation beobachtet, d.h. die Umsätze, welche die Geschäfte am Ende eines Simulationslaufes gemacht haben. Nun muss die Frage geklärt werden, ob es noch weitere relevante Beobachtungsebenen zwischen Makro- und Mikro-Ebene gibt.

Eine wichtige Zwischenebene ist die Ebene des resultierenden Agentenverhaltens (Ebene 2). Auf dieser Ebene wird beobachtet durch welche Werte die Agenten jedes einzelne Geschäft bewerten. Auf dieser Ebene sind die konkreten Aktionen/Interaktionen auf der Mikro-Ebene nicht relevant. Wichtig ist nur, was für Teilverhaltensweisen aus den Mikro-Aktionen entstanden sind. Auf Ebene 2 wird für das Einkaufsmodell daher nur das Ergebnis der Bewertungsberechnungen der Agenten beobachtet, aber nicht wie die Agenten die Werte konkret berechnet haben.

Auf Ebene 2 kann noch nicht genau bestimmt werden, ob das Gesamtverhalten (Geschäftsumsätze) aus den Teilverhaltensweisen (individuellen Geschäftsbewertungen) entstehen kann. Damit stellt Ebene 2 die Makro-Mikro-Lücke der Fallstudie dar. Wären Einkaufsentscheidungen, welche die simulierten Agenten für valides Gesamtverhalten treffen müssen, bereits bekannt, dann könnten die einzelnen Agentenmodelle unabhängig voneinander so kalibriert werden, dass jedes für sich die entsprechenden Entscheidungen trifft. Da dies nicht der Fall ist müssen theoretisch die Parameterwerte aller Einkaufsagenten gleichzeitig kalibriert und dann geprüft werden, ob ein Simulationslauf die Umsätze der

Geschäfte korrekt reproduzieren kann. Ebene 2 bietet aber eine Möglichkeit zur genaueren Analyse des entstehenden Verhaltens während eines Simulationslaufes.

**Teilverhaltensweisen:** Den identifizierten Beobachtungsebenen müssen nun die jeweiligen zu beobachtenden Teilverhaltensweisen zugeordnet werden. Auf der obersten Hierarchieebene (Ebene 1) wird global valides Modellverhalten beschrieben (a), das aus der korrekten Reproduktion der Geschäftsumsätze aus den Daten besteht. Auf Ebene 2 wird aggregiertes Agentenverhalten abgebildet, durch welches das Makro-Verhalten erzeugt wird. In der Fallstudie sind dies die während eines Simulationslaufes resultierenden Einkaufsentscheidungen der Agenten (b). Da die simulierten Agenten ihre Einkaufsentscheidungen auf der Grundlage einer Bewertung der Geschäfte durchführen sollen ist das Ergebnis des Entscheidungsprozesses durch die individuellen Gesamtbewertungen der Agenten für die einzelnen Geschäfte gegeben.

Auf der Mikro-Ebene des Agentenverhaltens geht es darum, wie die Geschäftsbewertungen von den Agenten konkret ermittelt werden, nämlich dadurch, dass die Einkaufsagenten die einzelnen charakteristischen Geschäftsmerkmale bewerten und verrechnen (c).

Mit den Verhaltensaspekten (a),(b) und (c) sind die relevanten und Teilverhaltensweisen, die für die Einkaufssimulation kalibriert und validiert werden müssen, bestimmt.

**Modellstrukturen:** Im nächsten Schritt werden die zugehörigen Modellstrukturen identifiziert. Zunächst muss ein valides Umweltmodell (d) entworfen werden, da der Entwurf und die Simulation aller anderen Modellbestandteile von diesem abhängt. Das Umweltmodell besteht aus der simulierten Region, den Geschäften, ihren charakteristischen Merkmalen und der Karte. Um aus den einzelnen Bewertungen der Geschäftsmerkmale (c) die Gesamtbewertungen der Geschäfte (b) berechnen zu können, müssen für die Agenten Strukturen zur Verrechnung der Einzelbewertungen modelliert werden (e). Die Einzelbewertungen der charakteristischen Merkmale werden von den Agenten dadurch berechnet, dass die relevanten Geschäftsmerkmale (g) durch Teilbewertungsfunktionen (f) bewertet werden. Mit diesem Schritt sind die wesentlichen Modellstrukturen identifiziert.

**Modellausprägungsunsicherheiten:** Für einige Modellstrukturen ist ihre konkrete Strukturprägung unklar. (h) zeigt verschiedene Alternativen zur Auswahl einer Verrechnungsfunktion für die Teilbewertungen. (i) zeigt Alternativen, mit denen die Teilbewertungen berechnet werden können. (j) schließlich identifiziert verschiedene charakteristische Merkmale, die möglicherweise in die Einkaufsentscheidungen der Agenten einfließen müssen. Mit (i), (h) und (j) sind Modellausprägungsprobleme dieser Fallstudie identifiziert worden.

**Heterogenitätsunsicherheiten:** Der vorletzte Schritt besteht in der Identifikation von Problemen der Heterogenitätsfestlegung in den Parameterwerten der Agenten. (k) beschreibt, dass unklar ist, ob alle Agenten die gleichen individuellen Parameterwerte für die Bewertungsfunktionen verwenden sollten oder ob heterogene Parameterwerte benötigt werden.

**Abhängigkeiten zwischen (Teil-)verhaltensweisen:** Im letzten Schritt werden die Abhängigkeiten zwischen den einzelnen Elementen abgebildet. Die Pfeile von (d) zeigen an, dass das Verhalten erst erzeugt werden kann, wenn ein valides Umweltmodell erstellt wurde. Das Gesamtverhalten kann erst erzeugt werden, wenn die individuellen Einzelbewertungen der Geschäfte ermittelt wurden (b) und diese können erst berechnet werden, wenn die konkreten individuellen Teilbewertungen auf der Mikro-Ebene ermittelt (c) und eine Verrechnungsfunktion (e) festgelegt wurde. Die Eignung einer Struktur für eine Teilbewertungsfunktion kann nur auf der Basis einer Auswahl relevanter Geschäftsmerkmale ermittelt werden (Pfeil (f) nach (g)). Der Pfeil von (k) nach (c) zeigt, dass die Heterogenitätsunsicherheit nur die Parameterwerte der Teilbewertungsfunktionen, nicht aber die Struktur-

auswahl selbst betrifft.

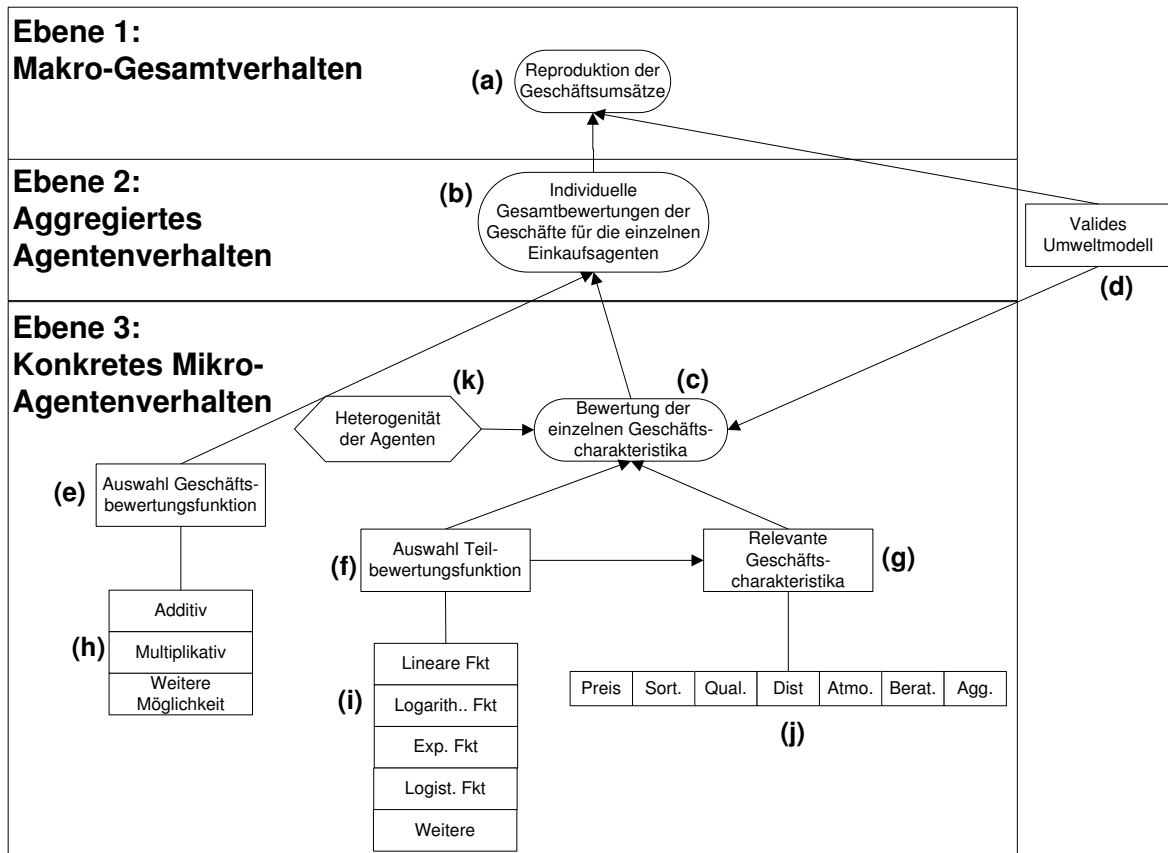


Abbildung 8.2: Graphische Analyse des Modellierungs- und Kalibrierungsproblems der Agenten-basierten Einkaufssimulation.

Zusammenfassend kann für die Fallstudie die ABS-Eigenschaft identifiziert werden, dass das Verhalten auf der Ebene des Gesamtverhaltens und auf der Ebene des individuellen Agentenverhaltens (= Geschäftsbewertungen) validiert und kalibriert werden muss. Beim Entwurf der Agentenmodelle gibt es einige Modellausprägungsprobleme. Zusätzlich stellt sich das Problem der Heterogenitätswahl in den Parameterwerten der Agenten.

Die Validierung des Simulationsverhaltens auf der Makro-Ebene ist dadurch gegeben, dass geprüft wird, wie gut die Umsätze der Geschäfte reproduziert werden. Die Validierung und damit die Kalibrierung des Mikro-Verhaltens ist schwieriger, da unklar ist, wie die individuellen Agenten die einzelnen Geschäfte bewerten sollten, damit valides Gesamtverhalten entsteht. Es besteht eine Makro-Mikro-Wissenslücke. Der Umgang mit diesem Problem wird in Kapitel 9.1.3 behandelt.

## Kapitel 9

# Das Makro-Mikro-Verfahren mit robuster Kalibrierung

In diesem Kapitel wird das entwickelte Makro-Mikro-Verfahren zur Kalibrierung von ABS und seine Erweiterung zu einem robusten Kalibrierungsverfahren bezüglich verrauschter Kalibrierungsdaten beschrieben. Das Kapitel geht von der Kalibrierung einer ABS mit der etablierten Black-Box-Kalibrierung aus, die aufgrund der in Kapitel 4.1 beschriebenen besonderen Eigenschaften von ABS-Kalibrierungsproblemen nicht zu einem zufriedenstellenden Ergebnis führt. Zunächst werden die Ursachen für das nicht zufriedenstellende Kalibrierungsergebnis aufgezeigt und dann das Makro-Mikro-Verfahren zur Lösung der identifizierten Ursachen eingeführt. In Abschnitt 9.1.3 wird das Makro-Mikro-Verfahren in seiner Basisvariante eingeführt, die grundlegende Funktionsweise erklärt und die Beschränkungen der Basisvariante aufgezeigt. Zur Lösung der Beschränkungen wird in Abschnitt 9.2 das Makro-Mikro-Verfahren durch ein Reverse-Engineering aggregierter Verhaltensmodelle von Teilmodellen verfeinert. Die durch das Reverse-Engineering ermittelten Verhaltensmodelle können zusätzlich genutzt werden, um das Makro-Mikro-Verfahren zu einem robusten Kalibrierungsverfahren zu erweitern. Dies wird in Abschnitt 9.3 beschrieben.

### 9.1 Black-Box-Kalibrierung und die Makro-Mikro-Lücke

Bisher wurden drei grundlegende Herangehensweisen zur Kalibrierung von Simulationsmodellen beschrieben: Black-Box-Ansatz, White-Box-Zerlegung und der wissensbasierte Kalibrierungsansatz. Von diesen Ansätzen ist der Black-Box-Ansatz der generellste und universal einsetzbarste, da zum Einsatz von Black-Box-Kalibrierung kein Vorwissen über die Struktur und Wirkungszusammenhänge des zu kalibrierenden Modells oder die Struktur des Parametersuchraums benötigt wird. Der entscheidende Vorteil hierbei ist, dass der Modellierer nach Fertigstellung der Modellstruktur nur die zu kalibrierenden Parameter und ihre Wertebereiche angeben muss und die Kalibrierung des Modells dann vollautomatisch erfolgen kann (die Verfügbarkeit eines entsprechenden Black-Box-Kalibrierungsverfahrens vorausgesetzt).

### 9.1.1 Black-Box-Kalibrierung von ABS

Möchte man eine ABS kalibrieren bietet es sich ebenfalls an, das Kalibrierungsproblem mit Black-Box-Methoden zu adressieren. Nach dem Ende der Kalibrierung muss untersucht werden, ob das erhaltene Kalibrierungsergebnis ausreichend valides Simulationsverhalten erzeugen kann oder ob das Simulationsverhalten so sehr zu wünschen übrig lässt, dass das Simulationsmodell so nicht wie geplant eingesetzt werden kann. Folgende Ergebnisse der Kalibrierung sind möglich:

1. Die Kalibrierung ist erfolgreich. Das erzeugte ABS-Verhalten ist auf Makro- und auf Mikro-Ebene valide. Damit ist die Kalibrierung beendet. In diesem Fall ist der Einsatz von Black-Box-Kalibrierung vollkommen ausreichend. Der Fall ist daher für diese Arbeit nicht interessant. Im Folgenden wird gezeigt, dass die Erwartung, dass dieses Ergebnis eintritt für größere, komplexere ABS nicht realistisch ist.
2. Es werden viele unterschiedliche Parametereinstellungen gefunden, die bezüglich des Gesamt-VK optimal sind, aber jeweils unterschiedliches Mikro-Verhalten erzeugen. Es ist unklar welche Parametereinstellungen als Kalibrierungsergebnis eingesetzt werden soll. Mögliche Ursachen hierfür sind:
  - (a) Das Kalibrierungsproblem ist unterbestimmt. Es müssen zu viele Parameterwerte eingestellt werden und die Suchraumstruktur bietet hierfür nicht genügend Randbedingungen, d.h. die verwendeten  $VK$  definieren valides Verhalten auf der Mikro-Ebene nicht genau genug.
  - (b) Die Annahme, dass es nur eine richtige Parametereinstellung oder ein Mikro-Verhalten, welches das Makro-Verhalten erzeugen kann, ist falsch.
3. Es wird keine Einstellung gefunden, die sowohl valides Verhalten auf Makro- als auch auf Mikro-Ebene generieren kann. Dies ist der Fall, der in dieser Arbeit betrachtet wird. Mögliche Ursachen für das Fehlschlagen der Black-Box-Kalibrierung sind:
  - (a) Der Parametersuchraum war zu groß und das Verfahren hat nicht mit der optimalen Einstellung terminiert. Für größere, komplexere ABS mit vielen Parameterwerten und langer Simulationslaufzeit besteht die Gefahr, dass in der für die Kalibrierung zur Verfügung stehenden Zeit nicht das optimale Kalibrierungsergebnis gefunden werden kann.
  - (b) Eine gemachte Homogenitätsannahme für die Parameterwerte der simulierten Agenten war nicht gerechtfertigt. In diesem Fall wurde zur Verkleinerung des Parametersuchraums eine falsche Annahme über die Struktur des Parametersuchraums gemacht und es somit unmöglich gemacht die optimale Parametereinstellung zu finden.
  - (c) Die angenommenen Validierungskriterien - einzelne oder ihre Gesamtheit - sind fehlerhaft. Ein Spezialfall hiervon sind fehlerhafte  $VK$  auf Mikro-Ebene, z.B. sind  $VK$  bezüglich einzelner Agenten falsch, wodurch diese Agenten trotz terminierter Black-Box-Kalibrierung nicht korrekt kalibriert sind.
  - (d) Die der Kalibrierung zu Grunde liegenden Modellstrukturen waren nicht angemessen. In diesem Fall muss das Modell selbst überarbeitet werden bevor eine erneute Kalibrierung durchgeführt werden kann.



### 9.1.2 Beispiel Fallstudie: Einkaufssimulation

Das beschriebene Problem kann an der in Kapitel 7.5 eingeführten Fallstudie nachvollzogen werden. Als erster Ansatz zur Kalibrierung des Modells wurde eine reine Black-Box-Herangehensweise mit evolutionärer-Algorithmens-Parametersuche gewählt. Auf der Makro-Ebene sollten Umsätze der Geschäfte reproduziert werden. Jeder Agent „entschied“ aufgrund seiner Bewertungsfunktionen wie er sein Einkaufsbudget auf die Geschäfte verteilen sollte. Die Parameter der Bewertungsfunktionen sollten auf der Mikro-Ebene kalibriert werden. Es wurde entschieden, dass die Agenten zunächst nur 3 Geschäftsmerkmale (Preis, Sortimentsgröße und Qualität) wahrnehmen sollten. Dies hätte bei heterogenen Parameterwerten und jeweils 9 Parametern pro Agent (siehe Kapitel 7.5) zu 9000 zu kalibrierenden Parameterwerten geführt. Da die gleichzeitige Kalibrierung dieser Parametermenge mit Black-Box-Suche im zeitlich beschränkten Rahmen des Projektes nicht machbar war wurde entschieden, dass homogene Parameterwerte für die Agenten kalibriert werden sollten. Das Ergebnis der Black-Box-Kalibrierung war, dass in allen Kalibrierungsläufen bezüglich der globalen Zielfunktion nie mehr als 58% Reproduktionsgenauigkeit der Geschäftsumsätze erreicht werden konnte. Dieses Ergebnis wurde als nicht valide genug eingestuft.

### 9.1.3 Makro-Mikro-Lücke in der ABS-Kalibrierung

Verantwortlich für die beschriebenen Probleme ist die sogenannte “Makro-Mikro-Lücke”, welche bei der Kalibrierung auf mehreren Beobachtungsebenen existieren kann. Darunter wird fehlendes Wissen verstanden, wie genau das Gesamtverhalten aus den individuellen Aktionen der simulierten Agenten auf der Mikro-Ebene entstehen soll. Es ist unklar welche Mikro-Ebenen-Aktionen zu welchem aggregierten Verhalten führen sollen, welches dann wiederum zu noch aggregierterem Verhalten führt und schlussendlich im Gesamtverhalten mündet. Für die Kalibrierung von ABS bedeutet dieses fehlende Wissen, dass wichtige Randbedingungen für das Kalibrierungsproblem fehlen.

In Kapitel 4.1 wurde darauf eingegangen, dass für ABS mit dem Auftreten der obigen und in Abschnitt 4.1.3.1 genauer beschriebenen Probleme bei der Kalibrierung mit Black-Box-Verfahren gerechnet werden muss. Es besteht die Gefahr, dass das Kalibrierungsergebnis nicht die Erwartungen des Modellierers erfüllen kann. Der Modellierer muss dann entweder dieses Ergebnis akzeptieren oder versuchen auf geeignete Weise zusätzliches Wissen in den Kalibrierungsprozess mit einzubringen, d.h. er muss versuchen die Makro-Mikro-Lücke zumindest so weit zu schließen, dass durch zusätzliches Wissen so viele zusätzliche Randbedingungen an das Modell gestellt werden, dass obige Probleme nicht mehr auftreten. Konkret bedeutet dies, dass er so viel zusätzliches Zusammenhangswissen in das Kalibrierungsproblem eingebracht werden muss, dass:

1. Die Suchraumstruktur so genau bestimmt wird, dass nur eine optimale Parameter-einstellung existiert, die zu validem Verhalten auf Makro- und Mikroebene führt.
2. Die Suchraumstruktur so genau bestimmt wird, dass das Kalibrierungsproblem geeignet in Teilprobleme zerlegt werden kann, welche dann mit insgesamt kürzerer Simulationslaufzeit gelöst werden können.

### 9.1.4 Das Basis-Makro-Mikro-Verfahren

In diesem Abschnitt wird das in dieser Arbeit entwickelte Makro-Mikro-Verfahren zur Kalibrierung von ABS in seiner Basisvariante vorgestellt. Die Idee des Makro-Mikro-Verfahrens besteht darin die Mehrebenen-Eigenschaft von ABS auszunutzen, um die Kalibrierungsqualität zu verbessern. Das Modellkonzept wird vom Modellierer mit dem in Kapitel 8.1 vorgestellten Verfahren analysiert. Durch die Analyse werden relevante Teilverhaltensweisen des Modells identifiziert, denen Teilmodelle des Mikro-Modells zugeordnet werden können. Auf Grundlage der identifizierten Teilverhaltensweisen werde eine oder mehrere Zwischenbeobachtungsebenen gebildet, welche den Zusammenhang zwischen Verhalten auf Mikro- und Makro-Ebene näher charakterisieren. Der Modellierer kann den Teilverhaltensweisen nun eigene zusätzliche Validierungskriterien zuweisen. Diese Validierungskriterien konnten bisher noch nicht in die Kalibrierung einbezogen werden, da die zugehörigen Betrachtungs(zwischen)ebenen nicht explizit im Modell behandelt wurden.

Die Schwierigkeit dabei besteht nun darin solche zusätzlichen Validierungskriterien für das Modell beschreiben zu können. Es können drei unterschiedliche Situationen auftreten:

1. Im besten Fall können vom Fachexperten Validierungskriterien für die Teilmodelle konkret spezifiziert werden. Durch die Identifikation der Teilverhaltensweisen auf den Zwischenbeobachtungsebenen kann bisher nur implizit vorhandenes Expertenwissen explizit gemacht und im Kalibrierungsprozess verwendet werden.
2. Im schlechtesten Fall ist beim Fachexperten kein Zusatzwissen vorhanden, um für die zusätzlichen Teilverhaltensweisen Validierungskriterien zu formalisieren. Damit in diesem Fall weitergemacht werden kann, müssen die Validierungskriterien „qualifiziert geraten“ werden. Werden die Validierungskriterien im Verlauf der weiteren Kalibrierung als ungeeignet identifiziert muss durch weiteres „qualifiziert raten“ versucht werden die *VK* geeignet abzuändern.
3. Ein Mittelweg zwischen den beiden Extremen ist dann möglich, wenn der Fachexperte zwar kein Wissen zur vollständigen Formalisierung der *VK* hat, es ihm aber möglich ist, zumindest Strukturen zur Beschreibung der *VK* für die Teilverhaltensweisen zu formalisieren (siehe dazu Kapitel 9.2).

Durch das Basis-Makro-Mikro-Verfahren können die ersten beiden Fälle adressiert werden. Der dritte Fall wird durch eine Erweiterung des Basis-Verfahrens behandelt.

#### 9.1.4.1 Informelle Beschreibung des Makro-Mikro-Verfahrens

Die Idee des Makro-Mikro-Verfahrens besteht darin, den Kalibrierungsprozess für ABS in zwei Bereiche aufzuspalten. Auf der Makro-Ebene wird versucht, unabhängig von den konkreten Mikro-Modellen der Agenten für die identifizierten Teilmodelle jeweils ein zugehöriges Modell validen aggregierten Teilmodellverhaltens durch ein ideales Verhaltensmodell *IVM* zu beschreiben. Dieses Verhaltensmodell dient dann als Validierungskriterium für das zugehörige Teilmodell auf der Mikro-Ebene. Auf der Mikro-Ebene wird danach versucht, die konkreten Parameterwerte des jeweiligen Teilmodells so einzustellen, dass es auf aggregierter Ebene dem des zugehörigen Teilverhaltensmodells entspricht. Ist die Kalibrierung auf der Mikro-Ebene abgeschlossen wird geprüft ob auch das Gesamtverhalten des Modells valide ist.

Im Optimalfall, d.h. wenn das Gesamtverhalten nun valide ist, ist die ABS-Kalibrierung damit beendet. Im schlechten Fall ist das Simulationsverhalten immer noch nicht zufriedenstellend, weil der Fachexperte die Validierungskriterien nur unzureichend schätzen konnte. In diesem Fall muss die Kalibrierung fortgeführt werden. Es wird nun wieder auf die Makro-Ebene gewechselt. Das aktuelle Gesamtverhalten muss analysiert und die Validierungskriterien für die Teilmodelle müssen auf der Basis dieser Analyse modifiziert werden. Das ganze wird so lange iterativ fortgesetzt, bis entweder nichts mehr an den Validierungskriterien geändert werden kann oder das erwünschte Simulationsverhalten eintritt.

Der iterative Schritt des Basis-Makro-Mikro-Verfahrens ist wenig befriedigend, da er auf ein wenig systematisches Ändern der Validierungskriterien bis das Gesamtverhalten der ABS stimmt hinaus läuft. In Abschnitt 9.2 wird eine Verfeinerung des Basis-Verfahrens zur Behandlung dieses Problems beschrieben.

Es sind aber auch Fälle denkbar in denen genau diese unsystematischen iterativen Änderungen der Validierungskriterien Anwendung finden. Dies ist z.B. der Fall, wenn das Makro-Mikro-Verfahren zur Optimierung statt zur Kalibrierung eingesetzt werden soll und noch nicht bekannt ist welche Teilverhaltensweisen zu optimalem Gesamtverhalten führen können. In diesem Fall sind die Optimierungskriterien für die Teilmodelle selbst Teil des Optimierungsproblems.

Im Folgenden wird das Basis-Makro-Mikro-Verfahren informell beschrieben. Abbildung 9.1 veranschaulicht den Prozess.

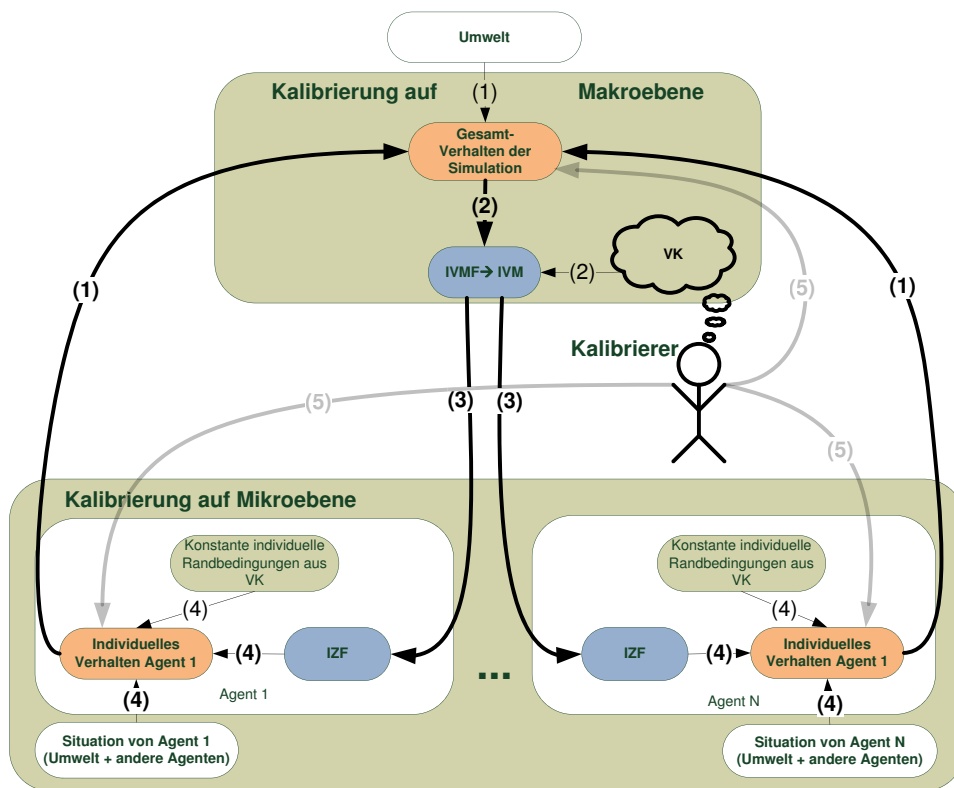


Abbildung 9.1: Praktischer Makro-Mikro-Kalibrierungsprozess.

**Eingaben des Makro-Mikro-Verfahrens:**

Das Makro-Mikro-Verfahren erhält als Eingabe zunächst das ursprüngliche Kalibrierungsproblem  $KP_{ABS}$ . Für das zu  $KP_{ABS}$  gehörende Modell wird die Analyse aus Kapitel 8.1 durchgeführt. Als Ergebnis der Analyse erhält man für das Gesamtverhalten der ABS relevante Teilverhaltensweisen auf Zwischenbeobachtungsebenen. Diese Teilverhaltensweisen können auf verschiedenen Aggregierungsebenen des Modellverhaltens liegen. Eine solche aggregierte Teilverhaltensweise kann z.B. aus dem Zusammenwirken von Agenten und ihrer Umwelt, Gruppen von Agenten oder im Idealfall aus dem Verhalten einzelner Agenten entstehen.

Basierend auf den identifizierten Teilverhaltensweisen muss der Modellierer oder Fachexperte im Basis-Makro-Mikro-Verfahren als zusätzliche Eingabe Modelle  $IVM$  (ideale Verhaltensmodelle) zur Beschreibung der Teilverhaltensweisen spezifizieren. Durch diese  $IVM$  wird für ein Teilmodell valides Verhalten beschrieben, das aus den Entscheidungen, Aktionen und Interaktionen der Agenten auf der Mikro-Ebene entstehen muss, um valides Gesamtverhalten der ABS zu erzeugen. Die direkte Vorgabe von  $IVM$  ist nur im Optimalfall möglich. Im schlechtesten Fall kann der Modellierer oder Fachexperte die  $IVM$  nur schätzen oder er kann nur  $IVMF$  - also Familien idealer Verhaltensmodelle - vorgeben, aus denen dann im Iterationsschritt des Makro-Mikro-Verfahrens die wirklich benötigten  $IVM$  ermittelt werden müssen.

**Makro-Schritt:**

Im Makro-Schritt wird nur der Zusammenhang zwischen  $IVM$  und der Validität des Gesamtverhaltens betrachtet. Das konkrete individuelle Agentenverhalten auf der Mikro-Ebene spielt auf dieser Kalibrierungsebene keine Rolle. Es geht nur darum die  $IVM$  so zu wählen, dass aus ihnen valides Gesamtverhalten entsteht. Im Optimalfall ist dieser Schritt durch die Vorgabe der  $IVM$  bereits erledigt. Es wird im Makro-Schritt nicht betrachtet, ob das durch  $IVM$  beschriebene Verhalten auf der Mikro-Ebene überhaupt erzeugt werden kann und ob auf Basis der  $IVM$  kalibrierte Teilmodelle auf der Mikro-Ebene auch bezüglich anderer Mikro-Validierungskriterien ausreichend valide sind. Solche Untersuchungen werden im zweiten Kalibrierungsschritt auf der Mikro-Ebene durchgeführt (Pfeile 3 in Abbildung 9.1).

**Mikro-Schritt:**

Für den Mikro-Schritt werden aus den im Makro-Schritt ermittelten  $IVM$   $IZF$  (Ideale ZielFunktionen)<sup>1</sup> erzeugt. In diesen  $IZF$  wird gemessen, wie genau das simulierte Verhalten eines Teilmodells auf der Mikro-Ebene mit dem Verhalten des zugehörigen  $IVM$  übereinstimmt. Die Kalibrierung der konkreten Agentenmodelle wird nun auf der Basis dieser  $IZF$  und sonstiger Validierungskriterien bezüglich der Mikro-Ebene aus der ursprünglichen Problembeschreibung durchgeführt (Pfeile 4).

<sup>1</sup>Nicht zu verwechseln mit den zuvor eingeführten  $IVK$ .  $IVK$  sind vorgegebene Validierungskriterien zur Bewertung individuellen Verhaltens auf der Mikro-Ebene.  $IZF$  sind zusätzliche Zielfunktionen zur Bewertung des Verhaltens von Teilmodellen auf der Basis zugehöriger idealer Verhaltensmodelle  $IVM$ .

Abhängig von den bei der Analyse identifizierten Teilverhaltensweisen kann das Kalibrierungsproblem auf der Mikro-Ebene zerlegt werden, indem das Gesamtkalibrierungsproblem in Teilkalibrierungsprobleme bezüglich der ermittelten *IZF* zerlegt wird oder das Wissen über Teilverhaltensweisen genutzt wird, z.B. um ein wissensbasiertes Kalibrierungsverfahren einzusetzen. Im schlechtesten Fall wird auf der Mikro-Ebene auf der Basis der zusätzlichen *IZF* mit Black-Box-Suche kalibriert. Im weiteren Verlauf des Abschnitts wird hierauf noch genauer eingegangen.

### **Bewertungsschritt:**

Nach dem Ende der Kalibrierung auf der Mikro-Ebene wird die Validität des erzeugten Simulationsverhaltens auf Makro- und Mikro-Ebene bewertet (Pfeile 1). Im Optimalfall entspricht das Verhalten der Agenten den *IVM*, ist bezüglich der übrigen Mikro-Validierungskriterien ausreichend valide und erzeugt valides Gesamtverhalten auf der Makro-Ebene. Aus verschiedenen Gründen kann es jedoch sein, dass das kalibrierte Verhalten nicht ausreichend valide ist. Beispielsweise kann es sein, dass die Parameter der Agenten nicht genau genug bezüglich der *IVM* eingestellt werden konnten. Eine andere Möglichkeit wäre, dass die Agenten zwar bezüglich *IVM* kalibriert wurden, aber das erzeugte konkrete Mikro-Verhalten bezüglich im ursprünglichen Kalibrierungsproblem vorgegebener *IPA* nicht ausreichend valide ist.

### **Iteration:**

Wird im Bewertungsschritt nicht ausreichend valides Simulationsverhalten auf Makro- und/oder Mikro-Ebene identifiziert, muss wieder zum Makro-Schritt zurückgekehrt werden. Das Simulationsverhalten muss analysiert werden (Pfeile 5) und es müssen geeignete Änderungen an den *IVM* vorgenommen werden (Pfeile 2). Auf Basis der angepassten *IVM* wird dann wieder zum Mikro-Schritt gewechselt (Pfeile 3) usw.

#### **9.1.4.2 Formale Beschreibung**

Ausgangspunkt der formalen Beschreibung ist eine durchgeführte Modellanalyse mit der Teilmodelle auf der Mikro-Ebene identifiziert wurden zu denen ein ideales Verhalten betrachtet werden kann. Das formale Vorgehen im Makro-Mikro-Verfahren besteht darin,  $KP_{ABS}$  in zwei voneinander abhängige Probleme  $KP_{MAKRO_{ABS}}$  und  $KP_{MIKRO_{ABS}}$  aufzuspalten.  $KP_{MAKRO_{ABS}}$  und  $KP_{MIKRO_{ABS}}$  werden im Makro-Mikro-Verfahren kombiniert behandelt (siehe Abbildung 9.2):

**Begriffsbestimmung 15: Makro-Mikro-Teilproblem  $KPMAKRO_{ABS}$** 

Das Kalibrierungsteilproblem auf der Makroebene wird beschrieben durch:

$$KPMAKRO_{ABS} = ((UMF, IVMF), VK) = ((\{UM\}, \{IVMF_1, \dots, IVMF_v\}), VK)$$

- $VK$ : Validierungskriterien aus  $KP_{ABS}$  für das Gesamtverhalten der Simulation.
- $IVMF = \{IVMF_1, \dots, IVMF_v\}$ : Menge an  $v \in \mathbb{N}$  Modellfamilie(n), zur Beschreibung **idealen Teilmodellverhaltens auf der Mikroebene**. Die  $IVM$  zu den  $IVMF$  sollen im Makro-Schritt so gewählt werden, dass ihr kombiniertes Verhalten zu Gesamtverhalten führt, das bezüglich  $VK$  ausreichend valide ist.

**$KPMAKRO_{ABS}$  behandelt also nicht die Parameter des eigentlichen  $KP_{ABS}$ . Es wird eine Zwischenebene idealen Teilmodellverhaltens betrachtet. Im Makro-Schritt müssen die  $IVM$  geeignet gewählt werden, um auf ihrer Basis den Mikro-Schritt durchführen zu können.**

Das zugehörige Kalibrierungsproblem  $KPMIKRO_{ABS}$  auf der Mikroebene ist eine Variante von  $KP_{ABS}$ , in der nicht mehr (nur) aufgrund der globalen  $VK$  kalibriert wird. Stattdessen werden die zu den  $IVM$  gehörigen Mikro-Teilmodelle so kalibriert, dass ihr aggregiertes Verhalten dem der in  $KPMAKRO_{ABS}$  gewählten  $IVM$  entspricht. D.h. die Mikro-Teilmodelle werden bezüglich zusätzlicher Zielfunktionen  $IZF$  kalibriert, die messen, wie genau das Mikro-Verhalten dem der  $IVM$  entspricht. Wie bereits geschrieben können einzelne Mikro-Teilmodelle beliebige Agenten und Umweltteile des Mikro-Modells enthalten. Im Folgenden wird der Mikro-Schritt für den Fall formalisiert, dass jeder Agent als eigenes Teilmodell betrachtet werden kann. Ist dies nicht möglich so können durch ein  $IAMF$  größere Teile als die Mikromodelle einzelner Agenten beschrieben werden.

**Begriffsbestimmung 16: Makro-Mikro-Teilproblem  $KPMIKRO_{ABS}$** 

Das Kalibrierungsteilproblem auf der Mikroebene wird beschrieben durch:

$$KPMIKRO_{ABS} = ((\{UM\}, \{IAMF_1, \dots, IAMF_n\}), VK_{Mikro})$$

- $VK_{Mikro}$ : Besteht aus den  $IZF$  der gewählten  $IVM$  und den in  $KP_{ABS}$  vorgegebenen übrigen Mikrovalidierungskriterien.
- $IAMF_i$ : Modellfamilie eines einzelnen Agenten auf der Mikro-Ebene  $i \in [1; n], n \in \mathbb{N}$

Die Schritte auf der Makro- und der Mikro-Ebene sind miteinander verknüpft. Jedes Mal, wenn auf der Makroebene die Wahl der  $IVM$  geändert wird erfordert dies eine Kalibrierung auf der Mikroebene. Konnte der Fachexperte die  $IVM$  nur schätzen kann erst nach der Lösung von  $KPMIKRO_{ABS}$  und der Analyse der Validität des resultierenden Gesamtverhaltens der Simulation bewertet werden:

1. ob aus den gewählten *IVM* valides Gesamtverhalten hervorgehen kann,
2. ob das Verhalten der Teil-Mikromodelle so kalibriert werden konnte, dass es auf aggregierter Ebene dem der zugehörigen *IVM* entspricht
3. ob das resultierende Gesamtverhalten des gesamten Mikromodells valide bezüglich der *VK*.

Ist einer der Punkte nicht erfüllt, so muss - unter der Annahme, dass die Modellstrukturen valide sind - geprüft werden, in welcher Weise die *IVM* modifiziert werden müssen. Abbildung 9.2 veranschaulicht den Prozess.

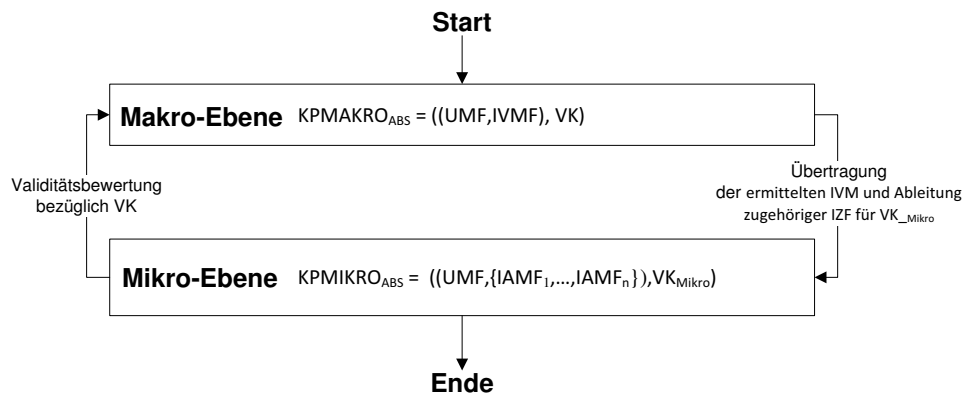


Abbildung 9.2: Allgemeiner Makro-Mikro-Kalibrierungsprozess.

Das Makro-Mikro-Verfahren endet, wenn die Parameter auf der Mikro-Ebene so eingestellt werden konnten, dass das Gesamtverhalten der Simulation valide ist und alle globalen und lokalen Validierungskriterien ausreichend gut erfüllt sind.

Ein entscheidender Punkt bleibt beim Basis-Makro-Mikro-Verfahren offen. Die Festlegung der *IVM* muss im Basis-Verfahren vom Fachexperten vorgegeben oder geschätzt werden. Auch die Änderung der *IVM* im Fall einer ungeeigneten Schätzung bleibt offen. Im nächsten Kapitel wird die Reverse-Engineering-Verfeinerung des Makro-Mikro-Verfahrens zur Behandlung des Problems der *IVM*-Wahl beschrieben.

### 9.1.5 Beispielanwendung: Basis-Makro-Mikro-Verfahren

In diesem Abschnitt wird das Basis Makro-Mikro-Verfahrens anhand der Fallstudie der Agenten-basierten Einkaufssimulation (siehe Abschnitt 7.5) veranschaulicht. Die Analyse des zugehörigen Kalibrierungsproblems kann Kapitel 8.1 entnommen werden. Betrachtet wird ein Kalibrierungsproblem  $KP_{EK}$  (Kalibrierungsproblem für das EinKaufsmodell), in dem in einer festgelegten Umwelt 1000 Einkaufsagenten *IAMF* kalibriert werden sollen:

$$KP_{EK} = ((\{UM\}, \{IAMF_1, \dots, IAMF_{1000}\}), \{ZF_{unquadrat}\} \cup IRB)$$

- $ZF_{unquadrat}$ : Die Zielfunktion zur Bewertung des Gesamtverhaltens. Es wird bewertet, wie genau die Umsätze der Geschäfte reproduziert werden können.

- *IRB*: Eine Menge von Randbedingungen an das Verhalten einzelner Agenten. Die charakteristischen Eigenschaften der Geschäfte und die Entfernung der Einkaufsagenten von den Geschäften bilden individuelle Randbedingungen an das Verhalten jedes simulierten Agenten. Die *IRB* fordern, dass sich die Eigenschaften der Geschäfte in den Entscheidungen der Agenten wiederfinden müssen, d.h. Geschäfte mit schlechteren Eigenschaften dürfen nicht besser bewertet werden als Geschäfte mit besseren Eigenschaften. Die Bewertung der Entfernung zu einem Geschäft wird von jedem Agenten bezüglich seines individuellen Wohnortes vorgenommen. Größere Entfernungen werden als schlechtere Geschäftseigenschaften bewertet.

In  $KP_{EK}$  wird nach Parameterwerten für die Bewertungsfunktionen der Einkaufsagenten gesucht, mit denen sie entscheiden, in welchen Geschäften sie einkaufen gehen (siehe Kapitel 7.5). Zur Validierung der Agenten auf der Mikro-Ebene stehen keine *IZF* zur Verfügung, d.h. es kann nicht angegeben werden, welche Agenten welche Geschäfte wie attraktiv finden müssen, um valides Gesamtverhalten zu erzeugen. Die Parameter der Bewertungsfunktionen der Agenten können daher im Grundproblem nur gleichzeitig bezüglich  $ZF_{unquadrat}$  und *IRB* kalibriert werden.

#### 9.1.5.1 Makroebene:

Die Modellanalyse aus Kapitel 7.5 zeigt, dass in  $KP_{EK}$  jeder Einkaufsagent als Teilmodell für die Kalibrierung betrachtet werden kann, da jeder Agent seine Einkaufsentscheidungen unabhängig von den übrigen Agenten trifft. Eine Funktion wie ein Agent sein Einkaufsbudget beim Einkaufen auf die einzelnen Geschäfte verteilt, d.h. wie attraktiv er die einzelnen Geschäfte findet (Ebene 2 in Abbildung 8.2), kann als aggregiertes Verhaltensmodell des Agenten verwendet werden. Vom Fachexperten vorgegebene  $IVM_{EK}$  müssen damit für jeden einzelnen Einkaufsagenten beschreiben, wie attraktiv er jedes wahrgenommene Geschäft finden muss, damit die Gesamtumsätze der Geschäfte reproduziert werden können, ohne dass die *IRB* verletzt werden.

Die Struktur  $IVMF_{EK}$  der  $IVM_{EK}$  ist leicht zu beschreiben. Sie besteht für jeden Agenten aus einer Liste von Attraktivitätsbewertungen für die vom Agenten wahrgenommenen Geschäfte. Das Verhalten der Einkaufsagenten ist im Beispiel statisch, d.h. es muss durch  $IVM_{EK}$  für jeden Agenten nur ein spezifisches Verhalten und keine Änderungen dieses Verhaltens abgebildet werden. Entsprechend besteht  $IVM_{EK}$  für jeden Agenten aus jeweils 132 Werten (für jedes simulierte Geschäft ein Attraktivitätswert). Zur Ausprägung der  $IVMF_{EK}$  zu  $IVM_{EK_{Valid}}$  muss der Fachexperte für jeden Agenten die genauen Werte der einzelnen Attraktivitätsbewertungen vorgeben. Dies ist aufgrund des Umfangs festzulegender Werte kaum möglich. Die  $IVM_{EK}$ -Werte müssen dann geschätzt werden.

In diesem Fall nimmt Modellierer/Fachexperte zu Beginn eine erste Schätzung der  $IVM$  der Agenten vor. In der Fallstudie der Agenten-basierten Einkaufssimulation würde z.B. als Schätzung festgelegt, dass alle Agenten alle Geschäfte proportional zum Zielumsatz der Geschäfte attraktiv finden sollen. Die Zielumsätze der Geschäfte sind diejenigen Daten, mit denen die simulierten Geschäftsumsätze in *GZF* verglichen werden. D.h. für jedes Geschäft werde der zugehörige Attraktivitätswert in  $IVM$  festgelegt durch:

$$Attrakt_{Geschäft_i} = \frac{Umsatz_{Geschäft_i}}{Max(Umsätze\ aller\ Geschäfte)}$$



### 9.1.5.2 Von der Makro-Ebene auf die Mikro-Ebene:

Beim Übergang von der Makro-Ebene auf die Mikro-Ebene werden für jedes *IVM* zugehörige *IZF* festgelegt. In der Fallstudie der Agenten-basierten Einkaufssimulation wird in den *IZF* berechnet, wie genau die auf Mikro-Ebene simulierte Geschäftsbewertungen eines Agenten für die einzelnen Geschäfte mit den zugehörigen Attraktivitätswerten in *IVM* übereinstimmen.

$$IZF_{Agent_i} = \sum_{j=1}^{\#Geschäfte} |BewFkt(Geschäft_j, Agent_i) - Attrakt_{iGeschäft_j}|$$

### 9.1.5.3 Mikroebene:

Auf der Mikroebene werden nun die individuellen Parameter der einzelnen Agenten auf der Grundlage der *IRB* aus *VK* und der vom Kalibrierungsschritt auf der Makroebene vorgegebenen *IZF* kalibriert. Je nachdem welche anderen Teile des Modells (z.B. Ausschnitt des Umweltmodells) für die Kalibrierung eines Agenten eine Rolle spielen, bilden diese ebenfalls Randbedingungen für die Kalibrierung des Agenten. Im Beispiel der Agenten-basierten Einkaufssimulation werden die Parameter der Bewertungsfunktionen jedes einzelnen Einkaufsagenten so eingestellt, dass seine daraus resultierenden Attraktivitätsbewertungen für die Geschäfte gemäß seiner *IZF* ausreichend passend sind. Zusätzlich dürfen keine seiner individuellen Randbedingungen aus *IRB* bezüglich der Geschäftseigenschaften verletzt werden.

Damit unterscheidet sich das Kalibrierungsproblem auf der Mikro-Ebene vom ursprünglichen *KPEK* durch zwei wesentlichen Punkte:

1. Durch die explizite Festlegung der *IVM* wurde zusätzliches Wissen in das Kalibrierungsproblem eingebracht, das zuvor nicht einbezogen werden konnte, weil die Ebene idealen Agentenverhaltens nicht im Modell betrachtet wurde. Unter der Annahme, dass die *IVM* korrekt sind, ist die Erwartung, dass hierdurch die Qualität des Ergebnisses erhöht und die Menge ähnlicher Mikro-Parametereinstellungen mit gleichem Makro-Gesamtverhalten reduziert werden kann. Der Grund hierfür ist, dass durch die zusätzlichen *IZF* die Struktur des Parametersuchraums, d.h. mehr invalide Parametereinstellungen direkt ausgeschlossen werden können.
2. Durch die Identifikation der einzelnen Einkaufsagenten als Teilmodelle deren Verhalten durch individuelle *IVM* beschrieben werden kann und durch die Vorgabe von *IZF* für diese Teilmodelle kann in diesem Beispiel das Kalibrierungsproblem in unabhängige Teilkalibrierungsprobleme aufgeteilt werden. Auf der Makro-Ebene sind die Gesamtumsätze der Geschäfte von den Entscheidungen aller simulierten Einkaufsagenten abhängig. Durch die Festlegung der *IVM* wurde diese Abhängigkeit aufgelöst. Der Grund hierfür ist, dass so für jeden Agenten einzeln festgelegt wurde, wie er sich Verhalten muss, damit insgesamt das erwünschte Simulationsverhalten entsteht. Die Teilkalibrierungsprobleme auf der Mikro-Ebene können nun unabhängig

voneinander gelöst werden. D.h. jedes Mikro-Agenten-Modell wird für sich kalibriert und die einzelnen Teilkalibrierungen können parallelisiert werden.

#### 9.1.5.4 Iteration:

Ist die Kalibrierung auf der Mikroebene beendet, wird das Gesamtmodell simuliert und auf der Makroebene das Gesamtverhalten anhand der ursprünglichen *VK* bewertet. Waren die Modellstrukturen, die *VK* und die *IVM* korrekt, so sollte das Gesamtverhalten nun ebenfalls valide sein. Ist dies nicht der Fall, weil die *IVM* nur geschätzt werden konnten, so muss unter der Annahme korrekter Modellstrukturen und *VK* geprüft werden, wie die *IVM* angepasst werden müssen. Hierzu muss das Simulationsverhalten analysiert und dann eine geeignete Anpassung der *IVM* vorgenommen werden. Die neuen *IVM* werden dann wieder der Kalibrierung auf der Mikroebene vorgegeben.

In Beispiel der Agenten-basierten Einkaufssimulation wird nach der Kalibrierung auf der Mikroebene geprüft, wie gut die Umsätze der Geschäfte reproduziert werden. Ist die Reproduktionsgenauigkeit zu schlecht, müssen in „geeigneter Weise“ Änderungen an den *IVM* Vorgaben für die Einkaufsagenten vorgenommen werden, d.h. die Zielattraktivitätswerte der Geschäfte für die einzelnen Agenten müssen modifiziert werden.

Die Änderung der *IVM* bleibt hier als „magischer“ Schritt offen und muss vom Modellierer/Fachexperten wie die Ausgangswerte geschätzt werden. In Kapitel 9.2 wird beschrieben, wie die Anpassung der *IVM* als Teil des Kalibrierungsprozesses durchgeführt werden kann.

Im Evaluationsteil wird die Makro-Mikro-Kalibrierung an der Fallstudie der Agenten-basierten Einkaufssimulation durchgeführt (siehe Kapitel 15).

## 9.2 Reverse-Engineering-Verfeinerung

Einen entscheidender Punkt blieb im Basis-Makro-Mikro-Verfahren offen, nämlich die Festlegung der *IVM* im Makro-Schritt des Verfahrens.

Hierzu wird in diesem Abschnitt eine Reverse-Engineering-Verfeinerung des Makro-Mikro-Verfahrens vorgestellt. Das Makro-Mikro-Verfahren wird so verfeinert, dass vom Modellierer oder Fachexperten nicht mehr gefordert wird, dass er konkrete *IVM* vorgeben muss. Stattdessen wird nur noch gefordert, dass er Strukturbeschreibungen *IVMF* für die relevanten *IVM* vorgeben kann. Durch die Reverse-Engineering-Verfeinerung wird die Ermittlung der *IVM* aus den *IVMF* ermöglicht. Das grundlegende Prinzip des Reverse-Engineering-Ansatz wurde von mir in [36] und [38] veröffentlicht.

### 9.2.1 Reverse-Engineering Verfeinerung für das Basis-Makro-Mikro-Verfahren

In diesem Abschnitt wird beschrieben, wie das Basis-Makro-Mikro-Verfahrens durch Reverse-Engineering so verfeinert werden kann, dass die Ermittlung von *IVM* und damit der *IZF* für die Kalibrierung auf der Mikro-Ebene ohne vollständige Vorgabe der *IVM* durch den Modellierer oder Fachexperten ermöglicht wird.

### 9.2.1.1 Reverse-Engineering von *IVM*:

Das Basis-Makro-Mikro-Verfahren wird nun so verfeinert, dass vom Modellierer bei der Kalibrierung nicht mehr gefordert wird, dass er vollständige *IVM* aggregierten Teil-Mikromodellverhaltens beschreiben muss. Stattdessen wird lediglich gefordert, dass er Strukturen *IVMF* zur Beschreibung aggregierten Teil-Mikromodellverhaltens spezifizieren muss. Dies bedeutet, dass der Modellierer nicht mehr genau wissen muss wie valides aggregiertes Mikroverhalten für Agenten und Teilmodelle auf der Mikro-Ebene aussieht. Er muss nur noch Modellstrukturen spezifizieren, auf deren Basis valides aggregiertes Verhalten ermittelt werden kann.

Im Beispiel der Fallstudie müssten die *IVM* bei vollständiger Vorgabe für jeden Einkaufsagenten genau abbilden wie er sein Geld in den simulierten Supermärkten ausgeben muss. *IVMF* zur strukturellen Beschreibung dieser *IVM* bestehen nur noch aus Datenstrukturen für jeden Agenten mit denen die *Geld* → *Geschäft* Abbildung beschrieben werden kann und Randbedingungen die bei der „Geldverteilung“ erfüllt sein müssen, aber nicht mehr aus den exakten Werten. D.h. der Modellierung benötigt „nur“ noch eine Idee der Attraktivitätsfunktion zu haben und muss nicht mehr die konkreten Werte vorgeben.

Damit das Makro-Mikro-Verfahren auch ohne vollständige *IVM*-Vorgabe durchgeführt werden kann, wird ein neuer Kalibrierungsschritt zur Ausprägung der *IVMF* zu *IVM* durchgeführt. Hierzu wird ein zusätzlicher Schritt in den Kalibrierungsprozess eingeführt, der als RE-Schritt (Reverse-Engineering-Schritt) bezeichnet wird. Der RE-Schritt ermittelt ideales Teil-Modellverhalten in Form von *IVM*. Im RE-Schritt wird das Modell nur auf der Basis der *IVM* und nicht durch die konkreten Mikro-Modelle der Agenten simuliert. Im neuen RE-Schritt wird das konkrete Mikromodellverhalten nicht betrachtet. In diesem Schritt wird nur geklärt, ob die vorgegebenen *IVMF* so zu *IVM* ausgeprägt werden können, dass:

1. Auf Basis der *IVM* valides Gesamtverhalten der Simulation bezüglich der Gesamt-*VK* entsteht.
2. Die *IVMF* so zu *IVM* ausgeprägt werden können, dass das aggregierte Teilmodellverhalten auf Makro-Ebene valide ist.

Der zusätzliche Kalibrierungsschritt umfasst also die Kalibrierung der *IVMF* bezüglich der *VK* ohne Betrachtung der Mikro-Ebene. Die *IVM* werden als eine Art Regressionsmodelle validen Teil-Mikromodellverhaltens auf der Grundlage des erwünschten Gesamtverhaltens ermittelt.

Ist der RE-Kalibrierungsschritt abgeschlossen - wurden die *IVMF* also zu *IVM* ausgeprägt - dann werden die *IVM* wie im Basis-Makro-Mikro-Verfahren zur Kalibrierung der Parameter auf der Mikroebene eingesetzt. Abbildung 9.3 veranschaulicht den Einsatz des Reverse-Engineering.

#### Ablauf:

Zunächst prüft der Modellierer oder Fachexperte im Makro-Schritt das aktuelle Gesamtverhalten des zu kalibrierenden Mikromodells. Auf Basis der Analyse aus Abschnitt 8.1 spezifiziert er durch welche Modellstrukturen valides aggregiertes Teilmodellverhalten

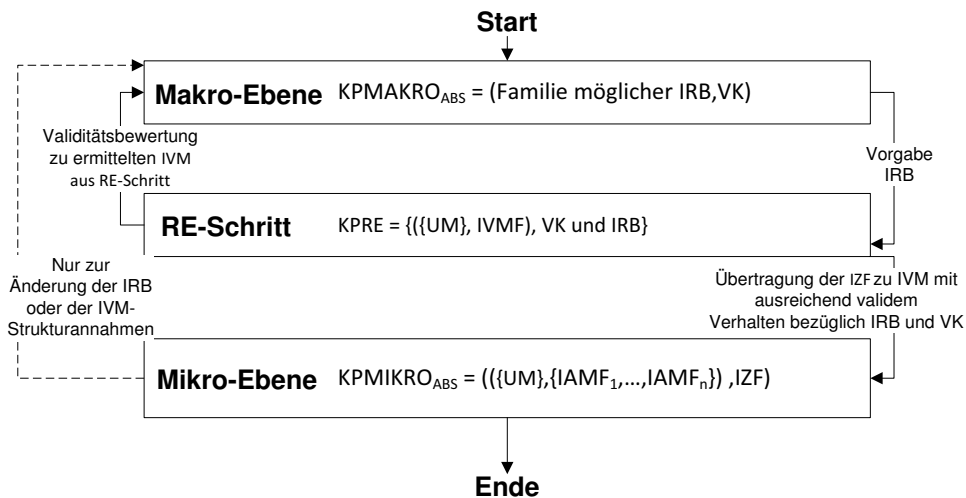


Abbildung 9.3: Makro-Mikro-Verfahren mit Reverse-Engineering.

beschrieben werden kann. Dies geschieht in Form der *IVMF* und weiterer Randbedingungen für ihre Festlegung. Im RE-Schritt wird nun lediglich die Makro-Ebene betrachtet. Die *IVMF* müssen so zu *IVM* ausgeprägt werden, dass das Simulationsverhalten auf Makro-Ebene valide ist. Dies geschieht durch Kalibrierung der *IVMF* auf Basis der Makro-*VK* und der zusätzlichen Randbedingungen an das Simulationsverhalten beim Reverse-Engineering. D.h. bei der Ermittlung der *IVM* wird das Modell nicht auf der Mikro-Ebene, sondern auf der Ebene der *IVM* simuliert. Als Ergebnis erhält man ein „Zwischen“-Modell bei dem valides Gesamtverhalten aus Beschreibungen aggregierten Teilmodellverhalten entsteht. Dabei ersetzt die Verhaltensbeschreibung der *IVM* das sonst auf der Mikro-Ebene erzeugte Verhalten.

Ist der RE-Kalibrierungsschritt abgeschlossen dann stehen die *IVM* für den Mikro-Kalibrierungsschritt des Basis-Makro-Mikro-Verfahrens zu Verfügung. Wie im Basis-Verfahren werden die *IZF* für die Mikroebenen-Kalibrierung dadurch ermittelt, dass gemessen wird, wie exakt das Mikro-Ebenen-Verhalten dem Verhalten in den zugehörigen *IVM* beschriebenen Verhalten entspricht.

Der Mikro-Kalibrierungsschritt wird nun wie im Basisverfahren durchgeführt. Das resultierende Gesamtverhalten wird auf der Makro-Ebene bewertet und gegebenenfalls eine neue Iteration auf der Makro-Ebene mit Reverse-Engineering angestoßen. Es werden jedoch nicht die *IVM* wie in Kapitel 9.1.3 „von Hand“ adaptiert. Stattdessen wird geprüft, ob:

1. Die *IVMF* valide Strukturbeschreibungen sind oder modifiziert werden müssen.
2. Die zusätzlichen Randbedingungen für die Kalibrierung im RE-Schritt modifiziert, entfernt oder erweitert werden müssen.

Dies bedeutet, dass vom Modellierer oder Fachexperten im Makro-Schritt keine direkten Änderungen an den *IVM* mehr gefordert werden. Bezüglich der *IVMF* und der Randbedingungen für die RE-Kalibrierung ist aber immer noch die Analyse und Modifikation durch den Fachexperten notwendig, wenn das Makro-Mikro-Verfahren nicht zum erwünschten Gesamtergebnis führt.

Eine Möglichkeit diesen Schritt auf der Makro-Ebene weiter zu automatisieren besteht darin, den Fachexperten bereits zu Beginn der Kalibrierung mehrere Kandidatenstrukturen für die *IVMF* spezifizieren zu lassen. Gibt es eine Reihe solcher Strukturen können diese während der Kalibrierung systematisch ausgetestet werden, indem in jedem Makro-Schritt eine andere *IVMF*-Kandidatenstruktur für die nächste Kalibrierungsiteration ausgewählt wird. Die Auswahl der nächsten Kandidatenstruktur kann dabei entlang einer Vorgabenliste oder mittels Techniken zum automatischen Modellentwurf erfolgen (siehe Kapitel 3.3).

### 9.2.1.2 Anwendungsbeispiel:

Die Lösung des Kalibrierungsproblems für die Fallstudie der Einkaufssimulation mit dem durch Reverse-Engineering verfeinerten Makro-Mikro-Verfahren sieht für das Beispiel aus Abschnitt 9.1.5 wie folgt aus (siehe Abbildung 9.4):

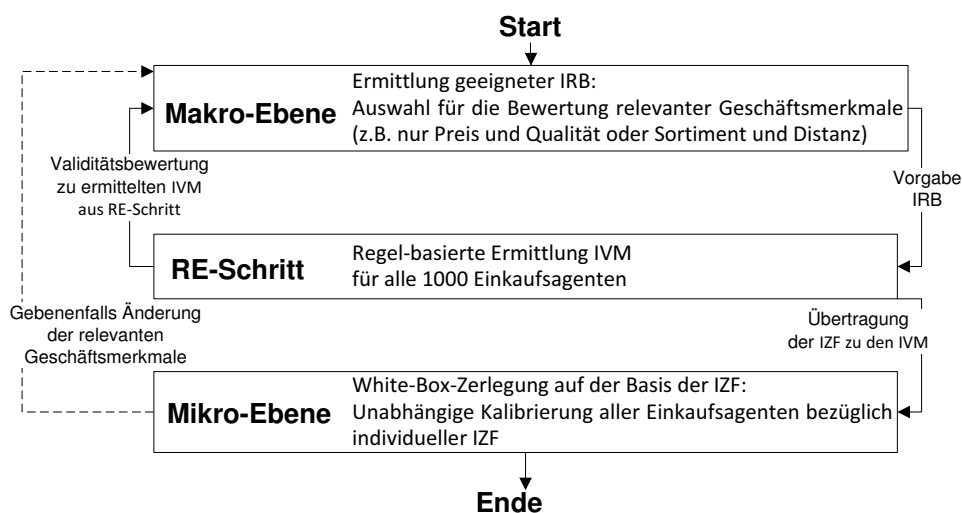


Abbildung 9.4: Schematische Kalibrierung der Agenten-basierten Einkaufssimulation mit dem Makro-Mikro-Verfahren und Reverse-Engineering.

#### 1. Start auf Makro-Ebene:

Beim Start auf der Makro-Ebene muss der Modellierer nicht mehr wie im Basis-Verfahren für jeden Einkaufsagenten in einem *IVM* vorgeben, wie attraktiv er welches Geschäft finden muss. Stattdessen muss er *IVMF* spezifizieren. Diese bestehen im Beispiel der Einkaufssimulation aus einer Liste für jeden Einkaufsagenten in der für jedes wahrgenommene Geschäft gespeichert werden kann, wie attraktiv er dieses Geschäft finden soll. Weiter muss der Modellierer Randbedingungen *IRB* vorgeben auf deren Grundlage im RE-Schritt die Werte der *IVMF* eingestellt werden sollen. Eine zentrale Randbedingung für die Fallstudie ist, dass ein Agent ein Geschäft nur dann als attraktiver als ein anderes empfinden darf, wenn dieses in keinem Geschäftsmerkmal (insbesondere der individuellen Entfernung des Agenten zu diesem Geschäft) schlechter ist als das andere ist. Für Geschäfte, die in Teilmengen der Merkmale besser und in anderen Teilen schlechter sind als ein anderes werden keine Vorgaben bezüglich der Attraktivität dieses Geschäfts im Vergleich zum anderen gemacht.

## 2. Kalibrierung im RE-Schritt:

Ein *IVMF* für einen Agenten besteht aus Attraktivitätswerten für jedes vom Agenten wahrgenommene Geschäft (bei 132 Geschäften also jeweils 132 Werte). Bei 1000 simulierten Agenten sind dies 132.000 Werte. Im Vergleich zum Ausgangsproblem ist die Größe des Parametersuchraums hierdurch zwar signifikant größer, die Kalibrierung im RE-Schritt kann jedoch mit einem wissensbasierten Verfahren sehr einfach durchgeführt werden: Zu Beginn werden die 132 Werte für alle Agenten homogen entsprechend der Zielumsätze der Geschäfte gewählt. Dann werden die Werte für jeden Agenten so angepasst werden, dass für diesen Agenten keine *IRB* verletzt werden. Nun wird ein erster Simulationslauf auf Makro-Ebene durchgeführt. Die Werte der *IVMF* werden während des folgenden Kalibrierungsteilschrittes so angepasst, dass sie die entsprechenden Geschäfte relativ zu deren Verschätzungsgrad mehr oder weniger „attraktiv finden“. Im zweiten Schritt werden die Werte der *IVM* jedes Agenten wieder so angepasst, dass für diesen Agenten keine *IRB* verletzt werden. Dieser Prozess wird so lange iteriert, bis die *IVM*-Werte konvergieren. Nachdem die Werte konvergiert sind hat man die *IVM* für die Kalibrierung auf der Mikro-Ebene ermittelt. Da im RE-Schritt die *IVMF* für alle Agenten gleichzeitig kalibriert werden, sind die *IVM* des Kalibrierungsergebnisses bereits so eingestellt, dass Abhängigkeiten zwischen den Agenten (aufgrund ihrer individuellen Entfernung zu den Geschäften) abbilden.

## 3. White-Box-Kalibrierung auf der Mikro-Ebene:

Die auf der Makro-Ebene ermittelten *IVM* bilden bereits die Abhängigkeiten zwischen den Agenten aufgrund ihrer unterschiedlichen räumlichen Lage ab, da die *IVM* im RE-Schritt gleichzeitig für alle Agenten ermittelt wurden. Da die *IVM* die Abhängigkeiten enthalten müssen diese auf der Mikro-Ebene nicht mehr explizit bei der Kalibrierung ermittelt werden (siehe hierzu auch Evaluationskapitel 15). Durch die *IVM* wird eine White-Box-Zerlegung des Kalibrierungsproblems auf der Mikroebene ermöglicht. Die Wahrnehmungsfunktionen jedes einzelnen Einkaufsagenten können unabhängig von den übrigen Agenten kalibriert werden. Dabei wird jedes Agentenmodell auf der Basis von *IZF* kalibriert. Die *IZF* messen den Unterschied zwischen den berechneten Attraktivitätswerten auf der Mikro-Ebene und den Vorgaben aus den *IVM*.

## 4. Iterationsschritt:

Eine Modifizierbarkeit der Randbedingungen für die Kalibrierung im RE-Schritt ist durch die Auswahl einer (Teil)menge an relevanten Geschäftsmerkmalen, bezüglich derer die Randbedingungen durchgesetzt werden, gegeben. Der Iterationsschritt für das Makro-Mikro-Verfahren wird nun als systematisches „Durchtesten“ verschiedener Kombinationen von Geschäftsmerkmalen umgesetzt, aufgrund derer die Agenten die Geschäfte bewerten. Damit kann in der Fallstudie der Einkaufssimulation das Makro-Mikro-Verfahren nach Spezifikation der Geschäftsmerkmal-kombinationen und Modellierung der *IVMF* automatisch durchgeführt werden. Das Verfahren endet, wenn für alle Kombinationen die Kalibrierung abgeschlossen ist.

Eine detaillierte Diskussion des hier beschriebenen Beispiels mit konkreten Zahlen erfolgt im Evaluationsteil dieser Arbeit, in Abschnitt 15.

### 9.2.2 Zusammenfassung

In diesem Kapitel wurde die Reverse-Engineering-Verfeinerung für das Makro-Mikro-Verfahren beschrieben. Bei der Integration von Reverse-Engineering und Makro-Mikro-Verfahren wird ein zusätzlicher Kalibrierungsschritt für ideales Teilmodellverhalten in den Makro-Mikro-Kalibrierungsprozess eingeführt. In diesem RE-Schritt werden zu auf der Makro-Ebene vorgegeben *IRB* theoretisch passende *IVM* ermittelt, ohne dass bei deren Ermittlung das konkrete Agentenverhalten auf der Mikro-Ebene betrachtet werden muss. Die ermittelten *IVM* werden dann durch zugehörige *IZF* zur Kalibrierung auf der Mikro-Ebene eingesetzt.

## 9.3 Robuste Kalibrierung auf der Mikro-Ebene

In diesem Abschnitt wird die Erweiterung des Makro-Mikro-Verfahrens mit Reverse-Engineering-Verfeinerung um eine robuste Kalibrierung auf der Mikro-Ebene beschrieben. Die Erweiterung nutzt die im RE-Schritt ermittelten *IVM*, um fehlerhafte Teilmodelle auf der Mikro-Ebene identifizieren und ausfiltern zu können.

Unter robuster Kalibrierung wird ein Verfahren zur Kalibrierung von ABS verstanden, das die Kalibrierung unter der Randbedingung durchführt, dass ein (unbekannter) Teil der zur Validierung und Modellierung verwendeten Daten deutlich stärker verfälscht ist als die restlichen Daten (siehe auch Abschnitt 4.1.3.4).

Sind die für die Kalibrierung eines Simulationsmodells verwendeten Daten über ein Originalsystem lücken- oder fehlerhaft, ist es möglich, dass trotz Erstellung geeigneter Modellstrukturen, fehlerfreier Implementation und der Verwendung eines effektiven Kalibrierungsverfahrens kein valides Simulationsverhalten erzeugt werden kann. Eine Ursache ist, dass die auf Daten basierenden *VK* aufgrund von Datenfehlern gar kein valides Simulationsverhalten beschreiben. Wird die Kalibrierung rein auf der Basis solcher *VK* durchgeführt, so wird sie durch die fehlerhaften *VK* so „gesteuert“, dass das resultierende Simulationsverhalten zwar optimal bezüglich *VK*, aber nicht valide bezüglich des Verhaltens des simulierten realen Systems ist.

Das Ziel der robusten Kalibrierung besteht darin, während des Kalibrierungsprozesses die (un)verfälschten Teile der Daten und Modellstrukturen zu identifizieren und die Kalibrierung nur auf der Basis der korrekten (bzw. nicht zu stark verfälschten) Daten durchzuführen. Im Verlauf des Kapitel wird eine Variante des RANSAC-Verfahrens [55] entwickelt, die zur robusten Kalibrierung von ABS angepasst wurde. Es wird beschrieben, wie das Verfahren in Kombination mit der Reverse-Engineering-Verfeinerung im Makro-Mikro-Verfahren genutzt werden kann.

Im ersten Teil des Abschnitts wird das Verfahren zur robusten Kalibrierung generisch beschrieben. Im zweiten Teil werden Anforderungen genannt, die erfüllt sein müssen, damit die robuste Kalibrierung im Rahmen des Makro-Mikro-Verfahrens eingesetzt werden kann. Im dritten Teil wird eine Beispielanwendung betrachtet. Nach der Klärung und dem Beispiel wird im vierten Teil der praktische Einsatz anhand der Fallstudie der Agenten-basierten Einkaufssimulation demonstriert.

### 9.3.1 Robuste Kalibrierung von ABS

In diesem Abschnitt wird die in dieser Arbeit entwickelte Variante des RANSAC-Verfahrens [55] zur robusten Kalibrierung von ABS beschrieben. Das Verfahren verwendet die im Makro-Mikro-Verfahren ermittelten  $IVM$ , um automatisch fehlerhafte Modellteile auf Mikro-Ebene zu identifizieren und in der Simulation durch ihre idealen aggregierten  $IVM$ -Pendants zu ersetzen. Das Original-RANSAC-Verfahren kann nicht ohne Weiteres zur Kalibrierung von ABS eingesetzt werden. Das RANSAC-Verfahren ist eigentlich für Kalibrierungen mit überbestimmten Gleichungssystemen entworfen, also für Systeme, in denen mehr Validierungsdaten als benötigt zur Verfügung stehen. Die Idee des RANSAC-Verfahrens zur Identifikation und Filterung von fehlerhaften Datenwerten besteht dann darin Teilmengen der verwendeten Daten aus dem Kalibrierungsprozess zu entfernen und z.B. nur minimal benötigte Teilmengen von Daten für die Kalibrierung zu verwenden, so dass eine große Chance besteht, dass diese Teilmenge keine fehlerhaften Daten enthält.

#### 9.3.1.1 $KP^{Robust}$ als RANSAC-Problem

Überträgt man  $KP^{Robust}$  aus Abschnitt 4.1.3.4 in ein Problem für den RANSAC-Ansatz, so entsprechen die  $IVK$  (Mikro-Validierungskriterien aus der Menge der Gesamt- $VK$ , d.h. keine generierten idealisierten Validierungskriterien) aus  $KP^{Robust}$  den in RANSAC verwendeten Datenpunkten. Es wird nun nach einem kalibrierten Simulationsmodell gesucht, das bezüglich aller korrekten  $IVK$  valide ist und nicht durch die fehlerhaften  $IVK$  verzerrt wird. Eine robuste Kalibrierung muss solche  $IVK$  innerhalb der Menge der  $UIVK$  (unsichere  $IVK$ ) identifizieren und die Kalibrierung unabhängig von diesen durchführen. Das Problem in der Anwendung des RANSAC-Verfahrens auf  $KP^{Robust}$  besteht darin, dass alle  $IVK$  zu bestimmten  $TM$  des Modells gehören und für deren Kalibrierung benötigt werden. Werden Daten- $IVK$  zur Kalibrierung auf der Mikro-Ebene entfernt, müssen theoretisch auch die zugehörigen  $TM$  entfernt werden. Dies ist in ABS aber nicht möglich, da die Gefahr besteht, dass das Gesamtverhalten der Simulation hierdurch verzerrt wird.

Als Folge ist es nicht einfach möglich, nur mit Teilmengen der  $IVK$  zu kalibrieren, was den allgemeinen RANSAC-Ansatz für ABS unmöglich macht.

#### 9.3.1.2 Idee der robusten Kalibrierung auf der Mikro-Ebene

Die Idee der robusten Kalibrierung für ABS beruht darauf, dass ABS auf verschiedenen Beobachtungsebenen analysiert, validiert und damit auch simuliert werden können. Im Makro-Mikro-Verfahren wurde explizit die Makro-Ebene der  $IVM$  als zusätzliche Beobachtungsebene aggregierten Mikro-Verhaltens behandelt. Das Makro-Mikro-Verfahren kann auch mehrstufig eingesetzt werden, indem das aggregierte Verhalten in zusätzliche Beobachtungsebenen immer weiter aggregiert wird. Als Ergebnis einer solchen mehrstufigen Anwendung hat man auf verschiedenen Beobachtungsebenen unterschiedliche  $IVM$ , welche valides Verhalten auf der Mikro-Ebene und Beobachtungsebenen in immer stärker aggregierten Varianten beschreiben. Stärker aggregierte  $IVM$  überdecken dabei zugehörige Modelle auf detaillierteren Ebenen, da sie vollständig das jeweilige valide Verhalten dieser Ebenen beschreiben.

Im Makro-Mikro-Verfahren werden zuerst geeignete  $IVM$  ermittelt und dann die zugehörigen Teilmodelle auf der Mikro-Ebene entsprechend kalibriert. Das Wechseln zwi-



schen Makro- und Mikro-Ebene entspricht einem vollständigen Ersetzen der Mikro-Teilmodelle durch ihre *IVM*-Pendants. Die Idee der entwickelten RANSAC-Variante besteht nun darin diese Ersetzung nicht für alle Mikro-Teilmodelle, sondern nur für ausgewählte Teilmengen an Teilmodellen durchzuführen. Teilmodelle auf der Mikro-Ebene, deren Daten-*IVK* im RANSAC-Vorgehen aus dem Kalibrierungsprozess entfernt werden sollen, werden nicht ganz entfernt, sondern durch ihre im Makro-Mikro-Verfahren ermittelten zugehörigen *IVM* ersetzt. Das Modell enthält dann verschiedene Typen von „Teilmodellen“ auf verschiedenen Aggregierungsebenen. Zum einen enthält es nicht entfernte Teilmodelle, die auf der Mikro-Ebene simuliert werden. Für entfernte Teilmodelle enthält es die zu den betroffenen zugehörigen *IVM*. Durch Simulation der (idealen) *IVM* anstelle der betroffenen Mikro-Teilmodelle fließt deren aggregiertes Verhalten, immer noch ausreichend valide in das Gesamtverhalten der Simulation ein, ohne dass das konkrete Mikro-Verhalten der Agenten auf der Basis der als problematisch eingestuften *VK* kalibriert werden muss.

Abbildung 9.5 veranschaulicht das Prinzip schematisch an einem Beispiel. Im Beispiel sei Validierungswissen auf vier Beobachtungsebenen verfügbar. Auf der Makro-Ebene, einer Zwischenebene bezüglich aggregierten Verhaltens mehrerer Agenten, der Ebene des aggregierten individuellen Agentenverhaltens und der Mikro-Ebene, auf der das konkrete Agentenverhalten simuliert wird. Werden bestimmte *AM* auf der Mikro-Ebene während einer robusten Kalibrierung aus dem Modell „entfernt“, werden sie durch Modelle ihres aggregierten Verhaltens ersetzt. Im Beispiel werden, jeweils die grau markierten Kästen im Modell simuliert. *AM1*, *AM2*, *AM7* und *AM8* bilden die minimale Teilmenge für die Kalibrierung auf der Mikro-Ebene. *AM3* wird nicht kalibriert oder konkret simuliert. Stattdessen wird das zugehörige ideale *IVM* simuliert, welches den Einfluss validen aggregierten Verhaltens von *AM3* auf die Simulation abbildet. *AM4* bis *AM6* werden ebenfalls nicht kalibriert. Sie werden durch ein zugehöriges *IVM* ersetzt, welches den kombinierte aggregierte Einfluss des Verhaltens dieser Agenten auf das Simulationsverhalten abbildet.

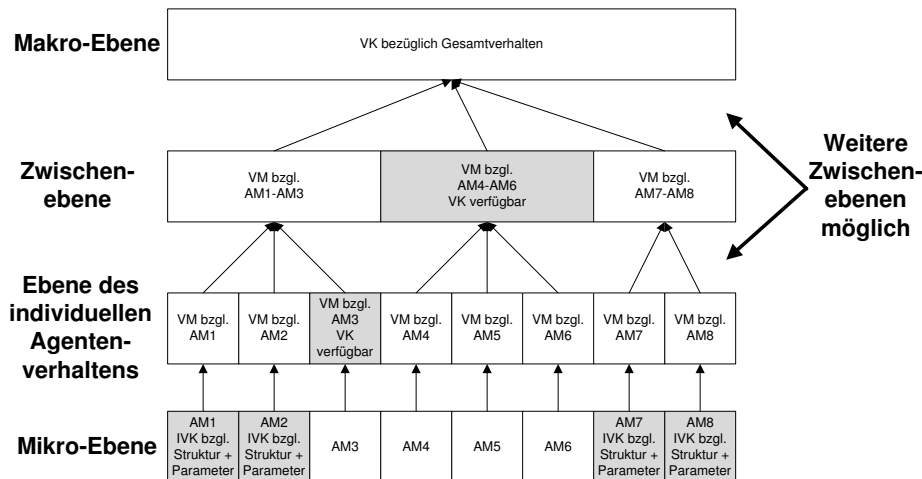


Abbildung 9.5: Beispiel möglicher Modellaggregationen auf Basis von *IVM* auf unterschiedlichen Beobachtungsebenen.

### 9.3.1.3 Umsetzung der robusten Kalibrierung auf der Mikro-Ebene

In diesem Abschnitt wird beschrieben, wie das Makro-Mikro-Verfahren um eine robuste Suche zur Kalibrierung auf der Mikro-Ebene erweitert werden kann. Bei der robusten Suche im Makro-Mikro-Verfahren wird nach Mikromodellteilen gesucht, die aufgrund fehlerhafter Daten nicht so kalibriert werden können, dass das erwünschte Gesamtverhalten entsteht. Diese Mikromodellteile werden „ausgefiltert“. „Ausgefiltert“ bedeutet in diesem Fall, dass die betroffenen Modellteile auf der Mikro-Ebene aus dem Modell entfernt und durch ihre zugehörigen *IVM* ersetzt werden. Ein Simulation des Mikro-Gesamtmodells nach der Filterung von Teilmodellen besteht dann aus einer Kombination von Modellteilen, die weiterhin auf der Mikro-Ebene simuliert werden, und anderen Modellteilen, die auf der Ebene des aggregierten validen Teilmodellverhaltens simuliert werden. Auf diese Weise können fehlerhafte Mikromodellteile entfernt werden, ohne dass der benötigte Einfluss dieser Teile auf das Gesamtverhalten der Simulation verloren geht. Die übrigen Modellteile werden nun wie bisher im Makro-Mikro-Verfahren kalibriert. Das Gesamtverhalten wird auf Basis der kalibrierten Teilmodelle und der *IVM* der ausgefilterten Teilmodelle bewertet. Als Kalibrierungsergebnis wird das Modell gewählt, welches eine möglichst geringe Anzahl ausgefilterter Teilmodelle hat und ein möglichst valides Gesamtverhalten aufweist.

Für die Anwendbarkeit der beschriebene robuste Suche gelten die folgenden Randbedingungen:

1. Es können nur Mikro-Teilmodelle ausgefiltert werden, zu denen zugehörige *IVM* validen aggregierten Verhaltens verfügbar sind. D.h. ein Teilmodell kann nur ausgefiltert werden, wenn auf aggregierter Ebene bereits ein zugehöriges valides Verhaltensmodell verfügbar ist.
2. Ein Mikro-Teilmodell kann nur ausgefiltert werden, wenn es im Mikro-Gesamtmodell einfach durch sein zugehöriges *IVM* ersetzt werden kann, während des Rest des Modells weiterhin auf Mikro-Ebene simuliert wird. D.h. nach der Ersetzung muss es möglich sein das gefilterte Gesamtmodell zu simulieren und auf Makro-Ebene zu validieren als sei es noch das Originalmodell. Damit die Ersetzung möglich ist müssen ein Teilmodell auf der Mikro-Ebene und sein zugehöriges *IVM* die definierte, identische Schnittstellen zum Rest der Simulation haben. Bei der Ersetzung wird das Teilmodell auf der Mikro-Ebene an diesen Schnittstellen entfernt und dafür das *IVM* eingesetzt.

Die beschriebene robuste Suche kann in das Makro-Mikro-Verfahren integriert werden. Im Makro-Mikro-Verfahren wird auf der Makro-Ebene das Modell nur auf *IVM*-Basis simuliert. Diese *IVM* bilden die Voraussetzung für Ersetzungen zugehöriger Teilmodelle bei robuster Kalibrierung auf der Mikro-Ebene.

Der Ablauf des Verfahrens wird in Abbildung 9.6 dargestellt. Zunächst werden, wie im Makro-Mikro-Verfahren mit RE-Verfeinerung *IVM* für die einzelnen Teilmikromodelle ermittelt oder vorgegeben. Das Verhalten der *IVM* muss als sicher valide bewertet werden, da die *IVM* sonst nicht zum Ausfiltern unsicherer Teilmodelle auf der Mikro-Ebene eingesetzt werden können. Im zweiten Schritt wird die eigentliche robuste Kalibrierung durchgeführt. In den einzelnen Iterationsschritten wird jeweils eine minimale Teilmenge an Teilmodellen auf einer Aggregierungsebene (z.B. der Mikro-Ebene) gewählt, die simuliert und deren zugehörige Parameterwerte kalibriert werden. Die übrigen Teilmodelle werden nicht auf dieser Aggregierungsebene simuliert und kalibriert. Stattdessen wird deren Einfluss auf das

Simulationsverhalten wann immer er benötigt wird durch die Simulation des zugehörigen *IVM* ermittelt. Dies ist z.B. der Fall bei der Simulation des Gesamtmodells oder wenn der Einfluss eines Teilmodells als Randbedingung für die Simulation eines anderen Teilmodells benötigt wird.

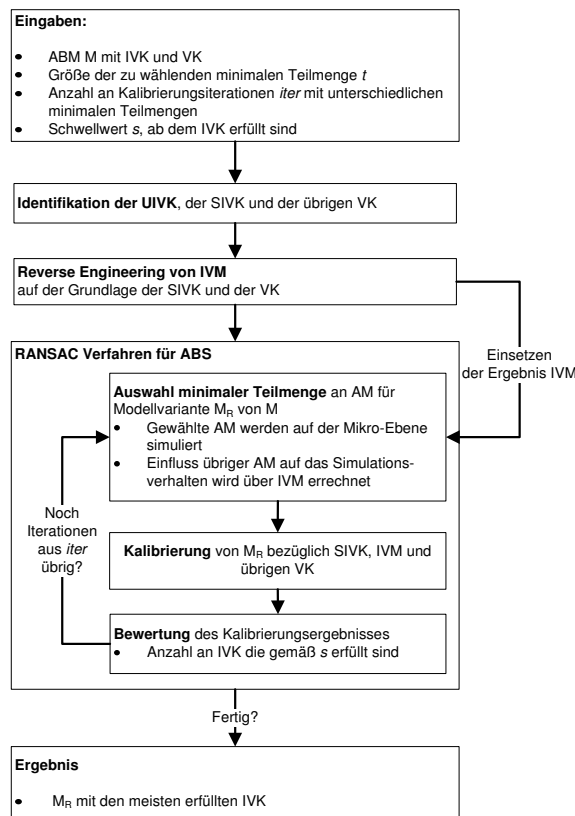


Abbildung 9.6: Robuste Kalibrierung von ABS.

### 9.3.2 Integration in das Makro-Mikro-Verfahren

In diesem Abschnitt wird beschrieben, wie die robuste Suche in das bisherige Makro-Mikro-Verfahren integriert werden kann. Im bisherigen Makro-Mikro-Verfahren diente die Ermittlung der *IVM* als Zwischenschritt zur Bestimmung benötigt *IZF* für die Kalibrierung von Teilmodellen auf der Mikro-Ebene. Auf der Makro-Ebene bildeten die *IVM* ein eigenes Modell aggregierten Verhaltens, eine konkrete Verbindung dieser Modellebene mit der Mikro-Ebene außer über die *IZF* war aber nicht zwingend notwendig. Die *IVM* konnten „für sich“ in einem eigenen aggregierten Modell simuliert und kalibriert werden, während die zugehörigen Teilmodelle auf der Mikro-Ebene im Originalmodell simuliert und kalibriert wurden.

Diese getrennte Behandlung der *IVM* und der zugehörigen Teilmodelle auf der Mikro-Ebene ist bei Verwendung der robusten Kalibrierung nicht mehr möglich. Damit während der robusten Suche dynamisch Teilmodelle auf der Mikro-Ebene durch ihre *IVM* ersetzt werden können, müssen die *IVM* als Modellteil an den Schnittstellen des zugehörigen Teil-

modells in das Originalmodell integriert werden. Dann muss ermöglicht werden, dass zwischen der Verwendung der originalen Teilmodelle auf der Mikro-Ebene und der Verwendung der zugehörigen *IVM* für jedes Teilmodell bei der Simulation des Gesamtmodells einfach „umgeschaltet“ werden kann. Eine mögliche Vorgehensweise hierzu ist:

1. **Erzeugung eines kombinierten Gesamtmodells**, das sowohl das Mikromodell als auch die *IVM* enthält. *IVM* und Original-Mikromodell können dabei parallel im Gesamtmodell repräsentiert sein.
2. **Integration der Modelle**. Für jedes Teilmodell auf der Mikro-Ebene, das durch sein zugehöriges *IVM* ersetzbar sein soll, muss ein Austauschmechanismus in das Gesamtmodell eingebaut werden. Dieser muss dafür sorgen, dass gegebenenfalls nicht das konkrete Teilmodell auf der Mikro-Ebene simuliert wird, sondern dessen aggregierter Verhaltenseinfluss auf das Gesamtverhalten und die anderen Teilmodelle das zugehörigen *IVM* simuliert wird. Wie ein *IVM* mit den übrigen Teilmodellen integriert werden muss hängt davon ab, welche anderen Teilmodelle Verhaltenseinflüsse aus *IVM* bei der Simulation erhalten müssen. Damit zwischen Simulation eines originalen Teilmodells auf der Mikro-Ebene und seinem *IVM* dynamisch gewechselt werden kann, werden Schalter zum Umschalten benötigt. Hierzu können z.B. für die einzelnen Teilmodelle auf der Mikro-Ebene zusätzliche boolesche Schalterparameter eingeführt werden. Ist der Schalterwert „wahr“, wird das originale Teilmodell auf der Mikro-Ebene simuliert. Ist der Schalterwert „falsch“, so wird das Teilmodell auf der Mikro-Ebene nicht simuliert sondern sein zugehöriges *IVM*.
3. **Kalibrierung auf der Makro-Ebene**. Die Validierung des Gesamtverhaltens auf der Makro-Ebene wird durch Bewertung des Gesamtverhaltens durchgeführt, wobei im simulierten Modell ausgefilterte Teilmodelle durch ihre *IVM* ersetzt wurden. Die Auswahl und Validierung der *IVM* verläuft wie für das Makro-Mikro-Verfahren beschrieben.
4. **Kalibrierung im RE-Schritt**. Das Reverse-Engineering verläuft wie für das Makro-Mikro-Verfahren beschrieben.
5. **Kalibrierung auf der Mikro-Ebene**. Die bisherige Kalibrierung des Makro-Mikro-Verfahrens auf der Mikro-Ebene wird nun durch die robuste Kalibrierung erweitert. D.h. die Kalibrierung auf der Mikro-Ebene wird nur für die nicht-ausgefilterten Teilmodell auf der Mikro-Ebene durchgeführt. Benötigte Einflüsse von ausgefilterten Modellteilen werden durch Simulation der zugehörigen *IVM* ermittelt.
6. **Unterstützung im Kalibrierungswerkzeug**. Zur Durchführung des RANSAC-Verfahrens ist es notwendig, dass iterativ zufällig „minimale Teilmengen“ (hier: der zu kalibrierenden Agentenmodelle) gewählt und „ausgefiltert“ werden. Die Filterung entspricht dem Umschalten zwischen originalem Teilmodell auf der Mikro-Ebenen und *IVM*. Damit dies automatisiert geschehen kann, muss das Vorgehen im verwendeten Kalibrierungswerkzeug unterstützt werden. Der für diese Arbeit entwickelte DAVINCI-Kalibrierer (siehe Kapitel 11) bietet eine entsprechende Unterstützung.

Oben wurde beschrieben, dass die *IVM* vor der Anwendung der robusten Suche bereits korrekt ermittelt worden sein müssen. Dies ist notwendig, da bei der robusten Suche fehlerhafte

Teilmodell auf der Mikro-Ebene nicht durch fehlerhafte *IVM* ersetzt werden sollen. Es ist aber auch möglich die robuste Suche in die Iterationen des Makro-Mikro-Verfahrens zu integrieren. In diesem Fall wird die robuste Kalibrierung jeweils mit den aktuellen Kandidatenmodellen der *IVM* durchgeführt, die während der robusten Suche als „ausreichend valide“ betrachtet werden. Dies erhöht den Gesamtaufwand für die Kalibrierung.

Im folgenden Abschnitt wird die praktische Umsetzung anhand eines Beispiels demonstriert.

### 9.3.3 Beispielanwendung: Einkaufssimulation

In Abschnitt 9.2 wurde das Makro-Mikro-Verfahren mit Reverse-Engineering-Verfeinerung am Beispiel der Agenten-basierten Einkaufssimulation erläutert. Das dort beschriebene Beispiel kann leicht zu einem Beispiel für die robuste Kalibrierung erweitert werden. Anstelle einer Kalibrierung auf der Mikro-Ebene, bei der alle Agenten bezüglich ihrer *IZF* kalibriert werden, werden nur Einkaufsagenten aus der minimalen Teilmenge kalibriert. Das Einkaufsverhalten der übrigen Agenten, also ihre Entscheidung welchen Anteil ihres Budgets sie in welchem Geschäft ausgeben, wird nicht als Wahrnehmung und Bewertung der charakteristischen Geschäftseigenschaften simuliert, sondern fließt direkt in Form der Attraktivitätswerte aus den *IVM* in die Simulation ein.

Praktisch geht der Nutzer dabei z.B. wie folgt vor:

1. Festlegung der *IVM* als Vektoren von Attraktivitätswerten. Wie bereits beschrieben repräsentiert jeder Attraktivitätswert eines Vektors wie attraktiv ein Agent *A* ein Geschäft *B* findet. Die Bestimmung der Werte in den *IVM* erfolgt wie zuvor beschrieben.
2. Integration: Eine Möglichkeit zur Integration ist gegeben, wenn jeder simulierte Agent seinen spezifischen *IVM*-Attraktivitätsvektor als zusätzliche Variable erhält. Ein Einkaufsagent kann Geschäftsbewertungen nicht auf der Basis wahrgenommener charakteristischer Geschäftsmerkmale ermitteln, sondern die Bewertungen direkt aus dem jeweiligen *IVM*-Vektor entnehmen. Der Wechsel zwischen echter Attraktivitätsberechnung und Verwendung vorgegebener Werte kann dann durch einen zusätzlichen booleschen Parameter für jeden Agenten gesteuert werden.
3. Durchführung der robusten Kalibrierung: Bei der Durchführung der robusten Kalibrierung nach RANSAC-Schema gibt der Nutzer nun eine Anzahl durchzuführender RANSAC-Iterationen und eine minimale Anzahl an Einkaufsagenten vor, die nicht durch *IVM* ersetzt werden sollen. Die robuste Kalibrierung wählt in jeder Iteration zufällig eine Teilmenge an Agenten aus und setzt deren Schalterparameter auf „verwende Mikroverhalten“. Für die übrigen Einkaufsagenten wird der Schalter auf „verwende Werte aus *IVM*“ gestellt. Nur für die nicht ausgefilterten Agenten werden die Parameterwerte kalibriert. Als Ergebnis erhält man diejenige Teilmenge an Einkaufsagenten mit kalibrierten Parametern für welche das valideste Gesamtverhalten der Simulation erzeugt werden konnte.

### 9.3.4 Zusammenfassung

In diesem Abschnitt wurde das Makro-Mikro-Verfahren um eine robuste Suche zur Kalibrierung von Parameterwerten auf der Mikro-Ebene erweitert:

- Das Verfahren ist eine Variante des in der Bildverarbeitung entwickelten RANSAC-Algorithmus.
- Im Gegensatz zum reinen RANSAC-Verfahren entfernt die robuste Kalibrierung für ABS keine Teilmodelle oder Validierungskriterien aus dem Modell, sondern ersetzt Teilmodelle auf Mikro-Ebene durch ihre zugehörigen *IVM*, für die eine höhere Verhaltensvalidität als den zugehörigen Teilmodell auf der Mikro-Ebenen angenommen wird.
- Das Verfahren zur robusten Kalibrierung von ABS kann einfach in das Makro-Mikro-Verfahren integriert werden und hilft dort, das Problem des Umgangs mit verrauschten Daten zu Kalibrierung und Modellierung auf der Mikro-Ebene zu lösen.

Die Einführung der robusten Suche schließt die Beschreibung des für diese Arbeit entwickelten Makro-Mikro-Verfahrens ab. Im nächsten Kapitel wird das Verfahren und seine Verfeinerungen hinsichtlich Anwendbarkeit, Randbedingungen und Nutzen diskutiert.

## 9.4 Diskussion des Makro-Mikro-Verfahrens

In diesem Abschnitt wird die Anwendbarkeit des Makro-Mikro-Verfahrens und seine Vorteile diskutiert. Die Diskussion wird entlang einer Beispielanwendung des Makro-Mikro-Verfahrens geführt. Es wird beschrieben welche Schritte durchgeführt werden sollen, welche Probleme dabei entstehen können, bzw. welche Voraussetzungen gelten müssen und welche Vorteile dabei im Vergleich zu Black-Box-Kalibrierung erzielt werden können. Als Zusammenfassung wird diskutiert, wann das Makro-Mikro-Verfahren einem Black-Box-Verfahren vorgezogen werden sollte. Eine Vergleich des Makro-Mikro-Verfahrens mit White-Box-Kalibrierungstechniken erfolgt nicht. Das Verfahren hat selbst die Anwendung von White-Box-Zerlegungstechniken als ein Ziel und kann in bestimmten Schritten die Anwendung von White-Box-Wissensbasierter-Kalibrierung erfordern. Daher wird statt einer Abgrenzung darauf hingewiesen, wenn während der Anwendung des Makro-Mikro-Verfahrens White-Box-Kalibrierungstechniken eingesetzt werden können.

### 9.4.1 Durchführung des Makro-Schrittes im Makro-Mikro-Verfahren

Der zentrale Schritt bei der Durchführung des Makro-Mikro-Verfahrens besteht darin, den Kalibrierungsschritt auf der Makro-Ebene, bzw. auf einer Zwischenebene aggregierten Verhaltens durchzuführen. Für diese Diskussion wird davon ausgegangen, dass nur eine Ebene aggregierten Verhaltens betrachtet wird. Für weitere Zwischenebenen gelten aber die gleichen Punkte der Diskussion.

#### 9.4.1.1 Definition der Ebene aggregierten Agentenverhaltens in Form von *IVM*

Zunächst muss ein Modell idealen aggregierten Verhaltens auf Basis von *IVM* für die Kalibrierung auf der Makro-Ebene erstellt werden.

Damit dies möglich ist muss die **Voraussetzung erfüllt sein, dass der Nutzer überhaupt eine Ebene aggregierten Verhaltens definieren kann**. Bisher wird valides Verhalten auf der Makro-Ebene lediglich durch Validierungskriterien in Form der globalen Zielfunktion *GZF* beschrieben.

Um den Makro-Kalibrierungsschritt des Basis-Makro-Mikro-Verfahrens durchführen zu können, muss der Nutzer nun eine Ebene idealen aggregierten Agentenverhaltens definieren können. Dies muss so geschehen, dass er zu den Agenten auf der Mikro-Ebene *IVM*s spezifizieren kann, die in einem Modell auf der Makro-Ebene als valide Repräsentation des aggregierten Zielverhaltens simuliert werden können.

Diese Voraussetzung zu erfüllen kann aus verschiedenen Gründen schwierig sein. Problematisch für die Anwendung des Makro-Mikro-Verfahrens sind diejenigen Fälle, in denen kein Wissen zur Definition aggregierter Ebenen vorhanden ist. Dabei können die folgenden Fälle unterschieden werden:

1. **Fehlendes Wissen über Existenz aggregierter Ebenen:** Dies ist der ungünstigste Fall für die Anwendung des Makro-Mikro-Verfahrens. Es ist keinerlei Wissen vorhanden, wie sich das erwünschte Makro-Verhalten aus dem Simulationsverhalten des Mikro-Modells ergeben soll. Dies könnte z.B. bei einem Modell der Fall sein, mit dem untersucht werden soll, ob bestimmte am Originalsystem beobachtete Mikro-Interaktionen ein bestimmtes aggregiertes Verhalten erzeugen können, ohne dass bekannt ist, ob dies überhaupt möglich ist, bzw. wie dies geschehen soll. Bei solchen Modellen besteht ein Simulationsziel darin, überhaupt erst zu untersuchen welche Wechselwirkungen und aggregierte Verhaltensweisen bei der Simulation eines entsprechenden Modells entstehen (siehe z.B. Axelrod [7]).
2. **Es existieren keine zusätzlichen Ebenen aggregierten Verhaltens oder der Modellierer findet sie nicht:** In diesem Fall gibt es nur eine Aggregierungsstufe für das Verhalten auf der Mikro-Ebene: die Ebene des Gesamtverhaltens. Es ist schwierig eine Aussage darüber zu treffen, ob/wie oft es solche Fälle gibt, da es immer möglich ist, dass eigentlich Fall 1 gilt. Selbst in sehr einfachen Modellen wie z.B. dem bekannten Boids-Modell<sup>2</sup> kann eine Zwischenebene mit Aufteilung in Bewegungsverhalten und Gruppierungsverhalten gebildet werden. Grundsätzlich ist aber klar, dass es für kleine Modelle - wenn man z.B. nur ein Teilmodell betrachtet - nicht unbedingt Zwischenebenen geben muss oder der Einsatz zusätzlicher Zwischenebenen bei der Kalibrierung keinen Vorteil bringt. Für komplexere Modelle mit verschiedenen Agentenklassen und Teilverhaltensweisen ist davon auszugehen, dass es Zwischenebenen gibt, da zumindest Ebenen für das Verhalten einzelner Agenten und ihrer Teilverhaltensweisen gebildet werden können.
3. **Aggregierte Ebenen existieren, es steht aber nicht genug Wissen zur Verfügung, um sie zu spezifizieren:** In diesem Fall können die *IVM* nicht spezifiziert oder nur geschätzt werden. Dies ist als einer der häufigsten Problemfälle bei der Anwendung des Makro-Mikro-Verfahrens anzusehen. Der Nutzer hat die Mikro-Ebene modelliert und eine Vorstellung davon, wie ideales aggregiertes Verhalten aussehen soll. Er kann zu dieser Vorstellung aber kein aggregiertes Modell auf der Basis von *IVM* ableiten. Eine mögliche Ursache hierfür ist, dass das Verhalten auf der Mikro-Ebene zu komplex ist. Dies kann z.B. der Fall sein, weil es Zufallseinflüsse im Modell oder Rückkopplungsschleifen in den Agenteninteraktionen erschweren, aus den Mikro-Aktionen aggregierte *IVM* ableiten zu können. Die Auswirkungen von Zufallseinflüssen und Rückkopplungsschleifen werden im Folgenden noch genauer betrachtet.

---

<sup>2</sup>Siehe Webseite von Craig Reynolds <http://www.red3d.com/cwr/boids/> (Stand: November 2010)

Kann keine Ebene aggregierten Verhaltens definiert werden oder können für diese Ebene weder *IVM* noch deren zugehörigen Strukturen *IVMF* für das Reverse-Engineering spezifiziert werden, dann kann das Makro-Mikro-Verfahren für das Kalibrierungsproblem nicht eingesetzt werden.

#### 9.4.1.2 Reverse-Engineering von IVM

Eine Möglichkeit mit dem Problem umzugehen, dass nicht genügend Wissen zur Spezifikation der *IVM* verfügbar ist, ist die Reverse-Engineering-Verfeinerung für das Basis-Makro-Mikro-Verfahren. Beim Einsatz von Reverse-Engineering muss der Nutzer nur noch die Strukturen der *IVMF* für die aggregierte Verhaltensebene spezifizieren. Diese sollen dann im Reverse-Engineering-Schritt so kalibriert werden, dass man die *IVM* erhält. Beim Reverse-Engineering wird also ein eigenes Kalibrierungsproblem für die aggregierte Ebene gelöst. Damit dies funktioniert müssen die folgenden Voraussetzungen erfüllt sein:

- Die *IVMF* beschreiben valide Strukturen für das benötigte aggregierte Verhalten: Bei der Entwicklung der *IVM* muss nicht nur ein Kalibrierungsproblem sondern auch ein Modellierungsproblem gelöst werden. Je nachdem wie komplex eine *IVMF* entworfen werden muss, um valides aggregiertes Agentenverhalten erfassen zu können, kann ihre Spezifikation unterschiedlich schwierig sein. Soll im Modell lediglich Agentenverhalten erfasst werden, das sich nicht dynamisch ändert, ist eine solche Spezifikation relativ gut möglich. Problematischer ist der Fall, wenn ein Agentenverhalten erfasst werden soll, das sich zur Laufzeit der Simulation aufgrund komplexer Interaktionen mit anderen Agenten dynamisch verändert.

Sind die Strukturen der *IVMF* ungeeignet oder stellen sie nicht ausreichend Randbedingungen an die aus ihnen erstellbaren *IVM*, dann besteht die Gefahr, dass während des Reverse-Engineering falsche *IVM* ermittelt werden. Ist dies der Fall kann das Makro-Mikro-Verfahren nicht fortgeführt werden oder liefert falsche Ergebnisse.

- Beim Reverse-Engineering werden die *IVMF* zu *IVM* kalibriert und müssen dabei hinsichtlich ihrer Validität bewertet werden. Damit dies möglich ist, muss es ein simulierbares auf den *IVMF* basierendes Modell für die aggregierte Ebenen geben, das kalibriert und dessen Verhalten validiert werden kann. Können nur *IVMF* spezifiziert werden, die zugehörigen *IVM* aber nicht simuliert werden, dann ist es nicht möglich das Reverse-Engineering durchzuführen.
- Es existieren ausreichend Randbedingungen für das Reverse-Engineering: Abhängig von der Struktur der *IVMF* und vorhandener Randbedingungen an die *IVM* kann die Kalibrierung der *IVMF* leichter oder schwieriger sein. Ist der Suchraum für die Kalibrierung der *IVMF* durch die Modellstrukturen und Randbedingungen stark beschränkt, dann kann die Kalibrierung leichter durchgeführt werden. In der Fallstudie der Agenten-basierten Einkaufssimulation wird in Kapitel 15 ein Fall betrachtet in dem der Suchraum für die *IVMF* kaum beschränkt ist und kaum Daten zur Formulierung von Randbedingungen zur Verfügung stehen. Als Konsequenz können die *IVMF* zu unterschiedlichen *IVM* ausgeprägt werden und es ist schwierig zu entscheiden, welche *IVM* valide sind.
- Es steht ein Algorithmus für das Reverse-Engineering zur Verfügung: Im einfachsten Fall kann das Reverse-Engineering mit Hilfe von Black-Box-Kalibrierung durchgeführt



werden. Ist zusätzliches Wissen über die Zusammenhänge auf der aggregierten Ebene verfügbar, dann kann das Reverse-Engineering mit wissensbasierter Suche sogar effektiver als mit Black-Box-Kalibrierung durchgeführt werden. In der Fallstudie in Kapitel 15 konnte festgestellt werden, dass dem Algorithmus für das Reverse-Engineering entscheidende Bedeutung für das Ergebnis des Reverse-Engineering zu kam. In der Fallstudie mussten für das Reverse-Engineering der *IVM* 132000-Parameterwerte kalibriert werden. Dies war für ein Black-Box-Verfahren nicht durchführbar. Es konnte aber ein eigener wissensbasierter Kalibrierungsalgorithmus zur Ermittlung der *IVM* definiert werden, mit dem das Reverse-Engineering durchgeführt werden konnte. Die Validität der ermittelten *IVM* hing dabei aber nun auch vom Reverse-Engineering-Algorithmus selbst ab. War das im Algorithmus umgesetzte Kalibrierungswissen ungeeignet, so waren es auch die ermittelten *IVM*.

Das Basis-Makro-Mikro-Verfahren kann nur dann ohne Reverse-Engineering durchgeführt werden, wenn der Nutzer die *IVM* direkt spezifizieren kann. Kann der Nutzer nur Strukturen *IVMF* angeben, so muss immer die Reverse-Engineering-Verfeinerung verwendet werden. Mit der Durchführung des Reverse-Engineering kann der Makro-Schritt des Makro-Mikro-Verfahrens abgeschlossen werden.

#### 9.4.2 Vorteile des Makro-Mikro-Verfahrens durch den Makro-Schritt

Der Makro-Schritt des Makro-Mikro-Verfahrens wird durchgeführt, um alle oder Teile der folgenden Vorteile zu erlangen:

- **White-Box-Zerlegbarkeit des Kalibrierungsproblems auf der Mikro-Ebene:** Im Idealfall erlauben die auf der aggregierten Ebene ermittelten *IVM* die Zerlegung des Mikro-Kalibrierungsproblems in unabhängige Teilprobleme für jedes *IVM*, die dann wieder zum Gesamtergebnis kombiniert werden können. Weiter gehört im Idealfall zur jedem einzelnen Agenten auf der Mikro-Ebene ein solches unabhängiges *IVM*, so dass die Kalibrierung jedes einzelnen Agenten unabhängig von den übrigen mit im Vergleich zur vollen Black-Box-Kalibrierung stark eingeschränktem Parameter-suchraum durchgeführt werden kann.

**Dies ist aber nur unter den folgenden Randbedingungen möglich:**

- **Unabhängigkeit der *IVM*:** Damit White-Box-Zerlegung auf Basis der *IVM* möglich ist, müssen die in Abschnitt 3.1.5.3 beschriebenen Bedingungen erfüllt sein. Sind sie nicht erfüllt, dann können die *IVM* nicht zur Zerlegung des Mikro-Kalibrierungsproblems genutzt werden. Die Bedingungen für White-Box-Zerlegbarkeit sind umso schwerer zu erfüllen, je komplexer die Aktionen der Agenten auf der Mikro-Ebene miteinander gekoppelt sind. Je stärker das Verhalten der Agenten auf der Mikro-Ebene durch ihre Interaktionen miteinander gekoppelt ist desto wahrscheinlicher ist es, dass ein zugehöriges *IVM* nur aggregiertes kombiniertes Verhalten der Agenten beschreiben kann. Je weniger Teil-Mikromodelle aber gebildet werden können, die auf der Makro-Ebene durch *IVM* repräsentiert werden, desto weniger kann das Mikro-Kalibrierungsproblem in Teilprobleme zerlegt werden. Im schlechtesten Fall können die Wechselwirkungen zwischen den Agenten auf der Mikro-Ebene so komplex sein, dass keine Zerlegung des Kalibrierungsproblems möglich ist. In diesem schlechtesten

Fall kann das Makro-Mikro-Verfahren zwar immer noch zur Überbrückung der Makro-Mikro-Lücke, aber nicht zur Zerlegung des Kalibrierungsproblems eingesetzt werden.

Besonders starke Wechselwirkungen sind Rückkopplungsschleifen in den Agenteninteraktionen. Unter einer Rückkopplungsschleife wird verstanden, dass ein Agent durch eine Aktion  $A$  seine Umwelt/andere Agenten verändert und diese veränderte Umwelt rückwirkend den Zustand des Agenten selbst beeinflusst. Als Folge hat dann bei erneuter Ausführung von Aktion  $A$  durch den Agenten diese einen anderen Effekt hat als beim ersten Mal. Das Vorhandensein von Rückkopplungsschleifen im Mikro-Modell hat für sich weder positive noch negative Auswirkungen auf die Anwendbarkeit des Makro-Mikro-Verfahrens:

1. Im Optimalfall ist die fortschreitende Zustandsentwicklung der Agenten und ihrer Umwelt selbst ein wichtiger Validitätsaspekt für das Simulationsverhalten und der Modellierer oder Fachexperte kann in *IVM* auf einem Detaillierungsgrad nahe der Mikro-Ebene genau spezifizieren, welche valide aggregierte Zustandsentwicklung vom Modell erzeugt werden soll. Je nach Granularität der *IVM* ist es dann auch im Falle von Rückkopplungsschleifen möglich auf Basis der *IVM* eine White-Box-Zerlegung des Kalibrierungsproblems vorzunehmen.
2. Im schlechtesten Fall enthält das Modell auf der Mikro-Ebene viele komplexe Rückkopplungsschleifen und es fehlt Wissen wie aggregiertes Verhalten aus dem Mikro-Verhalten entsteht, bzw. auf dieses abgebildet werden kann. Das hat zur Folge, dass nur schwierig individuelle unabhängige *IVM* definiert und das Mikro-Kalibrierungsproblem auf Basis der *IVM* kaum in Teilprobleme zerlegt werden kann. Im Extremfall kann hier nur eine Black-Box-Kalibrierung eingesetzt werden.

Beispiel: In Axelrods Koalitionsmodell [7] werden Rückkopplungsschleifen auf der Mikroebene entworfen, um ihre Auswirkungen auf resultierendes aggregiertes Verhalten zu untersuchen. Es wurde ein abstraktes Modell ohne Daten betrachtet in dem kein Wissen vorgegeben war, ob und welches aggregierte Verhalten aus dem Mikroverhalten entstehen soll. Ohne weiteres Zusatzwissen kann das Makro-Mikro-Verfahren zur Kalibrierung eines solchen Modells auf ein bestimmtes Gesamtverhalten hin nicht sinnvoll angewendet werden. Es werden weitere Vorgaben benötigt, welche Teilverhaltensweisen aus den Rückkopplungsschleifen entstehen und dann zum Gesamtverhalten führen sollen.

In der Praxis sind Mischformen zwischen den beiden geschilderten Extremen zu Wissen über Rückkopplungsschleifen zu erwarten, da davon ausgegangen werden kann, dass der Modellierer bestimmte Wechselwirkungsmechanismen auf der Mikro-Ebene einbaut, weil er bestimmte aggregierte (Teil-)Verhaltensweisen erzeugen möchte. Dieses Wissen kann dann wieder dazu genutzt werden, um zugehörige Teilmodelle aggregierten Verhaltens in den Kalibrierungsprozess zu integrieren. Abhängig vom konkreten Modell muss bei der initialen Analyse des Modells untersucht werden, ob es Rückkopplungsschleifen gibt, welche im Makro-Mikro-Verfahren trotzdem eine Definition unabhängiger *IVM* erlauben und ob es Rückkopplungsschleifen gibt, welche es für einen bestimmten Modell-

teil nur erlauben, es als Ganzes zu aggregieren.

- **Hohe Granularität der IVM:** Damit das Kalibrierungsproblem für jeden Agenten auf der Mikro-Ebene in ein unabhängiges Teilproblem zerlegt werden kann, muss es möglich sein, für jeden Agenten ein eigenes *IVM* zu spezifizieren. Das Makro-Mikro-Verfahren ist aber immer noch anwendbar, wenn *IVM* nur für größere Teilmodelle (z.B. Agentengruppen) spezifiziert werden können. Hierdurch werden aber die Effizienz-Vorteile des Makro-Mikro-Verfahrens, z.B. verkleinerte Suchräume, eingeschränkt.

- **Ermittlung zusätzlicher Randbedingungen an die Kalibrierung auf der Mikro-Ebene:** Selbst wenn die ermittelten *IVM* so starke Abhängigkeiten zueinander haben, dass keine White-Box-Zerlegung des Mikro-Kalibrierungsproblems möglich ist, dann können die *IVM* trotzdem als zusätzliche Randbedingungen an die Mikro-Kalibrierung zur Einschränkung des Suchraums verwendet werden.

Ein Ausgangsproblem der Black-Box-Kalibrierung von ABS war, dass mit steigender Anzahl und Komplexität an Interaktionen auf der Mikroebene die Wahrscheinlichkeit steigt, dass unterschiedliche Verhaltensweisen auf der Mikroebene zum gleichen Makroverhalten führen.

Durch das Makro-Mikro-Verfahren wird die explizite Einbeziehung von Wissen über aggregierte Zwischenverhaltensebenen in den Kalibrierungsprozess ermöglicht. Durch das zusätzliche Wissen kann während der Kalibrierung ein Mikro-Verhalten gefunden werden, das stärker dem Originalsystem entspricht und so die Gesamtqualität des Kalibrierungsergebnisses verbessert werden. Dies ist auch dann möglich, wenn aufgrund starker Wechselwirkungen zwischen den *IVM* keine Zerlegung des Kalibrierungsproblems auf der Mikro-Ebene möglich ist.

- **Reduktion der zur Kalibrierung benötigten Gesamtzahl an Simulationsläufen:** Oben wurde bereits beschrieben, dass das Makro-Mikro-Verfahren die Voraussetzungen für White-Box-Zerlegungen des Mikro-Kalibrierungsproblems schaffen kann. Weiter wurde in der Diskussion des Reverse-Engineering beschrieben und in einer Fallstudie im Evaluationskapitel 15 untersucht, wie im Rahmen des Makro-Mikro-Verfahrens durch Nutzung zusätzlichen Kalibrierungswissens in einem wissensbasierten Kalibrierungsalgorithmus die Menge an benötigten Simulationsläufen im Vergleich zur Black-Box-Kalibrierung reduziert werden konnte.

Da dieser Vorteil durch die Verwendung zusätzlichen Kalibrierungswissens gewonnen wird, kann er nicht auf alle mögliche Anwendungen des Makro-Mikro-Verfahrens verallgemeinert werden. Es ist daher nicht garantiert, dass das Makro-Mikro-Verfahren schneller ist als ein Black-Box-Verfahren. Im schlechtesten Fall kann es sein, dass das Makro-Mikro-Verfahren deutlich länger dauert als ein Black-Box-Verfahren. Richtig und auf ein geeignetes Modell angewendet, d.h. wenn vor der Anwendung des Makro-Mikro-Verfahrens die benötigten Voraussetzungen überprüft wurden und diese erfüllt werden können, kann das Makro-Mikro-Verfahren aber deutlich weniger Simulationsläufe benötigen als ein Black-Box-Verfahren.

### 9.4.3 Durchführung des Mikro-Schrittes im Makro-Mikro-Verfahren

Im Mikro-Schritt des Makro-Mikro-Verfahrens wird für jedes Teil-Mikro-Kalibrierungsproblem die Kalibrierung bezüglich einer *IZF* durchgeführt, die aus der zugehörigen *IVM* abgeleitet wurde. Die Kalibrierung wird mit Black-Box-Verfahren oder falls zusätzliches Kalibrierungswissen vorhanden ist mit Wissen-basierten Kalibrierungsregeln durchgeführt.

Damit der Mikro-Schritt des Makro-Mikro-Verfahrens durchgeführt werden kann müssen die folgenden Voraussetzungen erfüllt sein:

- **Aus den *IVM* müssen *IZF* ableitbar sein:** Eine *IZF* muss so definiert werden, dass sie ihren optimalen Wert annimmt, wenn sich das Mikro-Modell genauso verhält wie das *IVM*. Es muss also möglich sein das *IVM*-Verhalten durch eine zugehörige *IZF* vollständig zu erfassen. Ist das nicht, nur in Form von Randbedingungen oder nur teilweise möglich so wird die Kalibrierung auf der Mikro-Ebene erschwert. Im Evaluationskapitel 15 wird für die Einkaufssimulation gezeigt wie dort *IZF* aus *IVM* ermittelt werden können.
- **Das Mikro-Modell muss das *IVM*-Verhalten erzeugen können:** Ist dies nicht der Fall, dann kann das Mikro-Modell nicht so kalibriert werden, dass sein Verhalten dem des zugehörigen *IVM* entspricht. Ist dies der Fall steht man bei der Modellentwicklung vor einem Dilemma: Entweder das *IVM* bildet falsches ideales Verhalten ab oder die Modellstrukturen des Mikro-Modells sind nicht geeignet, um valides Verhalten zu simulieren. Zum Umgang mit diesem Dilemma gibt es im Makro-Mikro-Verfahren den Iterationsschritt zwischen Mikro- und Makro-Ebene, bei dem auf der Makro-Ebene die Validität der bisher gewählten *IVM* in Frage gestellt wird.

In einer Variante der Fallstudie der Agenten-basierten Einkaufssimulation, die in Kapitel 15 beschrieben wird, tritt genau dieses Problem auf.

### 9.4.4 Robuste Kalibrierung

In der robusten Suche werden die im Makro-Schritt festgelegten oder mittels Reverse-Engineering ermittelten *IVM* genutzt, um fehlerhafte Modellteile oder Daten durch die Verwendung idealer aggregierter Teilverhaltensmodelle auszugleichen.

Damit die robuste Kalibrierung anwendbar ist, müssen aber die folgenden Punkte beachtet werden:

- **Ersetzung von Mikro- durch ideale Teile:** Jede Ersetzung, die von der robusten Suche vorgenommen wird tauscht einen Mikro-Teil durch seinen entsprechenden Gegenstück für ideales Verhalten aus. Hierbei besteht die Gefahr, dass der Nutzen des ursprünglichen Mikro-Modells für seinen Anwendungszweck eingeschränkt wird. Vor dem Einsatz der robusten Suche muss daher immer genau geprüft werden, ob die Ersetzung das Mikro-Modell nur der Ebene idealen aggregierten Verhaltens ähnlicher machen würde, ohne dass etwas für die Aussagekraft der Simulation auf der Mikro-Ebene gewonnen wird. Kann problematisches Verhalten ausgefiltert werden, so dass der die Ergebnisse des Mikro-Modells wirklich verbessert werden, dann sollte die robuste Kalibrierung eingesetzt werden.

- **Anwendbarkeit für einzelne stark verrauschte Modellteile und Daten:** Die robuste Suche dient dazu, einzelne stark verrauschte Modellteile oder Daten auszufiltern. Sind aber alle zu untersuchenden Elemente gleichmäßig verrauscht, so ist die robuste Kalibrierung auf der Mikro-Ebene nicht sinnvoll.

Im Evaluationsteil (siehe Kapitel 17) wurde die Anwendung der robusten Kalibrierung auf die Fallstudie der Agenten-basierten Einkaufssimulation untersucht. Die Ergebnisse zeigten, dass die robuste Suche grundsätzlich auf die Fallstudie angewandt werden konnte. Es war zentral, dass die resultierenden Ergebnisse der robusten Kalibrierung genau analysiert wurden und die Ergebnisse bei unterschiedlichen Mengen ausgefilterter Elemente miteinander verglichen wurden. In der Fallstudie lag eine gleichmäßige Verrauschung der fehlerhaften Elemente vor. Daher konnte die robuste Suche hier nicht zur Verbesserung des Mikro-Modellverhaltens eingesetzt werden, sondern machte das Mikro-Modell nur ähnlicher zum idealen aggregierten Verhalten.

#### 9.4.5 Zufallseinflüsse im Makro-Mikro-Verfahren

Nachdem in der Diskussion bisher die Voraussetzungen, möglichen Probleme und Vorteile bei der Durchführung der konkreten Makro-Mikro-Verfahrensschritte diskutiert wurden, wird in diesem Abschnitt noch betrachtet welche Auswirkungen der Anteil an Zufallseinflüssen im Mikro-Modell auf die Anwendbarkeit des Makro-Mikro-Verfahrens haben kann.

##### 9.4.5.1 Zufall auf der Mikro- und Makro-Ebene

Die möglichen Auswirkungen von Zufallseinflüssen auf das Makro-Mikro-Verfahren zur Kalibrierung sind:

- **Struktur und Komplexität der Agenteninteraktionen:** Je mehr Agenteninteraktionen vom Zufall beeinflusst werden, desto schwieriger wird es für einzelne Agentenmodelle zu verstehen, was ihre Aktionen auf der Mikro-Ebene für Auswirkungen haben und als Konsequenz wird es immer schwieriger für sie, von den übrigen Agenten unabhängige *IVM* zu spezifizieren. Die Folge ist, dass aggregiertes Verhalten nur für Agentengruppen betrachtet und spezifiziert werden kann. Hierdurch wird die mögliche Granularität der spezifizierbaren *IVM* und damit die Zerlegbarkeit des Kalibrierungsproblems im Mikro-Schritt eingeschränkt.
- **Heterogene oder homogene Parameterwerte:** Je mehr Agenten eine Agentenklasse simuliert werden und je höher der Zufallsanteil ihrer Aktionen ist desto unwichtiger wird das individuelle Verhalten der einzelnen Agenten für das Gesamtverhalten.

Als Beispiel können zwei Agenten betrachtet werden, die sich zwei Wege entlang bewegen. Ein Weg wird von einem breiten Fluss gekreuzt, während der andere von einem schmalen Graben unterbrochen wird. Der Agent, der an den Fluss kommt, muss schwimmen können, um den Fluss zu überqueren. Der Agent, der an den Graben kommt muss nicht schwimmen, aber über den Graben springen können. Geht jeder Agent immer nur genau einen Weg entlang benötigen sie jeweils heterogenes Verhalten. Ist es zufällig welchen Weg die Agenten nehmen müssen sie sowohl schwimmen als auch springen können und benötigen daher homogenes Verhalten.

Betrachtet man nun eine Menge von Agenten einer Agentenmodellklasse gilt, dass mit steigendem Zufallsanteil im Agentenverhalten auf der Mikro-Ebene heterogene Parameterwerte der Agenten für das aggregierte Verhalten eine immer geringere Rolle spielen. Im Extremfall könnte jeder Agent zufällig zu jedem Zeitpunkt an jedem Ort eine Aktion ausführen.

Für diese Auswirkung von Zufallseinflüssen werden daher keine unabhängigen *IVM* für jeden einzelnen Agenten benötigt. Durch den Zufallseinfluss werden nur homogene Parameterwerte und damit nur ein gemeinsames *IVM* für alle Agenten der Klasse benötigt.

Als Konsequenz spielen Mikro-Zufallseinflüsse im Makro-Mikro-Verfahren für die Kalibrierung im Mikro-Schritt auf jeden Fall und für den Makro-Schritt abhängig von den Auswirkungen des Zufalls auf das aggregierte Gesamtverhalten eine Rolle.

Für das Makro-Mikro-Verfahren ergeben sich die folgenden Auswirkungen von Zufallseinflüssen:

- Im besten Fall kann trotz Zufallseinflüssen für jeden Agent ein eigenes *IVM* spezifiziert werden, welches die Mikro-Zufallseinflüsse so aggregiert, dass sie im Makro-Schritt nicht betrachtet werden müssen.
- Im schlechtesten Fall enthält das Modell wenige Agenten aus unterschiedlichen Agentenklassen, die untereinander komplex und zufallsabhängig interagieren. In diesem Fall besteht die Gefahr, dass auf der aggregierten Ebene ohne umfassendes Wissen über die Wirkungszusammenhänge im simulierten System keine unabhängigen *IVM* für die einzelnen Agenten sondern nur ein *IVM* für ihr aggregiertes Gesamtverhalten gebildet werden kann. In diesem Fall kann im Makro-Mikro-Verfahren keine White-Box-Zerlegung des Kalibrierungsproblems durchgeführt werden und es muss wie im Black-Box-Verfahren ein Modell betrachtet werden, das allerdings auf Makro- und Mikro-Ebene betrachtet wird. Zur Kalibrierung in einem solchen Fall kann im Makro-Mikro-Verfahren die in Kapitel 10 vorgestellte Technik der Kalibrierung mit Heterogenitätssuche eingesetzt werden.
- Im Falle vieler Agenten einer Klasse kann Zufallsverhalten ein Hinweis darauf sein, dass nur homogenes und kein heterogenes Verhalten der Agenten benötigt wird. Ist dies der Fall wird das Kalibrierungsproblem deutlich vereinfacht.

Zur Bestimmung der genauen Auswirkungen der Zufallseinflüsse muss das Simulationsverhalten des Modells aber zunächst analysiert werden. Dies ist für das Makro-Mikro-Verfahren nicht anders als für Black-Box-Verfahren. Vor der Durchführung des Makro-Schrittes muss betrachtet werden, ob Zufallseinflüsse auf der Mikro-Ebene auch Auswirkungen auf das aggregierte Verhalten im Makro-Schritt haben oder ob sie durch die „Durchschnittsbildung“ bei der Aggregation auf der Makro-Ebene keine Rolle mehr spielen.

Abhängig von den Auswirkungen der Zufallseinflüssen kann die Kalibrierung des Modells dadurch erschwert werden, dass mehr Simulationsläufe durchgeführt werden müssen, um die resultierende Verhaltensvalidität zu einer Parametereinstellung bewerten zu können.

### 9.4.6 Zusammenfassung: Anwendbarkeit und Nutzen des Makro-Mikro-Verfahrens

In diesem Kapitel wurde die Anwendbarkeit des Makro-Mikro-Verfahrens, Voraussetzungen, Probleme und seine Vorteile im Vergleich zu Black-Box-Verfahren diskutiert.

Der Kernvorteil des Makro-Mikro-Verfahrens gegenüber von Black-Box-Verfahren ist, dass es explizit zusätzliches Wissen über Beobachtungsebenen aggregierten Verhaltens in den Kalibrierungsprozess mit einbezieht, was bei Black-Box-Verfahren nicht möglich ist. Hierdurch wird es möglich dieses Wissen zu nutzen, um die Kalibrierungsergebnisse im Vergleich zu Black-Box-Kalibrierung zu verbessern und den benötigten Kalibrierungsaufwand in Form durchzuführender Simulationsläufe im Vergleich zu Black-Box-Kalibrierungen zu reduzieren:

- **Zusätzliche Zwischenebenen:** Durch spezielle Analysetechniken wird das Verständnis des Kalibrierungsproblem systematisch erhöht. Es werden explizit Zwischenebenen aggregierten Teil-Mikromodellverhaltens und ihr Einfluss auf das Gesamtverhalten analysiert. Diese Zwischenebenen wurden zuvor nicht betrachtet und konnten daher nicht während der Kalibrierung eingesetzt werden.
- **Zusätzliche Randbedingungen:** Durch die Einführung der *IVM* und Festlegung der zugehörigen *IZF* werden die Validierungskriterien von der Makro-Ebene auf die Mikro-Ebene verlagert. Dies hat den Vorteil, dass für die Kalibrierung auf der Mikro-Ebene im Vergleich zur Black-Box-Kalibrierung mehr Mikro-Randbedingungen zu Verfügung stehen, wodurch der Parametersuchraum weiter eingeschränkt werden kann.
- **White-Box-Zerlegungen:** Die Betrachtung des aggregierten Teil-Mikromodellverhaltens hat das Ziel den Einsatz von White-Box und wissensbasierten Verfahren zu ermöglichen, die Modularisierung des Kalibrierungsproblems zu unterstützen und die Wertebereiche der Parameterwerte auf der Mikro-Ebene zu beschränken. Die auf aggregierter Ebene ermittelten *IVM* und die zugehörigen *IZF* können zur Vereinfachung des Kalibrierungsproblems auf der Mikro-Ebene durch White-Box-Zerlegungen eingesetzt werden.
- **Wissensbasierte Ansätze:** Für den Makro-Schritt und die Teil-Kalibrierungen auf der Mikro-Ebene können wissensbasierte Kalibrierungsverfahren eingesetzt werden, welche das Kalibrierungsproblem mit weniger Simulationsläufen lösen als ein Black-Box-Ansatz.
- **Robuste Kalibrierung:** Durch das Ersetzen von Modellteilen durch zugehörige ideale Modelle aggregierten Verhaltens, die über regressionsähnliche Methoden aus den Validierungskriterien gewonnen wurden, kann die Analyse weiter verbessert und die Kalibrierung des Modells robust gegen fehlerhafte Modellteile gemacht werden.

Der Kernvorteil des Makro-Mikro-Verfahrens führt aber auch zur Hauptschwierigkeit, die sich bei der Anwendung des Makro-Mikro-Verfahrens stellt. Die Vorteile des Makro-Mikro-Verfahrens kommen nur zum Tragen, wenn dem Nutzer ausreichendes Wissen über aggregiertes Modellverhalten und Wirkungszusammenhänge des Modells auf der aggregierten

Ebene und auf der Mikro-Ebene zur Verfügung steht. In der Diskussion wurde aufgezeigt, wie fehlendes Wissen in bestimmten Schritten die Vorteile des Makro-Mikro-Verfahrens einschränken und die Anwendbarkeit des Verfahrens erschweren kann.

Demzufolge, ist es für ein neues Kalibrierungsproblem schwierig, ad hoc zu bestimmen, ob das Makro-Mikro-Verfahren Vorteile gegenüber einer Black-Box-Kalibrierung bringt oder nicht. Die Frage, ob sich in einem konkreten Fall der Einsatz des Makro-Mikro-Verfahrens im Vergleich zu einer einfachen Black-Box-Optimierung lohnt, muss ausgehend von den Stärken und Schwächen der Black-Box-Kalibrierung betrachtet werden.

Black-Box-Kalibrierungsverfahren sind für die Kalibrierung von Simulationen besonders geeignet, :

1. Wenn nur wenige Parameter zu kalibrieren sind, d.h. die Suchraumgröße relativ klein ist. Dies ist z.B. dann der Fall, wenn die Annahme gemacht wird, dass jeder Agent mit gleicher Struktur auch gleiche Parameterwerte hat (homogene Parameterwerte).
2. Wenn die Simulationslaufzeit des Modells so kurz ist, dass viele Parameter-einstellungen in kurzer Zeit evaluiert werden können.
3. Wenn die Struktur des Parametersuchraums einigermaßen bekannt ist und wir demnach einschätzen können, dass der Suchraum nur wenige lokale Optima enthält.

Ist unklar, ob die obigen Punkte erfüllt sind oder nicht, so lohnt sich immer ein erster Kalibrierungslauf mit Black-Box-Methoden. Liefert dann die Kalibrierung mit Black-Box-Verfahren kein zufriedenstellendes Ergebnis, d.h. kann kein ausreichend valides Gesamtverhalten gefunden werden, so sollte geprüft werden, ob zusätzliches Wissen zur Erfüllung der Voraussetzungen des Makro-Mikro-Verfahrens verfügbar ist und das Makro-Mikro-Verfahren sollte zur Verbesserung der Ergebnisqualität gestartet werden.

Das Makro-Mikro-Verfahren kann aber auch direkt verwendet werden, da sein Einsatz den Vorteil bringt das Verständnis für das Modell und die Zusammenhänge im Modell zu erhöhen.

#### 9.4.7 Das Makro-Mikro-Verfahren als Instanz des EM-Algorithmus?

Betrachtet man die Vorgehensweise im Makro-Mikro-Verfahren, so erscheinen diese recht ähnlich zu der des bekannten EM-Algorithmus (Expectation-Maximation Algorithmus) [98]. Der EM-Algorithmus ist ein generisches Suchverfahren aus der mathematischen Statistik, das viele konkrete Anwendungen (Instanzen), wie z.B. das  $k$ -Medoids-Clustern [66], hat. Das Makro-Mikro-Verfahren ähnelt in seinem Ablauf dem des EM-Algorithmus, es gibt jedoch einige Unterschiede zwischen den beiden Verfahren. In diesem Abschnitt wird zunächst kurz der allgemeine EM-Algorithmus beschrieben und dann das Makro-Mikro-Verfahren vom EM-Algorithmus abgegrenzt.

##### 9.4.7.1 Allgemeiner EM-Algorithmus

Der EM-Algorithmus dient dazu, zu einer gegebenen Menge von Elementen  $E$  einer mehrdimensionalen Zufallsvariablen  $\Phi$ , die Parameter der  $\Phi$  zugrunde liegenden Wahrscheinlichkeitsverteilung  $W(\phi)$  zu bestimmen. Das Problem bei der Parametersuche besteht darin, dass nicht alle Dimensionen von  $\Phi$  „beobachtbar“ sind. Die nicht beobachtbaren Werte sind die sogenannten „versteckten“ Werte der Elemente  $E$ .



Im Folgenden wird der Ablauf des Algorithmus beschrieben. Abbildung 9.7 visualisiert den Ablauf. Ein Vergleich dieser Abbildung mit Abbildung 9.2 lässt die Ähnlichkeit zwischen EM-Algorithmus und Makro-Mikro-Verfahren bereits erahnen.

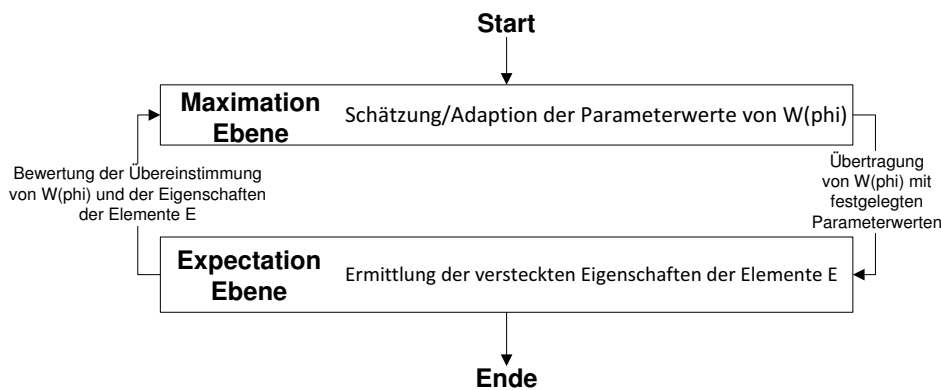


Abbildung 9.7: Ablauf des EM-Algorithmus.

### Ablauf:

Der EM-Algorithmus ist ein iteratives Verfahren zur Lösung obigen Problems. Zu Beginn wird eine erste Schätzung der unbekannt Parameterwerte von  $W(\phi)$  vorgenommen und damit  $W(\phi)$  vollständig festgelegt. Ausgehend von dieser Initialisierung arbeitet der EM-Algorithmus die folgende Iterationsschleife ab:

1. **„Expectation Ebene“:** Im ersten Schritt werden auf der Basis des aktuellen  $W(\phi)$  die Werte der Elemente  $E$  in den unbekannt Dimensionen ermittelt. Sie können bestimmt werden, da  $W(\phi)$  (vorläufig) festgelegt ist. Da die Werte von  $W(\phi)$  nur geschätzt sind, ist es wahrscheinlich, dass die ermittelten Werte der Elemente in  $E$  nicht optimal zu  $W(\phi)$  passen. Wären die Werte von  $E$  von Anfang an wie jetzt ermittelt bekannt gewesen, hätte man auf andere Parameterwerte für  $W(\phi)$  geschlossen. Daher wird nun ein zweiter Schritt ausgeführt.
2. **„Maximation Ebene“:** Durch die Berechnung der Werte der Elemente von  $E$  in den unbekannt Dimensionen im ersten Schritt können im zweiten Schritt nun umgekehrt neue Werte für die Parameter von  $W(\phi)$  ermittelt werden. Da die Eigenschaften der Elemente  $E$  vollständig festgelegt sind, können Maximum-Likelihood-Methoden (siehe Abschnitt 3.1.2.3) eingesetzt werden, um eine neue Schätzung der unbekannt Parameterwerte von  $W(\phi)$  vorzunehmen.

Der Algorithmus ist beendet, wenn sich von einer Iteration der beiden Schritte zur nächsten die Parameterwerte von  $W(\phi)$  nur noch geringfügig ändern. In diesem Fall sind durch die aktuelle Verteilung  $W(\phi)$  die Werte der Elemente  $E$  in den unbekannt Dimensionen so festgelegt, dass vom Algorithmus keine Einstellung mehr gefunden werden kann, bei der  $W(\phi)$  und die durch  $W(\phi)$  bestimmten unbekannt Werte der Elemente in  $E$  besser zueinander passen.

**Problemspezifische, „magische“ Schritte:**

Wie beim Makro-Mikro-Verfahren sind auch beim EM-Algorithmus in seiner allgemeinen Form einige entscheidende Punkte der beiden Iterationsschritte unbestimmt. Zum einen ist unbekannt, wie aus dem festgelegten  $W(\phi)$  die unbekanntesten Werte der Elemente in  $E$  hergeleitet werden sollen. Zum anderen ist unbekannt, wie zu festgelegten Elementen  $E$  passende Parameterwerte für  $W(\phi)$  gefunden werden können. Diese beiden Punkte sind problemspezifisch und müssen in den Instanzen des EM-Algorithmus für spezifische Probleme festgelegt werden.

**9.4.7.2 Vergleich mit dem Makro-Mikro-Verfahren**

Da der EM-Algorithmus auf einer sehr allgemeinen Ebene formuliert ist, wird für den Vergleich zusätzlich eine der bekanntesten Anwendungen des EM-Algorithmus, das  $k$ -Medoids Cluster Verfahren, herangezogen.

Das  $k$ -Medoids-Clusterverfahren wird im weiteren Verlauf dieser Arbeit auch noch in einer weiteren entwickelten Technik eingesetzt: Der Heterogenitätssuche in der Menge der Agenparameter (siehe Abschnitt 10).

**9.4.7.3  $k$ -Medoids-Clusterverfahren:**

Unter Clusterverfahren versteht man Techniken zur Datenanalyse, die aus einer Grundmenge von Objekten, die durch potentiell mehrere Variablen (multivariat) beschrieben werden, Gruppen (engl: Cluster) zusammengehöriger Elemente bilden [66]. Die Verfahren suchen nach Gruppierungen, so dass Elemente im gleichen Cluster bezüglich ihrer beobachtbaren Eigenschaften untereinander möglichst ähnlich, Elemente in unterschiedlichen Clustern möglichst unterschiedlich sind.

***Definition Cluster:** Ein Cluster ist eine Gruppe von Objekten. Die Objekte innerhalb eines Clusters sind bezüglich eines vorgegebenen Ähnlichkeitsmaßes zueinander ähnlicher, als zu Objekten in anderen Clustern.*

Das  $k$ -Medoids Clusterverfahren ist ein Algorithmus zur Einteilung einer Menge von Elementen  $E$  in  $k$  Gruppen. Dabei wird davon ausgegangen, dass den Elementen in  $E$  eine Einteilung in  $k$  Gruppen zugrunde liegt, welche durch die Eigenschaften der Elemente in  $E$  bestimmt ist. Diese Einteilung ist zunächst unbekannt, d.h. die Zuordnung eines Elementes in  $E$  zu einer Gruppe ist die unbekannteste Dimension des Elementes.

Eine festgelegte Einteilung entspricht einem festgelegten  $W(\phi)$  im EM-Algorithmus. Die gesuchten Parameter von  $W(\phi)$  sind die Repräsentanten der jeweiligen Cluster, die **Clusterzentren**. Im  $k$ -Medoids Algorithmus sind dies jeweils die genau mittleren Elemente eines Clusters. Die Ermittlung der unbekanntesten Dimensionen von  $E$  wird durchgeführt, indem jedes Element  $E$  dem Clusterzentrum zugeordnet wird, dem es am ähnlichsten ist. Die Adaption von  $P$  wird durchgeführt, indem zu einer gegebenen Clusterung jeweils innerhalb eines Clusters, das in der neuen Gruppierung genau mittlere Element als neues Clusterzentrum gewählt wird. Der Algorithmus endet, wenn sich die Clusterzentren von einer Iteration zur nächsten nicht mehr ändern.

Vergleichsaspekt	EM-Algorithmus	$k$ -Medoids-Clustering	Makro-Mikro-Verfahren
<b>Gesucht</b>	Parameterwerte $P$ von $W(\phi_i)$ , so dass aus $W(\phi_i)$ Eigenschaften der Elemente $E$ hergeleitet werden können, die optimal zu $W(\phi_i)$ passen.	Clusterzentren, die optimale Repräsentanten der durch sie erzeugten Cluster sind.	Individuelle Parameterwerte der Agenten ( $IAPW$ ), so dass Simulationsverhalten valide bezüglich $VK$ .
<b>Unbekannte bestimmende Eigenschaften</b>	Bestimmte Dimensionen von $E$	Clusterzugehörigkeit der zu gruppierenden Elemente $E$	$IZF$ der Agenten
<b>Initialisierung auf M-/Makro-Ebene</b>	Schätzung der $P$	Schätzung der Clusterzentren	Schätzung der $IVM$
<b>Schritt: E-/Mikro-Ebene</b>	Ermittlung der unbekanntenen Werte der $E$ auf der Basis der $P$	Zuordnung der Elemente zu Clusterzentren	Ermittlung der $IAPW$ auf der Basis von $IZF$ → Ermittlung des Mikro-Simulationsverhaltens zu den $IVM$
<b>Schritt: M-/Makro-Ebene</b>	Adaption der $P$ auf die vollständig festgelegten $E$	Auswahl neuer Clusterzentren auf Basis der generierten Clusterzuordnungen, die erzeugte Cluster besser repräsentieren.	Adaption der $IVM$ aufgrund Bewertung des Simulationsverhaltens bezüglich ursprünglicher $VK$
<b>Bewertung der Zwischenergebnisse</b>	Übereinstimmung zwischen vorherigen $P$ und neu adaptierten $P$	Waren die vorherigen Clusterzentren bereits optimale Repräsentanten der erzeugten Cluster?	Validität des Simulationsverhaltens bezüglich $VK$
<b>Bewertung erfolgt nach</b>	Adaption auf M-Ebene	Adaption auf M-Ebene	Ermittlung der $IAPW$ auf der Mikro-Ebene
<b>Garantierte Konvergenz im allgemeinen Fall</b>	Ja	Ja	Nein
<b>Bekanntes globales Optimum = Ende-Kriterium</b>	Nein	Nein	Ja
<b>Garantiertes lokales Optimum bei Algorithmusende</b>	Ja	Ja	Nein, bzw. stark abhängig von konkreter Umsetzung des Makro-Mikro-Verfahrens

Tabelle 9.1: Vergleich EM-Algorithmus und Makro-Mikro-Verfahren.

Vergleich:

Tabelle 9.1 zeigt die wichtigen Punkte des Vergleichs zwischen EM-Algorithmus und Makro-Mikro-Verfahren. Die zentralen Unterschiede der Verfahren liegen in den jeweils adaptierten Werten, den Bewertungszeitpunkten, der Konvergenz des Verfahrens und der Möglichkeit die globale Validität des Verhaltens einzuschätzen.



# Kapitel 10

## Heterogenitätssuche

Kapitel 9 widmete sich einem großen Problem bei der Entwicklung von ABS, der Makro-Mikro-Lücke. Kapitel 9.3 behandelte zusätzlich den Umgang mit verrauschten Daten im Rahmen dieses Problems. Im Folgenden wird das zweite große Problem, die unklare benötigte Heterogenität in den Parameterwerten der Agenten behandelt. Das von mir entwickelte Verfahren wurde in [36] bzw. [38] veröffentlicht.

Unter Heterogenitätssuche wird ein Verfahren zur Identifikation und Lokalisation geeigneter Heterogenität in den Werten von Agentenparametern verstanden. D.h. es wird für ein ABS-Kalibrierungsproblem untersucht:

- ob valideres Verhalten erreicht werden kann, wenn nicht alle Agentenmodelle auf homogene Parameterwerte festgelegt sind.
- welche Agentenmodelle zueinander homogene und welche zueinander heterogene Parameterwerte haben sollten, damit möglichst valides ABS-Verhalten erreicht werden kann.

Das Verfahren dient dazu, dem Nutzer einen automatisierten Ausweg aus dem Dilemma zu bieten, den Parametersuchraum so klein wie möglich halten zu wollen und gleichzeitig nicht zu wissen, wie groß der Suchraum eigentlich sein muss. Das grundsätzliche Problem des Umgangs mit Parameterheterogenität in der ABS-Kalibrierung wurde in Kapitel 4.1.3.2 dargelegt. Das Analyseverfahren für das Kalibrierungsproblem aus Kapitel 8.1 kann genutzt werden, um Probleme hinsichtlich der Festlegung der Heterogenität der Agenten zu erkennen. Abbildung 8.2 zeigt ein solches Heterogenitätsproblem für die Fallstudie der Agenten-basierten Einkaufssimulation.

In diesem Kapitel werden zunächst die wesentlichen Problemaspekte eines Kalibrierungsproblems mit zusätzlichem Heterogenitätsproblem identifiziert (Abschnitt 10.1). Im nächsten Teil (Abschnitt 10.2) wird dann ein Kalibrierungsverfahren mit Heterogenitätssuche zur Lösung des Problems beschrieben. Die Beschreibung endet mit einer Aufstellung der Anforderungen und Vorteile für das Verfahren (Abschnitt 10.2.5). Im letzten Teil des Kapitels (Abschnitt 10.3) wird die Einbettung des Verfahrens in das zuvor beschriebene Makro-Mikro-Verfahren dargelegt.

## 10.1 Lokalisierung von Heterogenität während der Kalibrierung

Das Ziel des entwickelten Verfahrens besteht darin, dem Nutzer einen geeigneten Mittelweg zwischen der Kalibrierung vollkommen heterogener und rein homogener Parameterwerte zu bieten, wenn unklar ist, wie die Heterogenität verteilt sein muss. Um dieses Ziel zu erreichen, versucht das Verfahren während des Kalibrierungsprozesses zum einen eine geeignete Ausprägung an Heterogenität in den Parameterwerten zu identifizieren und diese Heterogenität geeignet in den Parameterwerten der simulierten Agenten zu lokalisieren. Es wird also versucht Gruppen von Agenten zu bilden, wobei innerhalb der Gruppen alle Agenten homogene Parameterwerte haben und Agenten aus unterschiedlichen Gruppen unterschiedliche Parameterwerte erhalten können.

### 10.1.1 Gruppierung von $AMF$ :

Bisher wurde nur eine Einteilung einzelner Agentenmodelle in Klassen  $AMK$  betrachtet, d.h. Agenten wurden aufgrund homogener Modellstrukturen gruppiert. Es ist natürlich auch möglich, eine weitere Aufteilung der Agentenmodelle innerhalb einer  $AMK$  in Gruppen von Agenten dieser Klasse vorzunehmen. Nach Begriffsbestimmung 8 ist eine Gruppe von Agentenmodellen  $AMG$  eine Teilmenge von Agentenmodellen einer Agentenklasse, in der alle Agentenmodelle die gleichen  $PW$  haben. Dies bedeutet keine konkrete Festlegung der Parameterwerte, sondern lediglich, dass bei einer Konkretisierung der Agentenmodelle aus den jeweiligen  $IAMF$  alle Agentenmodelle einer Gruppe die gleichen Parameterwerte erhalten.

$$AMG \subseteq AMK \text{ mit } \forall IAM_i, IAM_j \in AMG : PW_{IAM_i} = PW_{IAM_j}, i, j \in \mathbb{N}$$

$AMG$  sind also Gruppen von Agenten mit identischen Parameterwerten. Im Fall homogener Agentenparameter sind alle  $IAM$  einer  $AMK$  in der gleichen  $AMG$  und im Falle vollkommen heterogener Parameterwerte bildet jedes  $IAM$  eine eigene  $AMG$ .  $AMK$  wird vollständig in  $AMG$  partitioniert und jedes Agentenmodell gehört zu genau einer  $AMG$ . Durch die Einteilung von  $IAM$  in  $AMG$  erhält der Nutzer die Möglichkeit, Zwischenstufen in der Heterogenität der Werte der Agentenparameter zu spezifizieren. Die Menge an gewählten  $AMG$  bedingt die Anzahl zu kalibrierender Parameterwerte, d.h. durch Veränderung der Anzahl an  $AMG$  für die Kalibrierung wird die Größe des Parametersuchraumes ebenfalls geändert.

### 10.1.2 $KP_{ABS}$ mit Heterogenitätsproblem

Die Verwendung von Gruppen für die Kalibrierung erlaubt es, die Größe des Suchraumes zu beschränken. Damit ein  $KP_{ABS}$  mit möglichst kleinem Parametersuchraum kalibriert werden kann, dessen Lösung trotzdem ausreichend valide ist, müssen die zugehörigen  $AMG$  bekannt sein. Sind die  $AMG$  aber nicht bekannt, so ist deren Identifikation ein nicht triviales Problem. Eine willkürliche Festlegung der  $AMG$  kann die gleichen Probleme bereiten, wie die Verwendung rein homogener Parameterwerte. Die Wahl sinnvoller  $AMG$  wird so selbst zum Teil des Kalibrierungsproblems.

Bei der Ermittlung von  $AMG$  für ein  $KP_{ABS}$  müssen zusätzlich zum eigentlichen Kalibrierungsproblem zwei weitere Aspekte geklärt werden:

1. **Benötigtes Ausmaß an Heterogenität:** Es muss geklärt werden, in wie viele *AMG* die Agentenmodelle eingeteilt werden müssen, damit das Kalibrierungsproblem mit ausreichender Validität des resultierenden Simulationsverhaltens lösbar ist.
2. **Lokalisierung der Heterogenität:** Neben der Anzahl an Gruppen muss auch noch eine sinnvolle Zuordnung der Agenten zu einer der Gruppen ermittelt werden. Wird beispielsweise die Entscheidung getroffen, die Anzahl zu kalibrierender Gruppen von zwei auf fünf zu erhöhen, muss in irgendeiner Form entschieden werden, in welcher Weise die Agenten in Gruppen aufgeteilt werden sollen. Diese Entscheidung kann nicht trivial sein, z.B. wenn eine große Menge von Agenten aufgeteilt werden muss oder die Grenzen zwischen einzelnen Gruppen unklar sind.

Sind beide Aspekte bekannt, kann auf ihrer Grundlage eine Zuordnungsfunktion

$$H_g : AM \rightarrow \{1, \dots, g\}, g \in \mathbb{N}$$

definiert werden.  $H_g$  bildet einzelne konkrete Agentenmodelle aus der Menge aller Agenten  $AM$  einer Agentenklasse  $AMK$  auf einen Indexwert ab. Agenten mit gleichem Indexwert werden einer gemeinsamen *AMG* zugeordnet. Entsprechend der Definition von  $H_g$  werden also  $g$  Agentengruppen gebildet. Nach dieser Zuordnung müssen die Parameterwerte der Agentengruppen und nicht der einzelnen Agentenmodelle kalibriert werden, da alle Agentenmodelle innerhalb einer Gruppe die gleichen Parameterwerte erhalten.

Sind die minimal benötigte Gruppenanzahl und die Heterogenitätslokalisierung jedoch nicht bekannt, so muss zur Einschränkung des Suchraumes zusätzlich zum eigentlichen Kalibrierungsproblem  $KP_{ABS}$  ein Heterogenitätsproblem  $HP$  zur Ermittlung der Zuordnungsfunktion  $H_g$  gelöst werden. Im Folgenden wird zunächst die entwickelte Lösungsidee für das Heterogenitätsproblem beschrieben und die benötigten Problemangaben abgeleitet. Auf dieser Grundlage wird dann das Heterogenitätsproblem genauer beschrieben.

### 10.1.2.1 Beispiel: Heterogenitätsproblem in der Fallstudie

In der Fallstudie zur Agenten-basierten Einkaufssimulation sind verschiedene Möglichkeiten zur Gruppenbildung denkbar (siehe Kapitel 7.5 und Abbildung 8.2). Als sinnvolle Möglichkeiten wurden im Projekt Gruppierungen aufgrund räumlicher Regionen, der Nähe der Agenten zueinander und die Gruppierung aufgrund individueller Agentenmerkmale betrachtet. Von diesen Möglichkeiten wurden nur die ersten beiden im Projekt näher untersucht (Ergebnisse siehe Evaluationsteil Abschnitt 16).

- **Räumliche Nähe:** Die Einkaufsagenten könnten aufgrund ihrer Nähe zueinander gruppiert werden. Diese Gruppierung folgt der Annahme, dass Agenten, deren Wohnorte nahe benachbart sind, ein ähnliches Einkaufsverhalten haben.
- **Region des Wohnortes:** Die Einkaufsagenten könnten aufgrund ihrer groben Wohnregion gruppiert werden, d.h. es gäbe z.B. Stadtagenten, Umlandagenten und Landagenten. Diese Art der Gruppierung folgt der Annahme, dass Agenten, die auf dem Land wohnen, aufgrund größerer Distanzen zu den Geschäften, ein anderes Einkaufsverhalten haben sollten, als Agenten, die in der Stadt leben.

- **Eigenschaften:** Die Einkaufsagenten könnten aufgrund persönlicher Merkmale, wie Alter oder Geschlecht gruppiert werden. Diese Gruppierung folgt der Annahme, dass das Einkaufsverhalten von Personen an ihren sozio-ökonomischen Merkmalen festgemacht werden kann.

Für jedes Gruppierungskriterium muss geklärt werden, wie viele solcher Gruppen gebildet werden sollen und welche Agenten in welche Gruppe eingeordnet werden sollten. Insbesondere an Grenzen zwischen AMG kann es für den Nutzer schwierig sein, eine geeignete Zuordnungsentscheidung vorzugeben.

## 10.2 Kalibrierung mit Heterogenitätssuche

In dieser Arbeit wird ein iteratives, automatisches Verfahren entwickelt, das den Nutzer bei der Lösung von  $HP$ , also der Klärung der beiden Aspekte „Ausmaß an Heterogenität“ und „Lokalisierung der Heterogenität“, zur Festlegung von  $H_g$  im Kalibrierungsproblem  $HP$  und der auf  $H_g$  basierenden Modellkalibrierung unterstützt.

### 10.2.1 Grundidee des Verfahrens

Müsste Heterogenitätssuche ohne irgendwelche Nutzervorgaben gestartet werden, so bliebe nichts anderes übrig, als Agentengruppen zufällig zuzuordnen, also zufällig zusammenzustellen. Zur Überprüfung, ob auf Basis der zufälligen Agentengruppen valides Simulationsverhalten entstehen kann, müsste jeweils eine Kalibrierung mit der entsprechenden Gruppierung durchgeführt werden. Die Kombination „zufällige Zuordnung“-Kalibrierung müsste so lange fortgesetzt werden, bis ausreichend valides Verhalten erreicht wurde. Ein solches Vorgehen kann sehr lang dauern und ist daher nicht befriedigend.

Der in dieser Arbeit entwickelte Lösungsansatz nimmt es dem Nutzer zwar nicht ganz ab Vorgaben an das Heterogenitätsproblem zu machen, es wird aber versucht die benötigte Genauigkeit der Vorgaben so zu gestalten, dass es dem Modellierer oder Fachexperten leichter wird, die entsprechenden Vorgaben zu spezifizieren:

- Anstelle einer konkreten Vorgabe eines Wertes für  $g$  wird lediglich ein möglicher Wertebereich gefordert. Im schlimmsten Fall muss der Wertebereich vom Modellierer geschätzt werden. Diese Schätzung hat aber eine höhere Chance den tatsächlichen Wert zu enthalten als die Schätzung eines einzelnen konkreten Wertes  $g$ .
- Anstelle einer konkreten Zuordnungsfunktion  $H_g$ , die jedes Agentenmodell genau einer Gruppe zuordnet, wird gefordert, dass der Nutzer eine Ähnlichkeitsfunktion  $\Omega$  spezifiziert.  $\Omega$  ist keine konkrete Zuordnung einzelner Agenten zu Gruppen, sondern lediglich ein Ähnlichkeitsmaß zum Vergleich zweier Agenten unabhängig von ihren Parameterwerten und sonstigen von den Parametern abhängigen Variablenwerten, die erst während eines Simulationslaufes ermittelt werden können. Aus diesem Grund wird  $\Omega$  vor oder nach, aber nicht während einer Simulation berechnet. Der Vorteil von  $\Omega$  ist zum einen, dass  $\Omega$  unabhängig von der Wahl von  $g$  ist. Der Nutzer muss nicht für jeden möglichen Wert von  $g$  eine eigene Zuordnungsfunktion spezifizieren. Zum anderen muss zur Spezifikation von  $\Omega$  nicht notwendigerweise jedes einzelne Agentenmodell, das einer Gruppe zugeordnet werden soll, betrachtet werden. Bei der direkten Spezifikation von  $H_g$  wäre dies der Fall.



Es wird nun für das betroffene  $KP_{ABS}$  (siehe Kapitel 4.1) ein erweitertes Kalibrierungsproblem betrachtet: **Das Heterogenitätsproblem HP**.

$$HP = (KP_{ABS}, \{(\Omega_{AMKF_1}, W_{g_{AMKF_1}}), \dots, (\Omega_{AMKF_k}, W_{g_{AMKF_k}}), \dots, (\Omega_{AMKF_n}, W_{g_{AMKF_n}})\}) \\ , n, k \in \mathbb{N}$$

- $n$ : Die Anzahl an verschiedenen zu kalibrierenden Agentenklassenfamilien  $AMKF$  für  $KP_{ABS}$ .
- $\Omega_{AMKF_k}$ : Eine Ähnlichkeitsspezifikation zum Vergleich jeweils zweier Agentenmodelle aus einer Familie  $AMKF_k$ .
- $W_{g_{AMKF_k}}$ : Ein Wertebereich für mögliche Werte von  $g_{AMKF_k}$  für  $AMKF_k$ .

### 10.2.1.1 Ähnlichkeitsfunktion $\Omega$ :

Der Nutzer formalisiert die Ähnlichkeit  $\Omega$  zwischen einzelnen Agenten  $Agent_i$  und  $Agent_j$  einer  $AMKF$  hinsichtlich ihrer Parameterwerte durch die Vorgabe der Ähnlichkeitsfunktion  $\Omega$ .  $\Omega$  bildet die Ähnlichkeit der Agenten auf einen numerischen Wert ab:

$$\Omega : (Agent_i, Agent_j) \rightarrow \mathfrak{R}$$

Wie der Ähnlichkeitswert konkret berechnet wird, ist Problem-spezifisch und liegt in der Hand des Nutzers.  $\Omega$  könnte die räumliche Distanz der Wohnorte der Agenten oder die Unterschiedlichkeit der Agenten hinsichtlich bestimmter Merkmale messen.

### 10.2.1.2 Gruppenspezifikation $W_g$ :

Im Folgenden wird zwischen Heterogenitätsstufen für  $AMKF$  und Heterogenitätsstufen für Kalibrierungsprobleme  $HP$  unterschieden.

Eine **Heterogenitätsstufe  $g$  für eine  $AMKF$**  beschreibt wie viele Gruppen von Agentenmodellfamilien innerhalb einer  $AMKF$  bezüglich der zugehörigen Ähnlichkeitsfunktion  $\Omega$  gebildet werden sollen. Für jede  $AMKF$ , für die eine Ähnlichkeitsfunktion spezifiziert wurde, werden eigene zugehörige Heterogenitätsstufen spezifiziert.

Wertebereiche für Heterogenitätsstufen einzelner  $AMKF$  können in dieser Arbeit auf zwei Arten festgelegt werden:

- **$W_g$  als Tupel von Intervallen**  $[MinGruppen; MaxGruppen]$  : Durch die Angabe eines Intervalls kann festgelegt werden, welche unterschiedlichen Anzahlen an Gruppen für eine Agentenklassenfamilie getestet werden sollen. *MinGruppen* beschreibt dabei die kleinste und *MaxGruppen* die größte Anzahl an Gruppen, die untersucht werden. Für zwei  $AMKF$  A und B könnte eine solche Vorgabe z.B.  $([1; 3], [1; 2])$  lauten. In diesem Fall würden für Klasse A ein bis drei Parametergruppen und für Klasse B ein bis zwei Parametergruppen gebildet und in verschiedenen Kombinationen getestet. Ausgeschrieben würden die Menge der Tupel aller möglichen Kombinationen für das Beispiel  $\{(1, 1), (1, 2), (2, 1), (2, 2), (3, 1), (3, 2)\}$  lauten. Diese ausgeschriebene Schreibweise wird im Folgenden als die Gruppenspezifikation  $W_g$  bezeichnet.

- **$W_g$  als konkrete Vorgaben:** An Stelle eines Intervalls können auch konkrete Vorgaben über Heterogenitätsstufen unterschiedlicher Agentenklassenfamilien gemacht werden, die kombiniert untersucht werden sollen. Für zwei AMKF A und B könnte eine solche Vorgabe z.B. (2, 5) lauten. In diesem Fall würden zwei Parametergruppen für Klasse A und fünf für Klasse B gebildet.

Als **Heterogenitätsstufe**  $g_{\text{Gesamt}} = \text{HET}(\text{KP}_{\text{ABS}}) \in \mathbb{N}$  eines **Kalibrierungsproblems**  $\text{KP}_{\text{ABS}}$  wird die Gesamtzahl aller AMGF der einzelnen AMKF in  $\text{KP}_{\text{ABS}}$  definiert.

Eine Kalibrierungsproblem  $\text{KP}_{\text{ABS}1}$  hat eine geringere Heterogenitätsstufe als ein  $\text{KP}_{\text{ABS}2}$ , d.h.  $\text{HET}(\text{KP}_{\text{ABS}1}) < \text{HET}(\text{KP}_{\text{ABS}2})$ , wenn gilt: Für alle AMKF aus  $\text{KP}_{\text{ABS}1}$  gilt, dass AMKF in höchstens genau so viele AMGF zerlegt ist wie in  $\text{KP}_{\text{ABS}2}$  und für mindestens eine AMKF gilt, dass AMKF in weniger AMGF zerlegt ist als in  $\text{KP}_{\text{ABS}2}$ .

### 10.2.2 Vorgehen: Kalibrierung mit Heterogenitätssuche

Das generische Vorgehen bei der Kalibrierung mit Heterogenitätssuche ist in folgendem Pseudocode dargestellt:

---

#### Algorithm 2: Pseudocode: Heterogenitätssuche

---

**Input:**  $\text{KP}_{\text{ABS}}$  und Menge von  $(\Omega_{\text{AMKF}_i}, W_{g_{\text{AMKF}_i}})$  für zu kalibrierende  $\text{AMKF}_i$

**Output:** Gruppierung  $G_{\text{best}}$  minimaler Heterogenitätsstufe und zugehörige Parameterwerte  $PW_{\text{best}}$

**Generiere** Gruppierungen  $G = \{G_1, \dots, G_j, \dots, G_m\}$  über alle AMKF durch ein Clusterverfahren auf Basis der  $(\Omega_{\text{AMKF}_i}, W_{g_{\text{AMKF}_i}})$ ;

**Generiere** aus  $\text{KP}_{\text{ABS}}$  und  $G$  die Menge

$\text{KP}_H = \{\text{KP}_{\text{ABS}1}, \dots, \text{KP}_{\text{ABS}j}, \dots, \text{KP}_{\text{ABS}m}\}$ , wobei in  $\text{KP}_{\text{ABS}j}$  jeweils die Agentenmodellfamilien gemäß  $G_j$  gruppiert sind;

**(Optional) Sortiere** die Menge  $\text{KP}_H = \{\text{KP}_{\text{ABS}1}, \dots, \text{KP}_{\text{ABS}j}, \dots, \text{KP}_{\text{ABS}m}\}$  nach aufsteigender Heterogenitätsstufe;

**while**  $\text{KP}_H$  enthält noch zu testende  $\text{KP}_{\text{ABS}j}$  **do**

    Wähle nächstes zu testendes  $\text{KP}_{\text{ABS}j}$ ;

    Führe automatische Kalibrierung zur Lösung von  $\text{KP}_{\text{ABS}j}$  durch und erhalte Parameterwerte  $PW_{G_j}$ ;

**if** Gesamtvalidität der ABS mit  $G_j$  und  $PW_{G_j}$  höher als bisheriges  $G_{\text{best}}$  und  $PW_{\text{best}}$

**then**

            Setze  $G_{\text{best}} = G_j$  und  $PW_{\text{best}} = PW_{G_j}$ ;

**end**

    (Optional) Prüfe, ob verbleibende noch zu testende Gruppierungskombinationen in  $G$  auf Grundlage der bereits getesteten  $G_j$  ausgefiltert werden können;

**end**

Gib beste bekannte Gruppierungskombination  $G_{\text{best}}$  und zugehörige Parameterwerte  $PW_{\text{best}}$  zurück;

---

### 10.2.3 Automatische Gruppengenerierung:

Im ersten Schritt werden aus den Eingaben alle möglichen relevanten Gruppierungskombinationen zu den zu kalibrierenden  $AMKF$  ermittelt. Durch die Vorgaben von  $\Omega$  und  $W_g$  kann für jede  $AMKF$  eine Menge möglicher  $H_g$  erzeugt werden. Für eine  $AMKF$  werden dabei die Agentenmodelle nicht zufällig in Gruppen sortiert. Die Agentenmodelle der  $AMKF$  werden auf Basis der für  $AMKF$  spezifizierten Ähnlichkeitsfunktion  $\Omega$  miteinander verglichen und dann aufgrund der berechneten Ähnlichkeitswerte in Gruppen eingeteilt. Die Gruppierung wird in zwei Schritten durchgeführt. Zunächst wird ermittelt werden welche Kombinationen von Gruppen untersucht werden müssen. Dann wird für jede Kombination von Gruppen eine Clusterung der Agentenmodelle durchgeführt und so die zu kalibrierenden  $IAMF$  einer Gruppe zugeordnet. Als Ergebnis erhält man eine Menge von Gruppierungen, die kalibriert und miteinander verglichen werden.

#### 1. Bestimmung benötigter unterschiedlicher Gruppenzahlkombinationen:

Wurden vom Nutzer konkrete Gruppenzahlkombinationen spezifiziert, wird dieser Schritt übersprungen.

Wurden vom Nutzer Intervalle für  $W_g$  spezifiziert, werden in diesem Schritt alle daraus resultierenden möglichen Kombinationen von Heterogenitätsstufen über alle zu kalibrierenden  $AMKF$  erzeugt. Als Ergebnis dieses Schrittes erhält man die Menge der Gruppenzahlkombinationen  $M = \{M_1, \dots, M_i, \dots, M_m\}$  und  $M_i = \{(g_{AMKF_{1_i}}, g_{AMKF_{2_i}}, \dots, g_{AMKF_{n_i}})\}$  mit  $i, n, m \in \mathbb{N}$ .  $M$  besteht also aus allen  $m$  zu untersuchenden Gruppenzahlkombinationen für die zu kalibrierenden  $AMKF$ . Wurde beispielsweise für eine Agentenklassenfamilie A ein Intervall  $[1; 2]$  und für eine Klassenfamilie B ein Intervall  $[1; 3]$  spezifiziert, werden 6 zu testenden Gruppenzahlkombinationen erzeugt, nämlich  $\{(1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 3)\}$

#### 2. Automatische Ermittlung der Gruppierungen:

Im zweiten Schritt werden aus den Gruppenzahlkombinationen in  $M$  die zugehörigen Gruppierungen ermittelt. Zur Durchführung wird für jedes  $M_i$  für jede  $AMKF_n$  für den zugehörigen Wert  $g_{AMKF_{n_i}}$  auf der Basis von  $\Omega_n$  ein Clusterverfahren [66] zur Gruppierung der Agentenmodelle angewendet. Die so ermittelte Zuordnung zerlegt  $AMKF_n$  in Gruppen. Bei mehreren zu kalibrierenden  $AMKF$  wird jede  $AMKF_n$  durch ein eigenes Clusterverfahren in Gruppen zerlegt. Die Kombinationen der erzeugten Gruppen der verschiedenen  $AMKF$  für ein  $M_i$  bildet jeweils eine zu testende Gesamtgruppierung  $G_i$  für  $KP_{ABS}$ . In obigem Beispiel würden 6 Gruppierungen  $G_1$  bis  $G_6$  ermittelt. Dabei enthielte z.B.  $G_3$  für  $AMKF$  A nur eine Gruppe (nämlich alle zu A gehörenden  $IAMF$ ) und für B drei Gruppen, die B gemäß des Ergebnisses des Clusterverfahrens partitionieren.

Im DAVINCI-Kalibrierer wurde dieses Verfahren mit Hilfe des Clusterverfahrens PAM von Kaufman und Rousseeuw [69] (Partitioning Around Medoids) und Standardparametereinstellung umgesetzt. Dieses Verfahren ist eine Variante des Clusterverfahrens k-Medoids (siehe Abschnitt 9.4.7), bei der vor der Zuordnung der Elemente zu den Clustern ein zusätzlicher Vertauschungsschritt für die Clusterzentren durchgeführt wird. Es kann jedoch auch ein anderes Clusterverfahren zur Gruppierung zum Einsatz kommen.

Als Ergebnis der automatischen Gruppengenerierung erhält man eine zu den Gruppierungskombinationen in  $M$  korrespondierende Menge zu testender Gruppierungen  $G = \{G_1, \dots, G_j, \dots, G_m\}$ . Zur Bewertung jeder Gruppierung  $G_j$  muss ein eigenes Kalibrierungsproblem auf Basis der Einteilung  $G_j$  gelöst werden. Damit muss zur Lösung des Gesamtproblems die Menge  $KP_H = \{KP_{ABS1} \dots KP_{ABS_m}\}$  von Kalibrierungsproblemen gelöst werden, für die jeweils die Zuordnung der  $IAMF$  zu Gruppen festgelegt sind. Die Lösung des Heterogenitätsproblems  $HP$  mit Heterogenitätssuche ist dann diejenige Lösung eines Kalibrierungsproblems in  $KP_H$ , welche bezüglich der Validitätskriterien die maximale Validität aufweist.

#### 10.2.4 Kalibrierung und Filterung

Der nächste Schritt besteht in der Lösung der Kalibrierungsprobleme  $KP_{ABS1}$  bis  $KP_{ABS_m}$  und der Auswahl der Lösung, die den höchsten Validitätsgrad aufweist. D.h. für jede generierte Gruppierungskonstellation wird ein eigener Kalibrierungslauf durchgeführt. Das Ergebnis der Gesamtkalibrierung ist die Gruppierungskonstellation  $G_{best}$  mit Ergebnisparametereinstellung  $PW_{best}$ , welche zum validesten Gesamtverhalten führt. Nach diesem Schritt ist die Kalibrierung beendet.

Für die Kalibrierung gibt es die folgenden Möglichkeiten:

##### 10.2.4.1 Möglichkeit 1: Vollständige Kalibrierung aller $KP$

Die offensichtliche Lösung von  $HP$  besteht darin, zunächst alle generierten Kalibrierungsprobleme  $KP_{ABS1}$  bis  $KP_{ABS_m}$  zu lösen und dann die Parametereinstellung, die über alle gelösten Kalibrierungsprobleme das valideste Verhalten erzeugen konnte, als Gesamtlösung zu verwenden. Dabei bestimmt sich der Gesamtaufwand zur Lösung von  $HP$  aus der Summe der Einzelaufwände der generierten Kalibrierungsprobleme. Eine vollständige Kalibrierung aller  $KP_{ABS1}$  bis  $KP_{ABS_m}$  ist nur dann attraktiv, wenn die Anzahl insgesamt benötigter Simulationsläufe zur Kalibrierung von  $KP_{ABS1}$  bis  $KP_{ABS_m}$  geringer ist als die bei der Kalibrierung des ursprünglichen  $KP_{ABS}$  benötigte Anzahl. Ist diese Bedingung nicht erfüllt, ist es effizienter, einen Kalibrierungslauf mit vollständiger Heterogenität in den Parameterwerten der Agenten durchzuführen, bei dem sich die richtige Heterogenität automatisch einstellen kann.

##### 10.2.4.2 Möglichkeit 2: Kalibrierung mit Abbruchkriterium

Um nicht alle möglichen generierten Gruppierungskonstellationen kalibrieren zu müssen, können auf der Basis bereits erzeugter Kalibrierungsergebnisse bestimmte Konstellationen ausgeschlossen werden. Hierzu werden zwei Abbruchkriterien zur Ausfilterung verbleibender zu testender Gruppierungen vorgeschlagen: Der Abbruch aufgrund eines Kriterium, das auf einer Monotonieannahme hinsichtlich des Zusammenhangs zwischen Heterogenitätssteigerung und Validitätsverbesserung beruht und ein Kriterium, das einen vorgegeben Schwellwerte für den zu erreichenden Validitätsgrad verwendet. Die beiden Kriterien werden im Folgenden beschrieben.

### Abbruchkriterium mit Monotonieannahme

Zum Verständnis der Verwendung des Abbruchkriteriums wird der Begriff der **irrelevanten Heterogenitätsstufe** eingeführt: Eine Heterogenitätsstufe ist irrelevant, wenn sie mehr Heterogenität in die Parameterwerte der Agenten bringt, aber keine Verbesserung der Validität des Simulationsverhalten erreicht werden kann.

Ein  $KP_{ABS_{n+1}}$  mit  $HET(KP_{ABS_n}) < HET(KP_{ABS_{n+1}})$  hat bezüglich  $KP_{ABS_n}$  eine relevante Heterogenitätsstufe, wenn gilt:

$$Valid(KP_n) < Valid(KP_{n+1}),$$

- mit  $Valid(KP)$ : Validität der Lösung von KP bezüglich zugehöriger VK.

Sonst hat  $KP_{ABS_{n+1}}$  bezüglich  $KP_{ABS_n}$  eine irrelevante Heterogenitätsstufe.

Das Abbruchkriterium basiert auf den folgenden Grundannahmen:

- **Die gesuchte Lösung ist diejenige mit der besten Verhaltensvalidität und der geringsten Heterogenitätsstufe.**
- **Strenge Monotonieannahme:** Für alle relevanten  $KP_1, \dots, KP_n, n \in \mathbb{N}$  und  $i, j, k \in [1; n]$  gilt:
  - $HET(KP_i) < HET(KP_j) \rightarrow Valid(KP_i) < Valid(KP_j)$

Wäre die Monotonieannahme nicht streng, so könnte nicht direkt abgebrochen werden, wenn das valideste Verhalten bereits gefunden wurde, aber für größere Gruppenzahlen genauso valides Verhalten erzeugt wird.

Zusammenfassend besteht die Grundannahme für die Verwendung des Abbruchkriteriums darin, dass angenommen wird, dass für alle relevanten Heterogenitätsstufen mit einer Erhöhung der Heterogenität in den Parameterwerten auch ein Anstieg der Validität des Gesamtverhaltens verbunden ist.

Das Kriterium kann auf zwei Arten eingesetzt werden:

- Wenn für  $HET(KP_k) < HET(KP_{k+1})$  kalibriert wurde und dabei  $Valid(KP_{k+1}) \leq Valid(KP_k)$  muss  $KP_{k+2}$  nicht mehr weiter kalibriert werden (unter der gemachten Monotonie-Annahme).
- Wenn das Kalibrierungsergebnis von  $HET(KP_{k+1})$  das gleiche Ergebnis wie  $HET(KP_k)$  liefert, muss nicht mehr weiter kalibriert werden. D.h. in diesem Fall gibt es zwar mehr Gruppen, aber die resultierenden Parameterwerte zeigen, dass die Parameterwerte wie im Falle der geringeren Gruppenzahl kalibriert wurden. Dies ist z.B. dadurch überprüfbar, dass für  $HET(KP_{k+1})$  und  $HET(KP_k)$  auf den Agentenmodellen Gruppierungen bezüglich der kalibrierten Parameterwerte gebildet werden. Diese Gruppierungen werden dann miteinander auf ihre Ähnlichkeit hin verglichen.

**Die gemachte Monotonieannahme für das Abbruchkriterium ist grundsätzlich nur eine Heuristik, die nicht notwendigerweise immer Gültigkeit besitzt.** Das Abbruchkriterium wurde erfolgreich erprobt (siehe Kapitel 16). Die Gefahr, die bei der Verwendung dieser

Annahme entsteht ist, dass der Suchprozess zu früh abgebrochen wird. Es ist z.B. denkbar, dass für eine Gruppenanzahl  $n$  das bis dahin beste Simulationsverhalten gefunden wird, sich aufgrund ungeeigneter Gruppierungen für  $n + 1$ ,  $n + 2$  und  $n + 3$  Gruppen schlechtere Validitätswerte ergeben, aber für  $n + 4$  Gruppen wieder ein valideres Verhalten ermittelt werden kann. In diesem Fall ist die Annahme falsch.

Es gibt verschiedene Heuristiken, die angewendet werden können, wenn die Annahme nicht getroffen werden kann. Z.B.:

- Können nach dem Entdecken der bisher validesten Gruppierungsanzahl  $x \in \mathbb{N}$  immer noch die nächsten  $x + 1$ ,  $x + 2, \dots, x + i$  Gruppierungsanzahlen betrachtet werden, bevor abgebrochen wird.
- Können nach dem Entdecken der bisher validesten Gruppierungsanzahl  $x \in \mathbb{N}$  immer noch bis zur nächsten  $x * 2$  Gruppierungsanzahl weiter Gruppierungen betrachtet werden. In diesem Fall würde die Annahme gemacht, dass mindestens bis zur theoretischen gleichmäßigen Aufteilung der jetzigen Gruppen gesucht werden muss (von der Anzahl der Gruppen her betrachtet).

Bei der Wahl des Abbruchkriteriums muss der Nutzer einen Trade/Off zwischen Sicherheit der Lösung und Anzahl an benötigten Kalibrierungsläufen machen.

### **Abbruchkriterium mit Schwellwert**

Wird die Kalibrierung bezüglich einer GZF durchgeführt, kann der Nutzer wie bei jeder anderen Kalibrierung einen Schwellwert angeben, ab dem er die Validität des Simulationsverhaltens als „ausreichend“ einstuft. Die Kalibrierung mit Heterogenitätssuche wird dann abgebrochen, wenn die Bewertung des Validitätsgrades für eine kalibriertes Modell zu einer bestimmten Heterogenitätsstufe über dem vorgegebenen Schwellwert liegt.

Die beiden vorgeschlagenen alternativen Abbruchkriterien werden im Evaluationsteil (siehe Kapitel 16) näher untersucht.

### **10.2.5 Anforderungen und Nutzen des Verfahrens**

Nach dem in den vorangegangenen Abschnitten die Funktionsweise des Kalibrierungsverfahrens mit Heterogenitätssuche beschrieben wurde, wird in diesem Abschnitt das Verfahren aus dem Blickwinkel des Ausgangsproblems bewertet und Anforderungen für seinen Einsatz beschrieben:

Die Heterogenitätssuche bietet dem Nutzer einen Mittelweg zwischen Kalibrierung im optimalen Kalibrierungsfall (vollständiges Wissen über benötigte Heterogenität in den Parameterwerten) und dem schlechtesten Fall (es muss „blind“ mit vollständiger Heterogenität kalibriert werden. Damit die Heterogenitätssuche durchgeführt werden kann muss die folgende Anforderung erfüllt sein:

Es wird gefordert, dass der Nutzer Wissen oder Vermutungen hat, wodurch die Heterogenität in den Parameterwerten bedingt wird. Dieses Wissen muss der Nutzer in ein Ähnlichkeitskriterium formalisieren können (Ähnlich einem Ähnlichkeitskriterium für

Clusterverfahren). Es wird also nicht gefordert, dass dem Nutzer die benötigte Heterogenität in den Parameterwerten bekannt ist.

Die Heterogenitätssuche stellt damit geringere Anforderungen als für den optimalen Kalibrierungsfall notwendig wären. **Der Nutzer muss nicht wissen wie viele Gruppen von Agenten mit homogenen Parameterwerten es geben und welcher Gruppe welcher Agent zugeordnet werden muss.**

Der Gewinn für den Anwender besteht somit darin, dass auf der Basis des Ähnlichkeitskriteriums automatisch nach der besten geeigneten Heterogenitätsverteilung und der zugehörigen Parametereinstellung gesucht wird. Es ist allerdings nicht trivial das benötigte Ähnlichkeitskriterium zu spezifizieren. Wird ein falsches Kriterium gewählt, kann die Kalibrierung nicht erfolgreich durchgeführt werden.

Da bei der Heterogenitätssuche iterativ nach einer geeigneten Heterogenitätseinstellung gesucht wird, müssen immer wieder Kalibrierungen durchgeführt werden. Hierzu stellt sich die Frage, ob der entstehende Gesamtaufwand (= benötigte Zahl an Simulationsläufen) nicht größer wird als der Aufwand direkt eine Kalibrierung mit vollständiger Heterogenität durchzuführen. Die Antwort auf diese Frage hängt vom jeweiligen zu lösenden Kalibrierungsproblem ab. Die Heterogenitätssuche bietet dem Nutzer mehr Vorteil bezüglich des Gesamtaufwandes, zu je größeren Gruppen mit gleichen Parameterwerten die Agentenmodelle zusammengefasst werden können. Um genauere Anhaltspunkte für eine Aussage zum Aufwand der Heterogenitätssuche zu erhalten wird im Evaluationsenteil der Arbeit (siehe Abschnitt 16) eine Vergleichsuntersuchung an einem künstlichen Beispielmmodell durchgeführt. In der Untersuchung wird für verschiedene vorgegebene benötigte Heterogenitätseinstellungen der Aufwand verglichen, diese bei Kalibrierung mit vollständiger Heterogenität und bei Kalibrierung mit Heterogenitätssuche zu finden. Die Untersuchung zeigt für das künstliche Modell, dass die Heterogenitätssuche im Vergleich zur Kalibrierung mit vollständiger Heterogenität vorteilhaft ist, wenn weniger als 50% Heterogenität in den Parameterwerten benötigt wird (d.h. wenn weniger als jedes zweite Agentenmodell eigene Parameterwerte benötigt).

Neben der Tatsache, dass es die Heterogenitätssuche ermöglicht Parameterwerte mit höherer Heterogenität bei möglichst geringem Kalibrierungsaufwand zu kalibrieren ist auch die Suche nach der „benötigten“ Heterogenität an sich ein Vorteil für den Nutzer. Axtell et al. [8] beschreiben die Entwicklung eines Modells zur Bevölkerungsentwicklung der Anasazi. Zunächst wird ihr Modell durch homogene Parameterwerte gesteuert. Sie konnten aber feststellen, dass die Einführung von zusätzlicher Heterogenität in den Parameterwerten großen Einfluss auf die Validität des Modellverhaltens hatte. Im Falle von [8] wurde diese Erhöhung der Heterogenität „von Hand“ durchgeführt und dann wieder kalibriert und das Verhalten bewertet. Wäre es Axtell et al. möglich das gesuchte valide Gesamtverhalten des Modells in eine messbare Zielfunktion abzubilden, dann wäre die Anwendung der Heterogenitätssuche zur Kalibrierung des Anasazi-Modells denkbar, da die Heterogenitätssuche genau die beiden Punkte „möglichst wenig zu kalibrierende Parameterwerte“ und „Suche nach benötigter Heterogenität in den Parameterwerten“ fokussiert.

Ein Punkt, der in dieser Arbeit nicht genauer untersucht wurde, aber eine mögliche Anwendung darstellt ist, dass die Heterogenitätssuche statt zur Kalibrierung auf der Basis eines

Ähnlichkeitskriteriums zur Suche nach einem entsprechenden Ähnlichkeitskriterium selbst, bzw. zur Verifikation eines vermuteten Ähnlichkeitskriteriums eingesetzt werden könnte. In diesem Fall würde der Nutzer verschiedene Ähnlichkeitskriterien aufstellen und mittels der Heterogenitätssuche analysieren, zu wie validem Gesamtverhalten die resultierenden Gruppierungen führen können. Dabei liegt dann der Hauptnutzen der Heterogenitätssuche in der Möglichkeit mit ihr Zusammenhänge im Modell analysieren zu können. Im Beispiel der Projektes zur Fallstudie der Agenten-basierten Einkaufssimulation wurden z.B. die Ähnlichkeitskriterien „räumliche Nähe“ und „regionale Zugehörigkeit“ miteinander verglichen.

#### 10.2.5.1 Identifikation relevanter Ähnlichkeitsfunktionen und Gruppennzahlen

Die Festlegung von Ähnlichkeitsfunktionen  $\Omega$  und die Auswahl zu untersuchender Gruppennzahlen in  $W_g$  sind wichtige Vorgaben für die Kalibrierung mit Heterogenitätssuche. Eine Möglichkeit Ähnlichkeitsfunktionen zu entwickeln besteht für den Nutzer z.B. darin, mit Hilfe von Clusterverfahren die Agenten auf der Grundlage einer möglichen Ähnlichkeitsfunktion zu gruppieren und das Ergebnis des Clusterverfahrens zu visualisieren. Auf diese Weise kann sich der Nutzer bereits einen Eindruck verschaffen, wie plausibel ihm die in der Kalibrierung verwendeten Gruppierungen erscheinen. Eine ermittelte Ähnlichkeitsfunktion kann dann bei der Heterogenitätssuche verwendet werden.

#### 10.2.5.2 Zusammenfassung

- Die Heterogenitätssuche sucht automatisch auf Grundlage einer Ähnlichkeitsfunktion und Validierungskriterien nach „benötigter“ Heterogenität im Modell zur Erreichung möglichst validen Gesamtverhaltens.
- Die Heterogenitätssuche kann dann weniger Kalibrierungsaufwand (= Simulationsläufe) benötigen als eine Kalibrierung mit vollständig heterogenen Agentenparametern, wenn die simulierten Agentenmodelle zu möglichst großen Gruppen mit homogenen Parameterwerten zusammengefasst werden können.
- Die Heterogenitätssuche stellt schwächere Anforderungen als zur Kalibrierung mit optimaler Heterogenitätsverteilung auf den Parameterwerten der Agenten benötigt würden.
- Die Heterogenitätssuche kann zur Analyse von Modellzusammenhängen eingesetzt werden.

Das Verfahren zur kombinierten Kalibrierung und Heterogenitätssuche bietet dem Nutzer einen Kompromiss zwischen dem Interesse, in möglichst kleinen Parametersuchräumen zu kalibrieren und der Notwendigkeit von Heterogenität in den Parameterwerten der simulierten Agenten. Der Nutzen des Verfahrens hängt von der Struktur des behandelten Kalibrierungsproblems und der (unbekannten) tatsächlich benötigten Menge an Heterogenität in den Parameterwerten ab.



## 10.3 Einbettung in das Makro-Mikro-Verfahren

Auch im Makro-Mikro-Verfahren hat die verwendete Heterogenität in den Parameterwerten der simulierten Agenten entscheidenden Einfluss auf die Schwierigkeit des Kalibrierungsproblems. Bisher war der Ausgangspunkt des Makro-Mikro-Verfahrens die vollständige Heterogenität der Agenten. Dies kann durch Verwendung der Kalibrierung mit Heterogenitätssuche im Makro-Mikro-Verfahren verbessert werden. Dabei kann die Kalibrierung mit Heterogenitätssuche zur Unterstützung von Heterogenitätsfestlegungen auf der **Makro-Ebene** eingesetzt werden: Die *IVMF* werden auf der Makro-Ebene zu *IVMGF* (ideale Verhaltensmodellgruppenfamilien) gruppiert, für welche identische *IVM* gesucht werden.

Auch eine Kalibrierung mit Heterogenitätssuche zur Unterstützung von Heterogenitätsfestlegungen auf der **Mikro-Ebene** kann sinnvoll sein: Kann beispielsweise bei der Kalibrierung im Makro-Schritt nur ein gemeinsames *IVM* für alle Agenten ermittelt werden, dann kann es auf der Mikro-Ebene immer noch notwendig sein, heterogene Parameterwerte für die Agenten zu ermitteln, damit diese trotz ihrer jeweiligen individuell unterschiedlichen lokalen Randbedingungen das gleiche Verhalten produzieren.

### 10.3.1 Übergeordnete Heterogenitätssuche

Die Heterogenitätssuche kann als übergeordnetes Verfahren zum Makro-Mikro-Verfahren eingesetzt werden. Der Makro-Schritt ersetzt dabei den Kalibrierungsschritt der Heterogenitätssuche. In diesem Fall werden bezüglich der generierten Heterogenitätseinstellungen identische *IVM* für die jeweiligen Agentengruppierungen kalibriert. Auf der Mikro-Ebene werden die *IAMF* bezüglich der zu den *IVM* gehörigen *IZF* ihrer jeweiligen Gruppe kalibriert. Es können aber heterogene Parameterwerte innerhalb der generierten Agentengruppen ermittelt werden, wenn durch nur auf der Mikro-Ebene vorhandene Randbedingungen unterschiedliche Parameterwerte für die Agenten benötigt werden, damit gleiches aggregiertes Verhalten entstehen kann.

Eine zweite Möglichkeit die Heterogenitätssuche als übergeordnetes Verfahren einzusetzen, wird in Abbildung 10.1 dargestellt. Hier wird die Heterogenitätssuche mit einem Makro-Mikro-Verfahren mit Reverse-Engineering kombiniert. Dabei werden auf der Makro-Ebene Gruppierungen für die Agenten generiert, welche auf allen Kalibrierungsebenen eingesetzt werden. Es werden *IRB* und zugehörige *IVMF* bezüglich der generierten Gruppierungen gesucht, für die Gruppierungen im RE-Schritt *IVM* und zugehörige *IZF* ermittelt und auf der Mikro-Ebene innerhalb der Gruppierungen homogene Parameter kalibriert. Die Kalibrierung im RE-Schritt ersetzt also in diesem Fall den Kalibrierungsschritt der Heterogenitätssuche. Damit wird in jedem Schritt der Heterogenitätssuche immer zunächst der nächste Schritt der Heterogenitätsfestlegung ausgeführt und dann die darunter angeordneten Suchverfahren.

### 10.3.2 Untergeordnete Heterogenitätssuche

Die Heterogenitätssuche kann auch auf der Mikro-Ebene des Makro-Mikro-Verfahrens eingesetzt werden. Dies ist ratsam wenn:

- Bei der Ermittlung der *IVM* auf der Makro-Ebene aufgrund der Komplexität der Agenteninteraktionen im schlechtesten Fall nur ein *IVM* für das gesamte Mikro-

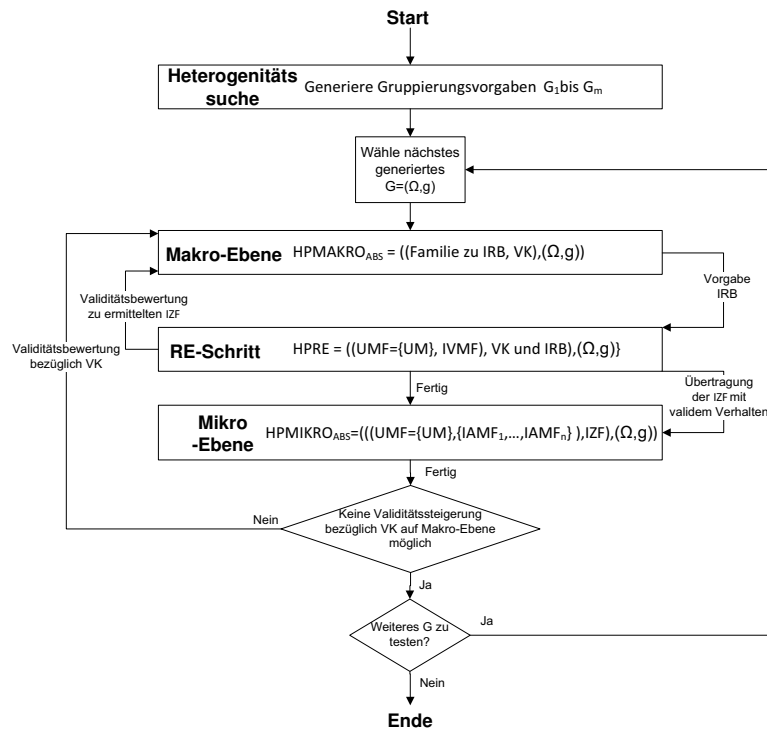


Abbildung 10.1: Makro-Mikro-Verfahren mit Reverse-Engineering. Die Heterogenitätssuche ist dem Verfahren übergeordnet und betrifft alle Kalibrierungsebenen.

Modell festgelegt werden konnte. Das gleiche ist der Fall, wenn zwar verschiedene *IVM* gebildet werden konnten, diese aber nicht unabhängig voneinander simuliert werden können, so dass keine Zerlegung des Kalibrierungsproblems auf der Mikro-Ebene möglich ist.

- Auf der Mikro-Ebene dennoch potentiell heterogene Agentenparameter kalibriert werden müssen, dies aber aufgrund der hohen Parameterzahl mit Black-Box-Verfahren nicht möglich ist.

Im beschriebenen Fall ist das Agentenverhalten so komplex, dass das Mikro-Kalibrierungsproblem nicht zerlegt werden kann und der Nutzer auf der Mikro-Ebene vor dem gleichen großen Parametersuchraum steht wie bei Black-Box-Kalibrierung des Ausgangsmodells. Der Einsatz der Heterogenitätssuche ist für den Nutzer dabei immer noch vorteilhaft. Die ermittelten *IVM* werden als Randbedingungen für die Kalibrierung mit Heterogenitätssuche verwendet. Die gleiche Einsatzmöglichkeit für die Heterogenitätssuche bietet sich auch ohne Makro-Mikro-Verfahren in direkter Kombination mit Black-Box-Verfahren auf der Mikro-Ebene.

**Teil III**

**Werkzeuge**



# Kapitel 11

## Der DAVINCI-Kalibrierer: Technik

Der DAVINCI-Kalibrierer (Developing Agent-based Simulations Via INtelligent CalIbration) ist ein Werkzeug zur Kalibrierung Agenten-basierter Simulationen und setzt die im theoretischen Teil der Arbeit beschriebenen Kalibrierungstechniken (siehe Kapitel 9 bis 10) und einige zusätzliche Techniken, die jedoch nicht im Fokus dieser Arbeit standen, um: DAVINCI kann dazu genutzt werden ABS im Makro-Mikro-Verfahren zu kalibrieren. Weiter integriert der DAVINCI-Kalibrierer Techniken zur Durchführung von Black-Box, wissensbasierten Suchverfahren und Verwendung von White-Box-Zerlegungen. Im Werkzeug werden dem Modellierer oder Fachexperten verschiedene Kalibrierungstechniken zur Verfügung gestellt. Diese müssen jedoch nicht alle getrennt voneinander oder alle gleichzeitig eingesetzt werden. Jede der einzelnen Techniken ist optional, so dass die Techniken vom Modellierer oder Fachexperten nach Belieben geeignet kombiniert werden können. Hierdurch erhält der Modellierer oder Fachexperte die Möglichkeit, zu einem spezifischen Kalibrierungsproblem ein spezielles Kalibrierungsverfahren zu entwerfen, das genau diejenigen Techniken einsetzt, die zur Lösung des Problems benötigt werden.

Zusätzlich zu den technischen Aspekten, die in diesem Kapitel dargelegt werden, wird in Kapitel 12 auf Techniken in DAVINCI eingegangen, die es dem Nutzer erlauben mit DAVINCI während der Laufzeit einer Kalibrierung zu interagieren.

Der DAVINCI-Kalibrierer ist ein Werkzeug zur automatischen Kalibrierung Agenten-basierter Simulationen auf der Basis formalisierbarer Validierungskriterien. Die Umsetzung des Kalibrierers erfolgte für das Simulationssystem SeSAM, das eine graphische Modellierungs- und Simulationsumgebung für Agenten-basierte Simulationen bietet.

Das weitere Kapitel ist wie folgt aufgebaut:

- In Unterabschnitt 11.1 werden kurz die zur Entwicklung von DAVINCI verwendeten Bibliotheken und das SeSAM-Simulationswerkzeug vorgestellt.
- In Unterabschnitt 11.2 werden zu den in dieser Arbeit entwickelten Kalibrierungstechniken zentrale Anwendungsfälle beschrieben und aus diesen der strukturelle Aufbau des DAVINCI-Kalibrierungswerkzeuges und seine benötigte Funktionalität abgeleitet.
- In Abschnitt 11.3 wird auf verschiedene technische Aspekte der internen Funktionsweise von DAVINCI eingegangen.

## 11.1 SeSAM und verwendete Programmierbibliotheken

### 11.1.1 Verwendete Programmierbibliotheken

Für die Umsetzung des DAVINCI Kalibrierers wurden verschiedene existierende freie Bibliotheken verwendet:

Die Clusteranalysen während der Heterogenitätssuche werden mit dem Statistikwerkzeug  $R^1$  berechnet. Die Anbindung von  $R$  erfolgt über die freie Bibliothek *JGR*<sup>2</sup> (Java GUI for R). Die Darstellung dreidimensionaler Plots von Punktmengen für die in der Analyse der bereits bewerteten Parametereinstellungen erfolgt über die Bibliothek *JMathTools*<sup>3</sup>, die unter der BSD-Lizenz steht.

### 11.1.2 SeSAM

SeSAM (Shell for Simulated Agent Systems) ist eine Modellierungs- und Simulationsumgebung für Agenten-basierte Simulationen, die am Lehrstuhl für künstliche Intelligenz und angewandte Informatik der Universität Würzburg entwickelt wird [75]. Das Werkzeug kombiniert die Konzepte einer deklarativen Modellrepräsentation auf abstrakter Ebene und visueller Programmierung. Auf diese Weise sollen Anfänger im Bereich der Modellierung und Simulation bei der Arbeit unterstützt werden. SeSAM wird erfolgreich in verschiedenen Bereichen eingesetzt. Neben der universitären Lehre gehören hierzu verschiedene interdisziplinäre Projekte, wie die Simulation sozialer Insektenstaaten oder Routenwahlmodelle in der Verkehrssimulation. Weiter bildet SeSAM die technische Grundlage für die Simulation virtueller Hochregallager, die für Tests von Steuerungssoftware und für Kundenpräsentationen eingesetzt werden.

#### 11.1.2.1 SeSAM Bestandteile

SeSAM Modelle werden zunächst deklarativ auf abstrakter Ebene entworfen, um eine Trennung zwischen dem eigentlichen Modell und dem Simulator zu ermöglichen. Das Modell besteht aus einer Reihe von Objekten: Einer Umwelt (nicht notwendigerweise räumlich explizit), Agenten und Ressourcen (alle Entitäten, die keine Agenten oder die Umwelt sind). Der Zustand eines jeden Simulationsobjektes wird durch eine Reihe von Zustandsvariablen beschrieben. Verhalten von Agenten und der Umwelt kann vom Nutzer mit Aktivitätsdiagrammen in UML spezifiziert werden. Eine Aktivität entspricht dabei einem Zustand, in dem sich ein Agent befindet. Durch Regeln kann definiert werden, wie zwischen verschiedenen Aktivitäten gewechselt wird. Die aktuelle Aktivität eines Agenten bestimmt die nächste Aktionsreihenfolge eines Agenten. Solche Aktionen können konkrete Handlungen des Agenten oder die Elemente von komplexem Entscheidungsverhalten sein. Bedingungen und Agentenaktionen können mittels sogenannter Primitive modelliert werden. Primitive sind zum einen elementare Aktionen, wie etwa das Setzen oder Auslesen eines Variablenwertes, können aber auch beliebig komplex werden (z.B. ein Primitiv zur Berechnung eines kürzesten Weges). Die von SeSAM zur Verfügung gestellte Primitivmenge ist *Turing-vollständig* und beliebig erweiterbar. Beispielsweise kann der Nutzer aus elementaren

---

<sup>1</sup><http://www.r-project.org/>

<sup>2</sup><http://rosuda.org/JGR/>

<sup>3</sup><http://jmathtools.sourceforge.net/>

Primitiven eigene komplexere Nutzerprimitive erstellen und diese dann bei der Modellierung verwenden. Der Entwurf von Agenten erfolgt in SeSAM auf Klassenebene, d.h. es werden immer Modellstrukturen für eine Klasse von Agenten entwickelt. Die Agenten in einer SeSAM-Simulation sind Instanzen solcher deklarierten Klassen, die individuelle Parameterwerte besitzen können. Startzustände für Simulationsläufe können in SeSAM durch die Deklaration von *Situationen* vorgegeben werden. Eine *Situationen* besteht aus einer konkreten Umweltkonfiguration und der Deklaration relevanter Objekt- und Agenteninstanzen mit individuellen Parameterwerten.

### **Interpreter und Simulator:**

Die deklarativ spezifizierten Modelle werden von SeSAM in eine lauffähige Simulation kompiliert. Ein Simulationslauf in SeSAM wird Zeitschritt-basiert berechnet. Parallelität in den Agentenaktionen wird dadurch simuliert, dass die Aktionen der Agenten in jedem Schritt in zufälliger Reihenfolge durchgeführt werden.

### **Kartenrepräsentation:**

In SeSAM können verschiedene Raumrepräsentationen zum Modell hinzugefügt werden. Der „Welt“ kann eine Karte und Agenten können Positionen und geometrische Informationen hinzugefügt werden. In der zugehörigen Laufzeitanimation ist es z.B. möglich, die Bewegungen der einzelnen Agenten zu verfolgen oder deren Variablenwerte zu betrachten.

### **Analysemöglichkeiten:**

SeSAM bietet dem Nutzer verschiedene Datendarstellungsmöglichkeiten. Für Analysen zur Laufzeit der Simulation werden Linien- und Balkendiagramme bereitgestellt. Eigenschaften zu konkreten Simulationszeitpunkten können mittels Analysetabellen veranschaulicht werden.

Zur Offline-Analyse des Simulationsverhaltens wird es ermöglicht aggregierte Verlaufsdaten abzuspeichern, um diese dann später mit externen Statistikwerkzeugen (z.B. Microsoft Excel<sup>4</sup> oder Statistica<sup>5</sup>) zu analysieren. Analysedaten können bereits während der Simulation in SeSAM aufbereitet werden (z.B. Mittelwertbildung).

Die Definition von Analysefunktionen erfolgt zur Modellierungszeit mittels Primitiven in ähnlicher Weise wie die Spezifikation von Agentenaktionen. Die Ergebniswerte der Analysefunktionen werden dann zur Laufzeit angezeigt bzw. abgespeichert.

### **Erweiterungsmöglichkeiten:**

Das SeSAM-System ist weitgehend frei vom Nutzer erweiterbar. Hierzu wird ein Plugin-Mechanismus bereitgestellt, über den ein Programmierer zusätzliche Fähigkeiten zum Simulationswerkzeug hinzufügen kann.

<sup>4</sup><http://office.microsoft.com/en-us/excel/microsoft-excel-spreadsheet-FX101825647.aspx> (November 2010)

<sup>5</sup><http://www.statsoft.de/> (November 2010)

## 11.2 Aufbau des DAVINCI-Kalibrierers

In diesem Abschnitt wird auf den Aufbau und die Funktionsweise des DAVINCI-Kalibrierers eingegangen. Zunächst werden zwei zentrale Anwendungsfälle für die Nutzung der in dieser Arbeit vorgestellten Techniken zur Kalibrierung von ABS dargestellt. Für jeden Anwendungsfall wird beschrieben, welche Aktionen der Nutzer auf Modellebene vornehmen muss und abgeleitet, welche Kalibrierungskomponenten von DAVINCI zur Verfügung gestellt werden müssen. Schließlich werden die identifizierten DAVINCI-Komponenten basierend auf den Anwendungsfällen zueinander in Beziehung gestellt. Dann wird aufgezeigt, wie aus den verschiedenen in DAVINCI integrierten Algorithmusbausteinen ein spezialisiertes Kalibrierungsverfahren für ein spezifisches Kalibrierungsproblem konfiguriert werden kann. Im letzten Teil wird die generelle Funktionsweise von DAVINCI während eines Kalibrierungslaufes erläutert.

### 11.2.1 Zentrale Anwendungsfälle für den DAVINCI-Kalibrierer

Damit DAVINCI zur Kalibrierung von ABS mit den in dieser Arbeit entwickelten Techniken eingesetzt werden kann orientiert sich die DAVINCI-Entwicklung an den folgenden beiden Anwendungsfällen und deren Erweiterungen:

1. Kalibrierung nur auf der Mikro-Ebene.
2. Makro-Mikro-Kalibrierung.

#### 11.2.1.1 Kalibrierung nur auf der Mikro-Ebene

Abbildung 11.1 veranschaulicht den Anwendungsfall aufgeteilt in Schritte, die der Nutzer auf Modellebene durchführt, Schritte, die der Nutzer bei der Nutzung von DAVINCI durchführen muss, und Schritte, bei denen DAVINCI die Modellkalibrierung durchführt. Ausgangspunkt ist ein ABS-Modell, das der Nutzer auf der Mikro-Ebene entworfen hat und eine Menge von  $VK$ , die zu einer globalen Zielfunktion aggregiert sind. Der Nutzer möchte dieses Modell auf der Mikro-Ebene unter Verwendung der auf der Mikro-Ebene verfügbaren Kalibrierungstechniken kalibrieren können.

#### Nutzervorgehen auf der Modellebene

- **Eingaben an die ABS-Kalibrierung:** Auf der Modellebene hat der Nutzer die ABS entworfen, die globale Zielfunktion spezifiziert. Diese Elemente sind Eingaben für den DAVINCI-Kalibrierer.
- **White-Box-Zerlegungstechniken:** Möchte der Nutzer White-Box-Zerlegungstechniken anwenden können, also zunächst Teilmodelle (entstanden aus Zerlegung oder Aggregation des Gesamtmodells) kalibrieren und deren Teilergebnisse dann zu einem Gesamtergebnis zusammenführen können, dann muss der Nutzer entsprechende Teilmodelle entwerfen, die von DAVINCI kalibriert werden können.



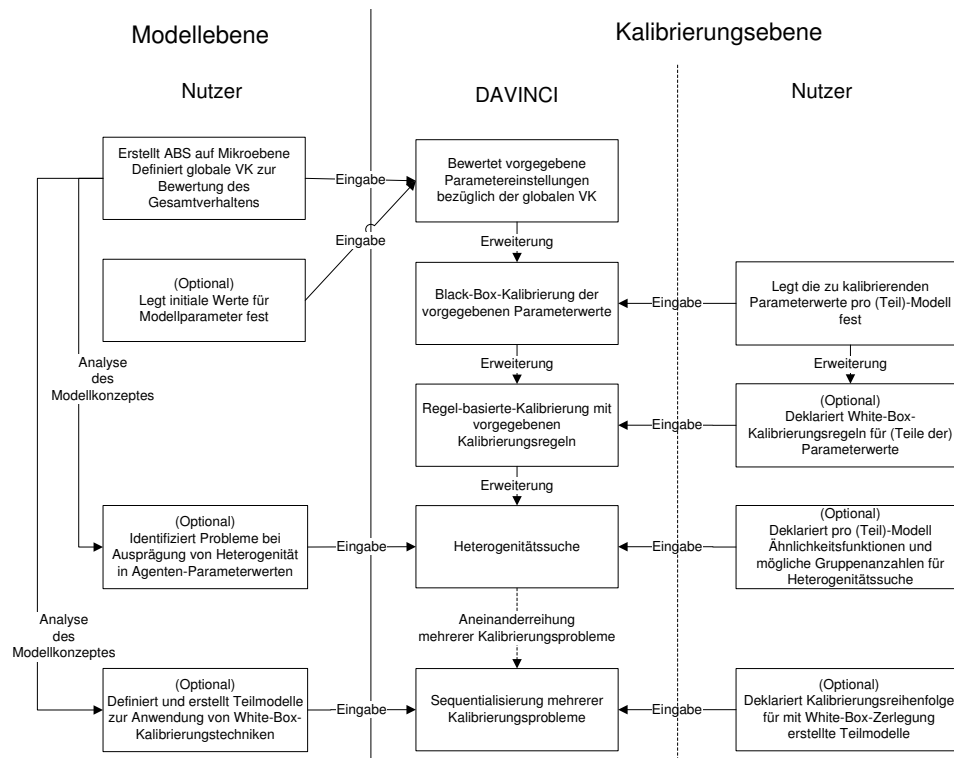


Abbildung 11.1: Anwendungsfall Kalibrierung auf der Mikro-Ebene.

### Nutzervorgehen auf der Kalibrierungsebene

- **Deklaration der Modellparameter:** Als weitere Eingaben an DAVINCI muss der Nutzer die von DAVINCI zu kalibrierenden Parameterwerte des Modells festlegen.
- **White-Box-Kalibrierungsregeln:** Möchte der Nutzer vorhandenes Wissen über Zusammenhänge zwischen Modellverhalten und benötigten Parameterwerten zur Kalibrierung der ABS nutzen können, so muss er in der Lage sein, dieses in Parameteränderungsregeln für die Kalibrierung zu spezifizieren, die einen (Teil-)Modellzustand auf eine Änderung von Parameterwerten abbilden.
- **Heterogenitätssuche:** Möchte der Nutzer Heterogenitätssuche zur Kalibrierung großer Anzahlen von Agentenparametern auf der Mikro-Ebene einsetzen, dann muss er pro relevanter Agentenklasse Ähnlichkeitsfunktionen und mögliche Gruppenanzahlen als Eingabe an DAVINCI spezifizieren.

### DAVINCI-Unterstützung auf der Kalibrierungsebene

- **Bewertung von Parametereinstellungen:** Auf der Kalibrierungsebene muss DAVINCI dem Nutzer mindestens die Möglichkeit bieten vom Nutzer vorgegebene Parameter-

einstellungen bezüglich der resultierenden ABS-Verhaltensvalidität zu bewerten. Hierzu wird eine Komponente in DAVINCI benötigt, die vorgegebene Parameterwerte in das ABS-Modell einsetzt, dieses simuliert und den zugehörigen Wert der Zielfunktion berechnet. Bei Einfluss von Zufallselementen muss die Komponente die Möglichkeit bieten für jede Parametereinstellung mehrere Bewertungsläufe durchzuführen und die Gesamtbewertung durch den Mittelwert oder Median der berechneten Einzelwerte zu ermitteln. Da es sich bei der Komponente um die grundlegendste in DAVINCI benötigte Technik zur Bewertung der Validität einer Parametereinstellung handelt wird sie im Weiteren als *Basisverfahren zur Validitätsbewertung* bezeichnet.

- **Black-Box-Kalibrierung:** Damit nicht nur einzelne vom Nutzer vorgegebene Parametereinstellungen getestet werden können, muss DAVINCI dem Nutzer zusätzlich die Möglichkeit bieten, mit Black-Box-Verfahren eine vollautomatische Kalibrierung der Parameter auf der Mikro-Ebene durchzuführen.
- **Regel-basierte-Kalibrierung:** Damit es dem Nutzer ermöglicht wird Regel-basierte Kalibrierung zur ABS-Kalibrierung einsetzen zu können, muss DAVINCI dem Nutzer die Möglichkeit bieten Kalibrierungsregeln zu spezifizieren. In diesen Kalibrierungsregeln können explizite Änderungen an Parameterwerten auf der Grundlage des Simulationszustandes bei Simulationseende vorgenommen werden. Zur Umsetzung wird eine Regel-basierte Suche für DAVINCI benötigt.
- **White-Box-Zerlegungstechniken:** Damit es dem Nutzer ermöglicht wird White-Box-Techniken zur Kalibrierung von Teilmodellen einsetzen zu können, muss DAVINCI dem Nutzer die Möglichkeit bieten verschiedene Teilmodelle getrennt zu kalibrieren und die kalibrierten Parameterwerte aus einem Teilmodell in ein anderes Teil- oder Gesamtmodell übernehmen zu können. Zur Umsetzung wird für DAVINCI die Möglichkeit zur sequenzialisierten Lösung verschiedener Kalibrierungsprobleme und Übertragung bereits ermittelter Parameterwerte benötigt.
- **Heterogenitätssuche:** Damit es dem Nutzer ermöglicht wird Heterogenitätssuche zur Kalibrierung großer Anzahlen von Agentenparametern auf der Mikro-Ebene einsetzen zu können, muss DAVINCI eine entsprechende Komponente zur Heterogenitätssuche enthalten, die mit den anderen Techniken kombiniert werden kann.

Je nach Art des konkreten Kalibrierungsproblems auf der Mikro-Ebene müssen die einzelnen Techniken separat zuschaltbar sein und der Nutzer muss jede Kalibrierungstechnik parametrisieren können.

#### 11.2.1.2 Makro-Mikro-Kalibrierung

Abbildung 11.2 veranschaulicht den Anwendungsfall der Durchführung des Makro-Mikro-Verfahrens aufgeteilt in Schritte, die der Nutzer auf Modellebene durchführt, Schritte, die der Nutzer bei der Nutzung von DAVINCI durchführen muss, und Schritte, bei denen DAVINCI die Modellkalibrierung durchführen soll.

Ausgangspunkt ist wieder ein ABS-Modell, das der Nutzer auf der Mikro-Ebene entworfen hat und eine Menge von  $VK$ , die zu einer globalen Zielfunktion aggregiert ist. Der Nutzer

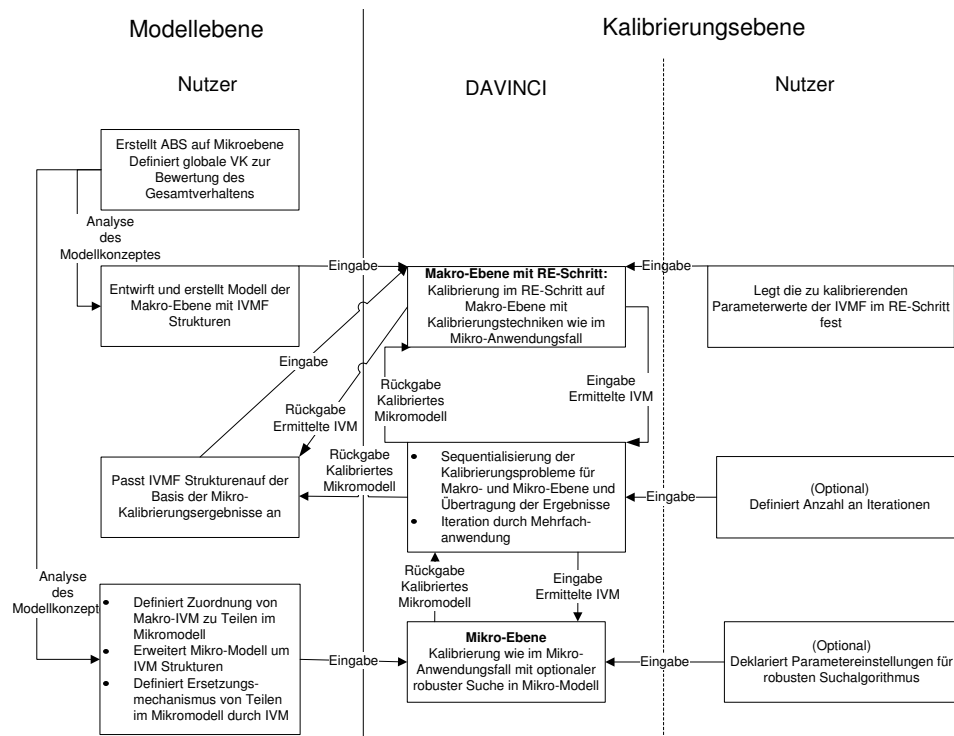


Abbildung 11.2: Anwendungsfall Makro-Mikro-Verfahren.

möchte dieses Modell im Makro-Mikro-Verfahren mit Reverse-Engineering unter Zuhilfenahme aller verfügbaren Kalibrierungstechniken kalibrieren können. Im Folgenden werden nur die Erweiterungen zu obigem Anwendungsfall beschrieben.

### Nutzervorgehen auf der Modellebene

- **Entwurf der Makro-Ebene:** Auf der Modellebene muss der Nutzer zur Anwendung des Makro-Mikro-Verfahrens zusätzlich ein Modell der Makro-Ebene erstellen, in dem das Gesamtverhalten der Simulation durch die idealen Verhaltensmodelle (*IVM* oder *IVMF*) der Agenten, oder Agentengruppen während der Simulation entsteht. Weiter muss der Nutzer eine Verknüpfung zwischen der *IVM* der Makro-Ebene und den Agentenmodellen im Mikromodell erstellen, d.h. er muss eine Zuordnung zwischen *IVM* und *IAMF* definieren und spezifizieren, wie die *IVM* in Zielfunktionen für das Mikro-Modell abgebildet werden können.
- **Einsatz robuster Kalibrierung:** Möchte der Nutzer auf der Mikro-Ebene robuste Kalibrierung nutzen, so muss er im Modell spezifizieren, wie der Einfluss einzelner Modellteile (z.B. Agentenmodelle) auf das Simulationsverhalten durch den Einfluss der zugehörigen *IVM* ersetzt werden kann. Weiter muss er im Mikro-Modell eine Möglichkeit schaffen, zwischen Simulation des tatsächlichen Mikro-Teilmodells und der Simulation des zugehörigen *IVM* durch boolesche Schalterparameter im Modell umschalten zu können.

### Nutzervorgehen auf der Kalibrierungsebene

- **Bereits beschriebene Schritte:** Für jedes Teilkalibrierungsproblem im Makro-Mikro-Verfahren muss der Nutzer die gleichen Schritte durchführen, wie bei der Kalibrierung auf der Mikro-Ebene, z.B. Deklaration zu kalibrierender Parameterwerte oder Festlegung von Algorithmeinstellungen.
- **Makro-Mikro-Iteration:** In DAVINCI muss der Nutzer die Iterationen zwischen Makro- und Mikro-Ebene explizit als Aneinanderreihung der zugehörigen Teilkalibrierungsprobleme deklarieren. Die Aneinanderreihung gleicht der Deklaration der Reihenfolge bei der sequentiellen Kalibrierung von mit White-Box-Zerlegung ermittelten Teilmodellen.
- **Robuste Suche:** Der Nutzer muss in DAVINCI die Parametereinstellungen des robusten Suchverfahrens definieren.

### DAVINCI-Unterstützung auf der Kalibrierungsebene

- **Kalibrierung auf der Makro-Ebene:** DAVINCI kann auf der Makro-Ebene globale Parameterwerte kalibrieren und das Reverse-Engineering der *IVMF* durchführen. DAVINCI unterstützt den Nutzer dabei bei der Kalibrierung jedes einzelnen Teilkalibrierungsproblems für das Reverse-Engineering. Dies ist durch Verwendung der gleichen Kalibrierungsregeln oder Black-Box-Verfahren wie auf der Mikro-Ebene möglich.
- **Übertragung der Ergebnisse von der Makro- zur Mikro-Ebene:** DAVINCI unterstützt bei der Übertragung der *IVM* auf die Mikro-Ebene, indem die im Modell der Makro-Ebene ermittelten *IVM* in das Modell der Mikro-Ebene eingesetzt werden, damit diese dort als Zielfunktionen dienen können. Dies geschieht durch Übertragung der Parameterwerte der *IVM* auf Makro-Ebene in das Mikro-Modell mit den gleichen Techniken, mit denen Parameterwerte zwischen Teilmodellen übertragen werden. Dabei müssen für DAVINCI das *IVM*-Modell der Makro-Ebene und das Originalmodell der Mikro-Ebene „nebeneinander“ in einem Gesamtmodell repräsentiert werden und es muss möglich sein zwischen den beiden Modellen durch einen Auswahlparameter umzuschalten.
- **Wechsel zwischen Makro- und Mikro-Ebene:** Der automatisierte Wechsel zwischen Makro- und Mikro-Ebene wird in DAVINCI durch die gleiche Sequentialisierungstechnik, wie beim Hintereinanderschalten verschiedener White-Box-Techniken umgesetzt. Die Makro-Kalibrierung und die Mikro-Kalibrierung werden in eigenen Kalibrierungsproblemen in DAVINCI gelöst. Die Kalibrierungen werden in DAVINCI sequenzialisiert und die jeweiligen Ergebnisse zueinander übertragen.
- **Robuste Suche auf der Mikro-Ebene:** DAVINCI enthält den Algorithmus zur robusten Suche, der basierend auf der Nutzerspezifikation nach Bedarf zwischen Mikro-Teilmodellen und zugehörigen *IVM* umschaltet.

11.2.2 Zusammenhang der DAVINCI-Komponenten

Abbildung 11.3 zeigt die verschiedenen Komponenten des DAVINCI-Kalibrierers, die sich aus den betrachteten Anwendungsfällen ergeben und ihre gegenseitige Verwendung in DAVINCI (Darstellung beginnend von links-unten im Bild). Weiter zeigt das Schaubild die Arbeitsschritte des Nutzers, bevor er die DAVINCI-Kalibrierung starten kann. Nach dem der Nutzer die (Teil-)Modelle vorbereitet hat muss er die Reihenfolge der Teilkalibrierungen, zusätzliche Parametereinstellungen, Regeln und weitere Vorgaben für die Suchverfahren spezifizieren und kann dann den Kalibrierungsprozess starten. Die einzelnen DAVINCI-Komponenten setzen jeweils eine der beschriebenen Suchtechniken um. Das Schaubild zeigt, dass die Suchtechniken hintereinandergeschaltet werden können, wobei einer Technik immer genau eine andere der Komponenten folgt. Am Schluss der Kette steht die Auswertung einer Parametereinstellung. Das Ergebnis dieser Auswertung kann dann in den Algorithmen der anderen Komponenten verwendet werden. Abbildung 11.3 fokussiert nur auf die mögliche Hintereinanderschaltung der DAVINCI-Komponenten, die zur Unterstützung der beschriebenen Anwendungsfälle benötigt wird. Im nächsten Abschnitt und Abbildung 11.4 wird die wechselseitige Benutzung und der Datenfluss zwischen den Komponenten beschrieben.

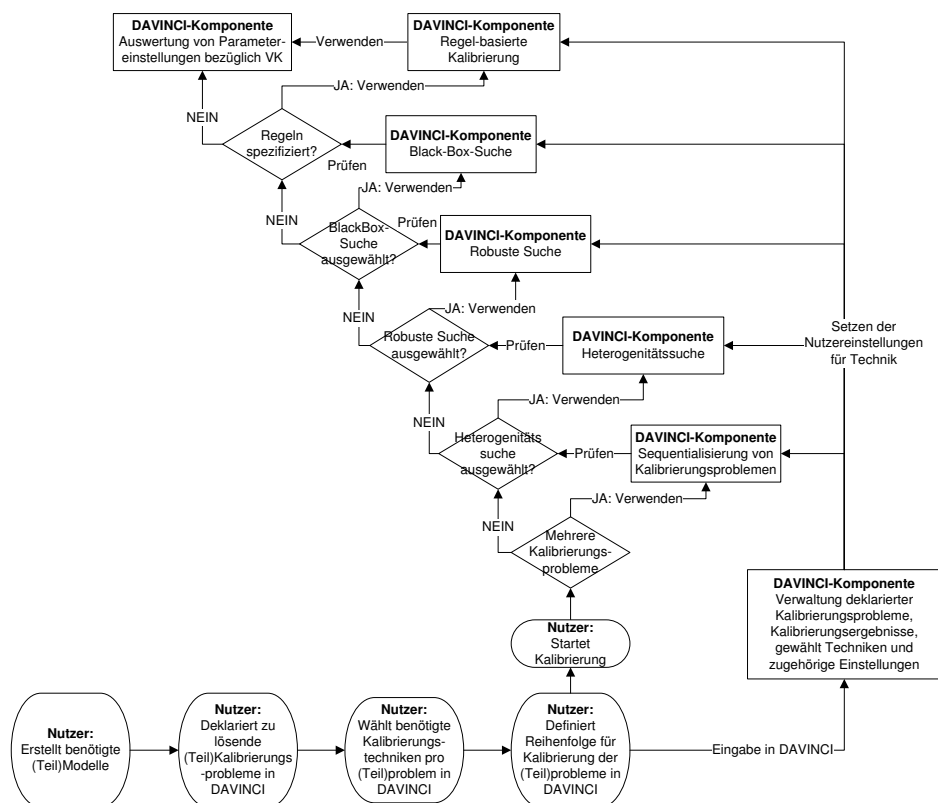


Abbildung 11.3: Anwendungsfall Makro-Mikro-Verfahren.

### 11.2.3 Implementierung der Komponenten im DAVINCI-Kalibrierer

Da nicht für jedes Kalibrierungsproblem alle in DAVINCI möglichen Kalibrierungstechniken benötigt werden, wurde der Kalibrierer so implementiert, dass beim Start eines Kalibrierungslaufes aus den Nutzer-Deklarationen zu den benötigten Techniken automatisch ein spezifischer Kalibrierungsalgorithmus „zusammengebaut“ wird. Die einzelnen Techniken sind in DAVINCI als Algorithmusbausteine realisiert, die nach den Nutzervorgaben miteinander kombiniert werden. Während eines Kalibrierungslaufes sind nicht alle gewählten Kalibrierungstechniken parallel tätig. Stattdessen berechnet immer genau einer der Techniken einen oder mehrere Algorithmusschritte und übergibt dann ihr aktuelles (Zwischen)ergebnis an eine andere Technik, die das Ergebnis benötigt, weiter.

Technisch beruht die Konfigurierbarkeit auf zwei Entwurfsmustern, dem Dekorator- und dem Strategiemuster (siehe Gamma [48]). Ausgehend von einem Basisverfahren, das nur zu einer gegebenen Parametereinstellung einen zugehörigen Zielfunktionswert ermitteln kann, werden die benötigten Techniken geeignet basierend auf dem Dekoratormuster hinzu konfiguriert. Gibt es für einzelne Techniken die Möglichkeit verschiedene Varianten (z.B. verschiedene Algorithmen bei Black-Box-Suchverfahren) anzuwenden, so sind die Varianten gemäß dem Strategiemuster austauschbar.

Abbildung 11.4 zeigt die in DAVINCI implementierte Reihenfolge, innerhalb der die einzelnen Kalibrierungstechniken miteinander kombiniert werden können und welche Daten die Komponenten untereinander austauschen.

Im Folgenden wird jede Technik kurz mit den vom Nutzer für die Technik zu spezifizierenden Parameterwerten erläutert.

#### 11.2.3.1 Basisverfahren zur Validitätsbewertung

Auf der untersten Ebene wird eine aktuelle Kandidateneinstellung der Modellparameter durch Simulation ausgewertet. Jede Bewertung einer Parametereinstellung während der Kalibrierung macht die Durchführung eines oder mehrerer Simulationsläufe (bei Zufallsinflüssen) notwendig, auf deren Basis die Parametereinstellung hinsichtlich der erzeugten Verhaltensvalidität bewertet wird. Dies geschieht im Basisverfahren von DAVINCI. Das Basisverfahren initialisiert das Modell mit der gegebenen Parametereinstellung, simuliert das Modell die gegebene Anzahl an Bewertungsiterationen und ermittelt das Validierungsergebnis entsprechend der vorgegebene Validierungskriterien. Als Rückgabewert erhält der Nutzer die vorgegebene Parametereinstellung und die zugehörigen Validitätsbewertungen gemäß der Validierungskriterien.

**Nutzereingaben:** Für das Basisverfahren spezifiziert der Nutzer wie oft ein Modell zur Validitätsbewertung simuliert werden soll und wie die Zielfunktionswerte mehrerer Simulationsläufe aggregiert werden sollen (Mittelwert oder Median). Optional kann der Nutzer eigene Parametereinstellungen spezifizieren und vom Basisverfahren auswerten lassen.

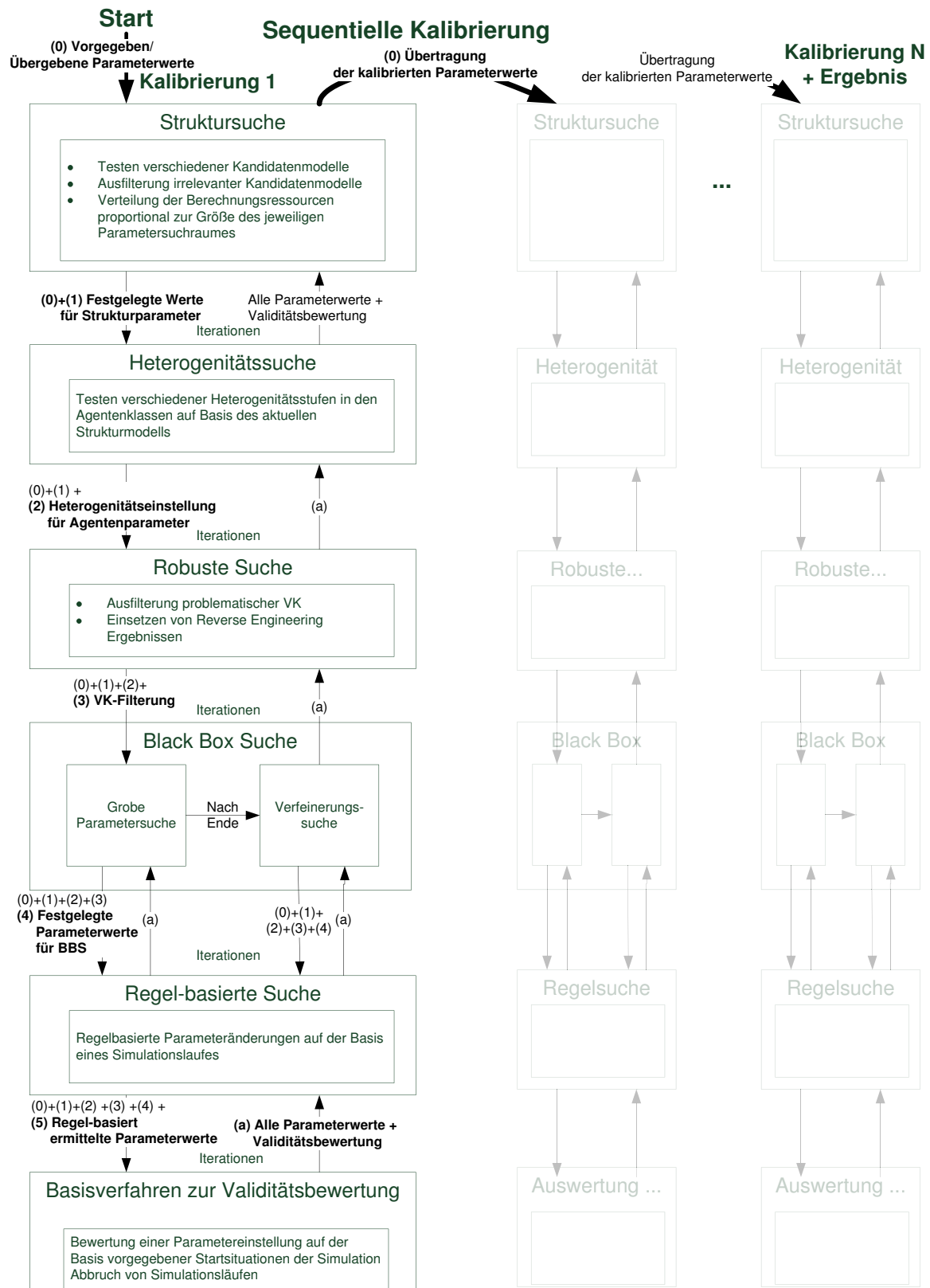


Abbildung 11.4: Schematische Skizze des entwickelten Kalibrierungswerkzeuges in Maximal Konfiguration.

### 11.2.3.2 Wissensbasierte Suche

Die DAVINCI-Komponente der wissensbasierte Suche besteht aus dem Einsatz von Regelbasierten Kalibrierungstechniken. Der Nutzer kann hier explizites Änderungswissen für die zu kalibrierenden Parameterwerte formalisieren. Dabei müssen die spezifizierten Regeln immer den Endzustand eines Simulationslaufes auf eine Änderung an Parameterwerten abbilden.

**Nutzereingaben:** Der Nutzer muss in DAVINCI Regeln spezifizieren, die den Zustand einer Simulation am Ende eines Simulationslaufes auf Änderungen an Parameterwerten des Modells abbilden.

### 11.2.3.3 Black-Box Suche

Die DAVINCI-Komponente der Black-Box Suche besteht aus dem Einsatz heuristischer und Gradienten-basierter Suchtechniken. Hierfür stehen verschiedene Verfahren zur Auswahl (siehe 11.4).

**Nutzereingaben:** Abhängig von gewählten Black-Box-Verfahren muss der Nutzer die Algorithmeinstellungen für das Verfahren in DAVINCI eingeben.

Regel-basierte und Black-Box Suche bilden damit den Kern des DAVINCI-Kalibrierers. Auf ihrer Basis können bereits Simulationsmodelle kalibriert werden, ohne dass Techniken wie Makro-Mikro-Verfahren, Heterogenitätssuche oder robuste Suche verwendet werden. DAVINCI ermöglicht auch den gemeinsamen Einsatz von wissensbasierten und Black-Box-Verfahren, d.h. der Nutzer kann einen Teil der Parameterwerte auf der Ebene mit Black-Box-Verfahren und einen anderen Teil mit wissensbasierter Suche einstellen.

### 11.2.3.4 Robuste Kalibrierung:

Die robuste Kalibrierung erweitert ein Parametersuchverfahren wie Black-Box oder wissensbasierte Suche (siehe Abschnitt 9.3). Der Einsatz der robusten Kalibrierung setzt voraus, dass der zu kalibrierende Parametersuchraum, d.h. die Heterogenität in den Parameterwerten, festgelegt ist. Zur Durchführung eines Bewertungsschrittes des robusten Suchmechanismus wird jeweils eine vollständige Kalibrierung der Parameterwerte benötigt. Aus diesem Grund ist die robuste Kalibrierung in der Anwendungsreihenfolge der Techniken in DAVINCI zwischen der Heterogenitätssuche und den Parametersuchverfahren angesiedelt.

**Nutzereingaben:** Der Nutzer muss zum einen das Modell so erweitern, dass zwischen Einfluss eines relevanten Teilmodells (z.B. Agent) und dem zugehörigen *IVM* auf das Simulationsverhalten umgeschaltet werden kann. Für DAVINCI muss der Nutzer dann einen booleschen „Schalterparameter“ spezifizieren, der die Umschaltung im Modell durchführt. DAVINCI wird diesen Schalter dann zum Wechsel zwischen Teilmodell und *IVM* in der robusten Suche verwenden. Weiter muss der Nutzer die Algorithmusparameter für die robuste Suche in DAVINCI eingeben (siehe Kapitel 9.3).

### 11.2.3.5 Heterogenitätsfestlegung:

Die DAVINCI-Komponente zur Heterogenitätssuche (ohne Parameterkalibrierung) ermittelt, für welche Gruppen von Agentenmodellfamilien Parameterwerte eingestellt werden müssen. Erst nach der Auswahl einer konkreten Heterogenitätsverteilung (Gruppierung)



ist die Struktur des Parametersuchraums festgelegt. Die Heterogenitätsfestlegung wird erst dadurch zur vollständigen Heterogenitätssuche, dass die zugehörigen Parameterwerte eingestellt werden. Zur konkreten Einstellung der Parameterwerte innerhalb einer Heterogenitätsfestlegung werden die Suchtechniken Black-Box-Verfahren oder wissensbasierte Suche (optional auch mit robuster Suche) eingesetzt. In jedem Schritt der Heterogenitätssuche wird also immer zunächst der nächste Schritt der Heterogenitätsfestlegung ausgeführt und dann (je nach Auswahl der Techniken) die in Abbildung 11.4 darunter angeordneten Suchverfahren.

**Nutzereingaben:** Für die Heterogenitätsfestlegung deklariert der Nutzer für jede zu kalibrierende Agentenmodellklasse eine Ähnlichkeitsfunktion und einen Wertebereich für mögliche Gruppierungsanzahlen. Weiter muss der Nutzer im Modell für jeden Agenten einen ID-Parameter deklarieren. Dieser Parameter wird von DAVINCI genutzt, um die generierte Gruppenzuordnung der Agenten zu speichern.

#### 11.2.3.6 Struktursuche:

DAVINCI bietet dem Nutzer auch die Möglichkeit automatisch verschiedene Modellstrukturen zu kalibrieren und miteinander zu vergleichen. Dabei werden Modellteile mit speziellen Schaltern „ein- und ausgeschaltet“. Die speziellen Schalter werden von der Struktursuche automatisch eingestellt. Diese Technik steht aber nicht im Fokus dieser Arbeit und wird daher nicht genauer betrachtet.

#### 11.2.3.7 Sequentielle Kalibrierung:

Die als sequentielle Kalibrierung bezeichnete übergeordnete Technik erlaubt die Hintereinanderschaltung mehrerer vollständiger Kalibrierungsprozesse. In DAVINCI werden White-Box-Techniken, bei denen mehrere Teilmodelle kalibriert werden, und das Makro-Mikro-Verfahren wie folgt umgesetzt. Zunächst deklariert der Nutzer für jedes Teilproblem ein eigenes Kalibrierungsproblem in DAVINCI und definiert eine Reihenfolge zur Lösung der Kalibrierungsprobleme. Die einzelnen Teilschritte werden durch Hintereinanderschaltung miteinander kombiniert. Dabei überträgt die sequentielle Kalibrierung jeweils die Kalibrierungsergebnisse des vorhergehenden Kalibrierungsproblems in das nächste. Iterationen können durch abwechselndes Hinzufügen der gleichen Kalibrierungsprobleme zur Sequenz umgesetzt werden.

**Nutzereingaben:** Der Nutzer muss jedes (Teil-)kalibrierungsproblem für sich in DAVINCI eingeben. Dann muss er in DAVINCI eine Hintereinanderschaltung der Teilproblemkalibrierungen definieren.

#### 11.2.3.8 Ausführung der Techniken zur Laufzeit der Kalibrierung

In diesem Abschnitt wird erläutert, wie in DAVINCI zur Laufzeit die einzelnen Kalibrierungstechniken angewendet werden. Abbildung 11.4 zeigt ein DAVINCI-Kalibrierungsverfahren in Maximalkonfiguration, d.h. in dem alle zur Verfügung stehenden Kalibrierungstechniken eingesetzt werden. Auf der obersten Ebene wird „Start aufgerufen“ (siehe Abbildung 11.4). Für jeden Schritt einer Technik auf einer höheren Ebene müssen einmal alle Schritte der Techniken auf darunter liegenden Ebenen durchgeführt werden. Wird

eine Technik nicht verwendet, so führen in Abbildung 11.4 die Pfeile aus der darüber liegenden direkt in die darunter liegende Ebene. Die abwärts gerichteten Pfeile in Abbildung 11.4 repräsentieren jeweils das Weiterreichen auf einer höher gelegenen Ebene generierter „Teilergebnisse“ an die Kalibrierungsverfahren auf einer darunter liegenden Ebene. Solche „Teilergebnisse“ können entweder in der Festlegung bestimmter Parameterwerte, der Ausfilterung bestimmter VK oder der Festlegung und Lokalisierung von Heterogenität in den Parameterwerten der Agenten bestehen. Die aufwärts gerichteten Pfeile repräsentieren die Rückgabe von Ergebnissen, die auf weiter unten liegenden Ebenen generiert wurden. Werden Werte an eine Komponente zurück gegeben, wird auf dieser Ebene geprüft, ob weitere Aktionen, wie Modifikation der Parameterwerte, Filterung anderer VK oder Änderung der Heterogenitätseinstellungen durchgeführt werden müssen. Werden Aktionen durchgeführt, so werden die angepassten Randbedingungen wieder an die darunter liegende Kapselungsstufe weitergereicht. Ansonsten werden die erhaltenen Ergebnisse an die darüber liegende Ebene geschickt. Die Kalibrierung ist beendet, wenn die aktuellen Kalibrierungsschritte abgeschlossen sind und die oberste Technik keine weiteren Kalibrierungsschritte mehr erfordert. Wird eine Sequenz mehrerer Teilkalibrierungen durchgeführt, werden die kalibrierten Parameterwerte an das nächste Teilproblem der Sequenz übergeben und dessen Lösungslauf gestartet.

#### 11.2.3.9 Verwaltung der verfügbaren Simulationsläufe

Nach der Deklaration der Kalibrierungsprobleme in DAVINCI wird die automatische Kalibrierung gestartet. Die verfügbaren Simulationsläufe werden von DAVINCI automatisch auf die einzelnen Teilkalibrierungen verteilt. Die Verteilung zwischen den einzelnen eingesetzten Techniken erfolgt in DAVINCI gleichmäßig, wobei in einem Kalibrierungsschritt nicht benötigte Ressourcen den weiteren Schritten zur Verfügung stehen. Es wäre auch denkbar, die verfügbaren Simulationsläufe immer abhängig von der Größe des jeweiligen Parameter-suchraumes aufzuteilen.

Nach dem Start der Kalibrierung kann der Modellierer oder Fachexperte entweder warten bis die Kalibrierung aller zu lösender (Teil-)Kalibrierungsprobleme beendet ist oder zur Laufzeit der Kalibrierung Analysen auf bereits von DAVINCI untersuchten Parameter-einstellungen durchführen und mit den Suchtechniken interagieren (siehe Abschnitt 12.2).

#### 11.2.3.10 Ende der Kalibrierung

Ist die automatische Kalibrierung beendet, kann der Modellierer oder Fachexperte die Ergebnisse untersuchen, abspeichern und in das Simulationsmodell übernehmen. Das Vorgehen für ein konkretes Kalibrierungsproblem hängt jeweils von den spezifischen Eigenschaften des Kalibrierungsproblems ab.

Wurde die Lösung mehrerer Kalibrierungsprobleme (z.B. Teil-Modelle bei einer White-Box-Kalibrierung) für DAVINCI in einer Kalibrierungssequenz vorgegeben, so stehen alle Teilergebnisse und das Endergebnis (=Ergebnis des letzten Kalibrierungsproblems) zur Verfügung.

Wurden nicht alle Teilkalibrierungsprobleme behandelt, wurde also z.B. in einem Kalibrierungslauf nur das Reverse-Engineering auf der Makro-Ebene durchgeführt, so kann

der Nutzer die Ergebnisse (die *IVM*) in ein neues Kalibrierungsproblem für die Mikro-Ebene übertragen und dieses dann kalibrieren lassen.

#### 11.2.4 Einsatz von DAVINCI für ABS-Kalibrierung

In diesem Abschnitt werden beispielhaft die einzelnen Schritte beim Einsatz von DAVINCI für die Anwendungsfälle der Kalibrierung auf der Mikro-Ebene und dem Makro-Mikro-Verfahren aufgelistet.

#### 11.2.5 Kalibrierung auf der Mikro-Ebene mit Heterogenitätssuche

1. Entwurf des Modellkonzeptes und Definition der globalen Validierungskriterien für das Modell.
2. Analyse des Modellkonzeptes: Identifikation von Problemen bei der Festlegung benötigter Heterogenität in den Agentenparameterwerten.
3. Modellierung des ABS-Modells.
4. Deklaration der zu kalibrierenden Agentenparameter.
5. Deklaration eines eindeutigen ID-Wertes für jedes zu kalibrierende Agentenmodell (für Details siehe nächster Abschnitt).
6. Deklaration der Ähnlichkeitsfunktion(en) zum Vergleich zweier Agenten in DAVINCI für jede zu kalibrierende Agentenklasse.
7. Deklaration eines Wertebereichs für mögliche Gruppenanzahlen in DAVINCI für jede zu kalibrierende Agentenklasse.
8. Deklaration eines zu verwendenden Suchverfahrens in DAVINCI (Black-Box oder Regel-basiert) und Festlegung zugehöriger Algorithmusparameter.
9. Start des Kalibrierungsprozesses mit Heterogenitätssuche.

#### 11.2.6 Kalibrierung im Makro-Mikro-Verfahren

1. Entwurf des Modellkonzeptes und Definition der globalen Validierungskriterien für das Modell.
2. Analyse des Modellkonzeptes: Identifikation der zusätzlichen Makro-Ebene (des Makro-Mikro-Verfahrens) und Strukturen der *IVM* zu Agenten(gruppen) auf der Mikro-Ebene.
3. Modellierung des ABS-Modells auf der Mikro-Ebene.
4. Modellierung der zugehörigen Modelle auf der Makro-Ebene des Makro-Mikro-Verfahrens mit *IVMF* Strukturen.
5. Erweiterung des Mikro-Modells, um die *IVM* und individuellen Zielfunktionen, die Agentenverhalten mit zugehörigem *IVM*-Verhalten vergleichen.

6. Deklaration Kalibrierungsproblem 1 für Makro-Ebene:
  - (a) Deklaration zu kalibrierender Parameter des Makro-Modells in DAVINCI.
  - (b) Deklaration eines zu verwendenden Suchverfahrens in DAVINCI (Black-Box oder Regel-basiert) und Festlegung zugehöriger Algorithmusparameter.
7. Deklaration Kalibrierungsproblem 2 für Mikro-Ebene:
  - (a) Deklaration zu kalibrierender Parameter des Mikro-Modells in DAVINCI.
  - (b) Deklaration neuer Zielfunktion auf Basis der individuellen Zielfunktionen, welche die globale Zielfunktion ersetzt.
  - (c) Deklaration eines zu verwendenden Suchverfahrens in DAVINCI (Black-Box oder Regel-basiert) und Festlegung zugehöriger Algorithmusparameter.
8. Hintereinanderschaltung von Kalibrierungsproblem 1 und 2.
9. Start des Kalibrierungsprozesses.
10. Manuelle Iteration: Wenn nach Ende das Gesamtverhalten nicht zufriedenstellend ist, dann Anpassung von Kalibrierungsproblem 1 und erneuter Start.

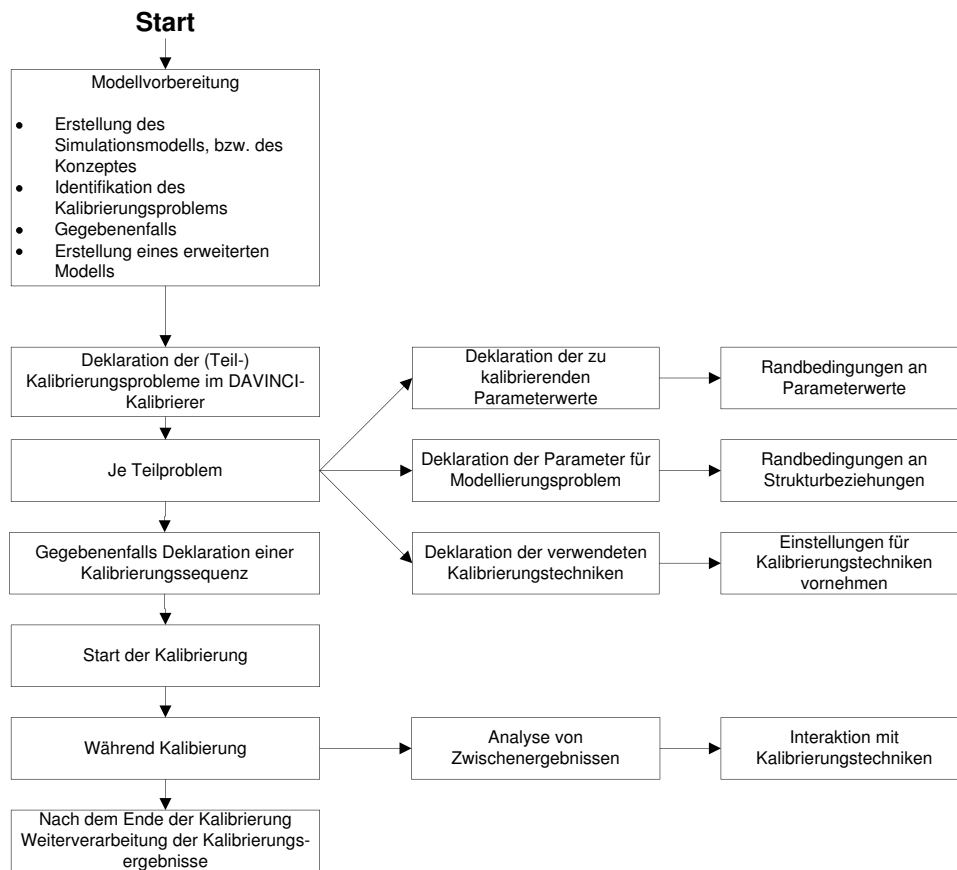


Abbildung 11.5: Zusammenfassung: Benutzung des DAVINCI-Kalibrierers.

### 11.2.7 Zusammenfassung: DAVINCI-Nutzung

Abbildung 11.5 zeigt zusammengefasst das Vorgehen bei der Anwendung des DAVINCI-Werkzeuges unabhängig von den beschriebenen Anwendungsfällen.

## 11.3 Weitere Aspekte der technischen Umsetzung von DAVINCI

In diesem Abschnitt wird auf verschiedene technische Aspekte der internen Funktionsweise von DAVINCI eingegangen. Die beschriebenen Aspekte betreffen technische Lösungen, die noch nicht in den bisherigen Kapiteln behandelt wurden.

### 11.3.1 Problemaspekte beim Management von Parameterwerten

Unter dem Management von Parameterwerten wird nicht die konkrete Kalibrierung verstanden, sondern wie die Parameterwerte und Heterogenitätseinstellungen während der Kalibrierung im Werkzeug verwaltet und während Simulationsläufen zur Bewertung von Parametereinstellungen in die Agenten eingesetzt werden. Die Unterschiede bei der Verwaltung und dem Einsetzen von Parameterwerten in ein Simulationsmodell bei der Kalibrierung von ABS im Vergleich zu anderer Simulationen resultieren aus den Problemen der potentiellen Offenheit von ABS (siehe Kapitel 4.1.3.3) und der unter Umständen notwendigen Identifikation und Lokalisierung benötigter Heterogenität in den Parameterwerten der Agenten (siehe Kapitel 10). Aus diesen Problemen resultieren einige Anforderungen an eine geeignete Verwaltung von Parameterwerten in einem Kalibrierungswerkzeug für ABS:

- **Dynamische Erweiterbarkeit des Parametersuchraumes zur Laufzeit:** Das Werkzeug muss in der Lage sein, die Menge der kalibrierten Parameterwerte zur Laufzeit der Simulation zu erweitern, nämlich dann, wenn ein neuer (bisher unbekannter) Agent in die Simulation kommt, dessen Parameterwerte kalibriert werden müssen.
- **Dynamisches Einsetzen von Parameterwerten zur Laufzeit:** Das Werkzeug muss in der Lage sein, Parameterwerte zur Laufzeit in simulierte Agenten einzusetzen, damit auch neu hinzugekommene Agenten korrekt initialisiert werden können.

### 11.3.2 Gruppen-basierte Verwaltung von Parameterwerten für Agenten

Im DAVINCI-Kalibrierer werden Parameterwerte nicht direkt für jeden Agenten gespeichert, sondern jeweils für Agentengruppen (siehe das Konzept der Agentengruppen in Definition 8). Parameterwerte werden den simulierten Agenten während eines Bewertungslaufes zu einer Parametereinstellung basierend auf ihrer Gruppenzugehörigkeit zugewiesen. Agenten mit gleicher Gruppenzugehörigkeit erhalten während der Kalibrierung die gleichen Parameterwerte zugewiesen.

### 11.3.3 Einsetzen von Parameterwerten

In DAVINCI definiert der Nutzer, zu welchen Zeitpunkten Parameterwerte während eines Simulationslaufes im Basisverfahren von DAVINCI auf Grundlage der Parametergruppen zugewiesen werden. Solche Zeitpunkte können z.B. sein, wenn:

- der Simulationslauf gestartet wird.
- ein Agent neu in die Simulation kommt.
- festgestellt wird, dass sich die Gruppenzugehörigkeit eines Agenten geändert hat. Dies ist z.B. dann der Fall, wenn ein Agent eine andere Rolle im simulierten System übernimmt und daher einer anderen Parametergruppe angehört. Eine entsprechende Änderung kann vom Modellierer oder Fachexperten modelliert werden, indem er den ID-Wert des jeweiligen Agenten anpasst, bzw. im Modell entsprechende Funktionalität eingeführt hat.

#### 11.3.4 Vorgabe von Startwerten für Agentenparameter

Möchte der Modellierer oder Fachexperte z.B. lokale Suchtechniken zur Kalibrierung verwenden, werden initiale Parameterwerte für die zu kalibrierenden Parameter benötigt. Dem Nutzer wird in DAVINCI die Möglichkeit gegeben, für jeden einzelnen simulierten Agenten eigene Parameterwerte vorzugeben. Dies ist natürlich bei vielen simulierten Agenten nicht realisierbar. Weiter kann es sein, dass der Nutzer nur für bestimmte Agenten Vorgaben für Parameterwerte machen möchte und für andere nicht. Um den Deklarationsprozess initialer Parameterwerte für die Agenten so flexibel und einfach wie möglich zu halten, werden in DAVINCI die folgenden Möglichkeiten bereitgestellt:

- **Keine Deklaration von Parameterwerten:** Der Nutzer kann sich entscheiden zufällige initiale Parameterwerte zu verwenden.
- **Deklaration von Parameterwerten auf Klassenebene:** Um nicht für jeden einzelnen Agenten individuelle Parameterwerte angeben zu müssen, kann der Modellierer oder Fachexperte für Klassen von Agenten homogene Parameterwerte deklarieren.
- **Deklaration von Parameterwerten auf Gruppenebene:** Die Deklaration von Parameterwerten kann bei Bedarf auch auf detaillierterer Ebene erfolgen. Dies geschieht, indem der Nutzer Parameterwerte bezüglich der Gruppen-ID-Werte spezifiziert.
- **Kombinierte Deklaration auf Klassen- und Gruppenebene:** Die Deklaration von Parametern auf Klassen- und Gruppenebene können auch gemischt verwendet werden. Auf diese Weise können für Teile der Agenten explizit individuelle Parameterwerte angegeben werden, während für den Rest homogene Parameterwerte verwendet werden.

#### 11.3.5 Dynamische Suchraum-Erweiterung

In diesem Abschnitt wird darauf eingegangen, wie in DAVINCI mit offenen ABS umgegangen wird. Die Menge zu kalibrierender Parameterwerte in DAVINCI kann während eines Simulationslaufes dynamisch wachsen. Es ist **nicht zwingend notwendig**, dass der Nutzer vor Beginn der Kalibrierung alle **Agenten-IDs, Gruppierungen und Startwerte für alle möglichen simulierten Agenten spezifiziert** hat. Während eines Simulationslaufes werden von DAVINCI grundsätzlich allen Agenten einer Klasse, für die Parameterwerte kalibriert werden sollen, auch Parameterwerte zugewiesen. Dabei wird wie folgt verfahren:

1. **Existenz eines Parameterwertes:** Zunächst wird geprüft, ob für den betroffenen Agenten bzw. die Gruppe, welcher der Agent zugeordnet wird, bereits ein Parameterwert gespeichert ist. Wenn ja, wird dieser Wert dem Agenten zugewiesen.
2. **Erzeugung eines neuen Parameterwertes:** Kann dem Agenten aus der Verwaltungsstruktur kein Parameterwert zugewiesen werden, wird für den Agenten eine neue Parametergruppe eingeführt. Gemäß der Vorgaben vom Nutzer wird ein initialer Parameterwert für die neue Gruppe generiert, gespeichert und dem Agenten zugewiesen. Der Parameterwert für die neu gebildete Gruppe des Agenten entspricht entweder einem vom Nutzer spezifizierten Startwert für die Klasse des Agenten oder wird zufällig bestimmt.

## 11.4 Black-Box Suche in DAVINCI

Der DAVINCI-Kalibrierer stellt eine Reihe von Black-Box-Kalibrierungsverfahren bereit. Im ersten Teil dieses Abschnitts wird auf die in DAVINCI eingesetzten Black-Box Kalibrierungsverfahren eingegangen und beschrieben, wie die einzelnen Techniken realisiert wurden. Die Verfahren wurden ausgewählt, um unterschiedliche Stärken globaler und lokaler Suchtechniken ausnutzen zu können.

In DAVINCI stehen dem Nutzer die folgenden Black-Box Verfahren zur Verfügung:

- **Zufallssuche:** Bei dieser einfachsten Form einer globalen Suchtechnik werden gleichverteilt zufällig Parametereinstellungen im gesamten Parametersuchraum ausgewählt und getestet. Der Nutzer gibt lediglich die Anzahl zu testender Parametereinstellungen und eine von ihm erwünschte Diskretisierung des Parametersuchraums, d.h. einen Mindestabstand für benachbarte Parameterwerte, vor.
- **Rasterexperimente:** Rasterexperimente (siehe Abschnitt 3.2.6) sind eine systematische Variante der Zufallssuche (globale Suchtechnik). In diesem Fall gibt der Nutzer die Granularität des Suchraumrasters und damit die Anzahl ausgewerteter Parametereinstellungen vor. Die Praktikabilität von Rasterexperimenten hängt von der Dichte des Rasters, der Anzahl der Parameterwerte und deren Wertebereich ab.
- **Evolutionäre Strategie:** Zur heuristischen Suche mit globaler Suchtechnik wird dem Nutzer eine evolutionäre Strategie (siehe Abschnitt 3.2.4.2) bereitgestellt. Zur Steuerung des Verfahrens kann der Nutzer die maximale Anzahl zu berechnender Generationen, die Größe der verwendeten Population, die Menge an Mutations- und Crossover-Operator-Anwendungen pro Generation und die Anzahl an Elterlösungen bei der Crossover-Anwendung einstellen. Die Auswahl der Eltern für die Anwendung des Crossover-Operators erfolgt nach dem „Survival of the Fittest“-Prinzip, d.h. Parametereinstellungen mit besserer Fitness werden wahrscheinlicher selektiert. Das Verfahren terminiert, wenn entweder die Bewertungen der Lösungspopulation konvergieren oder die maximale Anzahl an Generationen durchlaufen wurde.
- **Simulated Annealing:** Mit Simulated Annealing (siehe Abschnitt 3.2.4.1) wird ein Verfahren mit lokaler Suchtechnik bereitgestellt. Das Verfahren verwendet den grundlegenden Simulated Annealing Algorithmus mit Metropolis-Annahmekriterium und erweitert diesen um zusätzliche Eigenschaften:

- **Iterative Neustarts** erlauben es dem Verfahren lokalen Optima zu entkommen, wenn es diese nicht mehr aufgrund des Metropolis-Kriteriums schafft.
- Eine **Gedächtnis-Liste** sorgt dafür, dass bereits ausgewertete Parameterkonfigurationen vom Suchverfahren nicht erneut ausgewertet werden. Das Vorgehen unterscheidet sich von dem der Tabu-Suche, da es dem Verfahren erlaubt ist, bereits besuchte Suchraumpunkte erneut zu durchlaufen. Die Punkte werden lediglich nicht erneut durch Simulation bewertet.
- Zur besseren **Steuerung der Bewegungsrichtung** des Verfahrens wird eine einfache Heuristik zur Gradientenapproximation verwendet, die keine zusätzlichen Simulationsläufe als die bereits durchgeführten benötigt. Die Technik ist ähnlich dem SPSA-Verfahren und SPSA-Varianten (siehe Abschnitt 3.2.3), die Simulated Annealing Annahmekriterien verwenden [67], jedoch wird in keinem Fall explizit ein Gradient berechnet. Stattdessen wird bei Nicht-Aannahme einer Kandidatenparametereinstellung (Gradient fällt in Richtung dieser Parametereinstellung) als nächstes direkt die gegenüberliegende Parametereinstellung ausgewertet, falls diese bisher noch nicht ausgewertet wurde. Sonst wird ganz normal eine neue zufällige benachbarte Parametereinstellung gewählt.
- Um die **Konvergenz des Verfahrens** zu beschleunigen, können die Schrittweite und die Anzahl modifizierter Parameter beim Übergang zu benachbarten Kandidateneinstellungen automatisch adaptiert werden. Die in dieser Arbeit verwendete Adaption verwendet die folgende Heuristik: Wurden innerhalb der letzten  $n$  getesteten Kandidateneinstellungen mehr als  $y\%$  der Einstellungen abgelehnt, werden die Änderungsschrittweiten der einzelnen Parameter halbiert. Konnte weiter innerhalb einer Temperaturstufe keine Verbesserung der Zielfunktion erreicht werden, wird die Anzahl gleichzeitig modifizierter Parameter auf  $z\%$  des letzten Wertes reduziert.
- **Numerische Optimierung:** Als numerische lokale Suchtechnik (siehe Abschnitt 3.2.2) wird eine Implementierung des BFGS-Verfahrens nach Nocedal [86] bereitgestellt. Um dem Verfahren eine gewisse Unschärfe in der Suche zu geben, kann die Verschiebung  $h$  bei der Gradientenberechnung vorgegeben werden. Ein größerer Wert führt zu ungenauerer Schätzung, vermeidet aber unter Umständen lokale Optima.

Die einzelnen Black-Box-Suchverfahren sind beim Entwurf eines spezifischen Kalibrierungsverfahrens durch den Nutzer austauschbar. Innerhalb der Vorgabe **ein** Black-Box-Verfahren zu verwenden, kann der Nutzer gemäß dem Strategie-Muster verschiedene konkrete Black-Box-Verfahren auswählen.

## 11.5 Zusammenfassung

Bei der technischen Umsetzung des DAVINCI-Kalibrierers wurde speziell auf die Umsetzung und Kombinierbarkeit der in dieser Arbeit entwickelten Kalibrierungstechniken eingegangen. Zusätzlich wurde die Verwaltung der Agentenparameter in DAVINCI dargelegt und beschrieben, wie ein Kalibrierungslauf in DAVINCI abläuft. In Kapitel 12 wird nun ausführlich auf die Interaktionsmöglichkeiten des Nutzers mit DAVINCI während gestarteter Kalibrierungsläufe eingegangen.



## Kapitel 12

# Der DAVINCI-Kalibrierer: Interaktion

In Kapitel 11 wurde der Aufbau des DAVINCI-Kalibrierers beschrieben. In diesem Kapitel wird auf Interaktionsmöglichkeiten des Nutzers mit DAVINCI nach dem Start eines Kalibrierungsverfahrens eingegangen. Es werden die folgenden Interaktionsmöglichkeiten behandelt:

- Modifikation der verfügbaren Simulationsläufe, insgesamt und für einzelne Kalibrierungstechniken.
- Modifikation der Einstellungen einzelner Suchtechniken.
- Analyse der bereits bewerteten Parametereinstellungen.
- Hinzufügen von Nutzer-definierten Parametereinstellungen zur Laufzeit der Kalibrierung.
- Wiederholung der Kalibrierung einzelner Modellausprägungen zur Laufzeit der Gesamtkalibrierung.

Die einzelnen Interaktionsmöglichkeiten werden im Rahmen der Interaktionsdialoge beschrieben, mit denen der Nutzer zur Laufzeit von DAVINCI interagieren kann. Nach dem Start eines Kalibrierungslaufes gelangt der Nutzer in den globalen Übersichtsdialog. In diesem Dialog kann der Nutzer durch Betätigung des „Details“ Knopfes im Interaktionsbereich den Interaktionsdialog für das aktuell laufende Black-Box Kalibrierungsverfahren aufrufen.

Im Folgenden wird zunächst die interne Funktionsweise von DAVINCI bei der Nutzer-Interaktion mit Black-Box-Suchverfahren beschrieben und dann die weitere Beschreibung der Interaktionsmöglichkeiten ausgehend von diesem Interaktionsdialog durchgeführt.

### 12.1 Steuerung der Black-Box-Suchverfahren zur Laufzeit

Verwendet der Nutzer Black-Box-Suchverfahren für die Modellkalibrierung, kann die Wahl einer geeigneten Einstellung von Steuerungsparametern für die eingesetzten Verfahren für den Nutzer schwierig sein. Dies ist insbesondere dann der Fall, wenn der Nutzer mit den internen Funktionsweisen einzelner Suchverfahren nicht ausreichend vertraut ist. Wählt er die Steuerungsparameter für ein Suchverfahren ungünstig, kann dies dazu führen, dass das

Suchverfahren nur ein lokales Optimum des Suchraumes oder möglicherweise gar keine geeignete Parametereinstellung findet.

Zur Vereinfachung dieses Problems kann ein Nutzer das Verhalten eines gewählten Suchverfahrens zur Laufzeit der Kalibrierung modifizieren. Damit der Nutzer nicht mit den eigentlichen Algorithmusparametern umgehen muss, wird in DAVINCI eine vereinfachte Schnittstelle bereitgestellt. Jedes Black-Box Suchverfahren verfügt über eine für den Nutzer identische Algorithmusfassade [48]. Über diese Fassade kann der Nutzer jedes Black-Box-Suchverfahren über zwei einfache Anweisungen modifizieren:

- **Gröber Suchen:** Durch diese Anweisung wird das Suchverfahren veranlasst weniger lokale und eher globale (eher zufällige) Suchschritte durchzuführen.
- **Feiner Suchen:** Durch diese Anweisung wird das Suchverfahren veranlasst, hauptsächlich lokale Suchschritte, ausgehend von der besten bekannten Parametereinstellung, durchzuführen.

Die tatsächlichen Auswirkungen der gewählten Änderungsanweisung sind Algorithmus-spezifisch. Die beiden Änderungsanweisungen bilden daher im fest vorgegebene Algorithmusfassaden [48] für Änderungen an den Steuerungsparametern der Kalibrierungs-algorithmen. Die genauen im Werkzeug realisierten Auswirkungen werden im Folgenden für Simulated Annealing, Evolutionäre Strategien und Numerische Suche beschrieben.

### 12.1.1 Simulated Annealing

Bei Simulated Annealing hängt die Annahmewahrscheinlichkeit für eine schlechter bewertete Parametereinstellung vom aktuellen Wert der Temperaturvariablen ab. Die Algorithmusfassade für Simulated Annealing ist in DAVINCI so realisiert, dass der Nutzer interaktiv die Annahmewahrscheinlichkeit für bewertete Parametereinstellungen modifizieren kann. Hierzu wird in DAVINCI eine Modifikatorvariable  $vmod$  eingeführt, die immer mit dem Ergebnis des Metropolis-Kriteriums multipliziert wird. Initial wird die Wirkung von  $vmod$  neutral 1 gewählt. Bei Vergrößerung der Suche wird die Annahmewahrscheinlichkeit verdoppelt ( $vmod = vmod * 2$ ) und entsprechend bei Verfeinerung halbiert.

### 12.1.2 Evolutionäre Strategie

Für die Evolutionäre Strategie werden in DAVINCI die folgenden Auswirkungen für die Änderungsfassade verwendet:

- **Verfeinerung:** Die Anzahl an Mutationen wird verdoppelt und die Anzahl an Crossover-Operationen wird halbiert. Die Populationsgröße wird ebenfalls halbiert, bleibt jedoch immer mindestens 1.
- **Vergrößerung:** Die Anzahl an Mutationen wird halbiert und die Anzahl an Crossover-Operationen verdoppelt. Die Größe der Population wird ebenfalls verdoppelt. Neu hinzukommende Parametereinstellungen zu kalibrierender Agenten werden mit zufälligen Parameterwerten initialisiert.

### 12.1.3 Numerische Suche

Die Präzision der numerischen BFGS-Suche hängt stark davon ab, wie genau der Gradient an einer bestimmten Stelle des Suchraumes geschätzt wird. Die Schätzung des Gradienten hängt wiederum von der Wahl des Verschiebungswertes  $h$  ab. Ein größerer Wert führt dazu, dass kleine lokale Schwankungen im Suchraum nicht ins Gewicht fallen. Je größer der Wert jedoch wird, desto zufälliger werden die Suchraumbewegungen.

Entsprechend werden die Auswirkungen der Änderungsfassade in Modifikationen des  $h$  Wertes umgesetzt. Vergrößerung führt zu einer Verdopplung des  $h$  Wertes, während Verfeinerung ihn halbiert.

## 12.2 Laufzeitinteraktion mit dem DAVINCI-Kalibrierer

Nach dem Start eines Kalibrierungslaufes kann der Nutzer abwarten, bis die Kalibrierung beendet ist und dann die Kalibrierungsergebnisse analysieren. Zusätzlich bietet DAVINCI dem Nutzer aber auch die Möglichkeit während der Laufzeit der Kalibrierung mit dem Kalibrierungslauf zu interagieren, Einstellungen für Suchverfahren zu verändern, Zwischenergebnisse zu analysieren und zusätzliche Nutzerdefinierte Parametereinstellungen der laufenden Kalibrierung hinzuzufügen. Solche Nutzer-Interaktionen mit dem laufenden Kalibrierungsverfahren können dann wichtig sein, wenn der Nutzer unsicher ist, ob er alle Algorithmuseinstellungen für die Kalibrierung sinnvoll vorgenommen hat. Der DAVINCI-Kalibrierer bietet dem Nutzer die Möglichkeit während Kalibrierungsläufen das Verhalten der Algorithmen und die bereits getesteten Parametereinstellungen zu analysieren und zu adaptieren. Auf der Basis dieser Analysen kann der Nutzer in die Kalibrierung eingreifen, ohne dass der Lauf notwendigerweise neu gestartet werden muss.

### 12.2.1 Interaktionsdialog für das aktuell laufende Suchverfahren

Der Interaktionsdialog (siehe Abbildung 12.1) für das aktuell laufende Black-Box-Kalibrierungsverfahren bildet eine Fassade für alle im Werkzeug verfügbaren Suchverfahren, d.h. der Nutzer interagiert immer mit dem gleichen Dialog, egal welches Kalibrierungsverfahren er gewählt hat.

Der Interaktionsdialog ist in fünf Bereiche gegliedert:

- **Aktuelle Bewertung:** An oberster Stelle wird die aktuell beste Bewertung angezeigt.
- **Analysebereich:** Der zweite Bereich umfasst Möglichkeiten zur Analyse des Kalibrierungsfortschrittes. Über die Anzeige des Bewertungsverlaufes, kann der Nutzer Rückschlüsse über die Konvergenz des Suchverfahrens ziehen. Über den Knopf „Results“ gelangt der Nutzer in die Detailansicht für die bisher bewerteten Parametereinstellungen (siehe Abbildung 12.2). Der Analysebereich bildet die Grundlage für die Nutzung der übrigen Interaktionsbereiche. Nur durch Analyse des Verhaltens des Kalibrierungsalgorithmus und der bereits ausgewerteten Parametereinstellungen kann der Nutzer Rückschlüsse zur geeigneten Interaktion mit dem Suchverfahren ziehen.

- **Ressourcenbereich:** Der dritte Bereich umfasst Möglichkeiten die vom Suchverfahren verwendeten Berechnungsressourcen (= verfügbare Simulationsläufe) anzupassen. Die Änderung der Berechnungsressourcenzuweisung für ein Suchverfahren kann dazu dienen, Ressourcen des aktuellen Suchverfahrens aufzustocken, wenn der Nutzer erkennt, dass das Suchverfahren noch einige Iterationen länger suchen sollte. Umgekehrt kann ein Nutzer die noch verbliebenen Berechnungsressourcen des aktuell durchgeführten Suchverfahrens reduzieren oder das Verfahren ganz überspringen. Das aktuelle Ergebnis des Verfahrens und die übrig gebliebenen Simulationsläufe werden dann automatisch zur nächsten Kalibrierungstechnik übertragen. Nicht verwendete Simulationsläufe kommen wieder in den Pool insgesamt verfügbarer Berechnungsressourcen.
- **Nutzer-Vorgaben für Parameterwerte des Modells:** Der vierte Bereich erlaubt es dem aktuellen Black-Box Kalibrierungsverfahren zusätzliche Nutzer-definierte Parametereinstellungen hinzuzufügen. Werden einem Verfahren zusätzliche Nutzer-definierte Parametereinstellungen hinzugefügt, werden diese automatisch im Suchprozess mit geändert. In globalen Suchtechniken werden die zusätzlichen Parameterwerte in die Lösungspopulation des Verfahrens übernommen. In lokalen Suchtechniken werden die zusätzlichen Parameterwerte in Konkurrenz zum aktuellen Suchraum des Verfahrens ausgewertet. Durch die Eingabe Nutzer-definierter Parameterwerte in ein laufendes Suchverfahren, kann der Nutzer z.B. „vernachlässigte“ Suchraumteile untersuchen oder versuchen einzelne Schritte des Suchverfahrens zu überspringen, indem er Tendenzen im Verlauf des Suchraums selbst in Schätzungen vielversprechender Parametereinstellungen umsetzt. Wie der Nutzer entsprechende Analysen durchführen kann, wird im Folgenden beschrieben.
- **Nutzer-Algorithmus-Interaktion:** Im fünften Bereich kann der Nutzer das Verhalten des aktuellen Black-Box-Kalibrierungsverfahrens modifizieren (siehe Abschnitt 12.1).

### 12.2.2 Analysebereich: Konvergenzanalyse

Durch die Visualisierung des Verlaufs der Validitätsbewertungen der letzten  $n$  bewerteten Parametereinstellungen wird es dem Nutzer ermöglicht, sich ein Bild des Bewertungsverlaufs des von ihm verwendeten Suchverfahrens zu machen.

Zwei Beobachtungen sind hierbei relevant:

- **Ist das Verfahren weitgehend konvergiert** (kaum noch Änderungen in den Validitätsbewertungen), hat es aber noch keine Parametereinstellung mit ausreichend validem Verhalten gefunden, liegt die Vermutung nahe, dass das Verfahren auf ein lokales Optimum des Suchraums hin konvergiert.  
In diesem Fall sollte die Suchstrategie des Verfahrens wieder vergrößert oder die Kalibrierung mit anderen Einstellungen neu gestartet werden.
- **Verhält sich das Verfahren wie eine zufällige Suche** (große Sprünge in den Validitätsbewertungen), sollte aber einen bestimmten Bereich im Parametersuchraums genauer untersuchen, sollte die Suchstrategie des Verfahrens zu verfeinert und gegebenenfalls günstigere Startwerte für eine verfeinerte Suche gesetzt werden.

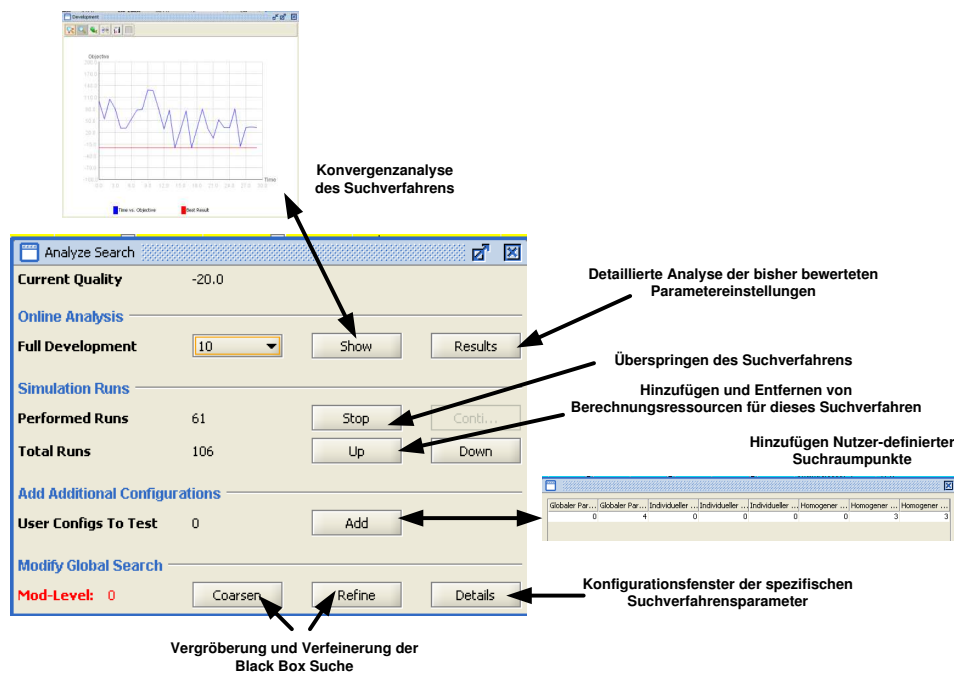


Abbildung 12.1: Dialogfassade für die Interaktion mit den Black-Box-Suchverfahren.

Suchraumanalysen können vom Nutzer durchgeführt werden, indem er den zugehörigen Detailanalysedialog aufruft.

### 12.2.3 Analysebereich: Detailansicht für Parametereinstellungen

Durch Betätigung des Knopfs „Results“ in Dialog 12.1 gelangt der Nutzer in die in Abbildung 12.2 dargestellte Detailansicht für die bisher bewerteten Parametereinstellungen. Hier kann der Nutzer verschiedene Techniken zur Analyse des bisher bekannten Parametersuchraumes durchführen. Im mittleren Bereich des Dialogs wird die Liste der bisher vom Suchverfahren ausgewerteten Parametereinstellungen und die zugehörigen Werte der Beobachtungsvariablen und Zielfunktion(en) angezeigt.

Diese kann im unteren Bereich nach unterschiedlichen Kriterien sortiert und gefiltert werden. Dies wird in den folgenden Abschnitten genauer erklärt. Zusätzlich kann die Liste der Parametereinstellungen in einer CSV-Datei (Comma-Separated-Values) abgespeichert werden, die dann in einem Statistikwerkzeug wie z.B. Statistica<sup>1</sup> oder Microsoft Excel<sup>2</sup> weiterverarbeitet werden kann.

Durch Analyse der bereits ausgewerteten Parametereinstellungen kann der Nutzer:

- erkennen, dass bestimmte **Bereiche des Suchraums vom Suchverfahren „vernachlässigt“** werden. In diesem Fall kann der Nutzer dafür sorgen, dass auch diese Bereiche durchsucht werden.

<sup>1</sup><http://www.statsoft.de/> (November 2010)

<sup>2</sup><http://office.microsoft.com/en-us/excel/microsoft-excel-spreadsheet-FX101825647.aspx> (November 2010)

**Analysebereich für bewertete Parametereinstellungen**

**Filterungs- und Sortierfunktionen auf der Basis unterschiedlicher Beobachtungs- und Zielfunktionsvariablen**

**Zwischenergebnisse können in CSV-Datei abgespeichert werden**

Abbildung 12.2: Detailansicht für die bisher bewerteten Parametereinstellungen (Ränder dienen der Markierung einzelner Bereiche).

- die **ungefähre Lage von Optima im Parametersuchraum erkennen**. In diesem Fall kann es von Interesse sein, explizit zu untersuchende Parametereinstellungen im Bereich der Optima vorzugeben oder die Suchstrategie des aktuellen Suchverfahrens zu verfeinern.

### 12.2.3.1 Visualisierung und Regressionsanalyse

Der obere Bereich der Detailansicht in Abbildung 12.2 bietet die Möglichkeit die Parameterkombinationen in ein- und zweidimensionalen Unterräumen des Parametersuchraumes zu visualisieren und zusätzlich verschiedene mögliche Verläufe der Suchraumoberfläche (Werte der Zielfunktion oder von Beobachtungsvariablen) darzustellen. Bei der Analyse wählt der Nutzer zunächst bis zu zwei Parameter und die erwünschte Zielfunktion, bzw. den Beobachtungswert, bezüglich der die Analyse durchgeführt werden soll. Der Nutzer hat entweder die Möglichkeit einfach einen entsprechenden 2D- oder 3D-Plot anzuzeigen, der die Daten aller bereits durchgeführten Parametereinstellungen enthält oder er lässt vom DAVINCI Kalibrierer zusätzlich ein Polynom vom Grad  $n$  zur Visualisierung einer Annäherung des Suchraumverlaufs schätzen und in der Anzeige darstellen.

Gemäß dem Prinzip der Response Surface Methodologie (siehe Abschnitt 3.2.7) wird es dem Nutzer in DAVINCI ermöglicht, automatisch mögliche Oberflächen des Parametersuchraumes zu visualisieren und so Rückschlüsse über mögliche Verläufe der Zielfunktion zu ziehen. Die Erzeugung der Suchraumboflächen erfolgt durch die automatische Schätzung von Regressionsmodellen für Polynome vom Grad  $n$ . Die geschätzten Polynome werden in DAVINCI zusammen mit den Parameterwerten dreidimensional visualisiert. Der Grad des erwünschten Polynoms ist vom Nutzer spezifizierbar, so dass unterschiedliche mögliche Suchraumboflächen verglichen werden können. Abbildung 12.3 zeigt zwei Ansichten aus einer dreidimensionalen Visualisierung zweier Parameterwerte und der Zielfunktion mit einer aus diesen Punkten geschätzten Polynomoberfläche vom Grad 2.

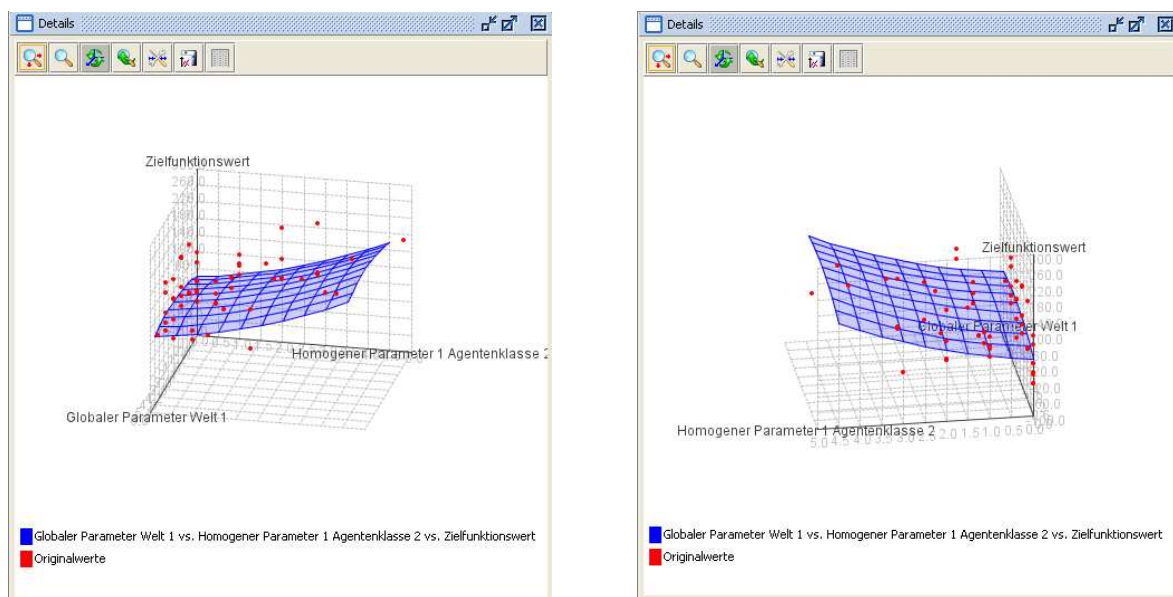


Abbildung 12.3: Zwei Ansichten einer dreidimensionalen Visualisierung des Parametersuchraumes mit einer Oberflächen-schätzung durch ein Polynom vom Grad 2.

Ziel der Analyse in der Detailansicht für die bisher bewerteten Parametereinstellungen ist es, während lange dauernder Kalibrierungsläufe (viele Parametereinstellungsauswertungen und lange Simulationsläufe) frühzeitig Tendenzen im Verlauf des Suchraums, vielversprechende, noch nicht ausgewertete Parametereinstellungen und „vernachlässigte“ Suchraumbereiche zu erkennen und geeignet Einfluss auf das Suchverfahren zu nehmen.

Ein weiteres Ziel besteht darin, die Kalibrierung selbst als Analysewerkzeug für das kalibrierte Modell einzusetzen. Dies kann dadurch geschehen, dass die Kalibrierung und die Analyse der bewerteten Parametereinstellungen eingesetzt wird, um ein allgemeines Verständnis für die Zusammenhänge des Parametersuchraumes zu erlangen.

### 12.2.3.2 Identifikation „interessanter“ Suchraumpunkte

Bei der Analyse der bereits ausgewerteten Parametereinstellungen ist es für den Nutzer nicht nur interessant, valides Verhalten zu identifizieren, sondern auch zu erkennen, ob eine Menge von unterschiedlichen Parametereinstellungen zu ähnlichem Simulationsverhalten

führt (multiple Optima des Suchraumes). Die Identifikation solcher unterschiedlicher lokaler Suchraumoptima innerhalb der Liste aller bisher bewerteten Parametereinstellungen kann schwierig sein, da die Liste viele sehr ähnliche Parametereinstellungen enthalten kann, von denen eigentlich nur derjenige mit der höchsten Validitätsbewertung interessant ist (z.B. Einstellungen, die von der Suche auf dem Weg zu einem (lokalen) Optimum bewertet wurden).

Abbildung 12.4 zeigt ein einfaches Beispiel für das Problem der Identifikation interessanter Optima (in diesem Fall Maxima der durch die Kurve dargestellten Zielfunktion). Im Beispiel wurden sieben Parametereinstellungen bewertet (diese werden also in der Liste angezeigt). Tatsächlich interessant sind für den Nutzer aber nur die Parametereinstellungen eins und sieben, da sie lokale Optima des Suchraums darstellen. Die übrigen Punkte sind für den Nutzer nicht interessant, da sie nur Zwischenergebnisse auf dem Weg zu eins und sieben sind. Insbesondere ist deshalb auch der Punkt zwei uninteressant, obwohl er in etwa gleich gut bewertet wird wie sieben.

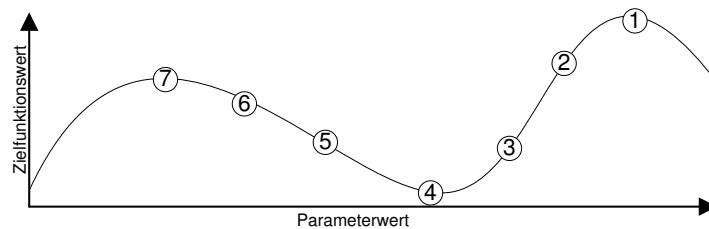


Abbildung 12.4: Filterung innerhalb von sieben bewerteten Parametereinstellungen.

Bei der Identifikation von interessanten Suchraumoptima wird der Nutzer im DAVINCI-Kalibrierer durch einen zusätzlichen Filtermechanismus unterstützt, der ausgehend von den bisher identifizierten validesten Punkten des Suchraumes alle Punkte, die in der Nähe oder auf dem Weg zu diesen Optima liegen, ausfiltert:



**Algorithmus Listenfilterung:****Eingabe:**

- Zu filternde Liste bewerteter Parametereinstellungen.
- Nachbarschaftskriterium für Parametereinstellungen.

**Ausgabe:**

- Gefilterte Liste, die alle Optima der Eingabeliste enthält.

**Hilfsvariable:**

- Liste ausgefilterter Elemente.

**Ablauf:**

1. Sortiere Eingabeliste absteigend nach der Validitätsbewertung.
2. Solange noch Elemente in der Eingabeliste sind
  - (a) Entferne vorderstes Element der Eingabeliste und setze es auf die Ergebnisliste.
  - (b) Durchlaufe die übrige Eingabeliste von vorne nach hinten
    - i. Ist das aktuelle Element in der Nachbarschaft eines Elementes auf der Ergebnisliste oder in der Nachbarschaft eines Elementes auf der Liste ausgefilterter Elemente und nicht besser bewertet als dieses, dann entferne es aus der Eingabeliste und füge es der Liste ausgefilterter Elemente hinzu.



# Kapitel 13

## Der DAVINCI Kalibrierer: Vergleich

In diesem Kapitel werden die Eigenschaften des DAVINCI-Kalibrierers mit denen existierender Werkzeuge verglichen, und DAVINCI so von diesen abgegrenzt. Da es zum Zeitpunkt der Erstellung dieser Arbeit keine existierenden Kalibrierungswerkzeuge für ABS gab, wurden als Vergleichswerkzeuge folgende, erfolgreiche existierende Kalibrierungs- bzw. Optimierungswerkzeuge für allgemeine Simulationen ausgewählt:

- **Witness Optimizer (Lanner)**<sup>1</sup>
- **RISK Optimizer (Palisade Corporation)**<sup>2</sup>
- **OptQuest (OptTek)**<sup>3</sup>
- **ISSOP (Dualis IT)**<sup>4</sup>

Die Informationen über diese Werkzeuge wurden den in den Fußnoten genannten frei verfügbaren Quellen entnommen.

### 13.1 Vergleichstabellen

Tabelle 13.1 vergleicht die Werkzeuge nach den verschiedenen Möglichkeiten Kalibrierungsprobleme zu deklarieren. Tabelle 13.2 stellt die verschiedenen möglichen Kalibrierungsalgorithmen und bereitgestellten Techniken der Werkzeuge gegenüber. Die letzte Tabelle 13.3 betrachtet die Möglichkeiten mit den Kalibrierungswerkzeugen zu interagieren.

### 13.2 Ergebnis des Vergleichs

Der Vergleich des DAVINCI-Kalibrierers mit den ausgewählten existierenden Kalibrierungs- bzw. Optimierungswerkzeugen kann wie folgt zusammengefasst werden.

---

<sup>1</sup><http://www.lanner.com/en/witness.cfm> (November 2010)

<sup>2</sup><http://www.palisade-europe.com/riskoptimizer/> (November 2010)

<sup>3</sup><http://www.opttek.com/Products/OptQuest.html> (November 2010)

<sup>4</sup><http://www.dualis-it.de/index.php?pageid=358> (November 2010)

Werkzeug	Fokus	Zielfunktionen	Parameterdeklaration	Randbedingungen
Witness Optimizer	<ul style="list-style-type: none"> <li>Fertigungsplanung</li> <li>Logistik</li> </ul>	<ul style="list-style-type: none"> <li>1 globale Zielfunktion</li> </ul>	<ul style="list-style-type: none"> <li>Globale Parameter, d.h. alle Parameter auf gleicher Ebene</li> </ul>	<ul style="list-style-type: none"> <li>Möglich (Typ unbestimmt)</li> </ul>
RISK Optimizer	<ul style="list-style-type: none"> <li>Fertigungsplanung</li> <li>Logistik</li> </ul>	<ul style="list-style-type: none"> <li>1 globale Zielfunktion</li> </ul>	<ul style="list-style-type: none"> <li>Globale Parameter, d.h. alle Parameter auf gleicher Ebene</li> </ul>	<ul style="list-style-type: none"> <li>Möglich</li> <li>Hart und Weich</li> <li>Auswertung vor und nach Simulationsläufen</li> </ul>
OptQuest	<ul style="list-style-type: none"> <li>Fertigungsplanung</li> <li>Logistik</li> </ul>	<ul style="list-style-type: none"> <li>1 globale Zielfunktion</li> </ul>	<ul style="list-style-type: none"> <li>Globale Parameter, d.h. alle Parameter auf gleicher Ebene</li> </ul>	<ul style="list-style-type: none"> <li>Möglich (Hart)</li> </ul>
ISSOP	<ul style="list-style-type: none"> <li>Fabricsimulation</li> <li>Fertigungsplanung</li> <li>Logistik</li> </ul>	<ul style="list-style-type: none"> <li>Multiple globale Zielfunktionen möglich (Pareto-Optimierung)</li> </ul>	<ul style="list-style-type: none"> <li>Globale Parameter, d.h. alle Parameter auf gleicher Ebene</li> </ul>	<ul style="list-style-type: none"> <li>Möglich (unbestimmt)</li> </ul>
DAVINCI	<ul style="list-style-type: none"> <li>ABS unterschiedlicher Domänen</li> </ul>	<ul style="list-style-type: none"> <li>Globale und lokale Zielfunktionen möglich</li> <li>Multiple Zielfunktionen möglich (Pareto-Optimierung)</li> </ul>	<ul style="list-style-type: none"> <li>Globale Parameterwerte</li> <li>Individuelle Parameterwerte</li> <li>Parameterwerte für Gruppen von Agenten</li> </ul>	<ul style="list-style-type: none"> <li>Möglich (Hart)</li> <li>Auswertung vor, während und nach Simulationsläufen</li> </ul>

Tabelle 13.1: Werkzeugvergleich: Kalibrierungsproblemdeklaration.

Werkzeug	Black-Box Verfahren	White-Box Verfahren	Wissensbasierte Verfahren	Mehr-Ebenen Probleme	Spezialalgorithmen
Witness Optimizer	<ul style="list-style-type: none"> <li>Zufallssuche</li> <li>Vollständige Suche</li> <li>Hill Climbing</li> <li>Simulated Annealing Variante</li> <li>Streuungssuche (Design of Experiments)</li> </ul>	<ul style="list-style-type: none"> <li>Keine</li> </ul>	<ul style="list-style-type: none"> <li>Keine</li> </ul>	<ul style="list-style-type: none"> <li>Keine Unterstützung</li> </ul>	<ul style="list-style-type: none"> <li>Six Sigma Algorithmus, zur automatischen Adaption eines Simulated Annealing Verfahrens zur Laufzeit</li> </ul>
RISK Optimizer	<ul style="list-style-type: none"> <li>Genetischer Algorithmus</li> </ul>	<ul style="list-style-type: none"> <li>Keine</li> </ul>	<ul style="list-style-type: none"> <li>Keine</li> </ul>	<ul style="list-style-type: none"> <li>Keine Unterstützung</li> </ul>	<ul style="list-style-type: none"> <li>Keine</li> </ul>
OptQuest	<ul style="list-style-type: none"> <li>Scatter-Suche</li> <li>Tabu Suche</li> <li>Ganzzahlige lineare Optimierung</li> </ul>	<ul style="list-style-type: none"> <li>Keine</li> </ul>	<ul style="list-style-type: none"> <li>Keine</li> </ul>	<ul style="list-style-type: none"> <li>Keine Unterstützung</li> </ul>	<ul style="list-style-type: none"> <li>Neuronales Netz als Filter für Parametereinstellungen.</li> </ul>
ISSOP	<ul style="list-style-type: none"> <li>Komponentenweise Enumeration</li> <li>Diskretes Gradienten-Verfahren</li> <li>Evolutionsstrategien</li> <li>Zufallssuche</li> <li>Genetischer Algorithmus</li> <li>Schwellwert Algorithmus</li> <li>Quaderalgorithmus</li> </ul>	<ul style="list-style-type: none"> <li>Keine</li> </ul>	<ul style="list-style-type: none"> <li>Keine</li> </ul>	<ul style="list-style-type: none"> <li>Keine Unterstützung</li> </ul>	<ul style="list-style-type: none"> <li>Quasi-gleichzeitige Durchführung aller Black-Box Verfahren.</li> <li>Neuronales Lern- und Adaptionssystem zur Algorithmensteuerung.</li> </ul>
DAVINCI	<ul style="list-style-type: none"> <li>Zufallssuche</li> <li>Evolutionsstrategien</li> <li>Simulated Annealing Variante</li> <li>Gradienten-basierte Suche (BFGS)</li> <li>Streuungsanalyse (Design of Experiments)</li> </ul>	<ul style="list-style-type: none"> <li>Sequenzialisierung von Kalibrierungsläufen zur Hintereinanderschaltung mehrerer Teilkalibrierungsläufe bei Einsatz von White-Box-Zerlegungen</li> </ul>	<ul style="list-style-type: none"> <li>Nutzerdefinierbare Änderungsregeln</li> <li>für beliebige Parameterwerte</li> <li>auf globaler oder lokaler Ebene</li> </ul>	<ul style="list-style-type: none"> <li>Unterstützung des Makro-Mikro-Verfahrens</li> </ul>	<ul style="list-style-type: none"> <li>Black-Box und wissensbasierte Ansätze kombinierbar</li> <li>Heterogenitätssuche</li> <li>Multiple Szenarien</li> <li>Robuste Kalibrierung</li> <li>Suchraumgrößen abhängige Verteilung der Simulationsläufe auf Teilkalibrierungen</li> </ul>

Tabelle 13.2: Werkzeugvergleich: Techniken.

Werkzeug	Abbruchkriterien Kalibrierung	Visualisierung	Online Interaktion	Integration
<b>Witness Optimizer</b>	<ul style="list-style-type: none"> <li>• Laufzeit</li> </ul>	<ul style="list-style-type: none"> <li>• Tabelle bewerteter Parametereinstellungen</li> <li>• Verlauf der Zielfunktion</li> <li>• Varianzgraph für iterierte Auswertungen von Parametereinstellungen</li> </ul>	<ul style="list-style-type: none"> <li>• Aufruf der Visualisierungen</li> <li>• Sortieren, Drucken der Parametereinstellungen</li> </ul>	<ul style="list-style-type: none"> <li>• WITNESS Simulator</li> </ul>
<b>RISK Optimizer</b>	<ul style="list-style-type: none"> <li>• Konvergenzüberwachung</li> <li>• Randbedingungen</li> <li>• Anzahl an Simulationen</li> <li>• Kalibrierungsdauer</li> </ul>	<ul style="list-style-type: none"> <li>• Tabelle bewerteter Parametereinstellungen</li> <li>• Verlauf der Zielfunktion</li> <li>• Einhaltung von Randbedingungen für Parametereinstellungen</li> </ul>	<ul style="list-style-type: none"> <li>• Aufruf der Visualisierungen</li> </ul>	<ul style="list-style-type: none"> <li>• Integration in Microsoft Excel</li> </ul>
<b>OptQuest</b>	<ul style="list-style-type: none"> <li>• Konvergenzüberwachung</li> <li>• Nutzerabbruch</li> </ul>	<ul style="list-style-type: none"> <li>• Tabelle bewerteter Parametereinstellungen</li> <li>• Verlauf der Zielfunktion</li> </ul>	<ul style="list-style-type: none"> <li>• Aufruf der Visualisierungen</li> </ul>	<ul style="list-style-type: none"> <li>• Universell einsetzbare Bibliothek. Anwendung z.B. in Anylogic.</li> </ul>
<b>ISSOP</b>	<ul style="list-style-type: none"> <li>• Keine Informationen</li> </ul>	<ul style="list-style-type: none"> <li>• Aktueller Status aller Verfahren.</li> <li>• Tabelle der ausgewerteten Parametereinstellungen.</li> <li>• Visualisierung der aktuellen Paretofläche.</li> </ul>	<ul style="list-style-type: none"> <li>• Aufruf der Visualisierungen</li> </ul>	<ul style="list-style-type: none"> <li>• Universell einsetzbare Bibliothek. Anwendung z.B. in eM-Plant, Enterprise Dynamics, ARENA, Automod und MATLAB</li> </ul>
<b>DAVINCI</b>	<ul style="list-style-type: none"> <li>• Randbedingungen</li> <li>• Anzahl an Simulationen</li> <li>• Kalibrierungsdauer</li> <li>• Nutzerabbruch</li> </ul>	<ul style="list-style-type: none"> <li>• Tabelle bewerteter Parametereinstellungen</li> <li>• Verlauf der Zielfunktion</li> <li>• 2D/3D-Projektionen von Suchraumteilen</li> <li>• Automatische Schätzung von Verlaufsmodellen für Teilsuchräume</li> </ul>	<ul style="list-style-type: none"> <li>• Sortierung und Filterung der Tabelle bewerteter Parametereinstellungen.</li> <li>• Aufruf der Visualisierungen</li> <li>• Manuelle Änderung der Suchalgorithmusparameter direkt oder über Fassade.</li> <li>• Deklaration zusätzlicher Nutzer-definierter Parametereinstellungen.</li> <li>• Hinzufügen/Entfernen von Simulationsläufen zu Kalibrierungen.</li> <li>• Überspringen von Teilkalibrierungsschritten.</li> <li>• Nutzerabbruch.</li> </ul>	<ul style="list-style-type: none"> <li>• SeSAM Simulator und Modellierungsumgebung</li> </ul>

Tabelle 13.3: Werkzeugvergleich: Interaktion.

### 13.2.1 Fokus der Werkzeuge

Die meisten der betrachteten Werkzeuge fokussieren auf die Optimierung von Simulationen zur Logistik und Fertigungsplanung. Die OptQuest Bibliothek wird auch im Simulationstool *Anylogic* eingesetzt, mit dem Agenten-basierte Simulationen durchgeführt werden. DAVINCI fokussiert auf die Kalibrierung Agenten-basierter Simulationen und adressiert spezifische Eigenschaften von ABS-Kalibrierungsproblemen.

### 13.2.2 Zielfunktionen

Alle Werkzeuge bis auf ISSOP und DAVINCI verwenden nur eine globale Zielfunktion. In ISSOP und DAVINCI ist es möglich mehrere Zielfunktionen in einer Pareto-optimalen Suche zu verwenden. Nur in DAVINCI können auch Zielfunktionen auf individueller Agentenebene deklariert werden.

### 13.2.3 Parameterdeklaration

In allen Werkzeugen außer DAVINCI müssen alle Parameter für sich deklariert werden. Die Parameter befinden sich alle auf der gleichen globalen Ebene. In DAVINCI können vom

Nutzer sowohl globale als auch individuelle Agentenparameter deklariert werden. Individuelle Parameter können, müssen aber nicht einzeln deklariert werden. Stattdessen kann der Nutzer auch Parameter für eine Klasse von Agenten deklarieren, aber die konkreten Konfiguration des Parameters (Wertebereich, Startwert. . .) auf Klassen oder Gruppenebene durchführen.

#### 13.2.4 Suchverfahren

Alle existierenden Suchverfahren sind reine Black-Box-Optimierungswerkzeuge, für die der konkrete Typ eines Simulationsmodells nicht relevant ist. Auch die Spezialverfahren der existierenden Werkzeuge sind Verfahren für Black-Box-Optimierung, d.h. adressieren keine spezifischen Eigenschaften einzelner Optimierungsprobleme.

DAVINCI unterstützt neben der Black-Box-Kalibrierung auch wissensbasierte Suchverfahren und die Zerlegung von Kalibrierungsproblemen mit White-Box-Methoden. Nur DAVINCI unterstützt Verfahren, die speziell die Probleme von ABS adressieren, nämlich die Kalibrierung auf verschiedenen Beobachtungsebenen, die Heterogenitätssuche, die parallele Kalibrierung in mehreren Szenarien, und die Suche nach geeigneten Modellausprägungen. Auch der Umgang mit verrauschten Daten mit Hilfe eines robusten Suchverfahrens existiert nur in DAVINCI.

#### 13.2.5 Laufzeitinteraktion

Die möglichen Interaktionen des Nutzers mit den existierenden Kalibrierungsverfahren beschränken sich auf die Anzeige von Visualisierungen zum Suchfortschritt und den Abbruch der Kalibrierungsverfahren. DAVINCI erlaubt zusätzlich die Interaktion mit den Kalibrierungsalgorithmen durch die Modifikation der zugehörigen Steuerungsparameter über einfache Algorithmusfassaden. Weiter ermöglicht es DAVINCI dem Nutzer während der Kalibrierung zusätzliche Parametereinstellungen zum Kalibrierungslauf hinzuzufügen und so aktiv in die Auswertung von Parametereinstellungen einzugreifen.

#### 13.2.6 Integration

Nur DAVINCI und der WITNESS Optimizer sind Kalibrierer, die für spezielle Simulationswerkzeuge entwickelt wurden. OptQuest und ISSOP sind universell anwendbare Optimierungsbibliotheken, die dadurch aber nur schwer auf spezifische Eigenschaften von Optimierungsproblemen angepasst werden können. RISK ist ein Optimierer für Microsoft Excel und entsprechend nur auf in der Tabellenkalkulation formulierte Simulationen anwendbar.

#### 13.2.7 Insgesamt

Insgesamt ist DAVINCI das einzige der untersuchten Kalibrierungswerkzeuge, das spezifische Kalibrierungsprobleme von ABS adressiert und diese löst. Die übrigen Werkzeuge fokussieren auf kein spezielles Simulationsproblem, sondern führen reine Black-Box-Optimierungen durch. Angewendet auf ABS sind sie daher mit den geschilderten Problem reiner Black-Box-Kalibrierung konfrontiert.

**Teil IV**

**Evaluation und Ausblick**





## Kapitel 14

# Evaluation: Einleitung und Überblick

In dieser Arbeit wurde ein Makro-Mikro-Kalibrierungsverfahren für Agenten-basierte Simulationen, und verschiedene Techniken zum Umgang mit fehlenden, unsicheren oder veräuschten Daten und Wissen bei der Entwicklung und Kalibrierung von ABS beschrieben. Die vorgeschlagenen Kalibrierungstechniken wurden in einem Werkzeug, dem DAVINCI-Kalibrierer, umgesetzt. In den folgenden Kapiteln werden die beschriebenen Verfahren und Techniken evaluiert. Die Evaluation ist in mehrere Kapitel gegliedert:

- **Kapitel 15: Makro-Mikro-Verfahren:**

Zunächst wird das Makro-Mikro-Verfahren (siehe Abschnitt 9.1.3) mit Reverse-Engineering (siehe Abschnitt 9.2) bewertet und mit Black-Box-Verfahren verglichen.

- **Kapitel 16: Heterogenitätssuche:**

Im zweiten Abschnitt werden Randbedingungen untersucht, innerhalb derer sich das Verfahren zur Heterogenitätssuche (siehe Abschnitt 10) gewinnbringend eingesetzt werden kann und das Verfahren an Fallstudien demonstriert.

- **Kapitel 17: Robuste Kalibrierung:**

Im dritten Teil der Evaluation wird anhand einer Fallstudie untersucht, wie durch die Anwendung robuster Kalibrierung (siehe Abschnitt 9.3) kritische Validierungskriterien identifiziert und entsprechend ausgeglichen werden können.

Die Evaluation wird zum einen anhand künstlicher Problemstellungen und zum anderen anhand eines konkreten Simulationsmodells durchgeführt: Der Agenten-basierten Simulation menschlichen Einkaufsverhaltens in Nordschweden.

Das Simulationsmodell wurde in Abschnitt 7.5 mit Modellbeschreibung, verwendeten Daten, Modellierungs- und Kalibrierungsproblemen und identifizierbaren Unsicherheiten eingeführt. Analysen der zugehörigen Kalibrierungs- und Modellausprägungsprobleme finden sich in Kapitel 8.1. In den einzelnen Teilabschnitten dieses Evaluationsteils wird die Lösbarkeit der identifizierten Kalibrierungs- und Modellierungsprobleme mit dem Makro-Mikro-Verfahren untersucht. Das Einkaufsmodell wurde in Zusammenarbeit mit dem geographischen Institut der Universität Würzburg entwickelt. Die Evaluation wird durchgeführt, indem die Verfahren mit Black-Box-Optimierungsverfahren verglichen werden. Ein expliziter Vergleich mit White-Box-Techniken wird nicht geführt, da für die Fallstudien keine White-Box-Kalibrierungsregeln oder Zerlegungen möglich waren. Es wird aber auf

Möglichkeiten zur White-Box-Zerlegung des Kalibrierungsproblems im Rahmen der Diskussion des Makro-Mikro-Verfahrens eingegangen.

Als Evaluationskriterien werden verwendet:

- Die **Anzahl benötigter Simulationsläufe** zur Kalibrierung des Modells. Je weniger Simulationsläufe zum Erreichen validen Simulationsverhaltens benötigt werden, desto besser.
- Die **Größe des zu durchsuchenden Parametersuchraumes**. Je kleiner der zu durchsuchende Parametersuchraum, desto einfacher ist es, eine geeignete Parametereinstellung zum Erreichen validen Simulationsverhaltens zu identifizieren.
- Die **Qualität der gefundenen Lösungen**. Je höher die Validität des Simulationsverhaltens eines kalibrierten Modells, desto geeigneter ist das Modell zur Beantwortung der Simulationsfragen.

## 14.1 Vorbemerkung zu verwendeten Suchverfahren und Validierungskriterien

### 14.1.1 Verwendete Black-Box-Suchverfahren

In Abschnitt 3.2 wurde auf die große Anzahl existierender Black-Box-Suchverfahren hingewiesen. Zur Evaluierung der vorgestellten Techniken in diesem Abschnitt werden im wesentlichen zwei Suchverfahren, nämlich **Simulated Annealing** und **Evolutionäre Strategien**, allein oder in Kombination eingesetzt. Diese Verfahren wurden gewählt, da es sich um etablierte heuristische Kalibrierungsverfahren handelt. Auf die Anwendung reiner Gradienten-basierter Black-Box-Kalibrierungsverfahren ohne heuristische Anteile zur Kalibrierung der Simulationsmodelle wurde verzichtet. Die Gründe hierfür sind zum einen, dass die erfolgreiche Anwendung von Gradienten-Verfahren stark von einer guten Ausgangsparametereinstellung abhängt und zum anderen, dass zur Berechnung jeder einzelnen Komponente eines Gradientenvektors ein eigener Simulationslauf durchgeführt werden muss, was die Gradienten-Berechnung für große Parameterzahlen nicht praktikabel macht.

### 14.1.2 Validierungskriterium auf der Makro-Ebene

In Abschnitt 7.5 wurden zwei Validierungskriterien für die Fallstudie der Agenten-basierten Einkaufssimulation vorgestellt, das „unquadratische Gütemaß“ und das „echte unquadratische Gütemaß“. Es wurde weiter beschrieben, dass bei der Verwendung des unquadratischen Gütemaßes aufgrund dessen Struktur

der Verschätzungsgrad von Geschäften mit sehr großem Zielumsatz wesentlich stärker einfließt als der von Geschäften mit kleinem Umsatz. Eine Folge dieser Eigenschaft ist, dass der zu  $ZF_{unquadrat}$  gehörende Suchraum viele lokale (und globale) Optima aufweist. Geschäfte mit kleinem Umsatz können auf unterschiedliche Art verschätzt werden, ohne dass dies großen Einfluss auf den Wert von  $ZF_{unquadrat}$  haben muss. Das Gütemaß wird in diesem Evaluationsteil dennoch verwendet, da es das von den Domänenexperten als geeignet bewertet und daher während des gesamten zur Fallstudie gehörenden Projektes eingesetzt wurde.

Untersuchte Verfahren	Durchgeführte Untersuchungen	Ergebnisse
Makro-Mikro-Verfahren (MMV) mit Reverse-Engineering (RE)	<ul style="list-style-type: none"> <li>• Diskussion MMV an künstlichem Beispielmodell.</li> <li>• Kalibrierung Fallstudie Einkaufssimulation (<math>F_{EK}</math>): Homogene Parameter.</li> <li>• Kalibrierung <math>F_{EK}</math>: Heterogene Parameter.</li> <li>• Vergleich MMV mit Black-Box-Suche.</li> <li>• White-Box-Zerlegung.</li> </ul>	<ul style="list-style-type: none"> <li>• MMV leicht bessere als <i>Black-Box-Suche</i>.</li> <li>• MMV weniger Simulationsläufe als <i>Black-Box-Suche</i>.</li> <li>• MMV ermöglichte <i>White-Box-Zerlegungen</i>.</li> <li>• RE ermöglichte Ermittlung von <i>IVM</i>, deren Ermittlung mit reiner <i>Black-Box-Suche</i> zu aufwendig war.</li> <li>• RE-Algorithmus und Randbedingungen an <i>IVM</i> haben entscheidenden Einfluss auf Qualität der <i>IVM</i>.</li> <li>• <i>IVM</i> und Mikro-Modelteile in starker Abhängigkeit. Können Mikro-Modelteile nicht passend zu <i>IVM</i> kalibriert werden, kann ist Kalibrierung nicht erfolgreich.</li> </ul>
Heterogenitätssuche mit verschiedenen Abbruchkriterien	<ul style="list-style-type: none"> <li>• Heterogenitätssuche (<i>HetS</i>) an Beispielmodell.</li> <li>• <i>HetS</i> für <math>F_{EK}</math> mit Gruppierungskriterium <i>Stadt/Umland/Land</i>.</li> <li>• <i>HetS</i> für <math>F_{EK}</math> mit Gruppierungskriterium <i>Nähe der Agenten zueinander</i>.</li> <li>• Vergleich Kalibrierungsaufwand <i>HetS</i> mit <i>Black-Box-Suche</i> vollständig heterogener Parameterwerte.</li> <li>• Wird optimale Heterogenitätseinstellung gefunden?</li> <li>• Untersuchung: versch. Abbruchkriterien.</li> <li>• Untersuchung: versch. <i>ZF</i>.</li> </ul>	<ul style="list-style-type: none"> <li>• In allen Untersuchungen fand die <i>HetS</i> für das gegebene Problem die optimale Heterogenitätseinstellung.</li> <li>• Im künstlichen Beispielmodell konnte durch <i>HetS</i> bis zu 50% an Simulationsläufen eingespart werden.</li> <li>• In <math>F_{EK}</math> hatten die Wahl des Gütemaßes und der Ähnlichkeitsfunktion Einfluss auf die Heterogenitätssuche.</li> </ul>
Robuste Kalibrierung	<ul style="list-style-type: none"> <li>• Robuste Kalibrierung der <math>F_{EK}</math> in Kombination mit <i>RE</i>.</li> <li>• Untersuchung, ob Daten bezüglich einzelner Geschäfte signifikant stärker verfälscht sind als die anderer, so dass aufgrund dieser Geschäfte nicht ausreichend valides Agentenverhalten erzeugt wird.</li> <li>• Untersuchung, ob resultierende fehlerhafte Verhaltensmodelle der Agenten geeignet ersetzt werden können.</li> </ul>	<ul style="list-style-type: none"> <li>• Verfahren zur robusten Kalibrierung in der Fallstudie einsetzbar.</li> <li>• Ergebnis der Untersuchung: Daten zu allen Geschäften in etwa gleich verrauscht waren.</li> <li>• Fallstudie zeigte: Die robuste Suche kann zur Analyse der Art des Datenfehlers verwendet werden.</li> </ul>

Tabelle 14.1: Überblick über die durchgeführten Evaluationsuntersuchungen.

## 14.2 Überblick über die Evaluation und ihre Ergebnisse

Tabelle 14.1 zeigt einen kurzer Überblick über die untersuchten Verfahren, die durchgeführten Untersuchungen und die Evaluationsergebnisse.



## Kapitel 15

# Evaluation: Makro-Mikro-Verfahren für ABS

In diesem Kapitel wird das entwickelte Makro-Mikro-Verfahren für ABS zunächst an einem einfachen Beispielmmodell diskutiert und die identifizierten Punkte anhand der konkreten Fallstudie der Agenten-basierten Einkaufssimulation evaluiert. Dabei wird das Makro-Mikro-Verfahren mit der Anwendung eines reinen Black-Box-Kalibrierungsansatzes verglichen. Ein getrennter Vergleich mit White-Box oder Regel-basierter Kalibrierung wird nicht geführt. Unabhängig vom Makro-Mikro-Verfahren konnten in der Fallstudie weder White-Box-Zerlegungen angewendet noch Kalibrierungsregeln spezifiziert werden. Im Rahmen des Makro-Mikro-Verfahrens wurden aber sowohl White-Box-Techniken als auch ein wissensbasierter Kalibrierungsalgorithmus eingesetzt. Diese werden im Rahmen der Diskussion des Makro-Mikro-Verfahrens beschrieben.

### 15.1 Evaluationsziele

Die Evaluationsziele dieses Abschnitts sind:

- Anhand eines künstlichen Beispielmmodells sollen Anforderungen zur Anwendbarkeit des Makro-Mikro-Verfahrens und Vor- und Nachteile des Verfahrens im Vergleich zur Black-Box-Kalibrierung untersucht werden.
- Am konstruierten Beispiel und der echten Fallstudie der Agenten-basierten Einkaufssimulation soll gezeigt werden, wie für allgemeine Black-Box-Kalibrierung formulierte Kalibrierungsprobleme für Agenten-basierte Simulationen in Probleme für das Makro-Mikro-Verfahren abgebildet werden können.
- Am Beispiel der Fallstudie der Agenten-basierten Einkaufssimulation soll untersucht werden, ob und inwiefern die Verwendung des Makro-Mikro-Kalibrierungsansatzes im Vergleich zu Black-Box-Ansätzen zu Vorteilen führen kann und wo die Grenzen der Anwendbarkeit des Verfahrens liegen.

## 15.2 Diskussion an künstlichem Evaluationsmodell

Im ersten Teil der Evaluation des Makro-Mikro-Verfahrens wird ein sehr einfaches künstliches Modell betrachtet, das die Kernelemente des Modells zur Fallstudie der Agentenbasierten Einkaufssimulation enthält. Der Vorteil der Evaluation anhand des künstlichen Modells ist, dass für dieses Modell vollständiges Wissen über valides Verhalten und Zusammenhänge des fiktiven realen Systems vorhanden ist. Auf dieser Grundlage können gut Anwendbarkeit und Vor- und Nachteile des Makro-Mikro-Verfahrens untersucht werden.

### 15.2.1 Modellbeschreibung

Das künstliche Modell besteht aus einer Umwelt ohne explizite räumliche Repräsentation. In der Umwelt befinden sich:

1. Zwei Geschäfte  $G_A$  und  $G_B$ . Jedes Geschäft hat zwei charakteristische Merkmalattribute  $Att1_A = 3$  und  $Att2_A = 2$  für  $G_A$  und  $Att1_B = 0$  und  $Att2_B = 2$  für  $G_B$ . Weiter hat jedes der beiden Geschäfte einen Zielumsatz, den es am Ende der Simulation haben soll:  $Z_A = 365$  für  $G_A$  und  $Z_B = 40$  für  $G_B$ .
2. Drei Einkaufsagenten  $EK_1$ ,  $EK_2$  und  $EK_3$ . Jeder Einkaufsagent hat ein Budget  $B_{EK_1} = 1000$ ,  $B_{EK_2} = 50$ ,  $B_{EK_3} = 100$  aus dem er Geld in den Geschäften ausgeben kann. Es muss aber nicht alles Geld ausgegeben werden. Weiter hat jeder Einkaufsagent  $EK_X$  zwei individuelle Präferenzparameter  $P_{Att1_{EK_X}}$  und  $P_{Att2_{EK_X}}$  für die beiden Attribute der Geschäfte.

Im Modell gibt jeder Einkaufsagent in jedem der beiden Geschäfte einen Teil seines Budgets auf der Grundlage der Merkmalattribute der Geschäfte aus. Die Berechnung des Budgetteils erfolgt auf Basis der Formel:

$$\text{Budgetteil}(EK_X, G_Y) = (Att1_Y \cdot P_{Att1_{EK_X}} + Att2_Y \cdot P_{Att2_{EK_X}}) \cdot 0.1 \cdot B_{EK_X}$$

mit  $x \in \{1, 2, 3\}$  und  $y \in \{1, 2\}$

Jeder Einkaufsagent multipliziert also für ein Geschäft dessen Attributwerte jeweils mit seiner individuellen Präferenz für das Attribut und summiert die Werte auf. Der Ergebniswert wird als Vielfaches von 10% des Einkaufsbudgets des Agenten im Geschäft ausgegeben.

Der Umsatz eines Geschäftes  $Y$  ergibt sich dann als:

$$\text{Umsatz}_Y = \text{Budgetteil}(EK_1, G_Y) + \text{Budgetteil}(EK_2, G_Y) + \text{Budgetteil}(EK_3, G_Y)$$

Das Gesamtverhalten der Simulation des Modells sei dann valide, wenn jedes Geschäft seinen Zielumsatz erreicht, d.h. es wird die folgende zu optimierende Gesamtziel­funktion verwendet:

$$\text{GZF} = -1 \cdot (|\text{Umsatz}_A - Z_A| + |\text{Umsatz}_B - Z_B|)$$

GZF hat den Optimalwert 0.

### 15.2.2 Kalibrierungsproblem

Das Kalibrierungsproblem besteht darin, für jeden Einkaufsagenten die Präferenzparameter  $P_{Att}$  so einzustellen, dass valides Gesamtverhalten entsteht. Damit müssen für die Einkaufsagenten bei homogenen Parameterwerten 2 und im heterogenen Fall 6 Parameterwerte kalibriert werden. Mit den Einkaufsagentenfamilien  $EKF$  (für jeden Agenten eine Familie) für die Kalibrierung ergibt sich dann das Kalibrierungsproblem.

$$KP_{KünstEK} = ((Umwelt, \{EKF_1, EKF_2, EKF_3\}), GZF)$$

Auf der Mikro-Ebene des Agentenverhaltens berechnen die Agenten mit der Formel *Budgetanteil* welchen Anteil ihres Geldes sie in welchem Geschäft ausgeben. Auf der Makro-Ebene des Gesamtverhaltens wird für alle Geschäft ermittelt wie genau die Zielsätze der Geschäfte in der Simulation erreicht wurden.

### 15.2.3 Black-Box-Kalibrierung

Für  $KP_{KünstEK}$  kann mit Black-Box-Kalibrierung leicht eine Lösung gefunden werden, welche den Wert von  $GZF$  optimiert. Wenn man die Kalibrierung aber mehrfach durchführt ist zu erkennen, dass  $GZF$  bei vielen verschiedenen Parametereinstellungen den maximalen Wert annimmt. Wie leicht erkennbar ist, ist das Gleichungssystem, das man zur Bestimmung der Parameterwerte lösen muss stark unterbestimmt. Eine Vielzahl unterschiedlicher Lösungen ist möglich.

Alle diese Lösungen führen zu bezüglich  $GZF$  vollständig validem Simulationsverhalten auf der Beobachtungsebene des Gesamtverhaltens der Simulation. Auf der Mikro-Ebene des konkreten Agentenverhaltens konnten verschiedene Parametereinstellungen gefunden werden, die zu validem Gesamtverhalten führen. Der Unterschied zwischen den Parametereinstellungen ist das jeweils aus der Berechnung auf Mikro-Ebene resultierende aggregierte Verhalten der Agenten, nämlich wie viel Geld jeder Agent in einem Geschäft ausgibt oder aus der Perspektive der Geschäfte, welchen Anteil seines Geschäftsumsatzes jedes Geschäft von welchem Einkaufsagenten erhält.

Für die weitere Diskussion wird ohne weitere Begründung festgelegt, dass die folgende Parametereinstellung Simulationsverhalten erzeuge, das auf allen Beobachtungsebenen, d.h. insbesondere der Ebene des aggregierten Agentenverhaltens valide ist:

Alle Agenten geben Geld in  $G1$  aus und nur  $EK_2$  und  $EK_3$  geben Geld in  $G2$  aus. Konkret:  
 $P_{Att1EK_1} = 1, P_{Att2EK_1} = 0, P_{Att1EK_2} = 1, P_{Att2EK_2} = 2, P_{Att1EK_3} = 1, P_{Att2EK_3} = 1$

Mit Black-Box-Kalibrierung kann in diesem Modell nur irgendeine der möglichen Lösungen gefunden werden. Es gibt im Beispielfall keine Möglichkeit sicherzustellen, dass immer obige Lösung gefunden wird.

**Als Folge bedeutet eine erfolgreiche Black-Box-Kalibrierung nicht, dass am Ende der Kalibrierung tatsächlich ein valides Mikro-Modell steht, bei dem auch das resultierende Verhalten der einzelnen Einkaufsagenten valide ist.**

### 15.2.4 Anwendung des Makro-Mikro-Verfahrens

Bei Anwendung des Makro-Mikro-Verfahrens zur Kalibrierung von ABS wird genau das beschriebene Problem der Black-Box-Kalibrierung auf der Mikro-Ebene behandelt.

#### 15.2.4.1 Kalibrierung auf der Makro-Ebene

Auf der Makro-Ebene wird zunächst die Fragestellung betrachtet „Wie soll das resultierende aggregierte Verhalten der Einkaufsagenten sein, das zu validem Gesamtverhalten führt?“.

**Vorteil des Makro-Mikro-Verfahrens** gegenüber Black-Box-Kalibrierung ist also, dass zuerst auf aggregierter Ebene geklärt wird, wie valides Agentenverhalten aussehen muss, das zu validem Gesamtverhalten führt. Im Makro-Schritt des Makro-Mikro-Verfahrens werden die  $IVM$  für die Kalibrierung auf der Mikro-Ebene ermittelt. Im Beispielmmodell wird auf der Ebene des idealen aggregierten Verhaltens für jeden Einkaufsagenten ermittelt wie viel Geld jedes Geschäft von ihm erhalten muss, damit einerseits  $GZF$  erfüllt wird und andererseits das Geldausgabeverhalten der Agenten valide ist. Welche Berechnungen die Einkaufsagenten auf der Mikro-Ebene anstellen müssen, damit das entsprechende Ausgabeverhalten entsteht, spielt hier noch keine Rolle. Für das Beispielmmodell können die zu wählenden  $IVM$  für die Einkaufsagenten als Listen von Ziel-Geld-Werten sortiert nach  $G1$  und  $G2$  angegeben werden:  $IVM_{EK1} = (300, 0)$ ,  $IVM_{EK2} = (15, 10)$ ,  $IVM_{EK3} = (30, 10)$ .

**Voraussetzung für die Anwendbarkeit des Makro-Mikro-Verfahrens** ist dabei, dass es dem Kalibrierer in Zusammenarbeit mit dem Modellierer und Fachexperten möglich ist, zumindest Strukturen für die  $IVMF_{EK}$  zu spezifizieren. Die Voraussetzungen wurden im Kapitel zum Makro-Mikro-Verfahren diskutiert. Insgesamt sind unter den beschriebenen Randbedingungen nicht die erforderlichen Voraussetzungen für die Anwendbarkeit des Basis-Makro-Mikro-Verfahrens auf das künstliche Beispielmmodell gegeben.

#### 15.2.4.2 Kalibrierung auf der Makro-Ebene mit Reverse-Engineering-Verfeinerung

Damit das Makro-Mikro-Verfahren auch angewendet werden kann, wenn die Anforderung des Basis-Makro-Mikro-Verfahrens nicht erfüllt werden kann, also wenn der Entwickler keine konkreten  $IVM$  auf der Ebene idealen aggregierten Verhaltens spezifizieren kann wurde das Makro-Mikro-Verfahren um die Reverse-Engineering-Verfeinerung erweitert.

Im Beispielmmodell wird bei Verwendung des Makro-Mikro-Verfahrens mit Reverse-Engineering-Verfeinerung gefordert, dass Strukturen für die  $IVMF$  definiert werden. Dies geschieht durch Spezifikation der Tupel  $IVMF_{EK1} = (x_1, y_1)$ ,  $IVMF_{EK2} = (x_2, y_2)$ ,  $IVMF_{EK3} = (x_3, y_3)$ . Hier besteht nun das Problem, wie die  $IVMF$  zu  $IVM$  kalibriert werden sollen. Hierfür wird ein Algorithmus benötigt, um das Reverse-Engineering durchzuführen.

Im Fall des Beispielmodells wäre eine Durchführung von Reverse-Engineering ohne zusätzliches Kalibrierungswissen nicht möglich. Die  $IVMF$ -Tupel sind genauso unbestimmt, wie die konkreten Parameter auf der Mikro-Ebene. Damit in diesem Fall das Reverse-Engineering durchgeführt werden könnte, müsste zusätzliches Kalibrierungswissen, z.B. in Form von Randbedingungen der Form, dass  $EK_1$  kein Geld in  $G2$  ausgeben soll, dass  $EK_2$  und  $EK_3$  beide Geld in  $G1$  und  $G2$  ausgeben, etc. vorhanden sein, auf dessen Basis dann das RE durchgeführt werden kann.

Für die weitere Diskussion wir nun davon vorausgesetzt, dass im Schritt auf der Makro-Ebene des Makro-Mikro-Verfahrens  $IVM$  für die Kalibrierung auf der Mikro-Ebene ermittelt werden konnten.



### 15.2.4.3 Kalibrierung auf der Mikro-Ebene

Nach dem Makro-Schritt stehen für die Kalibrierung auf der Mikro-Ebene die ermittelten *IVM* zu Verfügung.

#### Voraussetzung für Anwendbarkeit von White-Box-Zerlegungstechniken auf der Mikro-Ebene im Makro-Mikro-Verfahren:

Die *IVM* auf der Ebene idealisierten Verhaltens müssen das Verhalten der Agenten auf der Mikro-Ebene möglichst feingranular aggregieren. D.h. es sollte zu möglichst jedem einzelnen simulierten Agenten auf der Mikro-Ebene ein valides aggregiertes Verhalten spezifiziert werden.

Im schlechtesten Fall kann auf der Mikro-Ebene keine White-Box-Zerlegung durchgeführt werden, weil die *IVM* zu viele Wechselwirkungen untereinander haben. Beispielsweise sei es im Beispielmodell so, dass  $EK_1$  und  $EK_2$  auf der Mikro-Ebene erst untereinander Einkaufsverhandlungen führen und dann zusammen mit gemeinsamem kombinierten Einkaufsbudget einkaufen gehen. In diesem Fall wäre es schwierig, im Makro-Schritt des Makro-Mikro-Verfahrens noch *IVM* für jeden der Agenten zu spezifizieren. Dies wäre nur für die ganze gebildete „Einkaufsgruppe“ möglich. Es ist natürlich möglich die *IVM* für Gruppen von Agenten oder ganze Modellteile zu spezifizieren. Die White-Box-Zerlegung kann dann aber nur auf der Ebene dieser Teilmodelle erfolgen.

In diesem Fall bliebe von den Vorteilen des Makro-Mikro-Verfahrens auf der Mikro-Ebene nur die Verfügbarkeit zusätzlicher Randbedingungen and die Kalibrierung auf der Mikro-Ebene in Form der *IZF* übrig.

Für das betrachtete Beispielmodell ist die Anforderung an eine White-Box-Zerlegbarkeit auf der Mikro-Ebene im Rahmen des Makro-Mikro-Verfahrens erfüllt. Die Kalibrierung kann daher mit White-Box-Zerlegung durchgeführt werden. Im Folgenden werden für das Beispiel beide Fälle: optimaler Fall „White-Box“ und schlechtester Fall „Black-Box“ beschrieben.

- **Black-Box-Kalibrierung auf der Mikro-Ebene:** Die Black-Box-Kalibrierung der Mikro-Ebene wird für das Beispiel nicht mehr auf der Basis der *GZF*, sondern auf der Basis der Werte in den *IVM* durchgeführt. Die Zielfunktion lautet dann:

$$ZF = -1 \cdot (|Budgetteil(EK_1, G_1) - IVM_{1G_1}| + |Budgetteil(EK_1, G_2) - IVM_{1G_2}| + \dots + |Budgetteil(EK_3, G_2) - IVM_{3G_2}|)$$

Dabei ist  $IVM_{iG_j}$  der Ziel-Geld-Wert von Agent  $i$  für Geschäft  $j$ .

- **White-Box-Kalibrierung auf der Mikro-Ebene:** Die Kalibrierung auf der Mikro-Ebene kann für jeden Einkaufsagenten getrennt durchgeführt werden. Die individuelle Zielfunktion für einen Agenten  $EK_X$  ( $x \in \{1, 2, 3\}$ ) lautet dann:

$$IZF_X = -1 \times (|Budgetteil(EK_X, G_1) - IVM_{XG_1}| + |Budgetteil(EK_X, G_2) - IVM_{XG_2}| + \dots + |Budgetteil(EK_X, G_3) - IVM_{XG_3}|)$$

In jeder Teilkalibrierung müssen nur 2 statt 6 Parameterwerte kalibriert werden, da jeder Agent nur 2 Parameter hat. Die einzelnen Teilkalibrierungen können gleichzeitig durchgeführt werden. Die Teilergebnisse können dann direkt in das Gesamtmikromodell eingesetzt werden und die Kalibrierung ist abgeschlossen.

Mit dem Abschluss des Mikro-Schrittes ist die Anwendung des Makro-Mikro-Verfahrens auf das künstliche Beispielmodell abgeschlossen und die Modellparameterwerte gemäß Lösung 1 ermittelt.

#### 15.2.4.4 Übereinstimmung: *IVM* und zugehöriges Teilmodell auf der Mikro-Ebene

Ist der Makro-Schritt abgeschlossen gilt das in den *IVM* abgebildete Verhalten für den weiteren Verlauf des Makro-Mikro-Verfahrens als valide. Probleme bei der Kalibrierung auf der Mikro-Ebene werden daher zuerst als auf der Mikro-Ebene zu lösende Probleme eingestuft. Erst wenn dies nicht möglich ist werden die *IVM* in Frage gestellt und ein weiterer Iterationsschritt auf der Makro-Ebene durchgeführt.

Im Beispielmmodell könnte es z.B. sein, dass im Makro-Schritt ermittelt wird, dass  $EK_2$  50 Geld in Geschäft  $G_1$  ausgeben sollte. Im Mikro-Schritt wird dann festgestellt, dass das Mikro-Modell nicht entsprechend kalibriert werden kann.

Es muss nun geklärt werden, ob die *IVM* oder die Modellstrukturen auf der Mikro-Ebene angepasst werden müssen oder ob möglicherweise ein Fehler im zugrundeliegenden Datenmaterial vorliegt. Da das Verhalten der *IVM* auf der Makro-Ebene im Makro-Schritt bereits als valide bewertet wurde, kann dies aber zumindest als Indikation betrachtet werden, dass Daten und Mikro-Modellstrukturen überprüft und angepasst werden sollten.

Eine Überprüfung, wie gut bestimmte Teil-Mikro-Modell-Strukturen zur Erzeugung validen Gesamtverhaltens geeignet sind, war mit Black-Box-Kalibrierung zuvor nicht möglich. Man konnte lediglich feststellen, dass das gesamte Mikro-Modell das Zielverhalten in der Simulation erzeugen kann oder nicht. Daher wird die Möglichkeit der **Prüfung wie gut *IVM* und Teil-Mikro-Modelle zueinander passen als Vorteil des Makro-Mikro-Verfahrens** angesehen.

#### 15.2.4.5 Robuste Suche im Makro-Mikro-Verfahren

Die Evaluation der robusten Suche wird in Kapitel 17 diskutiert. In diesem Abschnitt wird nur kurz auf die Anwendbarkeit der robusten Suche im Beispielmmodell eingegangen.

Im Beispielmmodell würde z.B. Einkaufsagent  $EK_1$  durch sein  $IVM_1$  ersetzt. Dann würde während der Simulation des Mikro-Modells nicht mehr die Formel  $Budgetteil(EK_1, G_Y)$  zur Berechnung des ausgegebenen Geldes verwendet, sondern direkt die in  $IVM_1$  gespeicherten Geldwerte eingesetzt.

Im künstlichen Beispielmmodell wäre die robuste Suche damit einsetzbar. Kann das Mikro-Modell nicht entsprechend erweitert werden, so kann die robuste Suche nicht eingesetzt werden.

#### 15.2.5 Zusammenfassung: Diskussion des Makro-Mikro-Verfahrens

In diesem Abschnitt wurde ein einfaches künstliches Beispielmmodell verwendet, um das Makro-Mikro-Verfahren mit Reverse-Engineering-Verfeinerung entlang des Anwendung am Beispiel zu diskutieren. Dabei wurden die Vorteile des Verfahrens im Vergleich zur Black-Box-Kalibrierung und die Voraussetzungen, die zur Umsetzung des Verfahrens und zur Nutzung von sich daraus ergebenden Vorteilen notwendig sind aufgestellt.

Als Kernvorteile des Makro-Mikro-Verfahrens wurden genannt:

- Es wird ermöglicht explizit Wissen über aggregiertes Agentenverhalten auf der Makro-Ebene in den Kalibrierungsprozess von ABS auf der Mikro-Ebene einzubringen, was

in Black-Box-Verfahren nicht möglich ist.

- Durch die Ermittlung der *IVM* werden Voraussetzungen für die Anwendung von White-Box-Zerlegungstechniken auf der Mikro-Ebene geschaffen und damit für Vereinfachung und Zerlegung des Mikro-Kalibrierungsproblems im Vergleich zur Black-Box-Kalibrierung.
- Durch die Ermittlung der *IVM* werden Voraussetzungen für die Anwendung von robuster Suche in ABS geschaffen, was mit Black-Box-Kalibrierung nicht möglich war.

Die Hauptschwierigkeiten für die Anwendung des Makro-Mikro-Verfahrens sind:

- Dass zur Ausnutzung vieler möglicher Vorteile des Makro-Mikro-Verfahrens Voraussetzungen notwendig sind, die von der spezifischen Beschaffenheit des Modells und dem verfügbaren Wissen über die Wirkungszusammenhänge des modellierten Original-Systems abhängen.
- Damit kann die Anwendung des Makro-Mikro-Verfahrens nicht „blind“, sondern immer nur gut vorbereitet erfolgen. Nur dann schneidet das Verfahren für ein konkretes Kalibrierungsproblem erfolgreicher ab als eine Black-Box-Suche.
- Dass zur Anwendung des Makro-Mikro-Verfahrens noch zusätzlicher Modellierungsaufwand anfällt, um z.B. die Ebene der *IVM* zu modellieren oder das Mikro-Modell, um die *IVM* zu erweitern. Entsprechender Aufwand wird für eine Black-Box-Kalibrierung auf der Mikro-Ebene nicht benötigt.

Das Makro-Mikro-Verfahren kann Probleme bei der Kalibrierung von ABS lösen, die mit Black-Box-Kalibrierung nicht behandelbar sind. Es lohnt sich aber, zunächst eine Black-Box-Kalibrierung des Modells zu versuchen und deren Ergebnis zu bewerten, bevor zusätzlicher Erweiterungs- und Modellierungsaufwand für eine Makro-Mikro-Kalibrierung investiert wird.

In den folgenden Abschnitten wird die Anwendung des Makro-Mikro-Verfahrens auf die echte Fallstudie der Agenten-basierten Einkaufssimulation betrachtet und jeweils bewertet, ob und wie das Makro-Mikro-Verfahren eingesetzt werden konnte, wo die Grenzen lagen und welche Vorteile der Einsatz gegenüber der Black-Box-Kalibrierung bringen konnte.

### 15.3 Kurze Einleitung in die Fallstudie der Einkaufssimulation

Die Evaluation in diesem Abschnitt wird anhand der Fallstudie der Agentenbasierten Einkaufssimulation aus Abschnitt 7.5 durchgeführt. Im ersten Teil wird ein Kalibrierungsproblem  $KP_{GeoUmeaHom}$  (Geographisches Modell von Umea mit homogenen Parameterwerten) für ein Modell mit vollständig festgelegter Modellstruktur, fest vorgegebenen Strukturen für die IVMF und homogenen Agentenparametern betrachtet. Im zweiten Teil wird ein Kalibrierungsproblem  $KP_{GeoUmeaHet}$  mit heterogenen Agentenparametern betrachtet.

Im Folgenden werden kurz die gemeinsamen Merkmale des für die Kalibrierungsprobleme verwendeten Modells aufgezählt. Weitere Strukturelemente werden im Rahmen der Beschreibung der konkreten Kalibrierungsprobleme genannt:

- **Anzahl simulierter Geschäfte**

In  $KP_{GeoUmeaHom}$  und  $KP_{GeoUmeaHet}$  werden jeweils alle 132 Geschäfte der Region simuliert.

- **Anzahl simulierter Einkaufsagenten**

In  $KP_{GeoUmeaHom}$  und  $KP_{GeoUmeaHet}$  werden jeweils 1000 Einkaufsagenten simuliert.

- **Zeitfortschritt**

Der simulierte Zeitraum ist ein Jahr. Dies wird aber in einem einzigen Schritt berechnet.

- **Dynamik des Agentenverhaltens**

Das simulierte Agentenverhalten verändert sich nicht dynamisch. Das Verhalten der simulierten Einkaufsagenten ist vollständig durch ihre (z.B. von der Kalibrierung) fest vorgegebenen Parameterwerte bestimmt.

- **Validierungskriterien**

Als Validierungskriterium dient für  $KP_{GeoUmeaHom}$  und  $KP_{GeoUmeaHet}$  das unquadratische Gütemaß als globale Zielfunktion.

## 15.4 $KP_{GeoUmeaHom}$ : Problemstellung

Im ersten Teil dieses Abschnitts wird das Makro-Mikro-Verfahren im Vergleich zu Black-Box-Ansätzen anhand der Kalibrierung der Agenten-basierten Einkaufssimulation mit **homogenen Agentenparametern** evaluiert. Damit wird zunächst ein Kalibrierungsproblem betrachtet für das aufgrund der geringeren Zahl zu kalibrierender Parameterwerte Black-Box-Verfahren gut eingesetzt werden können.

### 15.4.1 Initiale Problemstellung für Evaluierung

Das initiale zu lösende Kalibrierungsproblem ist bezüglich einer typischen Black-Box-Kalibrierung des Modells gegeben (siehe Kapitel 4.1):

$$KP_{GeoUmeaHom} = ((UMF, AMF), VK)$$

- **UMF={UM}**: Das vorgegebene Umweltmodell der simulierten Region um die nord-schwedische Stadt Umea: Wohnorte der simulierten Einkaufsagenten und simulierte Geschäfte des Lebensmitteleinzelhandels mit ihren charakteristischen Eigenschaften und Standorten.
- **AMF = {AGF}**: Alle Agenten aus der Klasse der Einkaufsagenten sind Teil einer einzigen Gruppe, d.h. es werden homogene Parameterwerte für alle Agenten gesucht.
- **AGF = (S<sub>AG</sub>, V<sub>AG</sub>, P<sub>AG</sub>)**: Für die Einkaufsentscheidungen der Agenten seien die charakteristischen Geschäftsmerkmale **Preis, Sortiment, Qualität und Distanz** relevant.

- $S_{AG}$ : Die Struktur  $S_{AG}$  der Agentengruppenfamilie  $AGF$  ist gekennzeichnet durch logistische Wahrnehmungsfunktionen für die Preis, Sortiment und Qualität.

Die Geschäftsbewertung eines Einkaufsagenten  $X$  für ein Geschäft  $A$  erfolgt gemäß der Formel:

$$BewFkt(A, X) = \frac{\frac{10^{P_{Preis1}}}{1+10^{P_{Preis2} * e^{(-P_{Preis3} * Preis_A)}}}}{Rangdistanz \text{ von } X \text{ zu } A}} + \frac{\frac{10^{P_{Sort1}}}{1+10^{P_{Sort2} * e^{(-P_{Sort3} * Sort_A)}}}}{Rangdistanz \text{ von } X \text{ zu } A}} + \frac{\frac{10^{P_{Qual1}}}{1+10^{P_{Qual2} * e^{(-P_{Qual3} * Qual_A)}}}}{Rangdistanz \text{ von } X \text{ zu } A}}$$

Dabei seien  $Preis_A$ ,  $Sort_A$  und  $Qual_A$  jeweils die Preis-, Sortiments- und Qualitätsstufen dieser charakteristischen Merkmale von Geschäft  $A$ .

- Die Distanzwahrnehmung der Agenten sei als **Rang-basierte Wahrnehmung** fester Teil der Modellstruktur  $S_{AG}$ . Hierdurch bewertet potentiell jeder Einkaufsagent jedes Geschäft unterschiedlich, obwohl alle Einkaufsagenten zu einer  $AG$  gehören (Heterogene Distanzwahrnehmung aufgrund unterschiedlicher Wohnorte).
- $P_{AG} = \langle P_{Preis1}, P_{Preis2}, P_{Preis3}, P_{Sort1}, P_{Sort2}, P_{Sort3}, P_{Qual1}, P_{Qual2}, P_{Qual3} \rangle$ : Die Menge der zu kalibrierenden Parameter  $P_{AG}$  für  $AGF$  besteht aus den Steuerungsparametern für die logistischen Wahrnehmungsfunktionen. Für jeden Parameterwert  $PW$  der logistischen Funktionen werde als Wertebereich:  $PW \in [0,001; 5,0]$  festgelegt. Der Wertebereich ist empirisch durch verschiedene initiale Kalibrierungsversuche ermittelt.
- $VK = ((UM, AM), GZF)$  mit

$$- GZF = 1 - \frac{\sum_{i=1}^{\#Geschäfte} ZF_{GESCH_i}}{\sum_{i=1}^{\#Geschäfte} OrigUmsatz_{Geschäft_i}}$$

- $ZF_{GESCH_i}$  eine Teilbewertungsfunktion, welche die absolute Abweichung eines einzelnen der 132 simulierten Geschäfte  $GESCH_i$  misst:  $i$  von 0 bis 131.

$$ZF_{GESCH_i} = |SimUmsatz_{Geschäft_i} - OrigUmsatz_{Geschäft_i}|$$

#### 15.4.2 $KP_{GeoUmeaHom}$ : Suchraum

In  $KP_{GeoUmeaHom}$  müssen für 1000 Agenten die Parameterwerte der logistischen Wahrnehmungsfunktionen kalibriert werden. Da lediglich homogene Parameterwerte gesucht werden, müssen insgesamt neun Parameter kalibriert werden.

#### 15.4.3 $KP_{GeoUmeaHom}$ : Ausgangsvalidität

Da bei der Kalibrierung der Modellparameter keine initialen Schätzungen für die Parameterwerte verfügbar waren, wurden für die Kalibrierung zufällige Startwerte gewählt. Als Anhaltspunkt für den Validitätsgrad des Simulationsverhaltens vor Beginn der Kalibrierung sei der Validitätsgrad einer Simulation genannt, bei der für alle Parameterwerte jeweils der Wert 1,0 gewählt wurde. In diesem Fall war der Validitätsgrad des Simulationsverhaltens bezüglich  $GZF$  0,235. Das Simulationsverhalten wird bezüglich  $GZF$  als optimal valide bewertet, wenn  $GZF$  den Wert 1,0 annimmt.

Populationsgröße	Mutationen pro Individuum pro Runde	Elter pro Crossover-Anwendung	Anzahl Crossover pro Generation
20	3	2	20

Tabelle 15.1: Parameter der evolutionären Strategie.

Startpunkt	Temperatur-Start	Temperatur-Ende	Temperaturabschwächung	Gradientenschätzung	Schrittweitenadaptation
Zufall	0,05	0,005	0,9	Ja	Ja

Tabelle 15.2: Simulated Annealing Parameter.

## 15.5 $KP_{GeoUmeaHom}$ : Lösung mit Black-Box Ansatz

Im ersten Ansatz wird  $KP_{GeoUmeaHom}$  mit Hilfe eines Black-Box Kalibrierungsansatzes gelöst. In diesem Fall wird zur Bewertung des Gesamtverhaltens eine globale Zielfunktion  $GZF$  eingesetzt. Die zur Kalibrierung verwendeten Verfahren sind Evolutionsstrategie und das Simulated Annealing Verfahren. Zur Kalibrierung wird der DAVINCI-Kalibrierer eingesetzt.

Die Tabellen 15.1 und 15.2 fassen die wesentlichen Einstellungen für die Kalibrierungsverfahren zusammen. Zu Beginn der Evaluation wurden viele Versuche mit unterschiedlichen mehr und weniger restriktiven Algorithmusparametern und vielen Wiederholungen der Kalibrierungsläufe durchgeführt. Im Folgenden werden die Einstellungen beschrieben, mit denen die besten Ergebnisse bei möglichst wenigen für die Kalibrierung benötigten Simulationsläufen erzielt werden konnten.

Die evolutionäre Strategie verwendet eine relativ kleine Population und das Simulated Annealing Verfahren eine relativ strikte Temperaturskala. Auf diese Weise wird jeweils die Verbesserungs-„Gierigkeit“ der einzelnen Verfahren verstärkt und dafür gesorgt, dass das Kalibrierungsergebnis mit möglichst wenigen Schritten (= Simulationsläufen) erreicht wird. Pro Runde werden an jedem Individuum der Lösungspopulation 3 zufällig ausgewählte Parameterwerte mutiert. Beim Crossover werden pro Generation 20 Mal zufällig zwei Elter bestimmt und miteinander gekreuzt. Die 20 neuen Individuen kommen zur Gesamtpopulation hinzu. Als nächste Generation werden die 20 besten Individuen der Gesamtpopulation (aktuelle Population und neu erzeugte Individuen) gewählt.

Jedem der Verfahren wurden maximal 500 Simulationsläufe als Berechnungsressourcen während der Kalibrierung zur Verfügung gestellt.

### 15.5.1 Kalibrierungsergebnisse Black-Box-Ansatz

Mit beiden Verfahren wurden Kalibrierungsergebnisse mit annähernd gleichem Validitätsgrad bezüglich  $GZF$  erreicht. **Die im Test erreichte beste Validitätsbewertung eines Simulationslaufes mit kalibrierten Parameterwerten gemäß dem Black-Box Validierungskriterium  $GZF$  war 0,58.** Im Vergleich zur genannten möglichen Ausgangssituation der Kalibrierung ist dies eine Verbesserung um 0,345.

Parameter-nummer	Preis	Sortiment	Qualität	GZF-Bewertung	#Simulationen bis beste Einstellung
P1	1,90	2,14	1,30	0,575	350
P2	1,5	2,75	2,95		
P3	0,19	2,41	3,00		
P1	1,78	2,4	1,22	0,58	287
P2	2,5	2,9	0,97		
P3	0,001	1,26	2,67		
P1	2,28	2,56	1,34	0,571	420
P2	1,81	1,91	2,18		
P3	0,189	1,56	2,93		
P1	2,42	2,14	0,98	0,577	345
P2	2,93	2,14	0,67		
P3	0,16	1,39	0,001		

Tabelle 15.3: Resultierende Parametereinstellungen unterschiedlicher Black-Box-Kalibrierungen.

### 15.5.2 Evaluierungsergebnisse Black-Box Ansatz

- **Validitätsbewertung:** Der erreichte Validitätsgrad des Simulationsverhaltens bezüglich GZF war 0,58.
- **Identifizierte Parameterwerte:** Bei wiederholten Kalibrierungsläufen konnten verschiedene Parametereinstellungen mit annähernd gleicher Validität bezüglich GZF identifiziert werden, d.h. der zugehörige Suchraum enthält multiple lokale und globale Optima. Tabelle 15.3 zeigt einige der ermittelten Parametereinstellungen mit GZF-Wert.
- **Anzahl benötigter Simulationsläufe:** Während einer Black-Box Kalibrierung wurden jeweils maximal 500 Simulationsläufe zur Auswertung von Parametereinstellungen durchgeführt werden. Zur Evaluation wird nicht diese Maximalzahl, sondern die tatsächlich benötigte Anzahl an Simulationsläufen bis zur Identifikation der Ergebnisparametereinstellung betrachtet. Da die Kalibrierung mit Hilfe heuristischer Black-Box-Suchverfahren durchgeführt wurde, kann kein eindeutiger Wert für die Anzahl benötigter Simulationsläufe angegeben werden. In Tabelle 15.3 werden zu jedem Ergebnis, die in der zugehörigen Kalibrierung benötigten Simulationsläufe angegeben. Über 10 Testläufe wurden durchschnittlich 308 und immer mehr als 100 Simulationsläufe zur Kalibrierung der Parameterwerte zur Erreichung einer Validität bezüglich GZF von annähernd 0,58 benötigt. Der Maximalwert, der in Tests benötigt wurde war 490.

## 15.6 $KP_{GeoUmeaHom}$ : Lösung mit dem Makro-Mikro-Verfahren

In diesem Abschnitt wird die Lösung von  $KP_{GeoUmeaHom}$  mit Hilfe des in Abschnitt 9.1.3 beschriebenen Makro-Mikro-Verfahrens durchgeführt.

### 15.6.1 Übertragung des Problems auf das Makro-Mikro-Verfahren

In diesem Abschnitt wird zunächst die grundsätzliche Übertragung des Kalibrierungsproblems auf das Makro-Mikro-Verfahren beschrieben. Im nächsten Abschnitt wird auf die konkrete Umsetzung eingegangen und die Anwendbarkeit diskutiert.

Beim Einsatz des in Abschnitt 9.1.3 beschriebenen Makro-Mikro-Kalibrierungsansatzes wird  $KP_{GeoUmeaHom}$  in ein Kalibrierungsproblem auf Makroebene  $KPMAKRO_{GeoUmeaHom}$  und ein Kalibrierungsproblem auf der Mikroebene  $KPMIKRO_{GeoUmeaHomMikro}$  aufgeteilt.

**Auf der Makroebene:** Im ersten Schritt auf der Makro-Ebene des Makro-Mikro-Verfahrens muss zunächst valides aggregiertes Einkaufsverhalten der Agenten ermittelt werden.

Es gilt:

$$KPMAKRO_{GeoUmeaHom} = ((\{UM\}, \{IVMF_{AG}\}), ((Umweltmodell, \{Geschäfte\}), GZF))$$

**Auf der Mikroebene** werden Parameterwerte für die Modellfamilie  $AGF$  zur Beschreibung des konkreten Verhaltens der Gruppe aller Einkaufsagenten gesucht, welche valides Verhalten bezüglich  $GZF$  und entsprechender der auf der Makro-Ebene ermittelten  $IVM$  produzieren können. Die Validierung bezüglich der  $IVM$  soll mittels Zielfunktionen  $IZF_{AG}$  durchgeführt werden, mit deren Hilfe für jeden Einkaufsagenten bewertet wird, wie genau sein Verhalten seinem zugehörigen  $IVM$  entspricht. Das Kalibrierungsproblem  $KPMIKRO_{GeoUmeaHom}$  auf der Mikroebene ist dann gegeben durch:

$$KPMIKRO_{GeoUmeaHom} = ((\{UM\}, \{AGF\}), VK = \{GZF, (AG, IZF_{AG})\})$$

Im nächsten Abschnitt wird nun die konkrete Umsetzung der Makro-Mikro-Verfahrens betrachtet.

### 15.6.2 Durchführung der Kalibrierung auf der Makro-Ebene

Auf der Makro-Ebene des Makro-Mikro-Verfahrens müssen  $IVM$  für die Kalibrierung auf der Mikro-Ebene ermittelt werden. Damit das Makro-Mikro-Verfahren anwendbar ist, müssen zuerst die Voraussetzungen erfüllt sein, dass definiert werden kann, was aggregiertes Verhalten der Einkaufsagenten auf der Makro-Ebene ist und dass Modellstrukturen zur Beschreibung zugehöriger  $IVM$  spezifiziert werden können.

#### 15.6.2.1 Definition der $IVMF$

In der Fallstudie der Einkaufssimulation können die Strukturen der idealen Verhaltensmodelle  $IVM$  der Einkaufsagenten recht einfach aufgestellt werden. Die idealen Verhaltensmodellfamilien in der Menge  $IVMF$  beschreiben für jeden Agenten wie attraktiv er jedes Geschäft finden muss, damit ausreichend valides Gesamtverhalten entstehen kann. Da es in  $KP_{GeoUmeaHom}$  um die Kalibrierung homogener Agentenparameter geht, wird nach einem aggregierten Einkaufsverhalten gesucht, das für alle Einkaufsagenten gleich ist. Es werden also nicht für jeden Einkaufsagenten unterschiedliche  $IVM$  gesucht, sondern es werden für alle Einkaufsagenten identische  $IVM_{AG}$  gesucht. Die  $IVM_{AG}$  beschreiben dann aggregiertes Einkaufsverhalten, das für alle Einkaufsagenten auf der Mikro-Ebene homogen ist.



Die Struktur der zugehörigen  $IVMF_{AG}$  lässt sich jeweils durch einen 132-Werte (=Anzahl an Geschäften) Vektor beschreiben. Da sich das Einkaufsverhalten während der Simulation nicht dynamisch ändert, kann die Struktur für  $IVMF_{AG}$  hierauf beschränkt werden. Jeder der 132-Werte soll für jeweils ein Geschäft die valide homogene Attraktivitätsbewertung der Einkaufsagenten für das Geschäft beschreiben. Die Semantik der 132-Werte ergibt sich aus der auf der Mikro-Ebene verwendeten Bewertungsfunktion für die einzelnen Einkaufsagenten.

Der homogene Anteil des aggregierten Agentenverhaltens repräsentiert die Summe der Bewertungen  $LF_{Preis}(P_A) + LF_{Sortiment}(S_A) + LF_{Qualität}(Q_A)$ . Jeder der 132 Werte aggregiert also das Ergebnis der Summe der Preis-, Sortiment- und Qualitätswahrnehmungen für jeweils ein Geschäft. Diese Summe ist aufgrund der homogenen Agentenparameter für alle Einkaufsagenten gleich. Der Einfluss der Distanzwahrnehmung ist im Modell heterogen und durch die Rang-Distanz als Teil der Modellstruktur vorbestimmt.

Damit sind für  $KP_{GeoUmeaHom}$  alle Voraussetzungen zur Aufstellung der  $IVMF_{AG}$  erfüllt.

### 15.6.2.2 Ermittlung von $IVM$ zu $IVMF$

Damit der Schritt auf der Makro-Ebene abgeschlossen werden kann, werden die konkreten  $IVM_{AG}$ , also die konkreten 132 Attraktivitätswerte benötigt, welche die homogene kombinierte Wahrnehmung der Einkaufsagenten für Preis, Sortiment und Qualität der Geschäfte aggregieren.

Die Anforderung, dass die  $IVM_{AG}$  spezifiziert werden können, ist für  $KPMAKRO_{GeoUmeaHom}$  nicht erfüllt. Die konkreten Parameterwerte der  $IVMF_{AG}$  sind unklar. Daher kann in  $KPMAKRO_{GeoUmeaHom}$  der Kalibrierungsschritt auf der Makro-Ebene nicht ohne die Reverse-Engineering-Verfeinerung durchgeführt werden.

Im RE-Schritt der Makro-Ebene muss  $IVMF_{AG}$  zu  $IVM_{AG}$  kalibriert werden, bevor der Kalibrierungsschritt auf der Mikro-Ebene durchgeführt werden kann.

Das Kalibrierungsproblem für die Makro-Ebene mit RE-Verfeinerung ist gegeben durch:

$$KPRE = ((\{UM\}, IVMF_{AG}), \{(UM, \text{Geschäfte}), GZF\})$$

- **IVM<sub>AG</sub>**: Die Modellfamilien der homogenen Attraktivitätsbewertungen der Geschäfte bezüglich Preis, Sortiment und Qualität. In den zugehörigen  $IVM_{AG}$  werden nicht wie auf der Mikro-Ebene mit Hilfe der Wahrnehmungsfunktionen und der charakteristischen Geschäftsmerkmale Geschäftsbewertungen berechnet; stattdessen fließen bei Simulation des idealen aggregierten Verhaltensmodells auf der Makro-Ebene des Makro-Mikro-Verfahrens die in  $IVM_{AG}$  festgelegten Attraktivitätswerte direkt in die Simulation ein. Damit wird auf der Makro-Ebene für jedes  $IVM_X$  für ein Einkaufsagenten  $X$  die Gesamt-Attraktivität eines Geschäftes  $A$  als  $\frac{IVM_{AG_A}}{\text{Rangdistanz von } X \text{ zu } A}$  berechnet.
- **GZF**: Die globale Zielfunktion zur Bewertung der Schätzung aller Geschäftsumsätze.

In der Reverse-Engineering-Verfeinerung müssen die 132-Werte der  $IVM_{AG}$  kalibriert werden. Dabei bilden wie beschrieben die festgelegte heterogene Distanzwahrnehmung (Rangbasiert) der Agenten und die  $ZF_{GESCH_i}$  aus GZF die Randbedingungen für die Ermittlung der  $IVM_{AG}$ .

Effektiv wurde der Suchraum des Kalibrierungsproblem durch diesen Schritt zwar vergrößert (vorher: 9 homogene Werte, jetzt: 132 homogene Werte), zur Ermittlung der Werte von  $IVM_{AG}$  kann jedoch leicht eine spezifische Änderungsregel in Form eines einfachen Gradientenansatzes formuliert werden. Die Verwendung des spezifischen Algorithmus (Anwendung eines Regel-basierten White-Box-Verfahrens) erlaubt es die 132 Werte von  $IVM_{AG}$  signifikant schneller zu kalibrieren, als die 9 Parameter der Wahrnehmungsfunktionen auf der Mikro-Ebene mit Black-Box Ansätzen kalibriert werden können.

Grundsätzlich besteht bei der Anwendung von Gradienten- und wissensbasierten Verfahren die Gefahr, dass lediglich ein lokales Optimum des Suchraumes gefunden wird (siehe Abschnitt 3.1.5.2). Bei der Verwendung Gradienten-basierter Verfahren ist daher ein möglichst „guter“ Ausgangspunkt für die Kalibrierung entscheidend. In  $KP_{GeoUmeaHom}$  ist ein solcher Startpunkt durch die bekannten Zielumsätze der Geschäfte gegeben, die zu einem gewissen Grad die Vorlieben der Einkäufer widerspiegeln. Die initialen Werte der  $IVM_{AG}$  werden proportional zu den Zielumsätze der zugehörigen Geschäfte gewählt:

#### Algorithmus: Ermittlung der $IVM_{AG}$

1. Erstelle Variante  $V$  des eigentlichen Modells, in der die Agenten ihre Einkaufsentscheidung auf der Basis der Distanz und der  $IVM_{AG}$ -Werte treffen (direktes Einsetzen der  $IVM_{AG}$ -Werte ohne Berechnung der Wahrnehmungsfunktionen).

$$BewFkt(Geschäft_A, IVM_{AG}, Agent_X) = \frac{IVM_{AG}(A)}{\text{Rangdistanz von } X \text{ zu } A}$$

2. Initialisiere  $IVM_{AG}$  mit Werten, die proportional zu den Zielumsätzen der zugehörigen Geschäfte sind, d.h. für jedes Geschäft  $A$   $IVM_{AG}(A) = \text{Zielumsatz von } A$
3. Simuliere Modellvariante  $V$ .
4. Ermittle für jedes Geschäft  $A$  die relative Verschätzung  $RelAbw(A)$ :

$$RelAbw(A) = \frac{(\text{Zielumsatz}_A - \text{SimulierterUmsatz}_A)}{\text{Zielumsatz}_A}$$

5. Prüfe, ob der Umsatz aller Geschäfte ausreichend genau reproduziert wurde, d.h. prüfe, ob die Summe aller  $\frac{RelAbw(A)}{132}$  kleiner als ein Schwellwert  $s$  ist.:
  - (a) JA: Fertig.
  - (b) NEIN: Adaptiere die  $IVM_{AG}$ -Werte gemäß

$$IVM_{AG}(A)_{neu} = IVM_{AG}(A)_{alt} * (1 + RelAbw(A))$$

Gehe zu 3.

Durch Anwendung des oben beschriebenen Algorithmus konnten die 132 Werte von  $IVM_{AG}$  so kalibriert werden, dass auf der Makro-Ebene die Umsätze der Geschäfte mit einem Wert für  $GZF = 0,99$  reproduzieren konnten. D.h. wurde das aggregierte Modell auf der Makro-Ebene mit den ermittelten  $IVM_{AG}$  und der Berechnungsformel  $BewFkt(Geschäft_A, IVM_{AG}, Agent_X) = \frac{IVM_{AG}(A)}{\text{Rangdistanz von } X \text{ zu } A}$  simuliert, dann war der Wert des unquadratischen Gütemaßes 0,99 und somit das Simulationsverhalten auf der

Makro-Ebene bezüglich  $GZF$  valide.

Mit der Ermittlung der  $IVM_{AG}$  war nun der Kalibrierungsschritt auf der Makro-Ebene beendet. Die  $IVM_{AG}$  werden nun zu Kalibrierung auf der Mikro-Ebene genutzt.

### 15.6.3 Durchführung der Kalibrierung auf der Mikro-Ebene

Basierend auf den ermittelten  $IVM_{AG}$  muss nun auf der Mikro-Ebene das folgende Kalibrierungsproblem gelöst werden:

$$KPMIKRO_{GeoUmeaHom} = ((\{UM\}, \{AGF\}), VK = \{GZF, (AG, IZF_{AG})\})$$

Damit die Kalibrierung auf der Mikro-Ebene durchgeführt werden kann, muss es möglich sein die  $IVM_{AG}$  in  $IZF_{AG}$  abzubilden. Diese Abbildung ist in diesem Fall trivial möglich.  $IZF_{AG}$  misst:

$$IZF_{AG} = \sum_{i=1}^{\#Geschäfte} |IVM_{AG}(Geschäft_i) - (LF_{Preis}(P_A) + LF_{Sortiment}(S_A) + LF_{Qualität}(Q_A))|$$

Da nur homogene Parameterwerte kalibriert werden, ist  $IZF$  für alle Agenten gleich und kann unabhängig von konkreten Agentenmodellen betrachtet werden. Nun müssen auf der Mikro-Ebene die  $P_{AG}$  (der homogenen Parameter der logistischen Wahrnehmungsfunktionen für Preis, Sortiment und Qualität) nur auf Basis der  $IZF_{AG}$  kalibriert werden. Da die  $IVM_{AG}$  homogen und unabhängig von der Distanzwahrnehmung der Agenten sind, können die Parameterwerte der  $AGF$  aus den  $IVM_{AG}$  hergeleitet werden, ohne dass der Rest des Modells simuliert werden muss. Die Ermittlung der Parameterwerte kann durch Lösung eines Gleichungssystems der Form:

$$\begin{aligned} LF_{Preis}(Gesch_1) + LF_{Sort}(Gesch_1) + LF_{Qualität}(Gesch_1) &= IVM_{AG}(Gesch_1) \\ LF_{Preis}(Gesch_1) + LF_{Sort}(Gesch_1) + LF_{Qualität}(Gesch_1) &= IVM_{AG}(Gesch_2) \\ &\dots = \dots \\ LF_{Preis}(Gesch_{132}) + LF_{Sort}(Gesch_{132}) + LF_{Qualität}(Gesch_{132}) &= IVM_{AG}(Gesch_{132}) \end{aligned}$$

durchgeführt werden und benötigt keine Simulationsläufe mehr. Die Unbekannten des Gleichungssystems sind dabei die ursprünglich zu kalibrierenden neun Parameter der logistischen Funktionen. Da es neun Unbekannte und 132 Gleichungen gibt, ist das Gleichungssystem überbestimmt.

In der Fallstudie wurde die Ermittlung der Parameterwerte von  $AGF$  auf der Basis der  $IVM_{AG}$  mit Hilfe des Solver-Plugins für Microsoft Excel durchgeführt. Hierzu wurde eine Excel Tabelle aus den für die Studie verfügbaren Geschäftsdaten erstellt.

Die Tabelle enthält für jedes Geschäft:

- Dessen jeweiligen Stufenwert für jedes der relevanten Geschäftskriterien.

- Den ermittelten Attraktivitätswert für das Geschäft gemäß der  $IVM_{AG}$ .

Weiter enthält die Tabelle:

- Die kombinierte Wahrnehmungsfunktion  $LF_{Preis}(A) + LF_{Sort}(A) + LF_{Qualität}(A)$  der Agenten für die homogen wahrgenommenen charakteristischen Geschäftsmerkmale.

Die Tabelle muss keine Informationen bezüglich der heterogenen Distanzwahrnehmung der Agenten mehr enthalten, da der Einfluss dieser Werte bereits in den errechneten  $IVM_{AG}$  enthalten ist.

Mit Hilfe der Optimierung des Solver-Plugins können nun in wenigen Sekunden die Parameter von  $LF_{Preis}$ ,  $LF_{Sort}$  und  $LF_{Qualität}$  so eingestellt werden, dass die Verrechnungsfunktion die Stufenwerte der Geschäftskriterien für jedes Geschäft möglichst gut auf den Attraktivitätswert gemäß der  $IVM_{AG}$  abbildet.

Abbildung 15.1 zeigt einen Ausschnitt aus einer solchen Tabelle. Die resultierenden Pa-

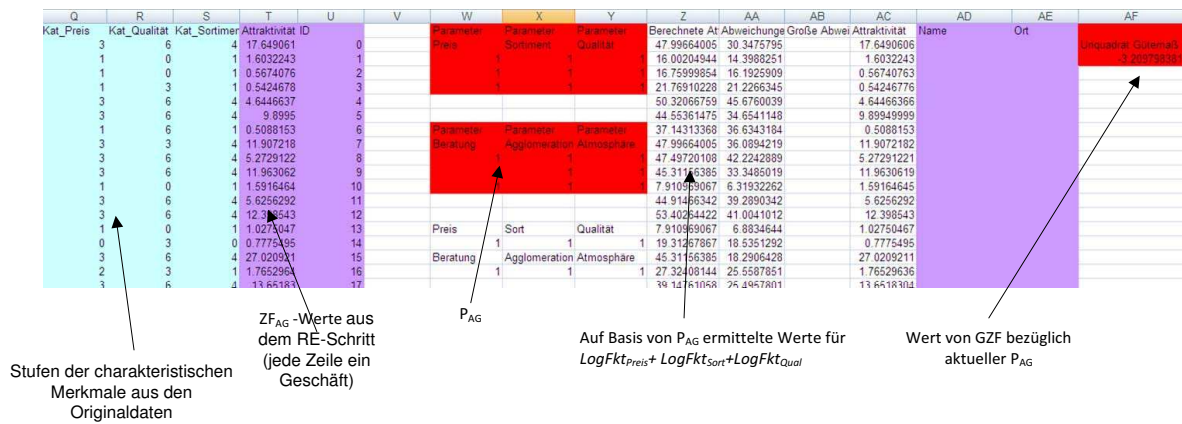


Abbildung 15.1: Veranschaulichung: Ausschnitt aus einer Excel-Tabelle zur Durchführung der Kalibrierung auf der Mikroebene.

parameterwerte sind die gesuchten Werte der Parameter von  $AGF$  und können direkt in das Simulationsmodell eingesetzt werden. Da die Parametersuche mit dem Solver-Plugin Gradienten-basiert ist, besteht auch hier die Gefahr, dass lokale Optima des Suchraums gefunden werden. Da die Berechnungsdauer für die Parameterwerte aber vergleichsweise sehr gering ist, können leicht mehrere Einstellungsläufe von verschiedenen Startpunkten aus durchgeführt werden, ohne dass dies wesentlich länger dauert. Tabelle 15.4 zeigt Ergebnisse der Lösung des Gleichungssystems, diese werden im nächsten Abschnitt diskutiert.

Die Parameter aus der Lösung des Gleichungssystems stellen nun das Ergebnis der Kalibrierung auf der Mikro-Ebene dar. Die Parameterwerte können in das Mikro-Modell zu  $KP_{GeoUmeaHom}$  eingesetzt werden und erzeugen in der Simulation den gleichen Wert des unquadratischen Gütemaßes wie er im Gleichungssystem berechnet werden konnte.

Dies schließt die Kalibrierung auf der Mikro-Ebene und damit die Kalibrierung mit dem Makro-Mikro-Verfahren ab. Auf der Mikro-Ebene wurden die Parameter der Agentenmodelle auf der Grundlage der ermittelten  $IVM_{AG}$  kalibriert. Zur Kalibrierung konnte ein spezialisiertes Lösungsverfahren in Form eines Gleichungssystems eingesetzt werden, wodurch keine weiteren Simulationsläufe zur Kalibrierung mehr notwendig waren.

Parameter-nummer	Preis	Sortiment	Qualität	GZF-Bewertung	#Simulationsläufe
P1	3,20	3,95	1,53	0,574	5
P2	2,32	4,17	3,55		
P3	1,00	1,40	2,28		
P1	3,25	4,46	0,0001	0,593	5
P2	2,13	4,65	1,96		
P3	0,88	1,37	2,81		
P1	3,15	3,91	0,0001	0,598	5
P2	1,98	4,28	2,34		
P3	0,86	1,48	0,05		

Tabelle 15.4: Makro-Mikro-Ergebnisse.

### 15.6.4 Kalibrierungsergebnis des Makro-Mikro-Verfahrens

In diesem Abschnitt wird das Ergebnis der Kalibrierung mit dem Makro-Mikro-Verfahren mit dem Ergebnis aus der Black-Box-Kalibrierung verglichen.

Tabelle 15.4 zeigt mit dem Makro-Mikro-Verfahren ermittelte Parametereinstellungen und zugehörige *GZF*-Bewertung des unquadratischen Gütemaßes. Dabei fallen zwei Punkte direkt auf:

1. Die Parametereinstellung, für die sich der beste *GZF*-Wert ergibt, ist mit 0,598 um 0,018 minimal besser als das Ergebnis mit der Black-Box-Suche.
2. Wie bei der Black-Box-Kalibrierung können in mehreren Kalibrierungsläufen unterschiedliche Parametereinstellungen mit annähernd gleicher *GZF*-Bewertung gefunden werden. Tabelle 15.4 zeigt drei mit dem Makro-Mikro-Verfahren ermittelte Beispielpametereinstellungen.

#### 15.6.4.1 Multiple Suchraumoptima

Wie beim Einsatz der Black-Box-Kalibrierungsverfahren konnten in der Fallstudie auch beim Einsatz des Makro-Mikro-Verfahrens mit Reverse-Engineering-Verfeinerung verschiedene Optima bezüglich *GZF* mit annähernd gleicher Validitätsbewertung identifiziert werden. Dies bedeutet, dass, obwohl durch die zusätzlichen *IVM* zusätzliche Randbedingungen an das Verhalten der Einkaufsagenten festgelegt wurden, der Suchraum immer noch verschiedene Parametereinstellungen mit sehr ähnlichem Simulationsverhalten enthielt. Zum einen liegt dies an der Verwendung homogener Parameter. Am Gleichungssystem im letzten Abschnitt sieht man, dass 132 Gleichungen zur Bestimmung von 9 Parametern verwendet werden, was stark überbestimmt ist. Als zweiter möglicher Grund kann die Verwendung des unquadratischen Gütemaßes angeführt werden, bei dem die Verschätzungsgrade kleinerer Geschäfte so wenig in die Bewertung eingehen, dass diese verschieden stark und auf verschiedene Art verschätzt werden können, ohne dass sich die zugehörige Bewertung mit dem unquadratischen Gütemaß wesentlich ändert.

## 15.7 $KP_{GeoUmeaHom}$ : Evaluationsergebnis Makro-Mikro-Verfahren

- **Anwendbarkeit des Makro-Mikro-Verfahrens:** Die Evaluation für  $KP_{GeoUmeaHom}$  hat gezeigt, dass das Makro-Mikro-Verfahren zur Kalibrierung eingesetzt werden konnte. In der Fallstudie war es nicht möglich den Makro-Schritt ohne Reverse-Engineering-Verfeinerung einzusetzen.
- **Zusätzliche Modellierungsarbeit:** Zur Kalibrierung der  $IVMF$  auf der Makro-Ebene musste ein angepasstes Modell aggregierten Agentenverhaltens auf Basis der  $IVM$  erstellt werden. Dies ist bei Black-Box-Kalibrierung nicht notwendig.
- **Einsatz spezialisierter Kalibrierungsverfahren:** Für das Reverse-Engineering konnte ein Optimierungsalgorithmus angegeben werden und auf der Mikro-Ebene konnten die Parameterwerte durch Lösung eines Gleichungssystems ermittelt werden, wodurch weniger Simulationsläufe benötigt wurden als bei Black-Box-Kalibrierung.
- **Größe des Parametersuchraumes:** Während der Kalibrierung im RE-Schritt müssen 132 positive Werte und während der Kalibrierung auf der Mikroebene neun Parameterwerte eingestellt werden. Im Vergleich zu einem Black-Box-Ansatz ist der Suchraum somit deutlich größer, da neben dem identischen Kalibrierungsproblem auf der Mikro-Ebene (neun Parameterwerte) ein zusätzliches Kalibrierungsproblem mit 132 Werte gelöst werden muss.
- **Validitätsbewertung:** Das Ergebnis der Makro-Mikro-Kalibrierung ist hinsichtlich des erreichten Validitätsgrades eines Simulationslaufes mit kalibrierten Parameterwerten  $PW_{AG}$  ähnlich dem der Black-Box-Kalibrierung. Der beste ermittelte Wert des Validierungskriteriums  $GZF$  lag bei 0,598 und war somit leicht besser als das Black-Box-Ergebnis von 0,58. Zusammenfassend kann gesagt werden, dass in der Fallstudie eine Kalibrierung mit Hilfe des Makro-Mikro-Verfahrens zu einem in etwa genauso guten Kalibrierungsergebnis wie das des Black-Box-Kalibrierungsansatzes geführt hat. Dieses Ergebnis ist nicht überraschend, da die Kalibrierung unter den gleichen Randbedingungen durchgeführt wurde.
- **Bewertungsverschlechterung beim Wechsel von Makro- zur Mikro-Ebene:** Ein interessanter Punkt ist allerdings die Bewertungsverschlechterung des resultierenden Simulationsverhaltens auf der Mikro-Ebene im Vergleich zu dem der Makro-Ebene. Nach Abschluss der Kalibrierung auf der Makro-Ebene war ein aggregiertes Modell basierend auf den  $IVM_{AG}$  gefunden worden, das auf Makro-Ebene valides Verhalten bezüglich  $GZF$  aufwies und homogene Parameterwerte hatte. Die Verschlechterung beim Schritt auf die Mikro-Ebene kann ihre Ursache daher nur darin haben, dass die Parameterwerte der Modellstrukturen der Mikro-Ebene nicht so kalibriert werden konnten, dass das Verhalten der Mikro-Modelle dem der aggregierten  $IVM$  entsprach. Die Ursache war nicht die Verwendung homogener Parameterwerte, da diese auch bei der Kalibrierung auf der Makro-Ebene verwendet wurden. Damit bleibt nur die Ursache übrig, dass die ermittelten  $IVM_{AG}$  und die auf der Mikro-Ebene definierten Strukturen der Einkaufsagentenmodelle nicht zueinander passten. Entweder waren die  $IVM$  ungeeignet und doch nicht valide oder die  $IVM$  waren valide und die ursprünglichen Mikro-Modell-Strukturen waren ungeeignet, um valides Agentenverhalten zu erzeugen.

Folgt man dem Vorgehen des Makro-Mikro-Verfahrens strikt, so durfte der Kalibrierungsschritt auf der Makro-Ebene nur beendet werden, wenn die Simulation der aggregierten *IVM* auf der Makro-Ebene zu validem Simulationsverhalten führten. Auf Basis dieses Ergebnisses muss unterstellt werden, dass die Strukturen der Einkaufsagenten auf der Mikro-Ebene ungeeignet waren. Diese strikte Annahme, dass die *IVM* absolut valide und daher die Strukturen der Einkaufsagenten ungeeignet waren, war in der Fallstudie jedoch nur schwer zu halten, da es nicht möglich war, die vollständige Validität des Verhaltens auf der Makro-Ebene zu prüfen. In der Fallstudie war das Verhalten auf der Makro-Ebene bezüglich *GZF* valide. Es war in der Fallstudie aber nicht möglich, zu überprüfen, in welchen Geschäften jeder einzelne Einkaufsagent wie viel Geld ausgeben musste. Hierfür fehlten entsprechende Daten, die in der Realität nicht erhoben werden konnten.

Die Fallstudie zeigte also, dass das Kalibrierungsergebnis des Makro-Mikro-Verfahrens stark davon abhängt wie gut ermittelte *IVM* und die zugehörigen Modelle auf der Mikro-Ebene aufeinander abbildbar sind.

- **Anzahl benötigter Simulationsläufe:** Der Vorteil beim Einsatz des Makro-Mikro-Verfahrens gegenüber der Black-Box Kalibrierung in der Fallstudie bestand in der geringeren Menge an benötigten Simulationsläufen zum Erreichen des Kalibrierungsergebnisses. Für die Ermittlung der *IVM<sub>AG</sub>*, die auf der Makro-Ebene zu annähernd vollständig validem Simulationsverhalten hinsichtlich des *GZF* führten (*GZF*-Wert: 0,99 bei Simulation der *IVM*), waren maximal 5 Iterationen, d.h. Simulationsläufe nötig. Die Berechnung der eigentlichen Parameterwerte der Agenten auf der Mikroebene konnte ohne weitere Simulationsläufe durchgeführt werden und war somit effektiv bezüglich benötigter Kalibrierungszeit „kostenlos“. Im Vergleich zum Black-Box-Ansatz (durchschn. 300 Simulationsläufe) kann die Kalibrierung also, aufgrund der Möglichkeit, ein effizientes Gradienten-basiertes Suchverfahren zu verwenden, wesentlich schneller durchgeführt werden.
- **Einschränkung der multiplen Optima bei Kalibrierung auf der Mikro-Ebene:** Die Fallstudie hat gezeigt, dass trotz Festlegung der *IVM* und Verwendung bei der Kalibrierung auf der Mikro-Ebene die Existenz von multiplen Optima des Suchraumes auf der Mikro-Ebene nicht ausgeschlossen werden kann.
- **Zerlegbarkeit des Kalibrierungsproblems:** Eine Zerlegung des Modells auf der Mikro-Ebene in Teilmodelle für jedes Einkaufsagentenmodell war aufgrund der homogenen Parameterwerte nicht notwendig. Die Zerlegung wäre aber möglich gewesen. Dabei wäre für jedes Agentenmodell die identische Kalibrierung wie für die Mikro-Ebene beschrieben durchgeführt worden.
- **Zusammenhang von *IVM* und Mikro-Verhalten:** Die Qualität des Ergebnisses des Makro-Mikro-Verfahrens hängt stark davon ab, wie gut die ermittelten *IVM* und die Modelle auf der Mikro-Ebene zueinander passen, d.h. ob die Mikro-Modelle so kalibriert werden können, dass ihr Verhalten dem der zugehörigen *IVM* entspricht.

Das Makro-Mikro-Verfahren hatte in der Fallstudie bei gleichbleibender Ergebnisqualität Vorteile bei der benötigten Anzahl an Simulationsläufen, die aber durch zusätzlichen Modellierungsaufwand und zusätzlichen Aufwand zur Implementierung der speziellen

Suchverfahren erkaufte werden mussten.

**Zentral für die Anwendbarkeit des Makro-Mikro-Verfahrens auf  $KP_{GeoUmeaHom}$  waren folgende Punkte. Wären diese Punkte nicht erfüllt gewesen hätte das Makro-Mikro-Verfahren nicht eingesetzt werden können:**

1. Die Struktur  $IVMF$  konnte für die Fallstudie spezifiziert werden, d.h. es konnte spezifiziert werden, wodurch aggregiertes Agentenverhalten auf der Makro-Ebene simuliert werden kann.
2. Es konnte für die Fallstudie ein Algorithmus zur Kalibrierung der  $IVMF$  in der Reverse-Engineering-Verfeinerung angegeben werden. Generell müssen die  $IVMF$  kalibrierbar sein. Wenn keine Möglichkeit gefunden wird die  $IVMF$  zu validen  $IVM$  zu kalibrieren kann das Makro-Mikro-Verfahren nicht angewendet werden.
3. Die in der Fallstudie ermittelten  $IVM$  wurden von Fachexperten als valide Modelle idealen Agentenverhaltens akzeptiert und konnten zur weiteren Kalibrierung auf der Mikro-Ebene eingesetzt werden. Wären die ermittelten  $IVM$  auf der Makro-Ebene nicht als valide angesehen worden, wären sie für die weitere Kalibrierung unbrauchbar.
4. Auf der Mikro-Ebene konnten leicht die  $IZF$  aus den  $IVM$  abgeleitet werden. Die ermittelten  $IVM$  sind nur dann zur Kalibrierung auf der Mikro-Ebene verwendbar, wenn klar ist wie die Mikro-Modelle gegen die  $IVM$  verglichen werden müssen, damit die  $IZF$  berechnet werden können. Für komplexe Modelle kann es analog zum Problem geeignete  $IVMF$  zu spezifizieren auch schwieriger werden, zu gewählten  $IVM$  geeignete  $IZF$  auf der Mikro-Ebene abzuleiten.

Damit ist die Evaluierung des Makro-Mikro-Verfahrens anhand der Fallstudie  $KP_{GeoUmeaHom}$  abgeschlossen.

## 15.8 $KP_{GeoUmeaHet}$ : Problemstellung

Im vorherigen Abschnitt wurde das Makro-Mikro-Verfahren im Vergleich zu Black-Box-Kalibrierung anhand der Kalibrierung der Agenten-basierten Einkaufssimulation mit homogenen Agentenparametern evaluiert. In diesem Abschnitt wird ein das Problem  $KP_{GeoUmeaHet}$  betrachtet, das sich von  $KP_{GeoUmeaHom}$  nur durch heterogene zu kalibrierende Agentenparameterwerte unterscheidet.

### 15.8.1 Problembeschreibung $KP_{GeoUmeaHet}$

Im Folgenden wird das zu lösende Kalibrierungsproblem  $KP_{GeoUmeaHet}$  beschrieben, dabei werden lediglich die Teile näher hervorgehoben, die sich von  $KP_{GeoUmeaHom}$  unterscheiden.

$$KP_{GeoUmeaHet} = ((\{UM\}, \{EKF_1, \dots, EKF_{999}\}), VK)$$

- $EKF_i$ : Eine Einkaufsagentenmodellfamilie, d.h. ein  $IAMF_i$  gemäß Kapitel 5.1.



- $EKF_i = (S_{AG}, V_{AG}, P_{EKF_i})$ : Die Definition einer Einkaufsagentenfamilie ist wie bei  $KP_{GeoUmeaHom}$  mit dem Unterschied, dass für jede  $EKF_i$  eigene Parameterwerte gesucht werden. Für die Einkaufsentscheidungen der Agenten bleiben wie zuvor die charakteristischen Geschäftsmerkmale **Preis, Sortiment, Qualität und Distanz** relevant.

## 15.9 $KP_{GeoUmeaHet}$ : Lösung mit Black-Box-Ansatz

Das Lösungsvorgehen zur Kalibrierung von  $KP_{GeoUmeaHet}$  mit Hilfe eines Black Box-Kalibrierungsansatzes ist identisch zum Lösungsvorgehen  $KP_{GeoUmeaHom}$ . Die Menge der zu kalibrierenden  $P_{EKF_i}$  ist aber für die 1000 Agenten jetzt um den Faktor 1000 erhöht, d.h. es müssen nicht mehr 9 sondern 9000 Parameterwerte kalibriert werden. Aufgrund dieses riesigen Parametersuchraumes ist  $KP_{GeoUmeaHet}$  mit einem Black-Box-Kalibrierungsansatz in begrenzter Zeit mit vorhandenen Computern nicht mit einem akzeptablen Ergebnis lösbar. Es konnte mit Black-Box-Kalibrierung kein Ergebnis gefunden werden, das eine ähnlich hohe  $GZF$ -Bewertung erhielt wie im homogenen Fall.

## 15.10 $KP_{GeoUmeaHet}$ : Lösung mit Makro-Mikro-Verfahren

Die Kalibrierung von  $KP_{GeoUmeaHet}$  wird mit dem Makro-Mikro-Verfahren wie für den homogenen Fall  $KP_{GeoUmeaHom}$  durchgeführt und  $KP_{GeoUmeaHet}$  in zwei zusammenhängende Kalibrierungsprobleme auf der Makro- und auf der Mikroebene aufgeteilt.

### 15.10.1 Kalibrierung auf der Makro-Ebene für $KP_{GeoUmeaHet}$

Im Makro-Kalibrierungsschritt für  $KP_{GeoUmeaHet}$  hat sich das Makro-Ebenen-Kalibrierungsproblem im Vergleich zu  $KP_{GeoUmeaHom}$  wie folgt geändert. Anstelle von

$$KPMAKRO_{GeoUmeaHom} = ((\{UM\}, \{IVMF_{AG}\}), ((Umweltmodell, \{Geschäfte\}), GZF))$$

gilt:

$$KPMAKRO_{GeoUmeaHet} = ((\{UM\}, \{IVMF_1 \dots IVMF_{999}\}), ((Umweltmodell, \{Geschäfte\}), GZF))$$

#### 15.10.1.1 Lösung von $KPMAKRO_{GeoUmeaHom}$ ist eine Lösung für $KPMAKRO_{GeoUmeaHet}$

Das Kalibrierungsproblem auf der Makro-Ebene ist stark unterbestimmt. Es werden zwar heterogene  $IVM$  mit viel mehr Parametern gesucht, aber es werden im Vergleich zum homogenen Fall keine zusätzlichen Randbedingungen an die Festlegung der  $IVM$  vorgegeben. Durch die Unterbestimmtheit können bei der Kalibrierung auf der Makro-Ebene viele mehr oder weniger zufällige Parametereinstellungen gefunden werden, welche gleich valides Simulationsverhalten der  $IVM$  erzeugen können. Es ist in diesem Fall nicht möglich zu entscheiden, welche dieser Einstellungen tatsächlich besser als eine andere, bzw. überhaupt am besten geeignet ist.

Als Beispiel wird wieder die Lösung von  $KPMAKRO_{GeoUmeaHom}$  betrachtet. Da sich am Makro-Kalibrierungsproblem keine  $VK$  oder sonstige Randbedingungen geändert haben, stellt die Lösung von  $KPMAKRO_{GeoUmeaHom}$  auch eine mögliche Lösung zu

$KPMAKRO_{GeoUmeaHet}$  dar. Der Grund hierfür ist, dass schon für  $KPMAKRO_{GeoUmeaHom}$  bei der Simulation der ermittelten  $IVM$  auf der Makro-Ebene heterogenes Verhalten der Einkaufsagenten simuliert wurde. Die  $IVM_{AG}$  waren zwar für alle Agenten identisch, aber durch den heterogenen Einfluss der festgelegten Rang-basierten Distanz im Modell war das Gesamtverhalten in der Simulation des aggregierten Verhaltens auf der Makro-Ebene heterogen.

Für  $KPMAKRO_{GeoUmeaHom}$  erzeugten die kalibrierten  $IVM$  bei Simulation des aggregierten Modells auf der Makro-Ebene ein bezüglich  $GZF$ -Bewertung vollständig valides Verhalten. Der  $GZF$ -Wert auf der Makro-Ebene war 0,99. Für  $KPMAKRO_{GeoUmeaHet}$  bedeutet dies, dass bei Kalibrierung heterogener  $IVMF$  kalibriert unter den gleichen Randbedingungen kein bezüglich  $GZF$  besseres Kalibrierungsergebnis als die Lösung von  $KPMAKRO_{GeoUmeaHom}$  gefunden werden kann. Die Lösung von  $KPMAKRO_{GeoUmeaHom}$  konnte direkt als Lösung für  $KPMAKRO_{GeoUmeaHet}$  eingesetzt werden indem einfach für jedes  $IVMF_i = IVMF_{AG}$  gesetzt wurde.

Aufgrund der starken Unterbestimmtheit war es ohne weitere Randbedingungen nicht möglich zu bestimmen welche Parametereinstellung das tatsächlich benötigte Ergebnis des Makro-Schrittes für das heterogene Kalibrierungsproblem sein musste. Damit war die Durchführung des Mikro-Kalibrierungsschrittes für  $KP_{GeoUmeaHet}$  unter den gegebenen Randbedingungen nicht möglich, auf der Mikro-Ebene hätte man ebenso beliebige Kalibrierungsergebnisse erhalten können wie auf der Makro-Ebene. Beispielsweise wäre auch das Kalibrierungsergebnis für den Mikro-Schritt ebenfalls aus  $KP_{MIKRO_{GeoUmeaHom}}$  ein mögliches Ergebnis für  $KP_{GeoUmeaHet}$  gewesen.

Da für  $KP_{GeoUmeaHet}$  im RE-Schritt auf der Makro-Ebene des Makro-Mikro-Verfahrens keine geeigneten  $IVM$  für den Mikro-Schritt ermittelt werden konnten, musste nun der Iterationsschritt des Makro-Mikro-Verfahrens zur Anpassung der  $IVM$ -Strukturannahmen auf der Makro-Ebene durchgeführt werden.

### 15.10.2 Bisheriges Evaluationsergebnis der Fallstudie zu $KP_{GeoUmeaHet}$

Für die Evaluation des Makro-Mikro-Verfahrens zeigte sich in der Fallstudie besonders stark die Abhängigkeit des Makro-Mikro-Kalibrierungsergebnisses von der Qualität der im Makro-Schritt ermittelten  $IVM$ . Können die  $IVM$  nicht eindeutig ermittelt werden, so kann das Makro-Mikro-Verfahren nicht durch die beiden Schritte „Kalibrierung auf der Makro-Ebene“ und „Kalibrierung auf der Mikro-Ebene“ abgeschlossen werden. Stattdessen wird der Iterationsschritt zur Anpassung der  $IVM$ -Strukturen auf der Makro-Ebene notwendig. Sind verschiedene mögliche Kandidatenstrukturen für  $IVMF$  oder Kandidatenmodelle von  $IVMs$  bekannt so bedeutet die Iteration ein iteratives Testen und Vergleichen der Kandidatenstrukturen, indem für jede Kandidatenstruktur der Makro- und der Mikro-Schritt durchgeführt werden. Das Makro-Mikro-Verfahren bietet für diesen Fall verschiedener konkurrierender Kandidatenmodelle aggregierten Verhaltens eine strukturierte Vorgehensweise zur Kalibrierung.

Im schlechtesten Fall, wenn pures Ausprobieren von  $IVM$ -Änderungen notwendig wird, bietet das Makro-Mikro-Verfahren nur noch dann Vorteile gegenüber Black-Box-Verfahren, wenn:

- der Parametersuchraum auf der Mikro-Ebene so groß ist, dass Black-Box-Verfahren nicht anwendbar sind
- oder wenn Validität der eigentlichen Mikro-Modellstrukturen selbst in Frage gestellt wird und zunächst auf aggregierter Ebene geklärt werden soll, wie das Ziel-Verhalten aussehen muss.

### 15.10.3 Anpassung Randbedingungen und Validierungskriterien für die Makro-IVM-Ermittlung in $KP_{\text{GeoUmeaHet}}$

Die Fallstudie zu  $KP_{\text{GeoUmeaHet}}$  wurde nun so fortgesetzt, dass im Schritt auf der Makro-Ebene des Makro-Mikro-Verfahrens zu den bisher ermittelten IVM weitere mögliche IVM gesucht wurden. Andere mögliche IVM-Lösungen können ermittelt werden, indem die Randbedingungen und die Kalibrierung der IVMF im Reverse-Engineering verändert werden und dann das Reverse-Engineering erneut durchgeführt wird.

Die Randbedingungen für das Reverse-Engineering in  $KP_{\text{GeoUmeaHet}}$  waren bisher gegeben durch:

- Die zu treffenden Zielumsätze der Geschäfte.
- Die festgelegte Rang-basierte Distanzwahrnehmung.
- Den zur Kalibrierung der  $IVM_{AG}$  eingesetzten Algorithmus.

Das Verhältnis der Geschäftsattribute zueinander, also z.B. ob ein Geschäft  $A$  eine höheren oder niedrigeren Preisattributswert hat als ein Geschäft  $B$  war bei der bisherigen Ermittlung der IVM noch nicht betrachtet worden.

Die Randbedingungen für die Ermittlung der IVM auf der Makro-Ebene wurden im Iterationsschritt nun wie folgt modifiziert:

1. Die Hauptänderung war, dass der bisherige Kalibrierungsalgorithmus so angepasst wurde, dass er die zusätzliche Randbedingung an das aggregierte Agentenverhalten durchsetzen sollte, dass ein Agent ein Geschäft  $A$  nicht attraktiver finden konnte als ein Geschäft  $B$ , wenn  $A$  in allen Geschäftsattributen höchstens gleich gut und in mindestens einem schlechter eingestuft war als  $B$ .
2. Dabei sollte die Distanzwahrnehmung der Agenten Rang-basiert sein, es sollte aber keine Festlegung mehr getroffen werden, wie die Rang-basierte Distanz in die Bewertung der Agenten einfließt. Stattdessen sollte der Distanzrang genau wie die übrigen Geschäftsattribute behandelt werden.

Entsprechend wurde der dritte Schritt des Kalibrierungsalgorithmus für das Reverse-Engineering für  $KP_{\text{GeoUmeaHet}}$  modifiziert.

#### Algorithmus 2: Ermittlung der IVM

(\*) Führe bisherige Schritte 1-3 aus dem Reverse-Engineering-Algorithmus für  $KP_{\text{GeoUmeaHet}}$  durch.

3. ... (fortgesetzt)

Für jede  $IAMF_i$  in  $IAMF$

- (a) Sortiere die Attraktivitätswerte in  $IVM_i$  aufsteigend gemäß den Relationen  $R<$ ,  $R=$  und  $R\neq$  mit:
- i.  $Wert_1 R < Wert_2$ , gdw. das zu  $Wert_2$  gehörige Geschäft in allen relevanten charakteristischen Geschäftsmerkmalen mindestens so hoch eingestuft wird, wie das zu  $Wert_1$  gehörende Geschäft.  
**Gilt  $Wert_1 < Wert_2$ , so muss  $Wert_1$  in der Sortierung weiter am Anfang stehen als  $Wert_2$ .**
  - ii.  $Wert_1 R = Wert_2$ , gdw. das zu  $Wert_2$  gehörige Geschäft in allen relevanten charakteristischen Geschäftsmerkmalen identisch zu dem zu  $Wert_1$  gehörenden Geschäft eingestuft wird.  
**Gilt  $Wert_1 = Wert_2$ , so müssen die beiden Werte in der Sortierung nebeneinander stehen.**
  - iii.  $Wert_1 R \neq Wert_2$ , gdw. das zu  $Wert_1$  gehörige Geschäft ist in einigen relevanten charakteristischen Geschäftsmerkmalen besser und in einigen schlechter eingestuft, als das zu  $Wert_2$  gehörige Geschäft. In diesem Fall sind die beiden Geschäfte nicht vergleichbar.  
**Gilt  $Wert_1 \neq Wert_2$ , so können die beiden Werte in der Sortierung beliebig zueinander positioniert sein.**
- (b) Durchlaufe aufsteigend alle Werte  $W_k (k \in [1; 132])$  des Vektors aus  $IVM_i$
- i. Ersetze Werte für Geschäfte mit gleichen Merkmalen durch einheitlichen Mittelwert:  
Wähle alle direkt auf  $W_k$  folgenden Werte  $W_{k+1}, W_{k+2}, \dots, W_{k+t}$  mit  $W_{k+t} R = W_k$ . Setze  $W_k$  bis  $W_{k+t}$  *Mittelwert*( $W_k, W_{k+1}, W_{k+2}, \dots, W_{k+t}$ ).
- (c) Durchlaufe aufsteigend alle Werte  $W_k$  aus  $IVM_i$
- i. Gilt für einen Wert  $W_k$ , dass der Attraktivitätswert des Nachfolgerwertes  $W_{k+1}$  kleiner ist als der von  $W_k$  (Verletzung von  $RB_{AMV}$ ), so ersetze den Wert von  $W_k$  durch den von  $W_{k+1}$  (wegen  $Wert_k R < Wert_{k+1}$  kann  $Wert_k$  höchstens so groß sein, wie  $Wert_{k+1}$ ).

4. Gehe zu Beginn von Schritt 3.

Basierend auf diesem angepassten Algorithmus wurde im Iterationsschritt die Ermittlung der  $IVM$  auf der Makro-Ebene des Makro-Mikro-Verfahren erneut durchgeführt.

### 15.10.3.1 Kalibrierung auf der Makro-Ebene

Der Kalibrierungsschritt auf der Makro-Ebene wurde wie für  $KP_{GeoUmeaHom}$  beschrieben durchgeführt, nur dass der oben beschriebene erweiterte Algorithmus 2 für das Reverse-Engineering der  $IVM$  verwendet wurde.

Ergebnis dieses Kalibrierungsschrittes waren heterogene  $IVM$  für alle 1000  $IVMF$  zu den Einkaufsagenten auf der Mikro-Ebene. Hauptgrund für die Heterogenität ist dabei, dass heterogene Rang-basierte Distanzattribut der Geschäfte explizit im Kalibrierungsalgorithmus verwendet wurde und nicht mehr als festgelegter Wert zu homogenen  $IVM$ -Werten

hinzugerechnet wurde.

Das Kalibrierungsergebnis auf der Makro-Ebene waren 1000-Mal 132 Werte für die heterogenen  $IVM$ . Der Algorithmus benötigte hierzu 3 Iterationen mit jeweils einem Simulationslauf und Parameteränderungsberechnungen was einer Laufzeit von ca. 5 Minuten auf einem Intel Centrino Duo Laptop mit T5500 (1.66GHz) Prozessor entsprach. Das Kalibrierungsergebnis bezüglich  $GZF$  bei Simulation der  $IVM$  auf der Makro-Ebene betrug 0,73. Im Vergleich zu den zuvor ermittelten  $IVM$  für  $KP_{GeoUmeaHom}$  war dies eine Verschlechterung um 0,26.

Unabhängig vom der Verschlechterung bezüglich des  $GZF$ -Wertes wurde das aggregierte Verhalten der  $IVM$ -Simulation von den Fachexperten auch auf seine Plausibilität hin bewertet und für plausibel befunden. Aus diesem Grund wurden die ermittelten heterogenen  $IVM$  als Ergebnis des Kalibrierungsschritts auf der Makro-Ebene akzeptiert und der Makro-Schritt abgeschlossen.

Zwei Ursachen kommen für die Verschlechterung in Frage: Zum einen kann es sein, dass die zusätzlichen Randbedingungen verhinderten, dass der Zielattraktivitätswert für ein Geschäft an den  $IVM$  Zielumsatz des Geschäftes angeglichen werden konnte als bei  $KP_{GeoUmeaHom}$ . Auf diese Weise wirkten sich Fehler in den verfügbaren Daten über die Geschäftsattribute stärker auf ermittelten  $IVM$  aus. Als zweiter Grund ist denkbar, dass der Algorithmus zur Durchsetzung der Randbedingungen ungünstig gewählt war.

### 15.10.3.2 Kalibrierung auf der Mikro-Ebene für $KP_{GeoUmeaHet}$

Gegeben war das Kalibrierungsproblem auf Mikro-Ebene durch:

$$KPMIKRO_{GeoUmeaHet} = ((\{UM\}, \{EKF_1, \dots, EKF_{999}\}),$$

$$VK = \{GZF, (Geschäfte, IZF_1, \dots, IZF_{999})\})$$

Auf Basis der ermittelten  $IVM$  und der zugehörigen  $IZF$  konnte das Mikro-Kalibrierungsproblem mit White-Box-Zerlegungstechniken vereinfacht werden. Es war eine Zerlegung des Kalibrierungsproblems in unabhängig kalibrierbare Teilprobleme  $KPMIKRO_{GeoUmeaHet_i} = (\{UM\}, \{EKF_i\}), VK = \{GZF, (Geschäfte, IZF_1)\}$  möglich. Dies war ein Vorteil der Anwendung des Makro-Mikro-Verfahrens bei der Kalibrierung im Vergleich zur Black-Box-Kalibrierung. Für jedes Einkaufsagentenmodell  $EKF_i$  wurde dabei ein eigenes unabhängiges Kalibrierungsproblem  $KPMIKRO_{GeoUmeaHet_i}$  betrachtet. In jedem  $KPMIKRO_{GeoUmeaHet_i}$  wurden die Parameterwerte eines einzelnen  $EKF_i$  auf seine individuelle Zielfunktion hin kalibriert. Als Technik konnte hierzu für jedes  $KPMIKRO_{GeoUmeaHet_i}$  die gleiche wie im Mikro-Schritt von  $KP_{GeoUmeaHom}$  genutzt werden.

Das Kalibrierungsergebnis auf der Mikro-Ebene war ernüchternd. Für die einzelnen Einkaufsagentenmodellfamilien konnten keine Parametereinstellungen gefunden werden, welche die Werte in den zugehörigen  $IVM$  reproduzieren konnten. Auch das resultierende Gesamtverhalten der Simulation ergab kein akzeptables Simulationsverhalten. Die ermittelten  $IVM$  waren nicht für die Kalibrierung auf der Mikro-Ebene geeignet. Damit konnte

der Mikro-Ebenen-Kalibrierungsschritt des Makro-Mikro-Verfahrens nicht erfolgreich abgeschlossen werden und das Makro-Mikro-Verfahren in diesem Fall nicht zur vollständigen Kalibrierung von  $KP_{GeoUmeaHet}$  eingesetzt werden.

### Diskussion des Ergebnisses auf der Mikro-Ebene

Der Grund, dass die Gesamt-Kalibrierung auf der Mikro-Ebene kein valides Mikro-Verhalten erzeugen konnte, ist, dass bereits die konkreten Agentenstrukturen der einzelnen Agenten in den Teilkalibrierungsproblemen auf der Mikro-Ebene nicht so kalibriert werden konnten, dass ihr aggregiertes Verhalten dem der ermittelten zugehörigen *IVM*-Modelle entsprach. Schon in der Fallstudie des homogenen Kalibrierungsproblems konnte beobachtet werden, dass zwar bei Simulation auf der Makro-Ebene von den *IVM* valides, d.h. ideales Verhalten bezüglich *GZF*, erzeugt werden konnte, aber die Modelle auf der Mikro-Ebene nicht so kalibriert werden konnten, dass sie das *IVM*-Verhalten erzeugten. Dies führte bei  $KP_{GeoUmeaHom}$  dazu, dass die Simulation der kalibrierten ABS auf der Mikro-Ebene ein um 0,4 schlechteres Verhalten bezüglich *GZF* lieferte als die Simulation der *IVM* auf der Makro-Ebene. Im nun betrachteten heterogenen Fall konnte schon auf der Ebene der *IVM* nur ein *GZF*-Wert von 0,73 erreicht werden. Wenn nun die Mikro-Modellstrukturen nicht passend zu den neuen *IVM*-Werten kalibriert werden können erhält man mit den ermittelten Parameterwerten auf der Mikro-Ebene ein noch schlechteres Simulationsverhalten.

Wie im homogenen Fall müssen die Ergebnisse der Fallstudie und mögliche Konsequenzen auf eine von zwei Arten interpretiert werden:

- **Die ermittelten *IVM* sind nicht valide:** Es besteht die Möglichkeit, dass die ermittelten *IVM* ungeeignet waren. Ihr *GZF*-Wert auf Makro-Ebene war schlechter als der im homogenen Fall und ihre Validität konnte sonst nur durch Plausibilität bewertet werden. Als Gegenmaßnahme wäre es notwendig weiter nach valideren *IVM* zu suchen, bevor die Kalibrierung auf der Mikro-Ebene durchgeführt werden kann.
- **Die Modellstrukturen auf der Mikro-Ebene sind nicht valide:** Macht man die Annahme, dass die *IVM* valide sind, weil sie auf Makro-Ebene als plausibel betrachtet wurden, dann bleibt nur die Möglichkeit, dass das von den Mikro-Modellstrukturen erzeugte Verhalten deshalb nicht zu den *IVM* passt, weil die Mikro-Modellstrukturen nicht valide sind. Als Gegenmaßnahme wäre es notwendig gewesen entweder pauschal für alle Einkaufsagenten oder möglicherweise für jedes einzelne Agentenmodell die Mikro-Strukturen so anzupassen, dass sie das Verhalten der *IVM* erzeugen können.

#### **15.10.4 Zusammenfassung: Evaluation des Makro-Mikro-Verfahrens anhand von $KP_{GeoUmeaHet}$**

Die Evaluation anhand der Fallstudie  $KP_{GeoUmeaHet}$  hat gezeigt, dass das Makro-Mikro-Verfahrens zwar grundsätzlich auf das Kalibrierungsproblem anwendbar ist und auch Vorteile gegenüber der Black-Box-Kalibrierung haben kann, dass aber in der praktischen Durchführung verschiedene Randbedingungen des Verfahrens nicht erfüllt werden konnten und so keine vollständige heterogene Kalibrierung von  $KP_{GeoUmeaHet}$  mit dem Makro-Mikro-Verfahren erreicht werden konnte.

#### 15.10.4.1 Vorteile des Makro-Mikro-Verfahrens in der Fallstudie

Positiv war, dass die grundsätzliche Anwendbarkeit des Makro-Mikro-Verfahrens auch für  $KP_{\text{GeoUmeaHet}}$  gezeigt werden konnte:

- **Zusammenhang homogener und heterogener Parameter:** Zunächst wurde gezeigt, dass mit dem Makro-Mikro-Verfahren bereits auf der Ebene der  $IVM$ , d.h. der Ebene des idealen Zielverhaltens, geklärt werden kann, dass das Kalibrierungsproblem  $KP_{\text{GeoUmeaHet}}$  so unterbestimmt war, dass kein eindeutiges ideales Verhalten für die Agenten ermittelt werden konnte. Auf der Grundlage dieses Ergebnisses wurde im Iterationsschritt die Ermittlung der  $IVM$  durch zusätzliche Randbedingungen so angepasst, dass für die Kalibrierung auf der Mikro-Ebene vom Experten als valide bewertete heterogene  $IVM$  ermittelt werden konnten.

Etwas Vergleichbares wäre beim direkten Einsatz eines Black-Box-Verfahrens auf der Mikro-Ebene nicht möglich gewesen. Die Kalibrierung wäre hier nur auf Basis der globalen  $GZF$  durchgeführt worden und als Kalibrierungsergebnis hätte man aufgrund der Unterbestimmtheit beliebige Parameterwerte und damit beliebiges Mikro-Verhalten ermittelt. Ohne die Betrachtung der zusätzliche Beobachtungsebene der  $IVM$  wäre es hier nicht möglich gewesen zu bestimmen wie das Mikro-Verhalten angepasst werden müsste, damit ein valideres Simulationsverhalten entsteht.

- **Ermittlung heterogener  $IVM$  auf der Makro-Ebene:** Im zweiten Teil der heterogenen Fallstudie konnte der Makro-Kalibrierungsschritt erfolgreich abgeschlossen werden und heterogene  $IVM$  mit 132-Werten für alle 1000 Einkaufsagentenmodelle auf der Mikro-Ebene ermittelt werden. Dies wäre mit einem Black-Box-Kalibrierungsverfahren in begrenzter Zeit nicht möglich gewesen.
- **Zerlegbarkeit von  $KP_{\text{GeoUmeaHet}}$  auf der Mikro-Ebene:** In der Fallstudie war es möglich, im Kalibrierungsschritt auf der Mikro-Ebene das Gesamtkalibrierungsproblem White-Box in unabhängige Teilprobleme für jeden Agenten zu zerlegen. Hierdurch waren die Parametersuchräume für jedes Teilproblem deutlich kleiner als im Gesamtproblem. Dies ist ein Vorteil gegenüber der Anwendung eines Black-Box-Kalibrierungsverfahrens, das mit der enormen Parameteranzahl nicht hätte sinnvoll umgehen können.

Zwar konnten im konkrete Problem der Fallstudie die Teilprobleme nicht gelöst werden, die Ursache hierfür lag aber in Inkonsistenzen in den verwendeten Geschäftsdaten. Die eigentliche Zerlegbarkeit des Kalibrierungsproblems ist ein Vorteil des Makro-Mikro-Verfahrens.

#### 15.10.4.2 Probleme und Abhängigkeiten des Makro-Mikro-Verfahrens in der Fallstudie

In der Fallstudie zeigten sich aber auch die Abhängigkeiten des Makro-Mikro-Verfahrens und daraus resultierende Probleme bei der Anwendung, die es letztendlich nicht möglich machen den Mikro-Kalibrierungsschritt des Makro-Mikro-Verfahrens für den zweiten Teil von  $KP_{\text{GeoUmeaHet}}$  abzuschließen:

- **Definition der Randbedingungen für das Reverse-Engineering:** Die Ermittlung der  $IVM$  auf der Makro-Ebene hängt stark von den im Reverse-Engineering verwendeten Randbedingungen ab. Zunächst waren die Randbedingungen so gewählt, dass es

keine heterogene Parametereinstellung geben konnte, die valideres Gesamtverhalten erzeugen konnte als die bereits bekannte homogene. Im zweiten Schritt wurden die Randbedingungen angepasst und heterogene *IVM* ermittelt. Dies zeigt wie groß der Einfluss der Randbedingungen im Reverse-Engineering ist und wie viel man bei ihrer Definition richtig oder falsch machen kann. Die Aufstellung der Randbedingungen ist daher selbst ein wichtiges Problem, das bei der Anwendung des Makro-Mikro-Verfahrens nicht unterschätzt werden darf.

- **Abhängigkeit von *IVM* und Mikro-Modellen:** Die Kalibrierbarkeit der Mikro-Modelle hängt stark davon ab, wie gut die *IVM* und die Mikro-Modelle zusammenpassen. Dieser Punkt wurde bereits im homogenen Fall diskutiert, wog aber im zweiten der heterogenen Fallstudie besonders schwer. Die Mikro-Modelle der einzelnen Agenten konnten nicht so kalibriert werden, dass ihr Verhalten dem der zugehörigen *IVM* entsprach.

## 15.11 Zusammenfassung der Evaluation des Makro-Mikro-Verfahrens

In der dreiteiligen Evaluation des Makro-Mikro-Verfahrens wurden zum einen die Anwendbarkeit des Verfahrens auf verschiedene Fallstudien, notwendige Randbedingungen zum Einsatz des Verfahrens und die Effizienz des Kalibrierungsansatzes im Vergleich zu Black-Box-Ansätzen untersucht.

- Es wurde gezeigt, wie die für Black-Box-Kalibrierung beschriebenen Kalibrierungsprobleme auf Makro-Mikro Kalibrierungsprobleme abgebildet werden können. Am künstlichen Beispielmmodell wurden die Voraussetzungen für das Makro-Mikro-Verfahren diskutiert.
- Es zeigte sich, dass die Kalibrierung des Einkaufsmodells mit dem Makro-Mikro-Verfahren mit einer deutlich verringerten Anzahl an Simulationsläufen im Vergleich zu Black-Box-Verfahren durchgeführt werden konnte. Dies konnte sowohl für den Fall homogener als auch für den Fall heterogener Agentenparameter beobachtet werden. Im Allgemeinen hängt die Effizienz des Makro-Mikro-Verfahrens für ein spezifisches Kalibrierungsproblem aber von der bewussten Anwendung des Verfahrens auf der Grundlage von zusätzlichem verfügbarem Kalibrierungswissens auf der Ebene des aggregierten Agentenverhaltens ab.
- Es wurde beobachtet, dass die Kalibrierung mit dem Makro-Mikro-Verfahren zu mindestens genauso validem Simulationsverhalten führte wie die Black-Box-Kalibrierung. Die Analyse der Ergebnisse zeigte auch, dass das Verfahren im Allgemeinen zu besseren Ergebnissen als die Black-Box-Kalibrierung kommen kann. Dies hängt stark davon ab, wie gut *IVM* ermittelt werden können und wie gut die Mikro-Modelle bezüglich der *IVM* kalibriert werden können.
- Es konnte für die Fallstudie gezeigt werden, dass das Makro-Mikro-Verfahren den Einsatz von White-Box-Zerlegungstechniken ermöglicht, was bei Black-Box-Verfahren nicht möglich war. Im Allgemeinen hängt der Einsatz der White-Box-Techniken von



der Komplexität des Mikro-Modellverhaltens und dem verfügbaren Wissen über die Wirkungszusammenhänge im Originalsystem ab.

- Ein weiterer wichtiger beobachteter Punkt war, dass die Randbedingungen, die für das Reverse-Engineering eingesetzt werden entscheidenden Einfluss auf die Generierung der *IVM* haben. Kleine Änderungen an den Randbedingungen führen zu andern *IVM*. Sind die Randbedingungen daher bei der Kalibrierung nicht vollständig klar ist es daher schwierig valide *IVM* zu ermitteln.
- Ein letzter Punkt der beobachtet werden konnte, war, dass es von entscheidender Wichtigkeit ist, Validität der *IVM* genau spezifizieren zu können. Ist dies nicht möglich oder ist man unsicher, wie sich valide *IVM* in der Simulation verhalten sollten, läuft man Gefahr, dass die *IVM* nicht präzise bestimmbar sind und das Makro-Mikro-Verfahren nur schwer einsetzbar ist.

Insgesamt hat die Evaluation gezeigt, dass das Makro-Mikro-Verfahren deutliche Vorteile gegenüber der Black-Box-Kalibrierung haben kann, aber der erfolgreiche Einsatz des Makro-Mikro-Verfahrens stark von der Erfüllung verschiedener Voraussetzungen abhängt. Die Voraussetzungen lassen sich darauf zurückführen, dass der Nutzer möglichst genau auf der Makro-Ebene des aggregierten Agentenverhalten spezifizieren können muss, wie valides Verhalten auf dieser Ebene aussieht und wovon es abhängt. Je weniger Wissen der Nutzer über das Verhalten auf der Makro-Ebene hat desto schwieriger und langwieriger wird der Einsatz des Makro-Mikro-Verfahrens. In einem solchen Fall verliert das Makro-Mikro-Verfahren immer mehr Vorteile, die es gegenüber dem Einsatz von Black-Box-Verfahren haben kann.



# Kapitel 16

## Evaluation: Heterogenitätssuche

In diesem Kapitel wird die in Kapitel 10 beschriebene Technik zur Heterogenitätssuche in den Agentenparametern evaluiert. Die Evaluierung ist in zwei Teile gegliedert:

1. **Untersuchung von Vorteilen der Heterogenitätssuche im Vergleich zur Kalibrierung mit vollständiger Heterogenität in den Parameterwerten.** Die Untersuchung wird an mehreren konstruierten Beispielen durchgeführt, für die jeweils bestmögliche Validität und benötigte minimale Heterogenität in den Parameterwerten der Agenten zum Erreichen dieses Validitätsgrades vorgegeben sind. Anhand dieser Beispiele wird untersucht, bis zu welchem benötigten Validitätsgrad die Heterogenitätssuche im Vergleich zu Kalibrierung mit vollständiger Heterogenität Vorteile hinsichtlich der insgesamt benötigten Anzahl an Simulationsläufen bietet.
2. Im zweiten Teil wird die **Heterogenitätssuche am Beispiel der Fallstudie der Agenten-basierten Einkaufssimulation demonstriert.** Die besondere Schwierigkeit dieses Problems besteht darin, dass die benötigte Ähnlichkeitsfunktion für die Heterogenitätssuche unbekannt ist. Dies würde eine Kalibrierung bei vollständiger Heterogenität notwendig machen, was aber aufgrund der riesigen Anzahl an Parameterwerten nicht möglich ist. Es wird daher anhand der Fallstudie demonstriert, wie die Heterogenitätssuche als Mittel zur Untersuchung verschiedener möglicher Ähnlichkeitsfunktionen eingesetzt werden kann.

### 16.1 Evaluierungsziele

Die Evaluation in diesem Abschnitt wird anhand mehrerer Varianten eines künstlichen Modells mit bekannter benötigter Heterogenität in den Parameterwerten und anhand der Fallstudie der Agenten-basierten Einkaufssimulation durchgeführt.

Die Evaluationsziele dabei sind:

- **Funktionalität des Vorgehens (konstruiertes Modell):**

Zunächst wird die Seite der Funktionalität des Vorgehens betrachtet. Es soll für verschiedene Varianten eines künstliches Evaluationsmodells gezeigt werden, dass jeweils die geringste Heterogenitätsstufe in den Agentenparametern, die zu „validem“ Simulationsverhalten (bezüglich einer gewählten Zielfunktion) führt, identifiziert werden kann. Dies wird für unterschiedliche Problemvarianten untersucht.

- **Verringerung der Anzahl benötigter Simulationsläufe:**

Auf der Seite der Sucheffizienz soll gezeigt werden, dass es Kalibrierungsprobleme gibt, in denen sich der Einsatz der Kalibrierung mit Heterogenitätssuche im Vergleich zu einer reinen Black-Box-Kalibrierung mit vollständig heterogenen oder homogenen Parameterwerten lohnt. Hierzu soll geklärt werden, bis zu welchem benötigten Maß an Heterogenität in den Parameterwerten der in Abschnitt 10 beschriebene Ansatz zur Heterogenitätssuche weniger Simulationsläufe bis zur Identifikation eines Modells mit validem Simulationsverhalten benötigt, als ein einzelner Kalibrierungslauf mit maximaler Heterogenität in den Parametereinstellungen. Wenn zu viele Parameter kalibriert werden müssen, kann die Heterogenitätssuche auch die einzige Möglichkeit sein, die Kalibrierung durchzuführen.

Dabei werden zwei Fälle betrachtet:

- Wie viele Simulationsläufe werden benötigt, um die geeignete Parametereinstellung zu finden (Abbruch der Suche durch Vorwissen über optimalen Wert der Zielfunktion)?
- Wie viele Simulationsläufe werden benötigt, um zu verifizieren, dass man mit einer höheren Heterogenitätsstufe keinen besseren Wert der Zielfunktion erreichen kann?

- **Funktionalität des Vorgehens (Fallstudie):** Evaluationsziele für die Fallstudie Einkaufssimulation: Der Evaluationsteil zur Fallstudie der Einkaufssimulation wird anhand zweier Heterogenitätsprobleme  $HP_{HetEKDist}$  und  $HP_{HetEKZone}$  durchgeführt, deren Ergebnisse jeweils mit dem Kalibrierungsergebnis verglichen werden, das bei der Kalibrierung homogener Parameter erreicht werden konnte (siehe z.B.  $KP_{GeoUmeaHom}$ ).

In den Fallstudien wird untersucht:

- Wie gut der Ansatz der Heterogenitätssuche funktioniert, wenn das Ähnlichkeitskriterium für die Heterogenitätssuche nicht bekannt ist.
- Welchen Einfluss das gewählte Gütemaß für die Kalibrierung auf den Verlauf der Heterogenitätssuche hat.

## 16.2 Künstliches Evaluationsmodell ( $HP_{HetAbs}$ )

In diesem Teilabschnitt wird die Technik der Heterogenitätssuche anhand verschiedener Varianten eines einfachen konstruierten Beispiels mit vorgegebenen optimalen Parametereinstellungen bewertet. Es werden mehrere solche Varianten des Beispielsmodells mit verschiedenen benötigten Heterogenitätseinstellungen in den Parameterwerten der Agenten verwendet. Die Varianten entstehen aus einem Grundmodell. Dabei wird jeweils für das Modell eine andere Zielheterogenitätsverteilung und unterschiedliche benötigte Zielparametereinstellungen für die Agenten gewählt. Im einfachsten Fall bilden alle Agenten eine Gruppe und es müssen homogene Parameterwerte gefunden werden. In komplexeren Beispielen werden für die Agenten unterschiedliche Zielgruppierungen mit unterschiedlichen Parameterwerten festgelegt. Die Heterogenitätssuche muss dann die richtigen hetero-

genen Parametereinstellungen für die (der Suche unbekannt) Zielgruppierungen finden. Das Grundmodell wird im Folgenden beschrieben.

### 16.2.1 Aufbau des konstruierten Evaluierungsmodells

Das konstruierte Grundevaluierungsmodell besteht aus neun in einer Umwelt verteilten Agenten, für die jeweils ein Parameter auf einen bestimmten (dem Kalibrierer unbekannt) Zielwert hin kalibriert werden müssen. Das Heterogenitätsproblem besteht darin, dass bei Start der Kalibrierung nicht bekannt ist, welcher Heterogenitätsgrad in den Parameterwerten tatsächlich benötigt wird. Die räumliche Anordnung der Agenten wird in Abbildung 16.1 dargestellt.

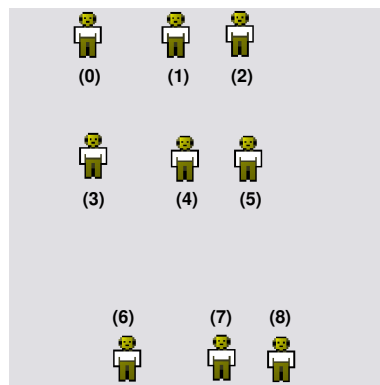


Abbildung 16.1: Künstliches Modell zur Heterogenitätsanalyse.

Im Beispielmodell wird die benötigte Heterogenität der Agenten durch die räumliche Nähe der Agenten zueinander bestimmt. Im homogenen Fall bilden alle Agenten eine gemeinsame Gruppe. In Varianten mit höherer benötigter Heterogenitätsstufe werden die Agenten aufgrund ihrer räumliche Nähe zueinander in die für die Variante vorgegebene Anzahl an Gruppen eingeteilt. In Abbildung 16.1 wurde den einzelnen Agenten jeweils eine Nummer zugeordnet, über die in Tabelle 16.1 referenziert werden kann, wie die Zielgruppierungen für die einzelnen Heterogenitätssuchen gewählt wurden.

Die im künstlichen Modell verwendeten Agentenmodelle sind degeneriert, da ein explizites Agentenverhalten im Beispielmodell nicht vorhanden ist. Ein solches wird für die hier beabsichtigte Untersuchung am künstlichen Modell nicht benötigt. Für die Auswertung des Validierungskriteriums kann angenommen werden, dass die Agenten sich dann valide verhalten, wenn ihre Parameterwerte den festgelegten Zielparameterwerten entsprechen. Auf diese Weise muss zur Auswertung der Validität nur der Vektor der Agentenparameterwerte mit den entsprechenden Zielwerten verglichen werden. Als Konsequenz gibt es auch keine wirklichen Simulationsläufe im künstlichen Modell. Der Ausgangszustand des Modells in der Simulation ist gleich seinem Endzustand.

In Tabelle 16.1 werden die untersuchten Varianten dargestellt. Jede Modellvariante zeichnet sich durch eine unterschiedliche benötigte Heterogenitätsstufe aus, die während der Kalibrierung mit Heterogenitätssuche identifiziert werden muss und deren Parameterwerte entsprechend eingestellt werden müssen. Die Modellvarianten reichen von einer Situation, in

Heterogenitätsstufe der Variante	Vorgegebene Ziel-Gruppen von Agenten mit homogenen Parameterwerten in den jeweiligen Varianten
1	(0,1,2,3,4,5,6,7,8)
2	(0,1,2,3,4,5) (6,7,8)
3	(0,1,2) (3,4,5) (6,7,8)
4	(0,1,2) (3,4,5) (6) (7,8)
5	(0,1,2) (3) (4,5) (6) (7,8)
6	(0) (1,2) (3) (4,5) (6) (7,8)
7	(0) (1) (2) (3) (4,5) (6) (7,8)
8	(0) (1) (2) (3) (4) (5) (6) (7,8)
9	(0) (1) (2) (3) (4) (5) (6) (7) (8)

Tabelle 16.1: Modellvarianten für die Evaluation der Heterogenitätssuche am künstlichen Modell.

der vollkommen homogenen Parameterwerte benötigt werden (in diesem Fall genügt die Kalibrierung eines einzigen Parameterwertes für alle Agenten) über verschiedene Heterogenitätskonstellationen bis zur Notwendigkeit vollkommen heterogener Parameterwerte (jeder Agent benötigt einen individuellen von den anderen Agenten unterschiedlichen Parameterwert), um eine optimale Bewertung der Zielfunktion zu erreichen.

### 16.2.2 Validierungskriterium

Als „Validierungskriterium“ für die einzelnen Modellvarianten wird eine Zielfunktion  $ZF_{Het.Abs}$  eingesetzt, welche die absolute Abweichung der ermittelten Parameterwerte der Agenten von den im der jeweiligen Variante vorgegebenen Zielparаметerwerten misst. Die konkreten Werte der Zielwerte sind für die Untersuchung nebensächlich und wurden frei gewählt. Es wurde aber sichergestellt, dass alle Agenten einer Gruppe die gleichen Zielwerte hatten und Agenten unterschiedlicher Gruppen verschiedene.

### 16.2.3 Evaluierungsbeispiele

Die Evaluation wird zunächst jeweils getrennt in den in Tabelle 16.1 dargestellten Varianten durchgeführt. Tabelle 16.1 zeigt verschiedene Heterogenitätsvarianten, in denen die Agenten des Grundmodells zu Gruppen mit gleichen benötigten Parameterwerten zusammengefasst werden. Die Zuordnung der Agenten zu Gruppen wird aufgrund der Identifikationsnummern aus Abbildung 16.1 durchgeführt. Die erste Spalte der Tabelle zeigt die Benennungen der einzelnen Varianten durch die jeweils benötigten Heterogenitätsstufen. Die zweite Spalte zeigt die in der jeweiligen Variante erforderliche Aufteilung der Agenten in Gruppen (homogene benötigte Parameterwerte innerhalb der Gruppen, heterogene zwischen den Gruppen). Geklammerte Agenten sind in einer Gruppe. Die Zuordnung der Agenten zu den Gruppen der einzelnen Modellvarianten erfolgt nicht willkürlich, sondern entfernungs basiert. Die Agenten wurden beim Entwurf der Varianten mit einem k-Medoids-Cluster-Verfahren aufgrund ihrer Entfernung zueinander in die jeweils für die Variante vorgesehene Anzahl an Gruppen eingeteilt.

### 16.2.4 Evaluierungsziele pro Modellvariante

In jeder Modellvariante besteht das Ziel darin:

- Die insgesamt „beste“ Parametereinstellung zu finden, d.h. diejenige Parametereinstellung, die der vorgegeben Ziel-Einstellung für die jeweilige Beispielvariante entspricht.
- Dabei möglichst wenige Simulationsläufe zu benutzen.
- Als Nebeneffekt der Parametersuche, die in der jeweiligen Variante vorgesehene Heterogenitätseinstellung der Agenten zu identifizieren.

**Insgesamt** wird dann durch den Vergleich aller Ergebnisse aus den Modellvarianten untersucht, bis zu welcher benötigten Heterogenitätsstufe die Heterogenitätssuche hinsichtlich der insgesamt benötigten Simulationsläufe Vorteile bezüglich der Kalibrierung bei vollständiger Heterogenität geboten hat.

### 16.2.5 $HP_{\text{HetAbs}}$ : Problembeschreibung

Die Beschreibung des Kalibrierungsproblems mit Heterogenitätssuche in diesem Abschnitt ist gegeben durch das folgende Heterogenitätsproblem:

$$HP_{\text{HetAbs}} = (((\{UM\}, AMF), VK), (\Omega, W_g))$$

- $AMF = \{IAMF_1, \dots, IAMF_9\}$ : Menge der Familien der Agentenmodelle
- $VK = \{(\sum_{i=0}^9 ZF_i)\}$
- $ZF_i$ : Absolute Abweichung des Parameterwertes von  $Agent_i$  vom in der jeweiligen Variante vorgesehen Ziel-Parameterwert für diesen Agenten.
- $\Omega = Dist(Agent_j, Agent_k)$
- $Dist(Agent_j, Agent_k)$ : Die Entfernung zweier Agenten  $j$  und  $k$  zueinander.
- $W_g = ([1; 9])$

## 16.3 $HP_{\text{HetAbs}}$ : Black-Box-Verfahren

$HP_{\text{HetAbs}}$  kann mit reinen Black-Box-Verfahren angegangen werden. In einem Black-Box-Verfahren, werden die Parameterwerte außerhalb der Simulation geändert und dann in die Simulation eingesetzt.

Hierfür gibt es in  $HP_{\text{HetAbs}}$  intuitiv zwei Möglichkeiten:

- Setze Wertebereich  $W_g = ([1; 1])$ , also homogene Parameterwerte für alle Agenten. In diesem Fall würden allen Agenten in der Black-Box die gleichen Parameterwerte eingesetzt.
- Setze Wertebereich  $W_g = ([9; 9])$ , also heterogene Parameterwerte für alle Agenten. In diesem Fall würden für jeden Agenten in der Black-Box individuelle Parameterwerte kalibriert.

Startpunkt	Temperatur-Start	Temperatur-Ende	Temperatur-abschwächung	Gradienten-schätzung	Schrittweiten-adaption
Zufall	0,005	0,0005	0,9	Ja	Ja

Tabella 16.2: Simulated Annealing Parameter.

Im Rahmen dieser Evaluation ist nur der zweite Fall interessant, da bei der Kalibrierung homogener Parameterwerte nur für die erste Variante eine geeignete Parametereinstellungen gefunden werden kann. Aus diesem Grund muss die Kalibrierung mit Black-Box-Verfahren unabhängig von der jeweiligen Variante immer durch Kalibrierung vollkommenen heterogener Agentenparameter durchgeführt werden.

### 16.3.1 Durchführung der einzelnen Kalibrierungsläufe

Zur Durchführung der Black-Box-Kalibrierungsläufe wurden das im DAVINCI Kalibrierer umgesetzte Simulated Annealing Verfahren eingesetzt. Die Einstellungen des Suchverfahrens können Tabelle 16.2 entnommen werden.

Da das Simulated Annealing Verfahren ein heuristisches Suchverfahren ist und die Parameterkalibrierung jeweils von einer zufälligen Anfangskonfiguration gestartet wurde, wurden sämtliche Kalibrierungsläufe 10 Mal wiederholt und zur Auswertung, die Anzahlen der jeweils benötigten Kalibrierungsläufe gemittelt.

### 16.3.2 $HP_{\text{HetAbs}}$ : Kalibrierungsergebnis Black-Box-Verfahren

Das Black-Box-Suchverfahren war bei der Suche nach vollständig heterogenen Parameterwerten in jeder Problemvariante in der Lage, die optimale Parametereinstellung hinsichtlich des Zielkriteriums zu finden. Entscheidend für den Vergleich mit der Kalibrierung mit Heterogenitätssuche ist, wie viele Simulationsläufe jeweils benötigt werden, um zum Kalibrierungsergebnis zu gelangen. Bei der Black-Box-Kalibrierung vollständig heterogener Parameterwerte wurden in jeder Variante bei 10 Wiederholungen durchschnittlich zwischen 230 und 260 Simulationsläufen zur Ermittlung der vorgegebenen Zielparametereinstellung benötigt. Die ermittelte Parametereinstellung entsprach jeweils genau den vorgegebenen Werten und entsprechend konnten die für die Variante vorgesehene Gruppeneinteilung der Agenten aus der ermittelten Parametereinstellung abgelesen werden. Die Anzahl pro Variante benötigter Simulationsläufe war unabhängig von der konkreten Heterogenitätsstufe der jeweiligen Problemvariante. Über alle Varianten gemittelt wurden durchschnittlich 247 Simulationsläufe zur Ermittlung der gesuchten Parametereinstellung benötigt. Die Detailergebnisse sind in Tabelle 16.3 dargestellt. Dabei ist in der Tabelle „HS“ die Heterogenitätsstufe, „# X“ die benötigte Anzahl an Simulationsläufen in Wiederholung  $X$ , „MW“ der Mittelwert aller Wiederholungen dieser Stufe und „STD“ die Standardabweichung.

## 16.4 $HP_{\text{HetAbs}}$ : Kalibrierung mit Heterogenitätssuche

Zur Durchführung der Kalibrierung mit Heterogenitätssuche wurde das im DAVINCI Kalibrierer umgesetzte Verfahren aus Abschnitt 10 eingesetzt. Innerhalb des Verfahrens zur Heterogenitätssuche wurden sämtliche Parameterkalibrierungen mit dem auch für die Black-



HS	# 1	# 2	# 3	# 4	# 5	# 6	# 7	# 8	# 9	# 10	MW	STD
1	233	212	196	200	212	198	269	236	263	290	230.9	33.29
2	304	266	264	282	223	235	319	217	243	211	256.4	37.05
3	245	300	258	275	209	237	237	296	248	237	254.2	28.57
4	266	255	271	257	269	236	231	240	246	271	254.2	15.16
5	293	231	239	228	271	217	224	225	229	257	241.4	24.49
6	238	295	249	208	250	214	268	228	301	243	249.4	31.06
7	258	305	233	230	225	203	228	263	251	227	242.3	28.29
8	289	270	263	252	220	238	206	222	217	231	240.8	26.79
9	227	256	301	201	267	273	221	269	255	211	248.1	31.79

Tabelle 16.3: Anzahl benötigter Simulationsläufe zur Lösung mit Black-Box-Verfahren.

Box-Suche eingesetzten Simulated Annealing Verfahren durchgeführt, für das die identischen Steuerungsparameterwerte wie im Black-Box-Ansatz gewählt wurden.

## 16.5 $HP_{\text{HetAbs}}$ : Ergebnis der Heterogenitätssuche

Die Kalibrierung mit Heterogenitätssuche war in jedem Problemszenario in der Lage:

- Die in der Variante vorgegebene Zielparametereinstellung zu finden.
- Die Zielparametereinstellung auf der minimalen in der jeweiligen Variante benötigten Heterogenitätsstufe zu identifizieren. Dies bedeutet, dass die Parametereinstellung (außer in Variante 9) nicht erst bei Kalibrierung mit vollständiger Heterogenität (wie beim Black-Box-Verfahren) erfolgreich war, sondern bereits zuvor, wenn die Agenten in die tatsächlich benötigten Gruppen eingeteilt wurden.

### 16.5.1 $HP_{\text{HetAbs}}$ : Evaluationsergebnisse der Heterogenitätssuche

Die Kalibrierung mit Heterogenitätssuche war in der Lage, eine Parametereinstellung zu finden, die zum besten Simulationsverhalten bezüglich des Bewertungskriteriums führte. Entscheidend für den Vergleich der beiden Ansätze ist daher, wie effizient das Kalibrierungsergebnis gefunden werden konnte, d.h. wie viele Simulationsläufe jeweils für die Kalibrierung nötig waren. Die Black-Box-Suche führte nur einen Kalibrierungslauf durch, musste jedoch einen sehr großen Suchraum durchsuchen. Die Kalibrierung mit Heterogenitätssuche führte mehrere Kalibrierungsläufe mit ansteigender Heterogenität in den Parameterwerten durch. Dafür waren jedoch die Suchräume der meisten Teilkalibrierungsläufe deutlich kleiner als bei der Black-Box-Suche. Um bewerten zu können, in welchen Einsatzszenarien sich die Verwendung der Kalibrierung mit Heterogenitätssuche lohnt, muss daher untersucht werden, in welchen Fällen der Aufwand mehrere einfachere Kalibrierungsläufe durchzuführen geringer ist als der für einen komplexeren Kalibrierungslauf.

#### 16.5.1.1 Evaluationsergebnis zur Anzahl benötigter Simulationsläufe

Abbildung 16.2 fasst die Ergebnisse zu den einzelnen Kalibrierungen der Varianten zusammen. Wie beim Black-Box-Verfahren wurden auch für die Heterogenitätssuchen jeweils 10 Wiederholungen durchgeführt. Auf der X-Achse wird die Heterogenitätsstufe der jeweiligen

Problemvariante dargestellt. Auf der Y-Achse der Mittelwert und die Standardabweichung der Anzahl benötigter Simulationsläufe bei Anwendung des in Abschnitt 10 beschriebenen Verfahrens bis zum erfolgreichen Abschluss der Kalibrierung.

Die **horizontale durchgezogene Linie** zeigt die in den jeweiligen Varianten benötigten Mittelwerte und Standardabweichungen der Anzahlen an Simulationsläufen, welche die Black-Box-Kalibrierung vollständig heterogener Parameterwerte benötigte.

Die **gestrichelte Linie** zeigt die Mittelwerte und Standardabweichungen der Anzahl benötigter Simulationsläufe bei Kalibrierung mit Heterogenitätssuche und automatischer Verifikation des Ergebnisses, d.h. bei Verwendung der Monotonieannahme als Abbruchkriterium. In diesem Fall wurde nach der Identifikation der besten Parametereinstellung immer noch ein zusätzlicher Kalibrierungslauf auf der nächsten Heterogenitätsstufe durchgeführt, an Hand dem die optimale Parametereinstellung verifiziert wurde (siehe Abschnitt 10, die Kalibrierung wird erst abgebrochen, wenn auf der nächsthöheren Heterogenitätsstufe keine Verbesserung hinsichtlich  $VK$  erzielt werden kann).

Die **gepunktete Linie** zeigt die Mittelwerte und Standardabweichungen der Anzahl benötigter Simulationsläufe, für den Fall, dass zu Beginn der Suche bereits der optimale Wert von  $VK$  bekannt ist. Es wurde also das schwellwertbasierte Abbruchkriterium verwendet. In diesem Fall kann Kalibrierung mit Heterogenitätssuche nach dem Ende des Kalibrierungslaufes, der das optimale Kalibrierungsergebnis identifiziert hat abgebrochen werden (siehe Abschnitt 10). Dies spart den zusätzlichen Kalibrierungslauf zur Verifikation.

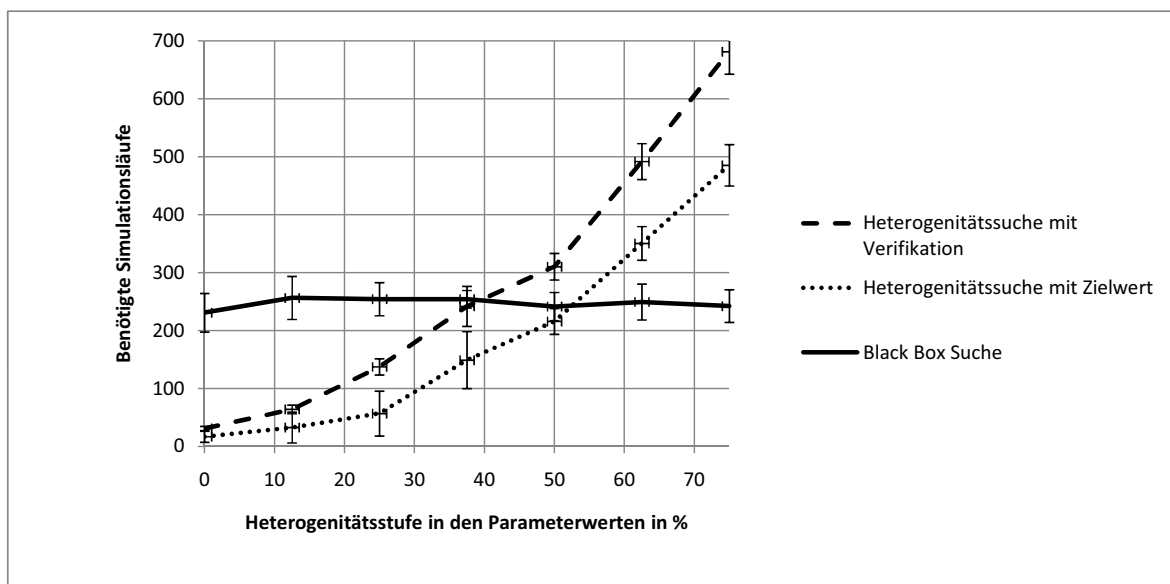


Abbildung 16.2: Evaluation der Kalibrierung mit Heterogenitätssuche am künstlichen Modell.

Abbildung 16.2 zeigt, dass im Evaluationsmodell bis zu einer benötigten Heterogenität in den Parameterwerten von 40 – 50%, d.h. Heterogenitätsstufe 4-5, durch das Kalibrierungsverfahren mit Heterogenitätssuche weniger Simulationsläufe benötigt wurden, als wenn direkt eine Kalibrierung mit maximaler Heterogenitätsstufe durchgeführt würde (im Falle von Heterogenitätssuche mit Verifikation des Ergebnisses knapp über Heterogenitätsstufe 4, ohne Verifikation knapp über Stufe 5).

Der Einsatz der Kalibrierung mit Heterogenitätssuche lohnt sich, wenn entweder eine be-

stimme Heterogenität in den Parameterwerten durchgesetzt werden soll (Heterogenität als Randbedingung) oder vermutet wird, dass eine im Vergleich zur maximal möglichen Heterogenität in den Parameterwerten geringere Heterogenität ( $< 50\%$ ) benötigt wird.

Weitere positive Aspekte des Verfahrens sind:

- die Möglichkeit explizite Randbedingungen an die Struktur der Parameterheterogenität durchsetzen zu können.
- Effizienzgewinne hinsichtlich der benötigten Simulationsläufe für die Parameterkalibrierung.
- Vereinfachung des Kalibrierungsproblems für die Parameterwerte einer sehr großen Agentenmenge.

## 16.6 Heterogenitätssuche in der Einkaufssimulation

Die Evaluation anhand der künstlichen Beispiele demonstriert den Nutzen des Verfahrens, wenn individuelle Agentenparameter kalibriert werden müssen, aber vermutet wird, dass nicht jeder einzelne Agent individuelle Parameterwerte benötigt. Eine Schwierigkeit bei der praktischen Anwendung des Verfahrens ist die Spezifikation einer geeigneten Ähnlichkeitsfunktion durch den Nutzer. Kann keine geeignete Ähnlichkeitsfunktion spezifiziert werden, sind die gebildeten Gruppen von Agenten unter Umständen ungeeignet.

Im Folgenden werden anhand der Fallstudie der Agenten-basierten Einkaufssimulation zwei Kalibrierungsprobleme mit Heterogenitätssuche untersucht.

- Im ersten Kalibrierungsproblem mit Heterogenitätssuche wird eine explizite Gruppenzuordnung auf der Menge der Agenten von den Domänenexperten als Randbedingung an die Kalibrierung vorgegeben und die Kalibrierung auf Basis dieser Randbedingung durchgeführt.
- Im zweiten Beispiel wird durch die Experten keine explizite Gruppierungsvorgabe, sondern lediglich ein Gruppierungskriterium und ein Wertebereich für mögliche Anzahlen an Agentengruppen vorgegeben. Dann wird mit Hilfe der in dieser Arbeit entwickelten Kalibrierung mit Heterogenitätssuche nach einer geeigneten Heterogenitätsstufe mit zugehöriger Parametereinstellung gesucht.

Ein Effizienzvergleich mit einer Kalibrierung bei vollständiger Parameterheterogenität wird aufgrund des zu großen Parametersuchraums nicht durchgeführt. Anhand der Fallstudie wird gezeigt, wie verschiedene Ähnlichkeitsfunktionen untersucht werden können und wie die Heterogenitätssuche für eine konkrete Ähnlichkeitsfunktion funktioniert. Am Ende der beiden Fallstudien werden die Evaluationsergebnisse zusammengefasst.

## 16.7 $HP_{EK3\text{Gruppen}}$ : Problembeschreibung

Im ersten betrachteten Teilproblem wird eine Variante von  $KP_{GeoUmeaHom}$  betrachtet, die um eine **von den Domänenexperten fest vorgegebene Heterogenitätsrandbedingung** erweitert wurde.

### 16.7.1 $HP_{EK3Gruppen}$ : Vorgegebene Randbedingung

Die in dieser Fallstudie verwendete Modellstruktur entspricht der aus  $KP_{GeoUmeaHom}$ . Die 1000 simulierten Einkaufsagenten nehmen die charakteristischen Merkmale Preis, Sortiment, Qualität und Distanz wahr. Preis, Sortiment und Qualität werden über logistische Funktionen wahrgenommen, deren Parameter kalibriert werden müssen (d.h. neun zu kalibrierende Parameterwerte  $P_{AGF}$ ). Die von den Domänenexperten vorgegebene Randbedingung an die Heterogenität der Agentenparameter teilte die Einkaufsagenten in drei Gruppen:

- $AGF_{Stadt}$ : Simulierte Agenten „wohnen“ in der größten Stadt der Region, Umea.
- $AGF_{Umland}$ : Simulierte Agenten „wohnen“ in den Randbezirken der Stadt Umea.
- $AGF_{Hinterland}$ : Simulierte Agenten „wohnen“ außerhalb der Stadt Umea.

Abbildung 16.3 zeigt eine Visualisierung einer möglichen Gruppeneinteilung.

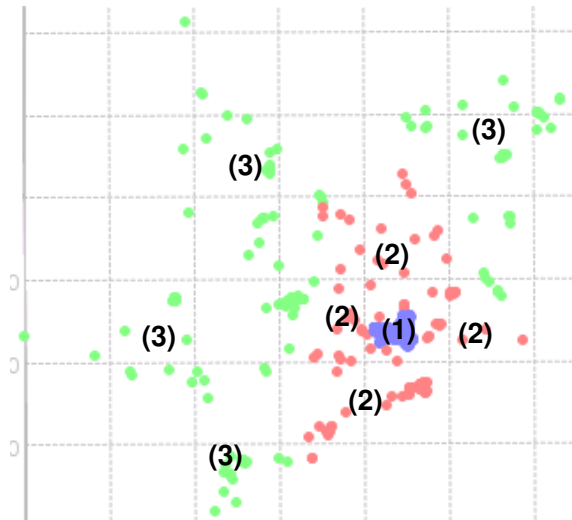


Abbildung 16.3: Von Domänenexperten vorgegebene Gruppierung der Einkaufsagenten in drei Gruppen: (1) Umea Stadt, (2) Umea Umland und (3) Hinterland.

### 16.7.2 $HP_{EK3Gruppen}$ : Problemformalisierung

$KP_{EK3Gruppen}$  wird formalisiert als:

$$HP_{EK3Gruppen} = (((\{UM\}), AMF, VK), (\Omega, W_g))$$

- $AMF = \{IAMF_0, \dots, IAMF_i, \dots, IAMF_{999}\} (i \in \mathbb{N})$ : Die Familien aller individuellen Agentenmodelle.
- $VK = \{GZF\}$ : Das unquadratische Gütemaß aus Abschnitt 7.5.

- $IAMF_i = (S_{AG}, P_{IAMF_i})$  mit  $S_{AG}$ : Die Struktur  $S_{AG}$  ist die gleiche wie in  $KP_{GeoUmeaHom}$  und  $P_{IAMF_i} = \langle P_{Preis_{G1}}, \dots, P_{Qualität_{G3}} \rangle$ . Es werden 27 Parameterwerte von Parametern logistischer Wahrnehmungsfunktionen kalibriert. 3 Parameter pro Wahrnehmungsfunktion für Preis, Sortiment und Qualität und dies jeweils für 3 Gruppen  $G1$ ,  $G2$  und  $G3$ .
- $\Omega$ : die  $IAMF_i$  werden von den Experten fest einer von drei Gruppen  $AGF_{Stadt}$ ,  $AGF_{Umland}$ ,  $AGF_{Hinterland}$  zugeordnet. Daher ist die Ähnlichkeitsfunktion in diesem Fall sehr degeneriert. Sie gibt für zwei Agenten 0 zurück, wenn sie zur gleichen Gruppe gehören und 1 sonst.
- $W_g = ([3; 3])$ .

## 16.8 $HP_{EK3Gruppen}$ : Kalibrierung mit Heterogenitätssuche

Die Kalibrierung von  $HP_{EK3Gruppen}$  wurde mit dem DAVINCI-Kalibrierer durchgeführt. Als Suchverfahren wurde eine Kombination aus Evolutionärer Strategie (Grobsuche) und Simulated Annealing (Feinsuche) eingesetzt.

Auch die wesentlichen Bestandteile der Deklarationen für den DAVINCI-Kalibrierer konnten von der Deklaration der Black-Box-Suche für  $KP_{GeoUmeaHom}$  übernommen werden. Insbesondere ist keine Änderung der in  $KP_{GeoUmeaHom}$  durchgeführten Spezifikation von Wertebereichen oder die Deklaration zusätzlicher zu kalibrierender Parameter notwendig.

**Bei der Deklaration des Kalibrierungsproblems musste vom Nutzer zusätzlich:**

- Eine degenerierte Ähnlichkeitsfunktion für die Agenten deklariert werden. In diesem Fall prüft die Funktion einfach, ob die Agenten zu selben Gruppierung gehören.
- Spezifiziert werden, dass Parameter für drei  $AGF$  gesucht werden sollen.

## 16.9 $HP_{EK3Gruppen}$ : Kalibrierungsergebnis mit Heterogenitätssuche

Bei der Kalibrierung auf der Basis des unquadratischen Gütemaßes konnte kaum Verbesserung der Validität des Simulationsverhaltens im Vergleich zum homogenen Fall festgestellt werden. Ein möglicher Grund ist die besondere Struktur des unquadratischen Gütemaßes, das speziell die Optimierung der Umsatzreproduktion von Geschäften mit großem Umsatz forciert. Da die meisten großen Geschäfte in der Stadt Umea liegen, kann vermutet werden, dass die Heterogenitätsrandbedingung zu  $HP_{EK3Gruppen}$  keinen großen Einfluss auf die Kalibrierung der Parameterwerte hat. Ein anderer möglicher Grund ist, dass die getroffene Gruppierung der Agenten ungeeignet war.

Tabelle 16.4 zeigt die Ergebnisse der Kalibrierung im Vergleich zu den bereits bekannten Ergebnissen der Black-Box-Kalibrierung von  $KP_{GeoUmeaHom}$  mit homogenen Parameterwerten. Da in der Heterogenitätssuche das gleiche Black-Box-Suchverfahren wie in der Fallstudie  $KP_{GeoUmeaHom}$  verwendet wird, wird in der Tabelle zum Vergleich das beste ermittelte Ergebnis mit dem Black-Box-Algorithmus und nicht das Ergebnis des Makro-Mikro-Verfahrens angegeben.

Heterogenität in den Parameterwerten	Unquadratisches Gütemaß
Eine homogene Gruppe	0, 58
Drei Gruppen gemäß $KP_{EK3Gruppen}$	0, 605

Tabelle 16.4: Evaluationsergebnisse zur Heterogenitätssuche mit drei Gruppen.

## 16.10 $HP_{EKDistGruppen}$ : Problembeschreibung

Im zweiten betrachteten Teilproblem wird eine Variante von  $KP_{GeoUmeaHom}$  untersucht, in der nicht wie in  $HP_{EK3Gruppen}$  vom Domänenexperten eine feste Gruppierungsvorgabe gemacht, sondern lediglich ein Ähnlichkeitskriterium für die Einkaufsagenten vorgegeben wurde: Die Entfernung der einzelnen Agenten zueinander. Zusätzlich wurde aufgrund der räumlichen Struktur der simulierten Region die Annahme gemacht, dass die Zahl möglicher Heterogenitätsgruppierungen der Agenten zwischen einer und acht Gruppen liegt.

### 16.10.1 $HP_{EKDistGruppen}$ : Formalisierung

$HP_{EKDistGruppen}$  wird formalisiert als:

$$HP_{EKDistGruppen} = (((\{UM\}, AMF), VK), (\Omega, W_g))$$

- $AMF = \{IAMF_0, \dots, IAMF_{999}\}$
- $VK$ : Das unquadratische Gütemaß aus Abschnitt 7.5.
- $\Omega$ : Zur Gruppierung der IAMF wird von den Experten ein Nähe-basiertes Ähnlichkeitskriterium vorgegeben:  

$$\Omega(Agent_1, Agent_2) = Distanz(Agent_1, Agent_2).$$
- $W_g = ([1; 8])$ .

Je nach gewählter Gruppierungsanzahl in der Heterogenitätsvorgabe liegt die Anzahl zu kalibrierender Parameter zwischen 9 und 72. Abbildung 16.4 zeigt farblich hervorgehoben die Ergebnisse einer Clusteranalyse für die verschiedenen möglichen Gruppenanzahlen auf der Basis des gewählten Ähnlichkeitskriteriums. Die Abbildung wurde für diese Arbeit zum einfacheren Verständnis mit zusätzlichen Beschriftungen versehen. Bei der Clusteranalyse wurden zunächst zufällig so viele Einkaufsagenten ausgewählt wie Gruppen gebildet werden sollten. Diese Agenten bildeten die „Repräsentanten“ der Gruppen. Dann wurden die übrigen Einkaufsagenten auf der Basis des Ähnlichkeitskriteriums jeweils der Gruppe des zu ihnen ähnlichsten Repräsentanten zugeordnet. Waren alle Agenten einer Gruppe zugeordnet so wurde für jede Gruppe der „mittelste“ Agent bezüglich des Wohnortes als neuer Repräsentant der Gruppe ausgewählt. Dann wurden die übrigen Einkaufsagenten wieder neu Gruppen zugeordnet. Dies wurde so lange iteriert, bis sich die Repräsentanten der Gruppen nicht mehr änderten. Die resultierenden Gruppen waren die gesuchten Cluster.

## 16.11 $HP_{EKDistGruppen}$ : Heterogenitätssuche

Die Kalibrierung von  $HP_{EKDistGruppen}$  wurde mit dem DAVINCI-Kalibrierer durchgeführt. Als Suchverfahren wurde eine Kombination aus Evolutionärer Strategie (Grobsuche) und

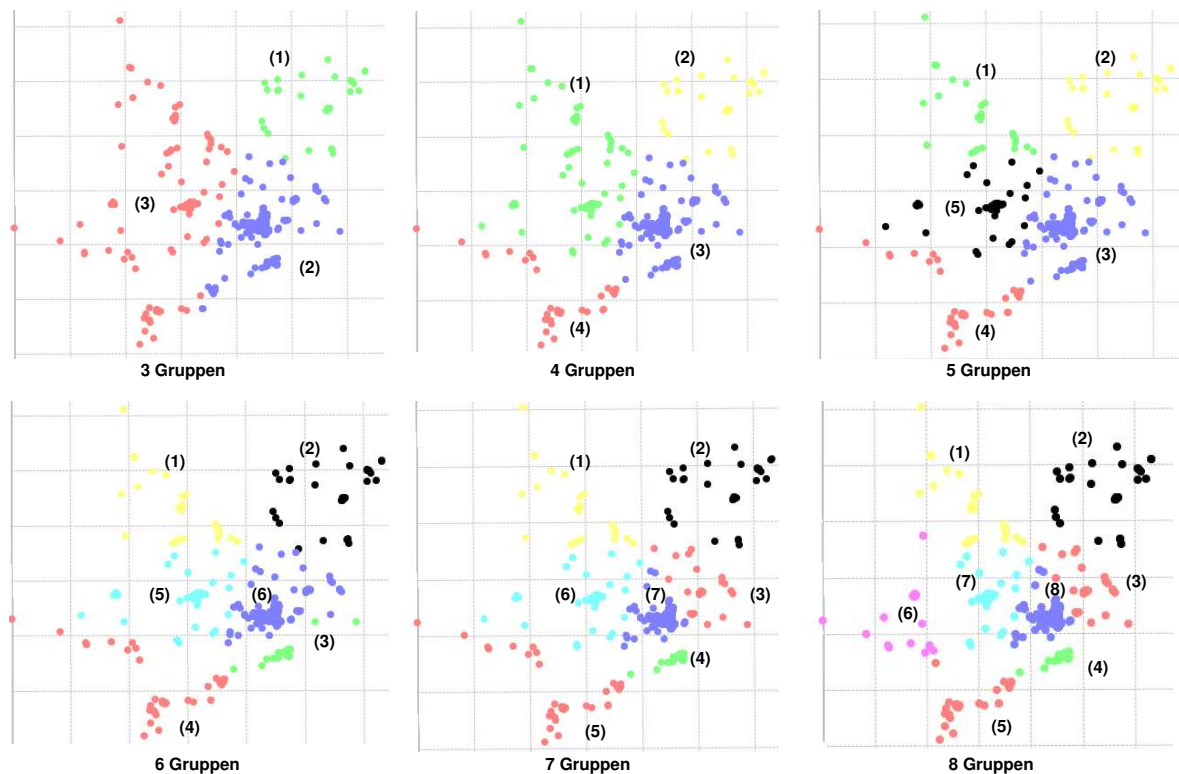


Abbildung 16.4: Verschiedene mögliche Gruppierungen der Agentenmodelle nach Nähe der Agenten zueinander.

Simulated Annealing (Feinsuche) eingesetzt. Da hier zum Teil sehr viele Parameterwerte kalibriert werden mussten, wurde den einzelnen Teilkalibrierungsläufen jeweils 5000 Simulationsläufe zur Verfügung gestellt.

### 16.11.1 Kalibrierungsergebnis mit unquadratischem Gütemaß

Zunächst wurde die Kalibrierung mit Heterogenitätssuche auf Basis des unquadratischen Gütemaßes durchgeführt  $ZF_{unquadrat}$ . Initial wurde dabei die Kalibrierung mit Heterogenitätssuche und aktiviertem automatischen Abbruchkriterium der Monotonieannahme durchgeführt. Die Heterogenitätssuche brach hier bereits nach der Kalibrierung zweier Parametergruppen ab. Dies zeigte, dass die Monotonieannahme des Abbruchkriteriums für die Fallstudie unter den gegebenen Randbedingungen nicht gültig war.

Das Experiment wurde dann wiederholt, wobei das automatische Abbruchkriterium deaktiviert wurde, d.h. für alle acht Heterogenitätsstufen wurde ein eigener Kalibrierungslauf durchgeführt. Tabelle 16.5 zeigt die Ergebnisse der Kalibrierung. Der Tabelle kann entnommen werden, dass es für unterschiedliche Heterogenitätsstufen praktisch keine Veränderungen im erreichten Validitätsgrad bezüglich  $ZF_{unquadrat}$  gab. Weiter kann der Tabelle entnommen werden, dass während der Kalibrierung der ersten sieben Gruppen jeweils ein Validitätsgrad von etwa 0,58 erreicht wurde. Erst für acht Gruppen stieg der erreichte Validitätsgrad auf ca. 0,62.

Hierfür gibt es verschiedene Erklärungen und sich daraus ergebende Gegenmaßnahmen:

Heterogenität in den Parameterwerten	$ZF_{unquadrat}$ -Wert
Eine homogene Gruppe	0,58
Zwei Gruppen	0,584
Drei Gruppen	0,581
Vier Gruppen	0,587
Fünf Gruppen	0,586
Sechs Gruppen	0,583
Sieben Gruppen	0,583
Acht Gruppen	0,617

Tabelle 16.5: Ergebnisse der Kalibrierung mit Heterogenitätssuche zu  $HP_{EK DistGruppen}$  bezüglich  $ZF_{unquadrat}$

- **Ungeeignetes Ähnlichkeitskriterium:** War das Ähnlichkeitskriterium falsch gewählt konnte Heterogenitätssuche kein Ergebnis finden. Das vom Fachexperten vorgeschlagene Ähnlichkeitsmaß gruppierte die Agenten nach räumlicher Nähe (Luftlinie), die aber die räumliche Struktur der unterschiedlichen Geschäfte und ihre tatsächliche Erreichbarkeit für die Einkäufer im Originalsystem nicht widerspiegelte. So ist z.B. die resultierende Gruppeneinteilung für drei Gruppen in Abbildung 16.3 und 16.4 deutlich unterschiedlich. Ein alternatives Ähnlichkeitsmaß könnte z.B. Bevölkerungsdichte am Wohnort sein. Entsprechende Daten waren aber in der Fallstudie nicht verfügbar.

- **Ungeeignete Gruppenzahlen:**

Unter Umständen konnte die Menge an verwendeten Gruppenanzahlen zu gering sein, so dass keine bessere Parametereinstellung als die homogener Parameterwerte gefunden werden konnte. In diesem Fall musste für die automatische Heterogenitätssuche mit einer höheren Zahl an Gruppen (also z.B. 9 Gruppen als Startwert) begonnen und größere Sprünge zwischen den einzelnen Gruppen (z.B. 1, 8, 16, 32, ...) verwendet werden.

- **Ungeeignetes Validierungskriterium:**

Die dritte Möglichkeit bestand darin, dass das verwendete Validierungskriterium  $ZF_{unquadrat}$  ungeeignet war. Wie beschrieben (siehe z.B. Kapitel 7.5), gehen in  $ZF_{unquadrat}$  Geschäfte mit großem Umsatz besonders stark ein und Geschäfte mit geringem Umsatz kaum. Möglicherweise hatte dies einen Einfluss auf das Ergebnis der Heterogenitätssuche.

Da keine Daten zur Verbesserung des Ähnlichkeitskriteriums zu Verfügung standen wurde unter der Annahme des ungeeigneten Validierungskriteriums, die automatische Heterogenitätssuche mit Abbruchkriterium bezüglich des VK  $ZF_{echt}$  (siehe Kapitel 7.5), in das die Verschätzung aller Geschäfte gleichmäßig eingeht, wiederholt.

### 16.11.2 Kalibrierungsergebnis mit echtem unquadratischem Gütemaß

Tabelle 16.6 zeigt die Ergebnisse der Kalibrierung. Für  $ZF_{echt}$  konnte während der Kalibrierung bei jeder Erhöhung der Gruppenanzahl ein Anstieg der Validität des Simulationsverhaltens nach der Kalibrierung festgestellt werden. D.h. für  $ZF_{echt}$  galt die Monotonie-Annahme des Abbruchkriteriums. Der höchste Validitätswert bezüglich  $ZF_{echt}$  wurde für acht Gruppen erreicht. Wie für  $ZF_{unquadrat}$  kann auch für  $ZF_{echt}$  der größte



Heterogenität in den Parameterwerten	$ZF_{echt}$ -Wert
Eine homogene Gruppe	0,477
Zwei Gruppen	0,492
Drei Gruppen	0,495
Vier Gruppen	0,504
Fünf Gruppen	0,506
Sechs Gruppen	0,511
Sieben Gruppen	0,513
Acht Gruppen	0,539

Tabelle 16.6: Ergebnisse der Kalibrierung mit Heterogenitätssuche zu  $HP_{EKDistGruppen}$  bezüglich  $ZF_{echt}$

Preis <sub>1</sub>	Preis <sub>2</sub>	Preis <sub>3</sub>	Sorti. <sub>1</sub>	Sorti. <sub>2</sub>	Sorti. <sub>3</sub>	Qualität <sub>1</sub>	Qualität <sub>2</sub>	Qualität <sub>3</sub>
3	3	1,52	2	2,727	3	1,52	0,001	2,72

Tabelle 16.7: Ergebnisparameterwerte für homogene Parameterwerte bezüglich des  $ZF_{echt}$ -Wertes 0,477.

„Validitätssprung“ beim Übergang von sieben auf acht Gruppen festgestellt werden. Die Kalibrierung mit Heterogenitätssuche und Abbruchkriterium lieferte in diesem Fall die Heterogenitätseinstellung „acht Gruppen“ mit einer Bewertung von 0,539 als Ergebnis. Die Tabellen 16.7 und 16.8 zeigen die bezüglich  $ZF_{echt}$  kalibrierten Parametereinstellungen für homogene Parameterwerte und acht Parametergruppen (siehe Abbildung 16.4).

In der Fallstudie mit  $ZF_{echt}$  zeigte sich, dass das Abbruchkriterium der Monotonieannahme für die Heterogenitätssuche für die Fallstudie funktionierte. Die beiden Ergebnisse für die Fallstudien mit  $ZF_{echt}$  und  $ZF_{unquadrat}$  zeigen, dass wenn das Monotonie-Abbruchkriterium verwendet wird trotzdem zur Überprüfung des Ergebnisses noch Kalibrierungen mit höheren Heterogenitätsstufen durchgeführt werden sollten.

Bezüglich des Vergleichs der Gütemaße  $ZF_{echt}$  und  $ZF_{unquadrat}$  konnten keine zusätzlichen Erkenntnisse gewonnen werden, da die Verbesserungen bei der Kalibrierung mit  $ZF_{echt}$  in höheren Heterogenitätsstufen ebenfalls nur gering waren.

## 16.12 $HP_{EK3Gruppen}$ und $HP_{EKDistGruppen}$ : Evaluationsergebnisse

Dieser Abschnitt fasst die Ergebnisse der Evaluation der Heterogenitätssuche anhand zweier Kalibrierungsprobleme der Fallstudie der Agenten-basierten Einkaufssimulation mit Heterogenitätsrandbedingung zusammen:

- In  $HP_{HetAbs}$  wurde an einem konstruierten Beispiel die theoretische Funktionalität der Kalibrierung mit Heterogenitätssuche demonstriert. Weiter wurde am künstlichen Bei-

Gruppe	Preis <sub>1</sub>	Preis <sub>2</sub>	Preis <sub>3</sub>	Sorti. <sub>1</sub>	Sorti. <sub>2</sub>	Sorti. <sub>3</sub>	Qualität <sub>1</sub>	Qualität <sub>2</sub>	Qualität <sub>3</sub>
1	2,89	3	1,347	0,427	1,054	0,101	0,279	1,684	2,416
2	2,585	2,883	2,1645	0,381	0,111	0,668	0,081	0,313	2,93
3	1,862	1,106	2,98	2,868	1,809	2,858	0,902	0,733	1,588
4	1,8612	1,106	2,98	2,868	1,809	2,858	2,115	2,727	0,014
5	0,637	2,199	2,92	0,943	2,935	2,551	2,91	2,389	1,865
6	1,929	2,603	0,786	0,126	0,979	0,807	0,051	2,858	3
7	2,72	3	1,67	0,204	1,583	0,415	0,196	2,447	1,677
8	0,106	1,376	1,374	2,93	3	3	0,001	0,929	0,001

Tabelle 16.8: Ergebnisparameterwerte für 8 Gruppen von Einkaufsagenten bezüglich des  $ZF_{echt}$ -Wertes 0,539.

spiel gezeigt, dass die Kalibrierung mit Heterogenitätssuche Vorteile hinsichtlich der zur Kalibrierung benötigten Simulationsläufe (= Gesamtdauer der Kalibrierung) bieten kann, wenn weniger als 50% Heterogenität in den Parameterwerten benötigt wird.

- In *HP<sub>EK3Gruppen</sub>* wurde demonstriert, dass die Kalibrierung von DAVINCI leicht genutzt werden kann, um explizite Randbedingungen an die Heterogenität der Parameterwerte der Agenten während der Kalibrierung einzubringen.
- In *HP<sub>EKDistGruppen</sub>* wurde die automatische Kalibrierung mit Heterogenitätssuche an einer realen Fallstudie durchgeführt. Es wurde beschrieben und gezeigt, welche Auswirkungen Fehler hinsichtlich verwendetem Ähnlichkeits- und Validierungskriterien auf die Heterogenitätssuche haben und welche Gegenmaßnahmen im konkreten Fall ergriffen werden können.

## Kapitel 17

# Evaluation: Robuste Kalibrierung

In diesem Abschnitt wird die in Abschnitt 9.3 beschriebene robuste Kalibrierung anhand der Fallstudie der Agenten-basierten Einkaufssimulation evaluiert. Das Ziel der robusten Kalibrierung besteht darin, automatisch einzelne besonders problematische Validierungskriterien (d.h. im Vergleich zu den übrigen VK falsche oder stark verrauschte Kriterien) für das Verhalten der Agenten zu identifizieren und diejenigen Modellteile, deren Kalibrierung von den problematischen Validierungskriterien betroffen ist, durch ihre idealen *IVM* zu ersetzen. Zur Durchführung der Evaluierung soll in der Fallstudie untersucht werden, ob es aufgrund eines besonders hohen Verrauschungsgrades einzelner zu Grunde liegender Daten, nur schwer möglich ist, valides Simulationsverhalten zu erzeugen. Es wird unterstellt, dass der in  $KP_{GeoUmeaHom}$  erreichte Validitätsgrad von 0,598 bezüglich des unquadratischen Gütemaßes nicht, wie in den bisherigen Ansätzen vermutet, in fehlerhaften Modellstrukturen, fehlender Heterogenität, ungeeignet gewählten Parameterwerten oder allgemeinem gleichmäßigen Datenrauschen begründet ist. Stattdessen wird angenommen, dass einzelne aus den Originaldaten ermittelte Zusammenhänge aus Geschäftswahrnehmung und Geschäftsumsatz für das Agentenverhalten besonders stark verfälscht sind und so das Gesamtverhalten des Modells verzerrt wird.

### 17.1 Evaluierungsziele

Die Evaluierungsziele in diesem Abschnitt sind, die Anwendbarkeit der robusten Suche zu untersuchen. Es soll gezeigt werden, dass die robuste Kalibrierung dazu genutzt werden kann, fehlerhafte *VK* zu identifizieren, die dazu führen, dass bestimmte Modellteile nach der Kalibrierung falsches Verhalten aufweisen. Weiter soll gezeigt werden, dass die robuste Suche dazu genutzt werden kann, die betroffenen Modellteile so geeignet zu ersetzen, dass der negative Einfluss der fehlerhaften *VK* kompensiert wird.

### 17.2 Problembeschreibung robuste Kalibrierung der Einkaufssimulation

Wir betrachten die Fallstudie des Einkaufsmodells. Es wird angenommen, dass die charakteristischen Eigenschaften einiger weniger Geschäfte falsch ermittelt wurden, und dass dies

das simulierte Einkaufsverhalten der Agenten so stark verzerrt, dass kein Simulationsverhalten mit einer höheren Validitätsbewertung als 0,598 erzeugt werden kann. Die Kalibrierung des Modells unter einer solchen Randbedingung steht vor folgenden Problemen:

- Die stark verrauschten Geschäfte sind unbekannt, müssen also zunächst identifiziert werden.
- Identifizierte stark verrauschte Geschäfte können nicht einfach aus dem Modell entfernt werden, da sonst das Gesamtverhalten der Simulation verzerrt wird.

Als Kalibrierungsproblem wird das Problem  $KP_{GeoUmeaHom}^{Robust}$  verwendet, das eine Variante von  $KP_{GeoUmeaHom}$  ist. Der Unterschied zwischen  $KP_{GeoUmeaHom}^{Robust}$  und  $KP_{GeoUmeaHom}$  liegt in der Hinterfragung bestimmter Validierungskriterien, auf deren Grundlage in  $KP_{GeoUmeaHom}$  bestimmte Teile der Modellstrukturen umgesetzt wurden.

Es wird nun vermutet, dass die Daten zu den charakteristischen Merkmalen mancher Geschäfte falsch sind. Als Folge können aus ihnen in der Simulation keine Agentenbewertungen erzeugt werden, die zu einer ausreichend genauen Reproduktion der Geschäftsumsätze führen (siehe Kapitel 9.3). Um diese Datengrundlage zu hinterfragen werden die charakteristischen Geschäftsmerkmale Sortimentsgröße, Preis, Qualität, etc. der einzelnen Geschäfte als zu überprüfende Randbedingungen an das Modellverhalten betrachtet („Das Geschäft X muss bestimmte charakteristische Merkmale erfüllen“). Damit ergibt sich das folgende Kalibrierungsproblem:

$$KP_{GeoUmeaHom}^{Robust} = ((\{UM\}, AMF), VK)$$

- $VK = \{GZF\} \cup RB$
- $RB = SRB \cup URB$
- $URB = \{RB_1, \dots, RB_i, \dots, RB_{132}\}$  mit  $i \in \mathbb{N}$
- $SRB$ : Menge an Randbedingungen, die beschreiben, wo im Modell die Wohnorte der Einkaufsagenten und die Geschäftsstandorte liegen. Diese werden nicht hinterfragt, da sie aus realen Karten extrahiert wurden.
- $URB$ : Menge an Randbedingungen, die beschreiben, welche Preis, Sortiments und Qualitätsstufen die einzelnen Geschäfte haben. Diese  $RB$  gelten als unsicher, da unklar ist, ob die Einkäufer im Originalsystem die Werte genauso wahrnehmen, wie sie aus der Statistik entnommen werden können, bzw. wie die Werte durch die Domänenexperten subjektiv klassifiziert wurden.
- $RB_i$ : Eine Randbedingung, welche für das  $i$ -te Geschäft festlegt, welche Werte Preis-, Sortiments- und Qualitätsstufen dieses Geschäftes haben müssen.

Zur Lösung von  $KP_{GeoUmeaHom}^{Robust}$  wird nach einem kalibrierten Modell gesucht, dass:

- Möglichst valides Simulationsverhalten aufweist.
- In dem möglichst viele der  $RB$  erhalten bleiben, d.h. in dem die Agenten ihre Geschäftsbewertungen soweit wie möglich auf der Grundlage der von den  $RB$  vorgegebenen Geschäftsmerkmalen treffen.

## 17.3 Lösung mit robuster Kalibrierung

Zur Lösung dieser Probleme wird die Einkaufssimulation mit Hilfe des in Abschnitt 9.3 beschriebenen robusten Kalibrierungsverfahrens des DAVINCI Kalibrierers kalibriert. Bei robuster Kalibrierung auf der Mikro-Ebene wird ein erweitertes Kalibrierungsproblem  $KPMIKRO^{Robust}$  betrachtet. Zur Lösung von  $KPMIKRO^{Robust}$  müssen diejenigen Randbedingungen  $RB_i \in URB$  gefunden werden, welche die Validität des Simulationsverhaltens negativ beeinflussen.

### 17.3.1 Realisierung der Filterung eines Elementes $RB_i$

In Kapitel 9.3 wurde beschrieben, dass während einer robusten Kalibrierung ausgefilterte Agenten durch ihre entsprechenden  $IVM$  ersetzt werden. In diesem Evaluationsbeispiel müssen nicht ganze Einkaufsagenten, sondern nur Teile ihrer Bewertungen einzelner Geschäfte ausgefiltert werden. Es werde beispielsweise die Randbedingung  $RB_X$ , d.h. die Daten zu den charakteristischen Merkmalen eines Geschäftes  $X$  angezweifelt.  $RB_X$  soll ausgefiltert werden. Dann dürfen die Agenten das Geschäft  $X$  nicht mehr aufgrund der von  $RB_X$  vorgegebenen Merkmalswerte bewerten, sondern müssen dieses Geschäft so attraktiv finden, dass ausreichend valides Gesamtverhalten im Rahmen des restlichen Modells entstehen kann. Die Geschäfte, deren zugehörige  $RB$  nicht ausgefiltert wurden, können weiterhin aufgrund ihrer Merkmale bewertet werden.

Wird also während der robusten Kalibrierung eine Randbedingung  $RB_i$  aus  $URB$  ausgefiltert, muss das in der Simulation so umgesetzt werden, dass die Einkaufsagenten die betroffenen Geschäfte nicht mehr gemäß der charakteristischen Geschäftsmerkmale in  $RB_i$ , sondern gemäß des jeweiligen Wertes des Geschäftes aus  $IVM_{AG}$  wahrnehmen (siehe Kapitel 9.3). Auf diese Weise werden die Geschäfte mit „ausgefilterten Geschäftseigenschaften“ von den Agenten valide gemäß dem jeweiligen Wert der idealen aggregierten Verhaltensmodelle wahrgenommen.

### 17.3.2 Auswahl der Menge zu in der robusten Kalibrierung zu filternder $RB$

Das Ziel der robusten Kalibrierung besteht darin, besonders stark verrauschte  $RB_i$  zu identifizieren und das Simulationsverhalten dadurch zu verbessern, dass auf diesen  $RB_i$  basierende Modellteile geeignet ersetzt werden. Da das Ergebnismodell immer noch auf einer möglichst großen Menge der ursprünglichen  $RB$  basieren soll, sollen nur die wirklich problematischen  $RB_i$  ausgefiltert werden. Die Menge ausgefilterter  $RB_i$  soll also möglichst gering sein.

Ist die Menge potentiell auszufilternder  $RB_i$  überschaubar, dann können durch systematische Tests alle möglichen Teilmengen von  $URB$  ausgefiltert und es kann geprüft werden, ob die  $RB_i$  in diesen Teilmengen wirklich fehlerhafte Geschäftsmerkmale vorgeben. Eine große Menge potentiell auszufilternder Randbedingungen und eine unklare Anzahl fehlerhafter  $RB_i$  macht es aber unmöglich systematisch alle möglichen Teilmengen von  $URB$  auszufiltern und den Einfluss auf die Validität des Simulationsverhaltens zu überprüfen.

Das in der robusten Suche verwendete RANSAC-Verfahren ist für den Fall entworfen worden, dass ein systematisches Testen nicht möglich ist und zufallsbasiert Teilmengen der  $URB$  ausgefiltert werden müssen. Die Durchführung der robusten Suche mit RANSAC ist umso effizienter, je sicherer eine Annahme über die Anzahl vorhandener problematischer

$RB_i$  gemacht werden kann. Dies ist deshalb der Fall, weil die Anzahl der auszufilternden  $RB_i$  und die Menge an Auswahliterationen für die Menge passend zur erwarteten Gesamtzahl problematischer  $RB_i$  gewählt werden muss.

Als einfaches Rechenbeispiel wird der (degenerierte) Fall betrachtet, dass nur zu einem Geschäft durch die zugehörige  $RB_i$  fehlerhafte Geschäftsmerkmale vorgegeben werden. In diesem Fall könnte die Größe für die Menge auszufilternder Randbedingungen als 10 gewählt und dann 10 Auswahliterationen für die Menge durchgeführt werden. Dann hat man eine 76% Chance, dass die fehlerhafte Randbedingung bei einer Auswahliteration ausgewählt wurde. Im direkten Vergleich müsste man beim systematischen Ausfiltern einzelner  $RB_i$  für die gleiche Trefferchance 76 Iterationen durchführen.

Ist die erwartete Anzahl problematischer  $RB_i$  nicht bekannt, muss mit der Annahme, dass nur wenige  $RB_i$  ausgefiltert werden müssen, gestartet werden. Kann keine Verbesserung des Ergebnisses erzielt werden, so muss die Annahme erhöht und die robuste Suche erneut gestartet werden.

In der Fallstudie war keine Vermutung über den Anteil problematischer  $RB_i$  an  $URB$  verfügbar. Daher musste für mehrere mögliche Anzahlen an problematischen  $RB_i$  (genauer: 10, 20, 30, 40, 50, 60, 70, 100 und 132) eine robuste Kalibrierung durchgeführt werden.

### 17.3.3 Ziel der robusten Kalibrierung

Das Ziel der robusten Kalibrierung von  $KP_{GeoUmeaHom}^{Robust}$  bestand darin:

- Festzustellen, ob es in der Menge der  $URB$  einzelne, besonders stark verrauschte Randbedingungen gab, die das Simulationsverhalten verzerrten. Konkret sollten Geschäfte identifiziert werden, deren aus Daten vorgegebene Geschäftsmerkmale so fehlerhaft waren, dass deren Wahrnehmung die Geschäftsbewertungen der Agenten so verzerrte, dass kein valides Simulationsverhalten entstehen konnte.
- Eine möglichst kleine Menge solcher  $RB_i$  auszufiltern und so Simulationsverhalten zu erzeugen, dass immer noch auf möglichst vielen Daten- $RB$  basierte, aber zugleich signifikant valider war. Konkret musste bei Ausfilterung einer  $RB_i$  dafür gesorgt werden, dass die Agenten das zugehörige Geschäft  $i$  nicht mehr auf Basis der Geschäftsmerkmale in  $RB_i$ , sondern auf Basis des zugehörigen Wertes in  $IVM_{AG}$  bewerteten.

### 17.3.4 Umsetzung im DAVINCI Kalibrierer

Die robuste Kalibrierung wurde im DAVINCI-Kalibrierer durchgeführt. Hierbei wurden im Rahmen der Lösung von  $KP_{GeoUmeaHom}$  folgende Techniken angewendet.

- **Reverse-Engineering:** Zunächst wurde das aggregierte Verhaltensmodell  $IVM_{AG}$  der Gruppe aller Einkaufsagenten im RE-Schritt bestimmt. Tatsächlich mussten diese Werte nicht neu berechnet werden, sondern konnten einfach von der Lösung von  $KP_{GeoUmeaHom}$  übernommen werden.
- **Robuste Kalibrierung:** Im Modell wurde für Anwendung von DAVINCI jedes Geschäft um eine zusätzliche boolesche Filtervariable für die robuste Kalibrierung erweitert. Diese Filtervariable wird von DAVINCI verwendet, um festzulegen, welche

Geschäfte während einer Kalibrierung von den Einkaufsagenten normal wahrgenommen werden und welche mittels des Wertes des Geschäftes in  $IVM_{AG}$  bewertet werden. Es wurden für 10, 20, 30, 40, 50, 60, 70, 100 und 132 Anzahlen möglicher problematischer  $RB_i$  robuste Kalibrierungen durchgeführt. Während jeder einzelnen robusten Kalibrierung (außer mit 132) wurden 10 Iterationen durchgeführt, innerhalb derer eine entsprechende Anzahl an  $RB_i$  zufällig ausgewählt und die Agentenwahrnehmung der zugehörigen Geschäfte durch den jeweiligen  $IVM_{AG}$ -Wert ersetzt wurde. Auf Basis der so ausgefilterten  $RB_i$  wurde dann in jeder Iteration ein Kalibrierungslauf durchgeführt, dessen Ergebnis nur von den nicht ausgefilterten  $RB_j$  abhängig war. Während der Entwicklung der robusten Suche waren die Tests auch mit zusätzlichen Iterationen durchgeführt worden. Es hatte sich aber gezeigt, dass sich mit mehr als 10 Iterationen keine Veränderung im Gesamtergebnis ergab.

- **Sequentielle Kalibrierung:** Die Verknüpfung der beiden Kalibrierungsschritte wurde durch sequentielle Kalibrierung im DAVINCI Kalibrierer (siehe Abschnitt 7) realisiert. Dabei wurde zunächst das Reverse-Engineering durchgeführt und dann automatisch die  $IMV_{AG}$  zur robusten Kalibrierung übertragen.

## 17.4 Kalibrierungsergebnisse

In diesem Abschnitt werden die Ergebnisse der robusten Kalibrierung von  $KP_{GeoUmeaHom}^{Robust}$  untersucht.

### 17.4.1 Ergebnisse der Kalibrierungsläufe

Abbildung 17.1 zeigt die Ergebnisse der robusten Kalibrierungen für verschiedene Anzahlen ausgefilterter  $RB$ -Elemente. Für jede Anzahl auszufilternder Elemente wurden 10 Iterationen durchgeführt, d.h. 10-Mal zufällig eine Menge auszufilternder  $RB$  gewählt und auf Basis dieser Ausfilterung eine Kalibrierung durchgeführt. Entlang der X-Achse werden aufsteigend die verschiedenen Anzahlen ausgefilterter Elemente angezeigt. Entlang der Y-Achse wird der in den jeweiligen robusten Kalibrierungsläufen erreichte Validitätsgrad des resultierenden Simulationsverhaltens aufgetragen. Der Validitätsgrad entspricht dabei jeweils dem Ergebnis der robusten Kalibrierung zur jeweiligen Anzahl auszufilternder  $RB$ , ist als jeweils der höchste Validitätsgrad der 10 durchgeführten Kalibrierungsläufe. Als erster Wert ist der Ausgangswert aus  $KP_{GeoUmeaHom}$  aufgetragen, da  $KP_{GeoUmeaHom}^{Robust}$  auf  $KP_{GeoUmeaHom}$  aufbaut.

Der Abbildung kann entnommen werden, dass jede Erhöhung der Anzahl ausgefilterter Elemente eine Verbesserung des Validitätsgrades des Simulationsverhaltens brachte. Abbildung 17.1 zeigt eine durchgezogene Linie, welche alle ermittelten Validitätswerte darstellt, eine gepunktete Trendlinie, welche die ermittelten Validitätswerte in 20er Schritten visualisiert und eine Trendlinie (die lineare Trendlinie ist automatisch aus den Ergebnisdaten berechnet und schneidet deshalb 1 bereits vor 132 ausgefilterten Elementen).

### 17.4.2 Interpretation

Mögliche Interpretationen der Ergebnisse waren:

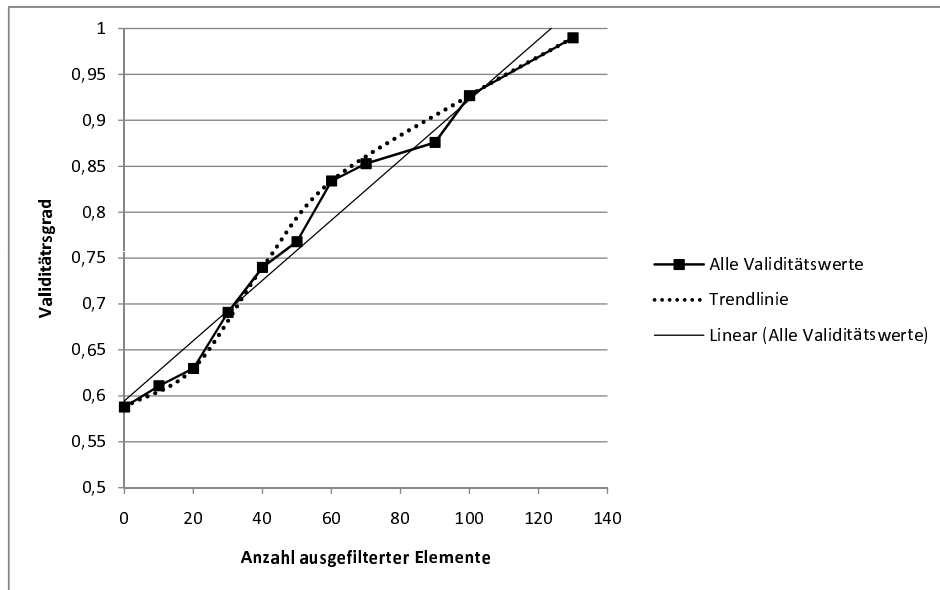


Abbildung 17.1: Kalibrierungsergebnisse der robusten Kalibrierungen.

- **Nur allgemeines Datenrauschen:**

Wären die Daten nur linear verrauscht, so müsste die Verbesserung des Validitätsgrades bei Ausfilterung größerer Anzahlen von *RB* in etwa linear ansteigen, d.h. in Abbildung 17.1 ungefähr durch die lineare Trendlinie beschrieben werden.

- **Starkes Datenrauschen in einzelnen *RB*:**

Wären einzelne *RB* besonders stark verrauscht, so müsste der Verlauf des Validitätsgrades an einer bestimmten Stelle einen deutlichen „Knicks“ haben, nämlich dann, wenn alle problematischen *RB* erfolgreich ausgefiltert werden konnten. Nach dieser Stelle müsste wieder ein linearer Verbesserungsverlauf des Validitätsgrades beobachtbar sein, da durch weitere Ausfilterungen lediglich allgemeines Datenrauschen beseitigt würde.

- **Zu geringe Anzahl an Ausfilterungsiterationen für kleine Filtergrößen:** Eine Möglichkeit für geringe Verbesserungen in den Testfällen mit weniger ausgefilterten Elementen ist, dass zu wenige Iterationen für die zufällige Auswahl der zu filternden Elemente durchgeführt wurden. Hierdurch war die Chance, dass alle fehlerhaften Elemente in der ausgefilterten Gruppen lagen zu gering. Entsprechendes ist aber in den Ergebnissen nicht zu beobachten, da die Validitätsbewertung nahezu linear mit dem Anzahl ausgefilterter Elemente ansteigt. Weiter haben vorausgehende Tests gezeigt, dass eine Erhöhung der Iterationszahl nicht zu unterschiedlichen Ergebnissen führte.

Im allgemeinen Fall ist die Gefahr bei geringen Ausfilterungsanzahlen nicht alle fehlerhaften Elemente auszuwählen der Grund, warum man die Anzahl ausgefilterter Elemente größer wählen muss als die Anzahl erwarteter fehlerhafter Elemente. Weiter muss bei unbekannter Anzahl fehlerhafter Elemente zunächst mit einer kleinen Ausfilterungsanzahl begonnen und diese dann gesteigert werden. Die Elemente



müssen dabei in allen Iterationen zufällig ausgewählt werden, da nicht bekannt ist, wie viele Elemente in einer Auswahl fehlerhaft sind. Wollte man den Auswahlprozess z.B. durch Tabu-Listen bereits getesteter Elemente verbessern, so wäre die nur unter der Annahme möglich, dass jedes Element nur einmal in eine Auswahlmenge aufgenommen werden muss. Diese Annahme gilt für größere Auswahlmengen nicht.

Kann starkes Datenrauschen einzelner *RB* identifiziert werden, kann eine möglichst kleine Menge auszufilternder *RB* an der Stelle des „Knick“ im Verlauf des Validitätsgrades abgelesen werden.

Die beiden in Abbildung 17.1 eingezeichneten Trendlinien geben den Trend jeweils unter den beiden Hypothesen „allgemeines Rauschen“ (durchgezogene Trendlinie) und „starkes Rauschen einzelner *RB*“ (gepunktete Trendlinie) wider.

#### 17.4.2.1 Vergleich mit gepunkteter Trendlinie

Die meisten der ermittelten Validitätsgrade liegen auf der gepunkteten Trendlinie. Die einzigen größeren Abweichungen der zwischen Werte von der Linie sind der Wert für 50 und für 90 ausgefilterte Elemente. Die Abweichungen sind nicht groß (0,03 für 50 und 0,02 für 90 Elemente), aber deutlich höher als bei den übrigen Elementen. Mögliche Erklärungen hierfür sind zum einen, dass das verwendete heuristische Suchverfahren mit jeweils 500 Suchschritten pro Iteration und jeweils 10 Iterationen in diesen Einzelfällen nicht genau genug kalibrieren konnte und zum anderen, dass erst durch eine weitere Erhöhung der Anzahl ausgefilterter Elemente wieder eine signifikante Steigerung des Validitätsgrades erreicht werden konnte. Zur genaueren Untersuchung wurden die robusten Kalibrierungen für 40 50 und 90 ausgefilterte Elemente jeweils drei Mal wiederholt, was die erzielten Validitätsgrade bestätigte. Im Bereich zwischen 30 und 60 ausgefilterten *RB* ist der Anstieg des Validitätsgrades deutlich stärker als davor und danach. Entsprechend könnte die Menge an 60 ausgefilterten *RB* (Knick der gepunkteten Trendlinie) als Ergebnis der robusten Kalibrierung interpretiert werden.

#### 17.4.2.2 Vergleich mit durchgezogener Trendlinie

Nur wenige der ermittelten Validitätsgrade liegen auf der durchgezogenen Trendlinie. Im Bereich von 0 bis 30 ausgefilterten *RB* liegt die Validitätskurve unter der durchgezogenen Trendlinie, im Bereich von 30 bis ausgefilterten Elementen darüber. Dennoch ist die Abweichung nicht so groß, dass dies nicht als „pendeln“ der Validitätswerte interpretiert werden könnte.

#### 17.4.2.3 Analyse der ausgefilterten Elemente

Untersucht man die Ergebnisse der robusten Kalibrierung hinsichtlich Übereinstimmungen in den im Ergebnis ausgefilterten Elementen, können hierdurch Rückschlüsse über die besonders problematischen Elemente gezogen werden, d.h. über die Elementen, die auf jeden Fall ausgefiltert werden müssen. In der Fallstudie nehmen die Einkaufsagenten 132 verschiedene Geschäfte wahr. Von diesen Geschäften sind 47% kleine Lebensmittelgeschäfte auf dem Land, 30% Tankstellen und „Tante Emma“ Läden mit Lebensmittelverkauf (alle diese Geschäfte haben nur geringe Umsätze) und nur 23% größere Supermärkte, sogenannte

Vollsortimenter (diese Geschäfte haben große Umsätze).

### Elemente mit potentiell größter Verrauschung

Bildet man die Schnittmenge aller Mengen ausgefilterter Elemente aus den einzelnen Kalibrierungsergebnissen besteht diese aus lediglich *RB* bezüglich vier Geschäften. **Einer Tankstelle mit Lebensmittelverkauf, einem „Tante Emma“-Laden, einem mittelgroßen Supermarkt und dem OBS Stormarknad, dem größten Supermarkt der ganzen Region.** Die ersten beiden Elemente spielen im Validitätskriterium des unquadratischen Gütemaßes keine Rolle. Entscheidend ist die Ausfilterung der größeren Supermärkte und insbesondere des OBS. Das Ergebnis zeigt, dass der OBS durch die verfügbaren Daten im Vergleich zu den übrigen Geschäften zu ungenau charakterisiert wird, so dass die Kalibrierung kein valideres Verhalten finden kann. Der Einfluss des OBS ist dabei insbesondere entscheidend, da er aufgrund seines Umsatzes den größten Einfluss auf den Wert des unquadratischen Gütemaßes hat.

### Zusammenhang zwischen Vollsortimentern und Validitätsgrad

Im Folgenden werden die einzelnen Mengen ausgefilterte Elemente betrachtet und hinsichtlich der enthaltenen Elemente analysiert. Abbildung 17.2 zeigt eine Analyse (für 40 -100 ausgefilterte Elemente) in der das Verhältnis an Vollsortimentern an der Menge ausgefilterter Elemente im Vergleich zur Steigerung der Validität des Simulationsverhaltens aufgetragen ist. Die Validitätswerte sind dabei die gleichen wie in Abbildung 17.1. Der Abbildung ist klar

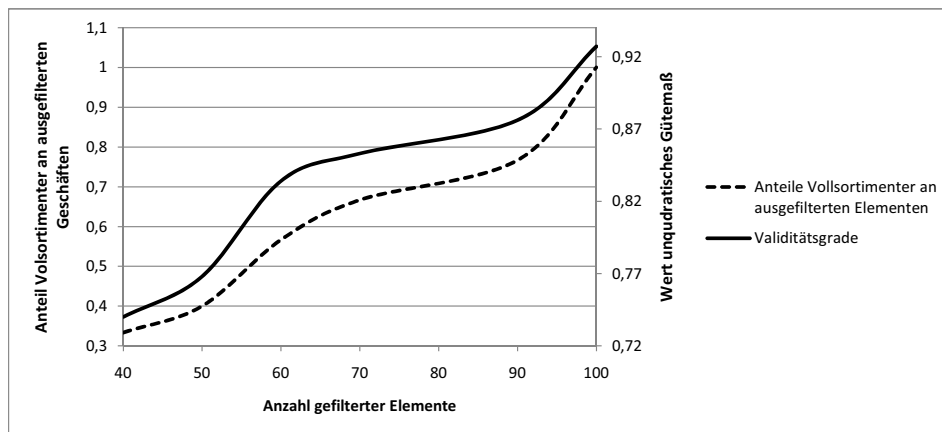


Abbildung 17.2: Anteil an Vollsortimentern an der Menge ausgefilterter Elemente im Vergleich zur Steigerung des Validitätsgrades.

ein starker Zusammenhang zwischen dem Anteil an Vollsortimentern und der Veränderung der Simulationsvalidität zu erkennen.

Dies lässt zwei mögliche Schlussfolgerungen zu:

1. Die Daten bezüglich der Vollsortimenter sind besonders verrauscht und müssen verbessert, bzw. neu erhoben werden.
2. Die Daten bezüglich der Vollsortimenter sind nicht notwendigerweise stärker verrauscht als die Daten bezüglich der übrigen Geschäfte. Das gewählte Validitätskriterium des unquadratischen Gütemaßes führt jedoch dazu, dass die Kalibrie-

rung besonders hinsichtlich dieser umsatzstärkeren Geschäfte durchgeführt wird.

#### 17.4.2.4 Schlussfolgerung aus Analyseergebnissen

Die Analyse des Zusammenhangs zwischen Geschäften mit großem Umsatz und dem Verlauf des Validitätskriteriums liefert eine gute Erklärung für den Verlauf des Validitätsgrades in Abbildung 17.1. Die robuste Kalibrierung von  $KP_{GeoUmeaHom}^{Robust}$  zeigte daher, dass das Rauschen in den Daten zu den charakteristischen Merkmalen der Geschäfte gleichmäßig ist. Aus diesem Grund konnte in der Fallstudie die Ausfilterung einzelner Geschäfte zur Verbesserung des Simulationsverhaltens keinen Gewinn für das Mikro-Modell bringen.

## 17.5 Diskussion der Evaluierung

In diesem Abschnitt wird das Ergebnis der Evaluierung der robusten Kalibrierung für Agenten-basierte Simulationen diskutiert:

- Mit der robusten Suche können potentiell verrauschte Teile der  $RB$ , auf denen die Agentenentscheidungen basieren, entfernt, d.h. auf ihnen basierende Modellteile geeignet durch ihre zugehörigen  $IVM$  ersetzt werden, ohne dass das Simulationsverhalten durch ihre Entfernung verzerrt worden ist.
- Die robuste Kalibrierung wurde als ein Werkzeug untersucht, um besonders problematischen Modellteile hinsichtlich validen Simulationsverhaltens zu identifizieren. Die Anwendung der robusten Suche in der Fallstudie zeigte, dass in der Fallstudie die Daten zu allen Geschäften gleichmäßig verrauscht waren. Als Konsequenz war die Ausfilterung von einzelnen Geschäften kein sinnvolles Mittel, um in der Fallstudie die Validität des Simulationsverhaltens zu erhöhen. Für die allgemeine Anwendung der robusten Suche zeigte dies, dass die robuste Suche zur Überprüfung der Vermutung, dass das Modell auf der Grundlage einzelner fehlerhafter Daten erstellt wurde, eingesetzt werden kann. Es muss jedoch geprüft werden, ob wirklich nur einzelne Daten und nicht alle Daten verrauscht sind, da sonst der Einsatz der robusten Suche zur Kalibrierung keinen Gewinn bringt.
- Die robuste Kalibrierung kann im Rahmen des Makro-Mikro-Verfahrens genutzt werden. Mit der robusten Kalibrierung kann durch die Ersetzungen von Mikro-Modellteilen, die auf ausgefilterten  $RB$  basieren, durch ihre zugehörigen  $IVM$  theoretisch ein Simulationsmodell mit erhöhter Verhaltensvalidität erzeugt werden, wenn dies den Nutzen des eigentlichen Mikro-Modells für seinen Anwendungszweck nicht einschränkt. Jede Ersetzung, die von der robusten Suche vorgenommen wird, tauscht einen Mikro-Teil durch seinen entsprechendes Gegenstück aggregierten idealen Verhaltens aus. Müsste ein Mikro-Modellteil ausgetauscht werden, der entscheidend ist für die Anwendbarkeit des Mikro-Modells, so kann der Austausch nicht vorgenommen werden, ohne dass das Mikro-Modell damit nutzlos wird.

Vor dem Einsatz der robusten Suche muss daher immer genau geprüft werden, ob die Ersetzung das Mikro-Modell nur dem idealen Verhaltensmodell der  $IVM$  auf der Makro-Ebene des Makro-Mikro-Verfahrens ähnlicher macht, ohne dass etwas für die Simulation auf der Mikro-Ebene gewonnen wird oder ob problematisches Verhalten

ausgefiltert werden kann, so dass der die Ergebnisse des Mikro-Modells wirklich verbessert werden.

- Die robuste Kalibrierung ist im Rahmen des Makro-Mikro-Verfahrens ein erfolgreicher Anwendungsfall für die beschriebene sequentielle Kalibrierung. In einem ersten Schritt werden die individuellen Zielfunktionen für die Agenten im RE-Schritt berechnet, im nächsten Schritt wird auf Basis dieser Zielfunktionen die robuste Kalibrierung durchgeführt.

Insgesamt kann die robuste Suche also genutzt werden, problematische *RB* auf der Mikro-Ebene auszufiltern, indem auf ihnen basierende Mikro-Modellteile durch ideale *IVM*, die im Makro-Schritt des Makro-Mikro-Verfahrens ermittelt wurden, ersetzt werden. Es muss jedoch immer beachtet werden, dass das Modell dabei vom Mikro-Modell zu einem aggregierteren Modell idealisierten Verhaltens hin verändert wird. Daher muss genau abgewogen werden welche Änderungen akzeptabel sind und das Simulationsmodell nützlicher machen und welche Änderungen zwar die Validität des Gesamtverhaltens bezüglich einer Makro-Zielfunktion erhöhen, aber auf Mikro-Ebene kein sinnvolles Modellverhalten mehr ermöglichen.

## Kapitel 18

# Zusammenfassung und Ausblick

Eine ABS erlaubt es Systeme mit komplexem Verhalten einzelner Agenten und deren Interaktionen auf hohem Detaillierungsgrad abzubilden und zu simulieren. Die Entwicklung solcher Agenten-basierter Modelle ist jedoch nicht trivial. Bei der Modellerstellung müssen konzeptionell valide Strukturen entworfen und bei der Kalibrierung die Parameterwerte des Modells so eingestellt werden, dass valides Simulationsverhalten entsteht. Sowohl die Lösung der Modellierungs- als auch die der Kalibrierungsprobleme ist bei der Erstellung von ABS problematisch.

Auf der Kalibrierungsebene entstehen Probleme, wenn die zur Kalibrierung und Validierung notwendigen Daten nicht in ausreichender Menge und Qualität verfügbar sind. Dies ist insbesondere dann der Fall, wenn durch die Kalibrierung die Validität des Modellverhaltens auf verschiedenen Beobachtungsebenen sichergestellt werden soll. Eine Kalibrierung rein auf Basis der Validität des Gesamtverhaltens oder der individuellen Verhaltensweisen der Agenten kann keine Garantie für die Validität auf anderen Beobachtungsebenen geben. Es werden daher Methoden benötigt, um auf verschiedenen Beobachtungsebenen zu kalibrieren. Solche Methoden sind besonders dann wichtig, wenn nicht auf allen relevanten Beobachtungsebenen geeignete Validierungskriterien für eine automatische Kalibrierung formalisiert werden können (z.B. fehlende oder verrauschte Validierungsdaten). Ein weiteres großes Problem der Kalibrierung von ABS sind die potentiell riesigen Parametersuchräume die bei der Kalibrierung individueller Agentenparameter entstehen. Die langen Simulationslaufzeiten von ABS kommen erschwerend hinzu. Es werden daher geeignete Methoden benötigt, um die Kalibrierung von ABS in begrenzter Zeit und begrenzten Berechnungsressourcen sinnvoll durchführen zu können.

Zur Behandlung von Kalibrierungsproblemen in ABS auch bei unvollständigen oder verrauschten Daten wurden in dieser Arbeit das Makro-Mikro-Verfahren mit Erweiterungen entwickelt und in ein Kalibrierungswerkzeug (DAVINCI) zur Unterstützung des Verfahrens umgesetzt. Die dabei vorgestellten Lösungsansätze umfassten:

- Ein Basis-Makro-Mikro-Verfahren zur Kalibrierung auf verschiedenen Beobachtungsebenen: Im Makro-Mikro-Verfahren wird, von der Makro-Beobachtungsebene kommend, zunächst versucht, Modelle idealen aggregierten Agentenverhaltens zu erstellen, die zu ausreichend validem Gesamtverhalten auf der Makro-Ebene führen. Aus diesen Modellen aggregierten Verhaltens werden dann Validierungskriterien für die Kalibrierung von Agenten und anderen Mikro-Modellteilen abgeleitet. Diese Validierungskriterien werden dann genutzt, um das konkrete Agentenverhalten auf

der Mikro-Ebene zu kalibrieren.

- Da es oft nicht einfach möglich ist, direkt ideale Modelle aggregierten Verhaltens zu erstellen wurde das Basis-Makro-Mikro-Verfahren durch das Reverse-Engineering verfeinert. Ihre Verwendung macht es nicht mehr notwendig vollständige Modelle idealen aggregierten Mikro-Verhaltens zu definieren. Stattdessen müssen „nur noch“ Strukturen zur Beschreibung aggregierten Agentenverhaltens spezifiziert werden. Der Schritt der Reverse-Engineering-Verfeinerung dient dann dazu, um auf der Basis gegebener Modellstrukturen die konkreten idealen Verhaltensmodelle auf der Makro-Ebene zu ermitteln.
- Ein Verfahren, das die im Makro-Schritt ermittelten idealen Verhaltensmodelle zur robusten Kalibrierung von ABS nutzt. Der Kalibrierungsansatz ist robust gegen die starke Verrauschung einzelner Kriterien zur Validierung individuellen Agentenverhaltens. Die Idee des Ansatzes besteht darin, für solche verrauschten Validierungskriterien die zugehörigen Agentenmodelle auf der Mikro-Ebene durch theoretisch zum Restmodell passende Modelle abstrahierten Agentenverhaltens zu ersetzen. Diese Modelle können mittels Reverse-Engineering ermittelt werden.
- Eine Kalibrierungstechnik, die das Makro-Mikro-Verfahren um eine Heterogenitätssuche erweitert. Der Ansatz nutzt, dass nicht notwendigerweise vollkommen heterogene Werte für die individuellen Agentenparameter kalibriert werden müssen und schränkt den Parametersuchraum während der Kalibrierung geeignet ein. Er bietet dem Nutzer damit einen Mittelweg zwischen der theoretischen Notwendigkeit, vollständig heterogene Agentenparameter zu kalibrieren und dem Interesse, den Parametersuchraum während der Kalibrierung möglichst klein zu halten.

Das Makro-Mikro-Verfahren und seine Erweiterungen wurden anhand von Problem-szenarien und einer Fallstudie evaluiert und als positiv bewertet. Das Verfahren und seine Erweiterungen können darüber hinaus als Ausgangspunkte für viele weitergehende Untersuchungen und Entwicklungen bei der Erstellung und Kalibrierung von ABS dienen, von denen im Folgenden einige vorgestellt werden.

## 18.1 Anwendung des Makro-Mikro-Verfahrens auf weitere ABS

Ein wichtiger nächster Schritt ist die Anwendung des Makro-Mikro-Verfahrens und seiner Erweiterungen auf möglichst viele weitere ABS aus möglichst vielen verschiedenen Anwendungsdomänen. Je nach Domäne gibt es bei der Entwicklung von ABS sehr unterschiedliche Ausgangssituationen was benötigtes Abstraktionsniveau, die Definition von ausreichend validem Verhalten und verfügbaren Daten über das Originalsystem angeht. Durch Anwendung in vielen unterschiedlichen Domänen kann die generelle Anwendbarkeit des Makro-Mikro-Verfahrens weiter verbessert und es können gegebenenfalls zusätzliche Techniken für Spezialprobleme in spezifischen Domänen entwickelt werden.

## 18.2 Verzahnung von Kalibrierung und Datenanalyse

Die vorgestellten Verfahren können nicht nur zur Lösung von Kalibrierungs- und Datenproblemen, sondern auch als Ausgangspunkt für die stärkere Verzahnung von Daten-

erhebung und Kalibrierung speziell für Entwicklung von ABS dienen.

### 18.2.1 Datenerhebung für die Kalibrierungsverfahren

Um die vorgestellten Kalibrierungsverfahren optimal nutzen zu können, werden Datenerhebungs- und Verarbeitungstechniken benötigt, die speziell auf die Anwendung der vorgeschlagenen Verfahren ausgerichtet sind. Beispielsweise wäre die Integration zusätzlicher Data-Mining-Techniken in den Kalibrierungsprozess als Hilfsmittel der Heterogenitätssuche wünschenswert. Im vorgeschlagenen Ansatz spezifiziert der Nutzer Ähnlichkeitsfunktionen zur Beschreibung von Parametergruppen. Hier könnte man einerseits versuchen, den Prozess der Identifikation geeigneter Ähnlichkeitsfunktionen zu automatisieren. Eine andere Möglichkeit wäre es, dem Entwickler z.B. durch Subgroup Mining [6] auf den Daten über die simulierten Agenten zu ermöglichen, „vielversprechende“ Gruppierungseigenschaften für die Agenten zu erkennen.

### 18.2.2 Vorhersage noch zu erhebender empirischer Daten

Die Validität eines Simulationsmodells hängt stark von den verfügbaren Daten und Theorien ab, auf deren Grundlage das Modell erstellt wird. Fehlen Daten kann unter Umständen kein ausreichend valides Modellverhalten erzeugt werden. Das Reverse-Engineering könnte nun genutzt werden, um Vorhersagen bezüglich noch benötigter Datenwerte zu machen, mit denen die Validität des Simulationsverhalten verbessert werden könnte. Dazu könnte der Modellierer oder Fachexperte z.B. eine neue Wahrnehmung für die Agenten einführen, für die noch keine entsprechenden Daten erhoben wurden. Mittels Reverse-Engineering könnten theoretisch benötigte Werte für diese Daten berechnet werden, die zu ausreichend validem Simulationsverhalten führen würden. Diese Werte könnten dann zur Prüfung der tatsächlich erhobenen Daten genutzt werden. Dabei könnte untersucht werden, ob die Daten ausreichend korrekt sind. Weiter könnte so direkt geprüft werden, ob die erhobenen Daten überhaupt zu validem Simulationsverhalten führen können oder ob Daten bezüglich anderer Wahrnehmungsaspekte benötigt werden, die eher den errechneten Theoriewerten entsprechen.

### 18.2.3 Kalibrierungsverfahren für Datenanalyse und -test

Die Lösungsansätze dieser Arbeit behandeln verschiedene Kalibrierungsprobleme für ABS, die aus der Problematik unvollständiger oder verrauschter Modellierungs- und Validierungsdaten resultieren. Entsprechende Probleme führen letztendlich dazu, dass ABS nur als Erklärungsmodelle, jedoch nicht zur Vorhersage, Planung oder Kontrolle des Verhaltens eines realen Systems eingesetzt werden können. Erst mit Modellen, die erfolgreich kalibriert wurden, können Aussagen über ein konkretes reales System gemacht werden. Die Durchführung des Makro-Mikro-Verfahrens, des Reverse-Engineering, der robusten Suche und der Heterogenitätssuche kann dem Modellierer oder Fachexperten Hinweise auf fehlende oder verrauschte Daten und Probleme bei der Beschreibung von Agentengruppen geben. Mit der Reverse-Engineering-Verfeinerung wurde in dieser Arbeit ein Verfahren vorgestellt, das es dem Modellierer oder Fachexperten ermöglicht, zur Modellerstellung die Korrektheit verwendeter Daten über das Originalsystem im Rahmen eines festgelegten Modells zu

Hinterfragen. Dieses Vorgehen könnte zu einer systematischen Konsistenzanalyse für empirische Daten ausgebaut werden. Dabei müsste der Modellierer oder Fachexperte zunächst eine Reihe von Annahmen über das Originalsystem aufstellen und diese in entsprechende Modelle umsetzen, die jeweils auf anderen Teilen der Daten beruhen. Auf der Grundlage dieser Annahmen können dann systematisch alle verwendeten Daten auf ihre Konsistenz untereinander hin geprüft werden.

### 18.3 Kalibrierung und interaktive Modelleingriffe

Die Interaktionsmöglichkeiten des Modellierers oder Fachexperten während eines Kalibrierungslaufes beschränken sich in den vorgeschlagenen Lösungsansätzen auf Vorgeben von Parameterwerten und Austesten von Änderungen an den verwendeten Kalibrierungsverfahren. Die strukturellen Modellausprägungsunsicherheiten werden vom Nutzer vor Beginn der eigentlichen Kalibrierung deklariert und vom System durchgespielt. Eine mögliche Erweiterung dieses Vorgehens wäre die Entwicklung eines kombinierten Modellierungs- und Kalibrierungsverfahrens, das es dem Modellierer oder Fachexperten erlaubt, zur Laufzeit einer Kalibrierung interaktiv Änderungen an den Modellstrukturen vorzunehmen. Das Problem bei solchen Online-Strukturänderungen in Bezug auf die Kalibrierung liegt darin, dass strukturelle Modelländerungen zugleich auch die Dimension und die Struktur des Parametersuchraums verändern können. Es kann also nicht davon ausgegangen werden, dass die für bereits getestete Parametereinstellungen ermittelten Bewertungen auch für die geänderte Struktur gelten. Effektiv wäre ein kompletter Neustart der Kalibrierung notwendig. Um dies zu vermeiden, müssten z.B. Methoden ermittelt werden, mit denen der Einfluss der durchgeführten Strukturänderungen auf Bewertungen der bereits getesteten Parametereinstellungen „geschätzt“ werden kann. Die Hoffnung bestünde dabei darin, eine entsprechende „Suchraumtransformation“ mit weniger Simulationsläufen ermitteln zu können als zum Testen der gleichen Parametereinstellungen bei einem kompletten Neustart nötig wären.

### 18.4 Verteilte Simulation und Kalibrierung

In den für diese Arbeit entwickelten Lösungsansätzen wird davon ausgegangen, dass dem Nutzer nur begrenzte Berechnungsressourcen, d.h. eine gewisse Anzahl an Simulationsläufen auf einem Computer, zur Verfügung stehen. Stehen mehrere Computer zur Verfügung bzw. verfügt der verwendete Computer über mehrere Prozessoren oder Kerne, bietet sich die Verteilung des Kalibrierungsprozesses auf mehrere Prozessoren oder Kerne an. Die Verteilung des Berechnungsaufwandes kann dabei zur Beschleunigung einzelner Simulationsläufe als auch zur verteilten Bewertung von Parametereinstellungen verwendet werden.

### 18.5 Modellierung entlang des Makro-Mikro-Verfahrens

Das in dieser Arbeit vorgestellte Makro-Mikro-Verfahren geht von einem auf Mikro-Ebene entworfenen Modell aus, für das ausreichende Verhaltensvalidität auf verschiedenen Beobachtungsebenen erreicht werden soll. Dabei wird vom Verhalten auf der Makro-



Ebene ausgegangen, dieses auf aggregierte Verhaltensmodelle für die Agenten „heruntergebrochen“ und daraus Zielfunktionen idealen Verhaltens für die Agentenmodelle auf der Mikro-Ebene abgeleitet. Die Beschreibung zeigt, dass das Makro-Mikro-Verfahren in ein Vorgehensmodell zur Entwicklung von ABS umgesetzt werden könnte. In einem solchen Vorgehensmodell würde der Entwurf und die Erstellung der Modellstrukturen dem Vorgehen zur Validierung und Kalibrierung im Makro-Mikro-Verfahren folgen. In einem solchen Top-Down-Ansatz würde zunächst ein aggregiertes Modell ausreichend validen Gesamtverhaltens erstellt. Auf der Basis dieses Modells würde versucht, ein Modell zu entwickeln, indem nur die aggregierten Agentenverhaltensweise modelliert werden. Kann dieses zweite Modell so erstellt werden, dass ausreichend valides Gesamtverhalten erzeugt wird, wird dann auf Basis der ermittelten Verhaltensmodelle nach konkreten Agentenmodellen (d.h. auch Strukturen) gesucht, durch die entsprechendes individuelles Verhalten erzeugt werden kann.

## 18.6 Kalibrierung und Modellausprägung mit Komponentenbibliotheken

In manchen Anwendungsfällen müssen nicht nur Parameter kalibriert werden, sondern es müssen auch Varianten bestimmter Modellstrukturen auf ihre Brauchbarkeit für einen bestimmten Anwendungsfall hin analysiert werden. Dies ist bei ABS häufig nötig, da in vielen Anwendungsbereichen von ABS noch keine Referenzmodelle existieren und somit unklar ist, wie bestimmte Modellkonzepte konkret realisiert werden sollen. Es ist möglich, die „Kalibrierung“ von Teilmodellvarianten durch spezielle Modellparameter für diese Teilmodelle zu automatisieren. Ein Vergleich von Varianten solcher Teilmodelle kann aber sehr aufwendig sein, wenn zunächst „von Hand“ die Modellstrukturen für verschiedene Teilmodellausprägungen modelliert werden müssen.

Wird es für bestimmte Anwendungsbereiche zumindest möglich, Referenz-Bibliotheken möglicher Teilmodellausprägungen zu formulieren, kann dies wesentlich zur Erleichterung der kombinierten Struktursuche und Kalibrierung für den Modellierer beitragen. Anstatt mögliche Modellausprägungen „von Hand“ zu modellieren, könnte der Modellierer lediglich bestimmte Einsetzpunkte im Modell kennzeichnen und im Kalibrierungswerkzeug auswählen, welche Modellausprägungen aus der Bibliothek näher untersucht werden sollten. Um dies zu ermöglichen, ist es zum einen zentral, Referenzkonzeptmodelle für bestimmte Anwendungsbereiche und Kataloge möglicher Modellausprägungen zu ermitteln. Zum anderen werden Komponenten-basierte ABS benötigt, die baukastenartig aus Modellausprägungsteilen zusammengesetzt werden können.

## 18.7 Übertragung auf die Kalibrierung und Optimierung realer Agenten

Ein weiteres vielversprechendes Gebiet ist die Übertragung der entwickelten Kalibrierungstechniken für ABS auf die Entwicklung und Optimierung „echter“ Agentensysteme. Insbesondere das Makro-Mikro-Verfahren mit Reverse-Engineering zur Ermittlung idealer aggregierter individueller Verhaltensmodelle und die kombinierte Kalibrierung mit Heterogenitätssuche bieten sich als Bestandteile eines Top-Down-Vorgehens zur Entwicklung

problemlösender Agentensysteme an.

Solche Agentensysteme sind darauf ausgerichtet ein bestimmtes Systemziel zu erreichen. Die Agenten müssen so entworfen werden, dass das Ziel optimal erreicht wird. Ausgehend vom globalen Ziel kann mit dem Makro-Mikro-Verfahren zunächst ermittelt werden, welches ideale Verhalten das Agentensystem haben muss, damit es sein Systemziel erreicht. Ist das ideale Verhalten für die einzelnen Agenten ermittelt, dann können diese so entworfen werden, dass sie sich entsprechend verhalten. Sollen gleichartige Agenten (eine Agentenklasse) unter unterschiedlichen Randbedingungen das gleiche Verhalten zeigen, kann eine Heterogenitätsanalyse über diesen Randbedingungen helfen vorab zu klären, welche Agenten homogen und welche heterogen entworfen werden müssen.

**Teil V**  
**Anhang**



# Anhang A

## Vorausgehende Fallstudien

Zur Vorbereitung auf die Entwicklung der in dieser Arbeit vorgestellten Techniken, bei der Entwicklung des DAVINCI-Kalibrierers und um ein möglichst gutes Verständnis über die Funktionsweise und Anwendbarkeit existierender Kalibrierungsverfahren auf ABS zu erhalten wurden für diese Arbeit verschiedene weitere Fallstudien durchgeführt. Bei diesen Studien kam das Makro-Mikro-Verfahren selbst nicht zum Einsatz. Es wurden Black-Box-Kalibrierungen mit dem DAVINCI-Kalibrierer durchgeführt und White-Box-Zerlegungstechniken angewendet. In diesem Abschnitt werden kurz vier dieser Fallstudien der Kalibrierung Agenten-basierter Simulationsmodelle genannt.

### A.1 Krankenhaussimulation

In einem Projekt von Rainer Herrler vom Lehrstuhl für Informatik der Universität Würzburg [58], ging es um die Optimierung der Terminplanung von ambulanten Patienten im Krankenhaus. Es sollten optimale Parametereinstellungen für die dynamische Umplanung von Terminen gefunden werden. Dynamische Umplanung wird immer dann vorgenommen, wenn die Abweichungen zwischen Plan und Ausführung besonders stark werden. Die Effektivität von verschiedenen Einstellungen des Planungsalgorithmus ist abhängig von Umwelteigenschaften (Notfallrate, Erreichbarkeit der Patienten), die von Krankenhaus zu Krankenhaus unterschiedlich sein können.

Ziel des Projektes war es, nachzuweisen unter welchen Bedingungen dynamische Terminplanung effektiv sein kann und unter welchen Bedingungen sie nur wenige Vorteile bietet.

#### A.1.1 Modellbeschreibung

Im Simulationsmodell wurden sechs Funktionseinheiten simuliert, die für eine Menge von Patienten Behandlungen planen (genauer siehe [58]). Die Zeit bis zur Entlassung der Patientenagenten aus dem Krankenhaus stand dabei in starker Abhängigkeit zur Terminplanung einzelner Funktionseinheiten und deren Wechselwirkungen. Beispielsweise konnte es sein, dass ein Patient an einem gegebenen Tag zu lange auf eine bestimmte Behandlung bei Funktionseinheit A warten musste, so dass er seine nächste Behandlung bei Funktionseinheit B an diesem Tag nicht mehr erhalten konnte.

### **A.1.2 Optimierungskriterium**

In diesem Modell wurde nicht nach validem Modellverhalten bezüglich bestimmter Daten, sondern nach optimalen Verhalten bezüglich einer Zielfunktion gesucht. Da die Optimierung und Validierung von Simulationsverhalten sehr ähnliche Probleme sind, konnte der DAVINCI Kalibrierer auch hier eingesetzt werden. Als Optimierungskriterien wurden die durchschnittliche Aufenthaltszeit der Patienten im Krankenhaus und die Auslastung und Überstunden der Funktionseinheiten gewählt. Die Auslastung der Funktionseinheiten sollte möglichst hoch, die durchschnittliche Aufenthaltszeit und die Überstunden möglichst niedrig sein.

### **A.1.3 Parameterwerte**

Während der Kalibrierung wurde versucht die Werte von zwei Optimierungsparametern auf der Basis von drei Experimentvariablen einzustellen. Als Optimierungsparameter wurden die Regelmäßigkeit mit der eine Umplanung der Termine in Betracht gezogen wurde und der Schwellwert für Terminverzögerungen, bei dessen Überschreitung eine Umplanung tatsächlich durchgeführt wurde, genutzt. Diese wurden unter verschiedenen Randbedingungen eingestellt. Solche Randbedingungen waren die Rate mit der Notfallpatienten in die Simulation eintraten, die Erreichbarkeitswahrscheinlichkeit für Patienten beim Umplanen und die Zeitschranke innerhalb der ein Patient mindestens von einem neuen Termin informiert werden muss.

### **A.1.4 Einsatz des Kalibrierers**

Der DAVINCI Kalibrierer wurde in diesem Projekt als Black-Box-Kalibrierer eingesetzt, um sowohl die Optimierungsparameter einzustellen, als auch die Experimente unter den verschiedenen Randbedingungen durchzuführen. Dabei wurden sämtliche der genannten Werte durch die evolutionäre Strategie des DAVINCI Kalibrierers eingestellt. Im Ergebnis des Kalibrierens und Experimentierens konnte nachgewiesen werden, dass die dynamische Umplanung besonders effektiv ist, wenn es wenige Notfälle gibt und die Umgebung gute Kommunikationsmöglichkeiten bereitstellt. Auch in anderen Fällen gab es Verbesserungen durch die dynamische Umplanung, die jedoch weniger deutlich waren.

## **A.2 Simulation kooperativer Ampeln**

In einem Projekt von Prof. Ana Bazzan von der UFRGS in Porto Alegre, Brasilien, ging es um Simulation, Kontrolle und Optimierung von Stadtverkehr mit Hilfe von Multiagentensystemen. Das Ziel des Projektes bestand darin, ein bestehendes proprietäres Stand-Alone System zur Verkehrssimulation (ITSUMO) zu erweitern. Dabei soll es das erweiterte Werkzeug neue Optimierungsalgorithmen, Verkehrskontrolle mittels Verstärkungslernen und Koordinations- und Verhandlungstechniken, wie verteilte Constraint-Optimierung (DCOP), ermöglichen.

### A.2.1 Modellbeschreibung

Geben sei ein Netzwerk aus 25 Kreuzungen (Knoten) und Gitter-artigen Verknüpfungen (Straßen) zwischen diesen. Auf jeder Kreuzung befindet sich eine Ampel. Jede Ampel hat einen Signalplan. Der den beiden von der Ampel kontrollierten Straßen alternierend eine Grünphase zuweist. Auf den Straßen des Netzwerks sind verschiedene Fahrer unterwegs (in der Simulation wurden 400 genutzt), die von einem beliebigen Startknoten zu einem beliebigen anderen Knoten fahren, d.h. die alle unterschiedliche Fahrtrouten haben. Dies bewirkt, dass an manchen Kreuzungen mehr Verkehr entsteht als an anderen. In diesem Fall sollten die Ampeln ihre Grünphasen geeignet anpassen. Die Grünphasen der einzelnen Ampeln haben dabei starke Auswirkungen auf den Verkehr an diesen Ampeln und an anderen Ampeln im Netzwerk. Wird beispielsweise die Grünphase einer Ampel, an der es einen Stau gibt, verändert, kann dies bewirken, dass sich der Stau in einen anderen Teil des Netzwerkes weiterbewegt.

### A.2.2 Optimierungskriterium

Während der Optimierung sollte die **durchschnittliche Reisezeit aller in der Simulation vorhandenen Fahrer minimiert** werden.

### A.2.3 Parameterwerte

Für jede Ampel mussten zwei Parameterwerte kalibriert werden. Der erste Parameterwert beschrieb die Gesamtzeit eines Rot-Grün Zyklus für die jeweilige Ampel und der zweite Parameterwert den Wechselzeitpunkt zwischen Rot und Grünphase innerhalb eines Zyklus. Bei der Kalibrierung mussten für jede der 25 Ampeln individuelle Parameterwerte ermittelt werden. Die Menge der Parameter war zu groß, um die einzelnen Werte „von Hand“ einzustellen.

### A.2.4 Einsatz des Kalibrierers

Der DAVINCI-Kalibrierer wurde in diesem Projekt als Black-Box-Kalibrierer dazu eingesetzt, eine feste Parametereinstellung für die individuellen Parameterwerte aller Ampeln zu finden. Eine Simulation des Modells mit den gefundenen Parameterwerten führte zur minimalen bekannten durchschnittlichen Reisezeit aller in der Simulation vorhandenen Fahrer. Dieses Ergebnis war besser als die vom Modellierer initialisierten Parameterwerte, welche die Zykluszeit einfach in zwei gleich große Hälften teilten. Im Projekt wurde der DAVINCI-Kalibrierer „nur“ dazu eingesetzt feste individuelle Parameterwerte zu suchen. In einem zukünftigen Projekt könnte der DAVINCI-Kalibrierer auch zur Kalibrierung eines Modells mit Ampeln, die sich adaptiv an unterschiedliche Verkehrsverhältnisse anpassen sollen, eingesetzt werden. In diesem Fall würden z.B. entweder mehr Parameter für unterschiedliche Verkehrssituationen oder andere Parameter, welche den Adaptionmechanismus der Ampeln steuern, kalibriert.

### **A.3 Projekt Evakuierungssimulation**

Das Ziel dieses Projektes bestand darin, verschiedene Modelle für Fußgängerverhalten in verschiedenen Anwendungsszenarien zu implementieren und zu vergleichen. Auf diese Weise sollte ein Überblick über die Leistungsfähigkeit verschiedener Standard-Modelle geschaffen werden.

#### **A.3.1 Modellbeschreibung**

Eines der verwendeten Standardmodelle war das Social Force Modell, bei dem eine Reihe von Parametern eingestellt werden mussten. Das Einstellen der Parameterwerte war nicht trivial. Die Verhaltensvalidität hing stark von den jeweils gewählten Parameterwerten ab. Der Grund hierfür war, dass es beim Social Force Modell keine harten Schwellwerte für Aktionen gibt. Hindernisse haben quantitative Einflüsse auf die Bewegungsrichtung bzw. Geschwindigkeit. Mit falsch eingestellten Parametern laufen die Fußgänger durch Wände hindurch, finden ihr Ziel nicht oder bleiben an Ecken und Kanten der Räume hängen usw.

#### **A.3.2 Validitätskriterium**

Aufgrund des Abstraktionsgrades des Modells konnte zur Kalibrierung kein echtes Validitätskriterium formalisiert werden. Stattdessen wurde eine Zielfunktion zur Beurteilung der Verhaltensplausibilität aufgestellt, in welcher die Anzahl aller Kollisionen von Agenten mit ihrer Umgebung gemessen wurde. Das Ziel war eine Einstellung der Parameter zu finden, die eine Simulation ohne Kollisionen ermöglichte.

#### **A.3.3 Parameterwerte**

Die zu kalibrierenden Parameter waren die Gewichte von verschiedenen Hinderniskategorien auf den Bewegungsvektor von Fußgängern in unterschiedlichen Szenarien. Es wurden homogene Parameterwerte für alle Agenten ermittelt.

#### **A.3.4 Einsatz des Kalibrierers**

Der DAVINCI-Kalibrierer wurde als Black-Box-Optimierer dazu eingesetzt, die Größen der Einflüsse von verschiedenen Hindernissen geeignet aufeinander abzustimmen, so dass das Social Force Modell plausibles Verhalten lieferte und sinnvoll mit den übrigen Modellen verglichen werden konnte.

### **A.4 Simulation von Käferwanderungen in ländlichen Regionen**

In diesem Projekt wurde zusammen mit Dr. Michael Neumann von der Universität Gießen ein Simulationsmodell zur Analyse unterschiedlicher räumlicher Bedingungen auf die Wanderung und das Überleben verschiedener Käferarten erstellt.



#### **A.4.1 Modellbeschreibung**

Das Simulationsmodell bestand aus einem Umweltmodell der simulierten Landschaft mit Jahreszeiten und den simulierten Käferagenten, die sich in der Landschaft bewegten und fortpflanzten. Dieses Landschaftsmodell war in unterschiedliche Zonen unterteilt, die jeweils anderen landschaftlichen Gegebenheiten, wie Feld, Wiese oder Dorf, entsprachen. Jede Zone hatte andere Attribute und damit unterschiedlichen Einfluss auf das Überleben der Käferagenten zu unterschiedlichen Jahreszeiten. Die Käferagenten bewegten sich im Jahresverlauf durch die Landschaft, starben während des Jahres, insbesondere im Winter, je nach Landschaftsgebiet mit unterschiedlicher Wahrscheinlichkeit und pflanzten sich im Sommer fort.

#### **A.4.2 Validitätskriterium**

Auch in dieser Studie war es schwierig ein sinnvolles Kriterium zur Beurteilung der Verhaltensvalidität der Simulation anzugeben. Es existierten zwar Daten, aus denen abgeleitet werden konnte, welche Anteile einer Käferart sich im Verlauf eines Jahres in welchen Gebieten aufhielten, jedoch war unbekannt, wie das Gebiet, in dem die Daten erhoben worden waren, hinsichtlich der landschaftlichen Zonen aufgebaut war. Als Validierungskriterium wurde dennoch die Abweichung der Verteilung der Käferagenten in der Simulation von den Daten des Originalsystems herangezogen. Die Struktur der simulierten Umwelt wurde dabei lediglich geschätzt.

#### **A.4.3 Parameterwerte**

Die zu kalibrierenden Parameter in diesem Modell waren die Präferenzen der Käferagenten für bestimmte Landschaftstypen während einer bestimmten Jahreszeit, d.h. jeder Käferagent „spürte“ zu jedem Zeitpunkt, wie wohl er sich seinen aktuellen Gebiet fühlte, und versuchte entsprechend dieses zu verlassen oder nicht. Es wurden homogene Parameterwerte für die Agenten gesucht.

#### **A.4.4 Einsatz des Kalibrierers und Ergebnis**

Der DAVINCI-Kalibrierer wurde als Black-Box-Optimierer dazu eingesetzt, die Präferenzparameter der Käferagenten so einzustellen, dass sich die Agenten im Verlauf eines simulierten Jahres entsprechend der erhobenen Daten auf die Landschaftsgebiete verteilten. Da im Modell nur ein abstraktes Landschaftsmodell simuliert wurde, das nicht exakt zu den erhobenen Daten passte, entsprachen die ermittelten Präferenzwerte im Wesentlichen genau den Anteilen von Agenten in den einzelnen Landschaftsteilen aus den Daten. Dieses Ergebnis war hinsichtlich der Einsatzmöglichkeiten des Simulationsmodells nicht befriedigend. Aufgrund der schlechten Datenlage war keine Verbesserung des Modells möglich, jedoch konnten Forderungen aufgestellt werden, unter deren Berücksichtigung vernünftige Prognosen möglich wären.

## A.5 Projekt Bienensimulation

In diesem Projekt wurde zusammen mit Projektpartnern des Biozentrums der Universität Würzburg (Marko Kleinhenz) eine Agenten-basierte Simulation zum Heizverhalten von Honigbienen entwickelt [35]. Im Brutnest des Bienenstockes werden die Larven der Bienen gepflegt. Die Larven entwickeln sich aus Eiern, die von der Bienenkönigin in einzelne Zellen der Brutnestwabe gelegt werden. Arbeiterinnen füttern die Larven zunächst und verschließen nach einiger Zeit die Zelle mit einem Deckel. Von nun an ist es für den weiteren Entwicklungsprozess der verpuppten Larve essentiell, dass die Umgebungstemperatur immer zwischen 33°C und 36°C liegt. Die Bienen erreichen dies, indem sie die Brutzellen geeignet beheizen. Hierzu bewegen die Bienen ihre Flugmuskulatur, ohne mit den Flügeln zu schlagen. Als Folge steigt die Thoraxtemperatur der Bienen bis auf Werte über 40°C an. Die Bienen pressen sich dann an einzelne Zellen und wärmen diese mit ihrem erhitzten Thorax. Um möglichst viel neue Bienenbrut aufziehen zu können, versuchen Imker Bienenvölker zu züchten, bei denen immer das gesamte Brutfeld mit neuer Brut gefüllt ist. Die entsprechenden Zuchtversuche waren jedoch bis jetzt nicht erfolgreich. Die Bienen lassen immer zwischen 5% und 20% leere Zellen als Lücken zwischen den mit Brut gefüllten Zellen frei. Das Ziel der Fallstudie bestand darin, am Modell eine mögliche Erklärung für diese Beobachtung zu untersuchen. Beobachtungen am Bienenstock zeigen, dass die Bienen nicht nur auf dem Deckel verschlossener Brutzellen, sondern auch in leeren Zellen, die zu Brutzellen benachbart sind, heizen. Dies führte zur Theorie, dass die Bienen in leeren Zellen ihre Heizwärme effizienter (d.h. Energie-sparender) an die Brutzellen abgeben können. Enthält das Brutnest jedoch zu viele leere Zellen würde es zu schnell auskühlen. Diese Theorie sollte mit einer Simulation zum Heizverhalten von Honigbienen überprüft werden.

### A.5.1 Modellbeschreibung

Das Modell besteht im Wesentlichen aus drei Bestandteilen. Der erste Bestandteil ist die Nestkonfiguration, das simulierte Brutnest. Der zweite Bestandteil ist ein physikalisches Temperaturflussmodell für die Wärmeübertragung zwischen einzelnen Zellen, zwischen heizenden Bienen und Zellen und für den Einfluss der Umgebungstemperatur. Der dritte Bestandteil sind die simulierten Heizbienen, die sich über das Brutnest bewegen und Brutzellen heizen. Die Bienenagenten waren die einzigen aktiven Simulationsbestandteile. Die Agenten bewegten sich zufällig über das Brutnest. Die Bienenagenten prüften, ob die Zelle, auf der sie sitzen, zu kalt war. War dies der Fall, begannen sie entweder auf dem Deckel der Zellen zu heizen oder kriechen, wenn möglich, in eine benachbarte leere Zelle und heizen dort. Beim Deckelheizen konnten die Bienenagenten ihre Wärme lediglich an die Oberfläche der beheizten Zelle abgeben. Heizten sie im Inneren einer Zelle, wurde die Heizwärme an alle umliegenden Zellen abgegeben. Die Bienen beendeten den Heizprozess, wenn die Temperatur der beheizten Zelle einen bestimmtem Schwellwert überschritt.

### A.5.2 Validitätskriterien

Die Validität des Simulationsverhaltens wurde getrennt bezüglich des Temperaturflussmodells (Datengrundlage: Messwerte), der Temperaturen der einzelnen beheizten Zellen (Makro-Ebene) (Datengrundlage: Messwerte) und hinsichtlich des Verhaltens der einzelnen

Bienenagenten (Mikro-Ebene) (Datengrundlage: Literaturwerte), z.B. der Heizdauern der Bienenagenten, bewertet.

### A.5.3 Parameterwerte

Im Temperaturflussmodell mussten fünf Parameterwerte eingestellt werden, d.h. ein Temperaturflusskoeffizient jeweils für Temperaturfluss zwischen Zellen, zwischen heizender Biene und beheizter Zelle, zwischen heizender Biene und benachbarten Zellen (Abstrahlwärme), zwischen Zellen und der kühlenden Umgebungsluft und ein Aufwärmfaktor für eine sich aufheizende Biene. Die Werte mussten so kalibriert werden, dass die Messwerte des Originalsystems in der Simulation reproduziert werden konnten. Für das Verhaltensmodell eines Bienenagenten mussten zwei Parameterwerte, die Schwellwerte für Beginn und Ende des Heizprozesses kalibriert werden. Die Parameterwerte der 200 simulierten Bienenagenten konnten für diese Simulationsstudie als homogen angenommen werden.

### A.5.4 Einsatz des Kalibrierers

Der DAVINCI-Kalibrierer wurde dazu eingesetzt, sowohl das Temperatúrausbreitungsmodell als auch das Verhaltensmodell der Bienenagenten zu kalibrieren. Der DAVINCI-Kalibrierer ermöglichte dabei die Durchführung einer White-Box-Zerlegung des Gesamtkalibrierungsproblems. Zunächst wurde das Modell für den Temperaturfluss zwischen den einzelnen Zellen durch Black-Box-Optimierung kalibriert. Die Ergebnisse wurden zum nächsten Kalibrierungsproblem übertragen. Dort wurden auf der Basis des ermittelten Zell-Zell-Temperaturflussmodells das Temperaturflussmodell zwischen Bienen und Zellen durch Black-Box-Optimierung kalibriert. Nachdem beide Temperaturflussmodelle erfolgreich kalibriert waren konnten nun die Parameter des Verhaltensmodells der Bienenagenten eingestellt werden. Die Dreiteilung des Kalibrierungsproblems vereinfachte die Kalibrierung des Gesamtproblems dadurch, dass die einzelnen Teilprobleme verkleinerte Parametersuchräume hatten. Die Teilkalibrierungsergebnisse konnten direkt zusammengesetzt werden, da folgende Kalibrierungsprobleme jeweils auf die Ergebnisse der vorangegangenen Kalibrierungen aufbauten. Die Lösung der drei Teilprobleme und das Zusammensetzen der Ergebnisse konnte mit Hilfe des DAVINCI-Kalibrierers automatisch durchgeführt werden. Auf diese Weise konnte ein Simulationsmodell erstellt werden, mit dem die Simulationsfragen erfolgreich beantwortet werden konnten [35].



# Abkürzungsverzeichnis

<i>ABM</i> .....	Agenten-basiertes Modell
<i>ABS</i> .....	Agenten-basierte Simulation
<i>AM</i> .....	Menge der Agentenmodelle eines ABM
<i>AMG</i> .....	Agentenmodellgruppe: Teilmenge der Agentenmodelle einer <i>AMK</i> mit homogenen Parameterwerten
<i>AMGF</i> .....	Modellfamilie einer AMG
<i>AMK</i> .....	Agentenmodellklasse: Teilmenge der Agentenmodelle eines <i>ABM</i> in der alle Agenten zur gleichen Modellfamilie gehören, d.h. in der alle Agenten die gleiche Struktur, gleiche Variablen und Parameter haben. Die Parameterwerte der Agenten können unterschiedlich sein
<i>AMKF</i> .....	Modellfamilie einer AMK
<i>AR</i> .....	Eine Menge von Änderungsfunktionen, die aus einer gegebenen Parameterbelegung <i>PW</i> und der zugehörigen Validitätsbewertung des Simulationsverhaltens bezüglich <i>VK</i> eine geänderte Parameterbelegung <i>PW</i> berechnen, die als nächstes getestet wird
<i>BewFkt</i> .....	Verrechnungsfunktion als internes Bewertungsmodell für die Agenten in der Fallstudie der Einkaufssimulation
<i>GA</i> .....	Genetischer Algorithmus: Ein Black-Box-Optimierungsverfahren
<i>GZF</i> .....	Globale Zielfunktion zur Bewertung der Validität des Gesamtverhaltens einer Simulation auf der Makro-Ebene
<i>H<sub>g</sub></i> .....	Zuordnungsfunktion bei der Kalibrierung mit Heterogenitätssuche. <i>H<sub>g</sub></i> bildet einzelne konkrete Agentenmodelle aus der Menge einer <i>AMK</i> auf einen Indexwert ab. Agenten mit gleichem Indexwert werden einer gemeinsamen <i>AMG</i> zugeordnet
<i>HP</i> .....	Heterogenitätsproblem: Um eine Heterogenitätssuche erweitertes Kalibrierungsproblem
<i>IAM</i> .....	Modell eines einzelnen Agenten
<i>IAMF</i> .....	Familie individueller Agentenmodelle
<i>IPA</i> .....	Plausibilitätsbewertung für das Verhalten eines Teilmodells
<i>IRB</i> .....	Randbedingung an das Verhalten eines Teilmodells
<i>IVK</i> .....	Validierungskriterien bezüglich des Verhaltens eines individuellen Agentenmodells
<i>IVM</i> .....	Modell idealen Verhaltens eines Teilmodells
<i>IVMF</i> .....	Familie von IVM
<i>IZF</i> .....	Ideale Zielfunktion für ein Teilmodell basierend auf dem zugehörigen <i>IVM</i>

$KP$ .....	Kalibrierungsproblem: Ein Kalibrierungsproblem $KP$ ist durch eine zu kalibrierende Modellfamilie $MF$ und eine Menge von Validierungskriterien $VK$ gegeben
$KP_{ABS}$ .....	Kalibrierungsproblem für Agenten-basierte Simulationen
$KP_{EK}$ .....	Kalibrierungsproblem der Fallstudie der Einkaufssimulation
$KP_{MAKRO_{ABS}}$	Kalibrierungsteilproblem auf der Makroebene im Makro-Mikro-Verfahren
$KP_{MIKRO_{ABS}}$ .	Kalibrierungsteilproblem auf der Mikroebene im Makro-Mikro-Verfahren
$M$ .....	Simulationsmodell: $M = (S, V, P, PW)$
$MF$ .....	Modellfamilie: $MF = (S; V; P)$ . Eine Modellfamilie $MF$ umfasst die Menge aller Simulationsmodelle $M_i$ mit gleichen Modellstrukturen $S$ , Parameter $P$ und gleichen Modellvariablen $V$
$P$ .....	Der Vektor der Modellparameter.
$P_i$ .....	(Bezeichner; $W(P_i)$ ): Einer der Modellparameter, beschrieben durch einen Bezeichner (symbolischer Name) und einen Wertebereich $W(P_i)$
$PW$ .....	Parameterwerte: Vektor zugehöriger Wertbelegungen für Modellparameter $P$
$RB$ .....	Randbedingung(en) zur Validitätsbewertung von Simulationsverhalten
$RE$ .....	Reverse-Engineering-Verfeinerung des Basis-Makro-Mikro-Verfahrens
$S$ .....	Struktur eines Simulationsmodells
$SA$ .....	Stochastic Approximation: Ein Black-Box-Optimierungsverfahren
$SAN$ .....	Simulated Annealing: Ein Black-Box-Optimierungsverfahren
$SRB$ .....	Als sicher korrekt errachtete Menge von Randbedingungen
$SV_M$ .....	Modellzustandstrajektorie: Eine zeitlich geordnete Menge an Zuständen, die das Modell $M$ während eines Simulationslaufes durchläuft
$SVK$ .....	Als sicher korrekt errachtete Teile der Validierungskriterien
$TKP$ .....	Teilkalibrierungsproblem eines $KP$
$TKP_{AM}$ .....	Teilkalibrierungsproblem auf der Menge der Agentenmodelle
$TKP_{IAM}$ .....	Teilkalibrierungsproblem bezüglich eines individuellen Agentenmodells
$TKP_{UM}$ .....	Teilkalibrierungsproblem für das Umweltmodell
$TM$ .....	Teilmodell eines Modells
$TMF$ .....	Teilmodellfamilie einer $MF$
$U$ .....	Umwelt eines Systems mit Agenten
$UM$ .....	Umweltmodell
$UMF$ .....	Familie von Umweltmodellen
$URB$ .....	Als unsicher, d.h. potentiell falsch, errachtete Menge von Randbedingungen
$UVK$ .....	Als unsicher, d.h. potentiell falsch, errachtete Teile der Validierungskriterien
$V$ .....	Der Vektor der Modellvariablen
$VK$ .....	Validierungskriterien: Die Bewertungsaspekte, auf deren Basis die Validierung des Simulationsverhaltens durchgeführt wird. Ein Validierungskriterium bewertet immer ein Teilmodell $TM$ mit Hilfe einer Validierungstechnik $VT$

$VK_i$ .....	$i$ -tes Validierungskriterium
$VT$ .....	Validierungstechnik: Dies können Zielfunktionen $ZF$ oder Randbedingungen $RB$ sein
$VW$ .....	Variablenwerte: Vektor zugehöriger Wertbelegungen für Modellvariablen $V$
$W(P_i)$ .....	Der Wertebereich des Parameters $P_i$
$W(V_i)$ .....	Der Wertebereich des $i$ -ten Modellvariablen
$Z_M$ .....	Die Menge aller möglichen Zustände eines kalibrierten Modells $M$ . Einen Zustand $z$ eines Modells $M$ erhält man durch Festlegung der Werte für die Modellvariablen $V$ von $M$
$ZF$ .....	Zielfunktion(en): Mit Hilfe einer Zielfunktion der Simulation kann der Validitätsgrad des Simulationsverhalten auf einen numerischen Wert abgebildet werden





# Literaturverzeichnis

- [1] R. Akcelik and M. Besley. Microsimulation and analytical methods for modelling urban traffic. 2001.
- [2] Vito Albino, Nunzia Carbonara, and Ilaria Giannoccaro. Innovation in industrial districts: An agent-based simulation model. *International Journal of Production Economics*, Volume 104, Issue 1, pages 30–45, 2006.
- [3] Sigrun Andradottir. A review of simulation optimization techniques. In *WSC '98: Proceedings of the Winter Simulation Conference*, pages 151–158. IEEE Computer Society, 1998.
- [4] E. Angun, J. P. C. Kleijnen, D. D. Hertog, and G. Gurkan. Response surface methodology revisited. In *WSC '02: Proceedings of the 2002 Winter Simulation Conference (WSC'02) - Volume 1*, pages 377–383, Washington, DC, USA, 2002. IEEE Computer Society.
- [5] Jay April, Fred Glover, James P. Kelly, and Manuel Laguna. Simulation-based optimization: practical introduction to simulation optimization. In *WSC '03: Proceedings of the Winter Simulation Conference*, pages 71–78, 2003.
- [6] Martin Atzmueller. *Knowledge-Intensive Subgroup Mining – Techniques for Automatic and Interactive Discovery*, volume 307 of *Dissertations in Artificial Intelligence-Infix (Diski)*. IOS Press, March 2007.
- [7] Robert Axelrod. A model of the emergence of new political actors. Working Papers 93-11-068, Santa Fe Institute, November 1993.
- [8] Robert L. Axtell, Joshua M. Epstein, Jeffrey S. Dean, George J. Gumerman, Alan C. Swedlund, Jason Harburger, Shubha Chakravarty, Ross Hammond, Jon Parker, and Miles Parker. Population growth and collapse in a multiagent model of the kayenta anasazi in long house valley. *PNAS*, 99(90003):7275–7279, 2002.
- [9] F. Azadivar and G. Tompkins. Simulation optimization with qualitative variables and structural model changes: A genetic algorithm approach. *European Journal of Operational Research*, 113(1):169–182, February 1999.
- [10] Farhad Azadivar. Simulation optimization methodologies. In *WSC '99: Proceedings of the Winter Simulation Conference*, pages 93–100, 1999.
- [11] Osman Balci. Principles and techniques of simulation validation, verification, and testing. In *WSC '95: Proceedings of the 27th conference on Winter simulation*, pages 147–154, 1995.

- [12] Philip S. Barry and Matthew T. K. Koehler. Leveraging agent based simulation for rapid course of action development. In *WSC '05: Proceedings of the 37th conference on Winter simulation*, pages 1001–1007. Winter Simulation Conference, 2005.
- [13] Michel Bierlaire and Michaël Thémans. Algorithmic developments for the estimate of advanced discrete choice models. In *Proceedings of the 5th Swiss Transportation Research Conference*, Ascona, Switzerland, 2005.
- [14] Riccardo Boero and Flaminio Squazzoni. Does empirical embeddedness matter? Methodological Issues on Agent-Based Models for Analytical Social Science. *Journal of Artificial Societies and Social Simulation*, 8(4), 2005.
- [15] Justin Boesel, Royce Bowden, Fred Glover, James P. Kelly, and Erik Westwig. Panel: simulation optimization: future of simulation optimization. In *WSC '01: Proceedings of the Winter Simulation Conference*, pages 1466–1469, 2001.
- [16] Eric Bonabeau. Agent-based modeling: Methods and techniques for simulating human systems. In *Proceedings of the National Academy of Science*, volume 99, pages 7280–7287. National Academy of Science, 2002.
- [17] Eric Bonabeau, Guy Theraulaz, and Marco Dorigo. *Swarm Intelligence. From Natural to Artificial Systems*. Oxford University Press Inc, USA, 1999.
- [18] Hartmut Bossel. *Modellbildung und Simulation - Konzepte, Verfahren und Modelle zum Verhalten dynamischer Systeme*. Vieweg Verlag, Braunschweig/Wiesbaden 1994, 1994.
- [19] A. G. Brandstein and G. E. Horne. Data farming: A meta-technique for research in the 21st century. *Maneuver Warfare Science*, pages 93–99, 1998.
- [20] Thomas Brenner and Claudia Werker. Empirical calibration of simulation models. *Computing in Economics and Finance 2004*. 89, Society for Computational Economics., August 2004. available at <http://ideas.repec.org/p/sce/scecf4/89.html>.
- [21] Richard H. Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comput.*, 16(5):1190–1208, 1995.
- [22] Benoit Calvez and Guillaume Hutzler. Automatic tuning of agent-based models using genetic algorithms. In Luis Antunes and Jaime Semaio Sichman, editors, *Proceedings of the 6th International Workshop on Multi-Agent-Based Simulation 2005*, 2005.
- [23] Benoit Calvez and Guillaume Hutzler. Ant Colony Systems and the Calibration of Multi-Agent Simulations: a New Approach. In *Multi-Agents for modelling Complex Systems (MA4CS'07)*, 2007.
- [24] Andre M. C. Campos, Anne M. P. Canuto, and Jorge H. C. Fernandes. Towards a Methodology for Developing Agent-Based Simulations: The MASim Methodology. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi-agent Systems - Volume 3*. IEEE Computer Society, Washington, DC, USA, 2004.

- [25] Lianyu Chu, H.X. Liu, Jun-Seok Oh, and W. Recker. A calibration procedure for microscopic traffic simulation. In *Intelligent Transportation Systems, 2003. Proceedings. 2003 IEEE*, pages 1574–1579. IEEE Computer Society, 2003.
- [26] Thomas M. Cioppa, Thomas W. Lucas, and Susan M. Sanchez. Military applications of agent-based simulations. In *WSC '04: Proceedings of the Winter Simulation Conference*, pages 171–180, 2004.
- [27] François E. Cellier. *Continuous System Modeling*. Springer Verlag New York Inc., 1991.
- [28] J. Cote, S. Gravel, A. Methot, A. Patoine, M. Roch, and A. Staniforth. The operational CMC-MRB global environmental multiscale (GEM) model. Part 1: design considerations and formulation. *Monthly Weather Review*, 126:1373—1395, 1998.
- [29] Vince Darley. Emergent phenomena and complexity. In R. Brooks and P. Maes, editors, *Artificial Life IV, Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems*, pages 411–416. MIT Press, 1994.
- [30] Jim Doran. *Agent-Based Modelling of Ecosystems for Sustainable Resource Management*, pages 383–403. Springer-Verlag Berlin Heidelberg, 2001.
- [31] Anna Dornhaus, Franziska Kluegl, Frank Puppe, and Jürgen Tautz. Task selection in honey bees - experiments using multi-agent simulation. In *Proceedings of the GWAL'98, Bochum*. Verlag Harry Deutsch AG, 1998.
- [32] Darren T. Drewry, Paul F. Reynolds Jr., and William R. Emanuel. Optimization and response surfaces: an optimization-based multi-resolution simulation methodology. In *WSC '02: Proceedings of the Winter Simulation Conference.*, pages 467–475, 2002.
- [33] Alexis Drogoul, Diane Vanbergue, and Thomas Meurisse. Multi-Agent Based Simulation: Where are the Agents? In *Proceedings of the Workshop on Multi-Agent-Based Simulation 2002*, 2002.
- [34] Giorgio Fagiolo, Paul Windrum, and Alessio Moneta. Empirical validation of agent-based models: A critical survey. LEM Working Paper 2006/14, Laboratory of Economics and Management, Sant'Anna School of Advanced Studies, Pisa, Italy, 2006.
- [35] Manuel Fehler, Marko Kleinhenz, Franziska Klügl, Frank Puppe, and Jürgen Tautz. Caps and gaps: a computer model for studies on brood incubation strategies in honeybees (*apis mellifera carnica*). *Naturwissenschaften*, 2007.
- [36] Manuel Fehler, Franziska Klügl, and Frank Puppe. Approaches for resolving the dilemma between model structure refinement and parameter calibration in agent-based simulations (extended version). In *EUMAS '05: Proceedings of the Third European Workshop on Multiagent Systems*, pages 148–1160, 2005.
- [37] Manuel Fehler, Franziska Klügl, and Frank Puppe. Techniques for Analysis and Calibration of Multi-agent Simulations. In M.-P. Gleizes, A. Omicini, and Franco Zambonelli, editors, *ESAW 04, LNAI 3451*, pages 305–321. Springer-Verlag Berlin Heidelberg 2005, 2005.

- [38] Manuel Fehler, Franziska Klügl, and Frank Puppe. Approaches for resolving the dilemma between model structure refinement and parameter calibration in agent-based simulations. In *AAMAS '06: Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 120–122, 2006.
- [39] Jacques Ferber. *Multi-Agent Systems - An Introduction to Distributed Artificial Intelligence*. Addison Wesley Longman Limited, 1999.
- [40] Paul Fishwick. *Simulation Model Design and Execution: Building Digital Worlds*. Prentice Hall, 1995.
- [41] Frederick K. Frantz. A taxonomy of model abstraction techniques. In *WSC '95: Proceedings of the 27th conference on Winter simulation*, pages 1413–1420, 1995.
- [42] Henrik Friman and Gary E. Horne. Using agent models and data farming to explore network centric operations. In *WSC '05: Proceedings of the 37th conference on Winter simulation*, pages 1094–1100. Winter Simulation Conference, 2005.
- [43] Michael C. Fu. A tutorial review of techniques for simulation optimization. In *WSC '94: Proceedings of the 26th conference on Winter simulation*, pages 149–156, San Diego, CA, USA, 1994. Society for Computer Simulation International.
- [44] Michael C. Fu. Simulation optimization. In *WSC '01: Proceedings of the 33rd conference on Winter simulation*, pages 53–61, Washington, DC, USA, 2001. IEEE Computer Society.
- [45] Michael C. Fu. Feature article: Optimization for simulation: Theory vs. practice. *INFORMS J. on Computing*, 14(3):192–215, 2002.
- [46] Michael C. Fu, Sigrún Andradóttir, John S. Carson II, Fred Glover, Charles R. Harrell, Yu-Chi Ho, James P. Kelly, and Stephen M. Robinson. Integrating optimization and simulation: research and practice. In *WSC '00: Proceedings of the Winter Simulation Conference.*, pages 610–616, 2000.
- [47] Michael C. Fu, Fred Glover, and Jay April. Simulation optimization: a review, new developments, and applications. In *WSC '05: Proceedings of the Winter Simulation Conference*, pages 83–95, 2005.
- [48] Erich Gamma, Richard Helm, Ralph E. Johnson, and John M. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1995.
- [49] Nigel Gilbert and Pietro Terna. How to build and use agent-based models in social science. *Mind & Society*, 1(1):57–72, 2000.
- [50] Nigel Gilbert and Klaus G. Troitzsch. *Simulation for the Social Scientist, 2nd Edition*. Open University Press, 2005.
- [51] M. Gilli and P. Winkler. A global optimization heuristic for estimating agent based models. *Computational Statistics and Data Analysis*, 42(3):299–312, 2003.
- [52] Fred Glover, Manuel Laguna, and Rafael Martí. Scatter search. pages 519–537, 2003.
- [53] David E. Goldberg. *Genetic Algorithms - in Search Optimization & Machine Learning*. Addison Wesley Publishing Company Inc., 1989.

- [54] M. Hare and P. Deadman. Further towards a taxonomy of agent-based simulation models in environmental management. *Mathematics and Computers in Simulation, Volume 64, Issue 1*, pages 25–40, 2004.
- [55] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.
- [56] Brian Heath, Raymond Hill, and Frank Ciarallo. A survey of agent-based modeling practices (january 1998 to july 2008). *Journal of Artificial Societies and Social Simulation*, 12:4, 2009.
- [57] Alexander Helleboogh, Guiseepe Vizzari, Adelinde M. Uhrmacher, and Fabien Michel. Modeling dynamic environments in multi-agent simulation. *Journal of Autonomous Agents and Multi-Agent Systems*, 14(1):87–116, 2007.
- [58] Rainer Herrler. *Agentenbasierte Simulation zur Ablaufoptimierung in Krankenhäusern und anderen verteilten, dynamischen Umgebungen*. PhD thesis, Julius-Maximilians-Universität Würzburg, 2007.
- [59] Frank Hoffmeister and Thomas Baeck. Genetic algorithms and evolution strategies - similarities and differences. In *PPSN I: Proceedings of the 1st Workshop on Parallel Problem Solving from Nature*, pages 455–469, London, UK, 1991. Springer-Verlag.
- [60] Marko Hofmann. On the complexity of parameter calibration in simulation models. *JDMS: The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 2(5):217–226, 2005.
- [61] John Henry Holland. *Emergence: from chaos to order*. Addison-Wesley, 1998.
- [62] Holger Hoos and Thomas Stützle. *Stochastic Local Search - Foundations and Applications*. Elsevier Inc., 2005.
- [63] Andy Ilachinski. Irreducible semi-autonomous adaptive combat (isaac): an artificial life approach to land combat. *Military Operations Research*, 5, pages 29–46, 2000.
- [64] L. Ingber. Simulated annealing: Practice versus theory. *Mathl. Comput. Modelling*, 18(11):29–57, 1993.
- [65] Yumi Iwasaki and Alon Y. Levy. Automated model selection for simulation. In *National Conference on Artificial Intelligence*, pages 1183–1190, 1994.
- [66] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.
- [67] Matthew H. Jones and Jr. K. Preston White. Stochastic approximation with simulated annealing as an approach to global discrete-event simulation optimization. In *WSC '04: Proceedings of the 36th conference on Winter simulation*, pages 500–507. Winter Simulation Conference, 2004.
- [68] Leslie Pack Kaelbling, Michael L. Littman, and Andrew P. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

- [69] L. Kaufman and P.J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, New York, 1990.
- [70] James Kennedy and Russel Eberhart. Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks*, pages 1942–1948. IEEE International Conference, 1995.
- [71] Jumi Kim and S. Olafsson. Two-stage np method with inheritance. In *WSC '02: Proceedings of the 2002 Winter Simulation Conference (WSC'02) - Volume 1*, pages 279–284, Washington, DC, USA, 2002. IEEE Computer Society.
- [72] Seong-Hee Kim and Barry L. Nelson. Selecting the best system: selecting the best system: theory and methods. In *WSC '03: Proceedings of the 35th conference on Winter simulation*, pages 101–112. Winter Simulation Conference, 2003.
- [73] Seong-Hee Kim and Barry L. Nelson. *Chapter 17 Selecting the Best System*, pages 501–534. Elsevier B.V., 2006.
- [74] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, Number 4598, 13 May 1983, 220, 4598:671–680, 1983.
- [75] F. Klügl, R. Herrler, and M. Fehler. Sesam: implementation of agent-based simulation using visual programming. In *AAMAS '06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 1439–1440, New York, NY, USA, 2006. ACM Press.
- [76] Franziska Kluegl. *Multiagentensimulation Note: In German*. Addison-Wesley Verlag, 2001.
- [77] Franziska Klügl. A validation methodology for agent-based simulations. In *SAC '08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 39–43, New York, NY, USA, 2008. ACM.
- [78] Franziska Klügl, Manuel Fehler, and Rainer Herrler. About the Role of the Environment in Multi-agent Simulations. In Danny Weyns, H. Van Dyke Parunak, and Fabien Michel, editors, *Environments for Multi-Agent Systems: First International Workshop, E4MAS 2004, New York, NY, July 19, 2004, Revised Selected Papers*. Springer-Verlag GmbH, 2004.
- [79] Franziska Klügl-Frohnmayr. *Agent-Based Simulation Engineering. Habilitation Thesis*. Faculty of Mathematics and Computer Science, University of Würzburg, 2009.
- [80] Averill M. Law and W. David Kelton. *Simulation Modeling and Analysis, 3rd Edition*. The McGraw-Hill Companies, 2000.
- [81] M. Locatelli. On the multilevel structure of global optimization problems. *Comput. Optim. Appl.*, 30(1):5–22, 2005.
- [82] Thorsten G. Müller. *Doktorarbeit: Modeling complex systems with differential equations*. Fakultät für Mathematik und Physik, Albert-Ludwigs-Universität Freiburg, 2002.

- [83] Youssef Monsef. *Modelling and Simulation of Complex Systems - Concepts, Methods and Tools*. Society for Computer Simulation International, San Diego, Erlangen, Ghent, Budapest, 1997.
- [84] Douglas C. Montgomery. *Design and Analysis of Experiments*. John Wiley & Sons, Inc., 1997.
- [85] Robin P. Nicolai, Rommert Dekker, Nanda Piersma, and Gerrit J. van Oortmarssen. Automated response surface methodology for stochastic optimization models with unknown variance. In *WSC '04: Proceedings of the 36th conference on Winter simulation*, pages 491–499. Winter Simulation Conference, 2004.
- [86] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer Verlag New York, Inc., 2000.
- [87] Christoph Oechslein. *A Process Model with Integrated Specification- and Implementation Language for Multi-Agent Simulation. Note: In German*. Shaker Verlag GmbH, 2004.
- [88] S. Olafsson and Jumi Kim. Towards a framework for black-box simulation optimization. In *WSC '01: Proceedings of the 33th conference on Winter simulation*, pages 300–306. Winter simulation Conference, 2001.
- [89] Sigurdur Olafsson. Two-stage nested partitions method for stochastic optimization. *Methodology and Computing in Applied Probability*, 6(1):5–27, 2004.
- [90] Liviu Panait and Sean Luke. Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, 11(3):387–434, 2005.
- [91] Panayiotou, Cassandras, and Gong. Model Abstraction for Discrete Event Systems Using Neural Networks and Sensitivity Information. In *Proceedings of the Winter Simulation Conference 2000*, 2000.
- [92] J. L. Paris, L. Tautou-Guillaume, and H. Pierreval. Dealing with design options in the optimization of manufacturing systems: an evolutionary approach. *International Journal of Production Research*, 39(6):1081–1094, April 2001.
- [93] Jutta Pichitlamken and Barry L. Nelson. A combined procedure for optimization via simulation. *ACM Trans. Model. Comput. Simul.*, 13(2):155–179, 2003.
- [94] H. Pierreval, C. Caux, J. L. Paris, and F. Viguier. Evolutionary approaches to the design and organization of manufacturing systems. *Comput. Ind. Eng.*, 44(3):339–364, 2003.
- [95] Henri Pierreval and Jean Luc Paris. From 'simulation optimization' to 'simulation configuration' of systems. *Simulation Modelling - Practice and Theory* 11, Issue 1, pages 5–19, 2003.
- [96] Jim Pugh and Alcherio Martinoli. Multi-robot learning with particle swarm optimization. In *AAMAS '06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 441–448, New York, NY, USA, 2006. ACM Press.
- [97] Ingo Rechenberg. *Evolutionstrategie '94*. Frommann-Holzboog, Stuttgart, 1994.

- [98] Stuart Russel and Peter Norvig. *Artificial Intelligence - A Modern Approach*. Pearson Education, Inc., 2003.
- [99] Lamjed Ben Said, Thierry Bouron, and Alexis Drogoul. Agent-based interaction analysis of consumer behavior. In *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 184–190, New York, NY, USA, 2002. ACM Press.
- [100] M. Sambridge. Geophysical inversion with a neighbourhood algorithm –ii. *Geophys. J Int.*, 138:727–746, 1999.
- [101] M. Sambridge. Geophysical inversion with a neighbourhood algorithm—i. searching a parameter space. *Geophys. J Int.*, 138:479, 1999.
- [102] Susan M. Sanchez and Thomas W. Lucas. Exploring the world of agent-based simulations: simple models, complex analyses: exploring the world of agent-based simulations: simple models, complex analyses. In *WSC '02: Proceedings of the 34th conference on Winter simulation*, pages 116–126. Winter Simulation Conference, 2002.
- [103] Robert G. Sargent. Validation and verification of simulation models. In *Winter Simulation Conference*, pages 104–114, 1999.
- [104] Tilman Schenk, Franziska Klügl, Günter Löffler, Frank Puppe, and Rauh Jürgen. Multiagentensysteme zur Simulation von Konsumentenverhalten. *Hallesches Jahrbuch der Geowissenschaften*, pages 117–130, 2004.
- [105] Tilman A. Schenk, Jürgen Rauh, Manuel Fehler, Franziska Klügl, and Frank Puppe. *Der simulierte Konsument*. Deutschen Gesellschaft für Geographie, 2006.
- [106] Carles Sierra, Jordi Sabater, Jaume Agustí-Cullell, and Pere Garcia. Evolutionary programming in SADDE. In *AAMAS '02: Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1270–1271. 2003.
- [107] J. C. Spall. Overview of the simultaneous perturbation method for efficient optimization. *Hopkins APL Technical Digest.*, 19:482–492, 1998.
- [108] James C. Spall. Stochastic optimization and the simultaneous perturbation method. In *WSC '99: Proceedings of the Winter Simulation Conference.*, pages 101–109, 1999.
- [109] N. Subramanian, A. Yajnik, and R. S. R. Murthy. Artificial neural network as an alternative to multiple regression analysis in optimizing formulation parameters of cytarabine liposomes. *APS PharmSciTech.*, 5(1), 2004.
- [110] Richard S. Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44, 1988.
- [111] James R. Swisher, Paul Hyden, Sheldon H. Jacobson, and Lee Schruben. Simulation optimization: a survey of simulation optimization techniques and procedures. In *WSC '00: Proceedings of the Winter Simulation Conference*, pages 119–128, 2000.
- [112] James R. Swisher, Sheldon H. Jacobson, and Enver Yücesan. Discrete-event simulation optimization using ranking, selection, and multiple comparison procedures: A survey. *ACM Trans. Model. Comput. Simul.*, 13(2):134–154, 2003.



- [113] Aimo Toern and Sami Viitanen. Topographical global optimization using pre-sampled points. *Journal of Global Optimization.*, 5(3):267–276, 1994.
- [114] Aimo Törn, Montaz Ali, and Sami Viitanen. Stochastic global optimization: Problem classes and solution techniques. *Journal of Global Optimization*, 14(4):437–447, 1999.
- [115] Klaus G. Troitzsch. Validating simulation models. In *18th European Simulation Multi-conference*, 2004.
- [116] K. Tumer and D. Wolpert. A survey of collectives. In K. Tumer and D. Wolpert, editors, *Collectives and the Design of Complex Systems*, pages 1–42. Springer, 2004.
- [117] Kagan Tumer and David Wolpert. Collective intelligence and braess' paradox. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 104–109. AAAI Press / The MIT Press, 2000.
- [118] A. M. Uhrmacher. Dynamic structures in modeling and simulation: a reflective approach. *ACM Trans. Model. Comput. Simul.*, 11(2):206–232, 2001.
- [119] Adelinde Uhrmacher, Daniela Degenring, and Bernard P. Zeigler. Discrete event multi-level models for systems biology. *T. Comp. Sys. Biology*, 1:66–89, 2005.
- [120] Rosemary H. Wild and Jr. Joseph J. Pignatiello. Finding stable system designs: a reverse simulation technique. *Commun. ACM*, 37(10):87–98, 1994.
- [121] David Wolpert and Kagan Tumer. Optimal payoff functions for members of collectives. *Advances in Complex Systems*, 4(2–3):265–279, 2001.
- [122] David H. Wolpert and William G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, April 1997.
- [123] Bernhard Zeigler. *Theory of Modelling and Simulation*. Malabar, Krieger, 1976.