

Optimierung und Simulation

Von

Dr. Jörg Biethahn

o. Professor für Wirtschaftsinformatik
Georg-August-Universität Göttingen

Dipl. Kfm. Andreas Lackner

Georg-August-Universität Göttingen

und

Dipl. Kfm. Michael Range

Georg-August-Universität Göttingen

unter Mitwirkung von

Dipl. Kfm. Ole Brodersen

Georg-August-Universität Göttingen

R. Oldenbourg Verlag München Wien

Inhaltsverzeichnis

Inhaltsverzeichnis	III
Abbildungsverzeichnis.....	IX
Tabellenverzeichnis.....	XIII
Abkürzungsverzeichnis	XVII
1 Einleitung.....	1
2 Vorgehensweise und traditionelle Verfahren für die zielfunktionale Simulation	5
2.1 Simulationsanalysen im entscheidungstheoretischen Zusammenhang	5
2.2 Simulation als Methode der Optimierung	14
2.3 Vergleich von Alternativen	18
2.4 Traditionelle Optimierungsverfahren für die zielfunktionale Simulation	25
2.4.1 Eindimensionale Suchstrategien.....	25
2.4.1.1 Einschachtelung des Minimums	25
2.4.1.2 Fibonacci-Suche	26
2.4.1.3 Teilung nach dem Goldenen Schnitt.....	28
2.4.1.4 Parabolinterpolationsstrategie.....	28

2.4.2	Direkte Suchstrategien.....	28
2.4.2.1	Einzelfaktormethode.....	29
2.4.2.2	Spiegelverfahren.....	30
2.4.2.3	Complex-Methode von M. J. Box	31
2.4.2.4	Pattern Search.....	32
2.4.2.5	Strategie der rotierenden Koordinaten und DSC-Strategie.....	33
2.4.3	Gradientenmethoden.....	34
2.4.3.1	Verfahren zur Schätzung von Gradienten.....	35
2.4.3.2	Schätzung über Differenzenquotienten.....	35
2.4.3.3	Perturbation Analysis	36
2.4.4	Quasi-Newton-Strategien	36
2.4.5	Regressionsmethoden.....	38
2.4.5.1	Klassischer Ansatz.....	38
2.4.5.2	Response Surface Methodology (Reaktionsoberflächen-Methode)	39
2.4.6	Vergleich der traditionellen Verfahren.....	41
3	Metaheuristische Verfahren für die zielfunktionale Simulation.....	43
3.1	Tabu Search.....	44
3.1.1	Generierungs- und Selektionsstrategien	47
3.1.2	Tabulistenmanagement.....	48
3.1.3	Aspirationskriterien	51
3.1.4	Diversifikation und Intensivierung	52

3.2	Evolutionäre Algorithmen.....	53
3.2.1	Genetische Algorithmen.....	54
3.2.1.1	Codierung.....	62
3.2.1.2	Selektion.....	64
3.2.1.3	Mutation.....	67
3.2.1.4	Crossover.....	68
3.2.2	Evolutionäre Strategien.....	75
3.2.2.1	Mutation.....	80
3.2.2.2	Rekombination.....	82
3.2.2.3	Selektion.....	84
3.2.3	Genetische Programmierung.....	84
3.2.3.1	Mutation.....	93
3.2.3.2	Automatisch definierte Funktionen (ADF).....	93
3.3	Simulated Annealing.....	95
3.3.1.1	Allgemeine Einstellungen.....	101
3.3.1.2	Problemspezifische Einstellungen.....	103
3.3.1.3	Erweiterungen.....	105
3.4	Ameisenalgorithmus.....	108
3.4.1	Ant Colony Optimization (ACO-Metaheuristik).....	112
3.5	Vergleich ausgewählter Optimierungsverfahren.....	119
3.5.1	Anmerkungen zum Vergleich von Optimierungsverfahren.....	119
3.5.2	Vergleich ausgewählter klassischer Optimierungsverfahren nach J. Biethahn.....	122
3.5.2.1	Verwendetes Testproblem.....	122
3.5.2.2	Getestete Verfahren.....	125
3.5.2.3	Ergebnisse.....	125

3.5.3	Vergleich metaheuristischer Verfahren nach V. Nissen	126
3.5.3.1	Verwendetes Testproblem	127
3.5.3.2	Getestete Verfahren	127
3.5.3.3	Ergebnisse.....	128
3.5.4	Vergleich nach D. E. Smith	130
3.5.4.1	Getestete Verfahren	131
3.5.4.2	Kriterien zur Auswahl der Testfunktionen	132
3.5.4.3	Ergebnisse.....	133
3.5.5	Vergleich nach H.-P. Schwefel.....	135
3.5.5.1	Getestete Verfahren	135
3.5.5.2	Ergebnisse.....	137
3.5.6	Vergleich nach D. Brüne	142
3.5.6.1	Verwendete Testfunktion	142
3.5.6.2	Getestete Verfahren	142
3.5.6.3	Ergebnisse.....	143
4	Anwendung ausgewählter Optimierungsverfahren auf das Biethahn'sche Lagerhaltungsmodell	147
4.1	Simulationssoftware Arena als Basis	147
4.2	Implementierung des Lagerhaltungsproblems in die Simulationssoftware Arena	151
4.2.1	Parameterkonstellation I (Biethahn'sches Lagerhaltungsmodell).....	153
4.2.2	Parameterkonstellation II (Modifikation des Biethahnschen Lagerhaltungsmodells)	155
4.3	Implementierung der Parameterkonstellationen des Modells in Arena.....	155
4.3.1	Gesamtmodell.....	157
4.3.2	Arena Modell.....	158

4.3.3	Interne Modelllogik	160
4.3.4	Entwicklung der ausgewählten Algorithmen.....	165
4.3.4.1	Local Search	166
4.3.4.2	Gradientenverfahren	167
4.3.4.3	Simulated Annealing	168
4.3.4.4	Genetischer Algorithmus	169
4.3.4.5	Tabu Search	174
4.3.4.6	Modifizierter Ameisenalgorithmus mit konstanter Populationsgröße.....	175
4.3.4.7	Modifizierter Ameisenalgorithmus mit linearem Wachstum der Population.....	176
5	Darstellung und Beurteilung der Ergebnisse der Anwendung.....	177
5.1	Enumeration	177
5.2	Gradientenverfahren.....	180
5.3	Simulated Annealing.....	182
5.4	Genetischer Algorithmus	184
5.5	Tabu Search.....	189
5.5.1	Einfaches Tabu Search	189
5.5.2	Tabu Search mit 97% Aspirationskriterium	192
5.5.3	Tabu Search mit 100% Aspirationskriterium	195
5.6	Modifizierter Ameisenalgorithmus mit konstanter Populationsgröße.....	198
5.7	Modifizierter Ameisenalgorithmus mit linearem Wachstum der Population	202

5.8 Vergleichende Beurteilung der untersuchten Algorithmen	205
5.8.1 Parameterkonstellation I	205
5.8.2 Parameterkonstellation II	212
5.8.3 Zusammenführung der Ergebnisse	218
6 Zusammenfassung und Ausblick	223
Anhang A: Ergebnistabellen	227
Anhang B: Visual Basic Quelltext	249
Interne Modelllogik	249
Dialoge	258
Records	260
Gradientenverfahren	261
Simulated Annealing	268
Genetischer Algorithmus	274
Tabu-Suche (einfach)	280
Tabu-Suche mit Aspirationskriterien	289
Schwarm-Algorithmus mit linearem Wachstum	298
Literaturverzeichnis	305