

Schriften zur quantitativen Betriebswirtschaftslehre und
Wirtschaftsinformatik

herausgegeben von Prof. Dr. Stefan Voß

Lutz Sondergeld

**Performance analysis methods
for heuristic search optimization with
an application to cooperative agent algorithms**

Shaker Verlag
Aachen 2001

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Sondergeld, Lutz:

Performance analysis methods for heuristic search optimization with an application to cooperative agent algorithms / Lutz Sondergeld.

Aachen: Shaker, 2001

(Schriften zur quantitativen Betriebswirtschaftslehre und
Wirtschaftsinformatik; Bd. 2)

Zugl.: Braunschweig, Techn. Univ., Diss., 2000

ISBN 3-8265-8203-9

Copyright Shaker Verlag 2001

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the publishers.

Printed in Germany.

ISBN 3-8265-8203-9

ISSN 1616-1920

Shaker Verlag GmbH • P.O. BOX 1290 • D-52013 Aachen

Phone: 0049/2407/9596-0 • Telefax: 0049/2407/9596-9

Internet: www.shaker.de • eMail: info@shaker.de

Contents

List of symbols.....	vii
Preface.....	1
<u>Part I</u>	
1 Introduction.....	3
2 Definition of terms.....	4
3 Algorithmic properties.....	8
3.1 Accuracy.....	9
3.1.1 Reference values.....	9
3.1.2 Measurement indices.....	10
3.1.3 Evaluation of heuristic accuracy.....	14
3.1.3.1 Analytical methods.....	14
3.1.3.2 Empirical analysis.....	16
3.1.3.3 Probabilistic analysis.....	16
3.2 Computational effort.....	17
3.2.1 Classification of effort.....	17
3.2.2 Measuring computational effort.....	18
3.2.2.1 Direct measurement.....	18
3.2.2.2 Implicit measurement by computation counts.....	19
3.2.3 Complexity analysis.....	20
3.3 Efficiency.....	20
3.4 Robustness.....	21
3.5 Adaptability.....	22
3.6 Reliability.....	22
3.7 Ease-of-use.....	22
4 Experiments.....	23
4.1 Pattern of experimental phases.....	23
4.2 Preparing the experiment.....	24
4.2.1 Experimental objectives.....	24
4.2.2 Measurements.....	25
4.2.3 Factors.....	25
4.2.3.1 Problem-related factors.....	26
4.2.3.2 Algorithmic factors.....	28
4.2.3.3 Factors of the test environment.....	29
4.2.4 Test log book.....	30
4.2.5 Statistic log book.....	30
4.2.6 Definition of a utility function.....	31
4.2.7 Design of experiments.....	31
4.2.7.1 Modes of experimentation.....	31
4.2.7.2 Design points.....	33
4.3 Performing the experiment.....	34
4.4 Presenting the experiment.....	34
4.4.1 Choice of appropriate presentation forms / modes of presentation.....	34
4.4.2 Textual presentation.....	36

4.4.3	Graphical presentation techniques	37
4.4.4	Conclusion of an experiment	39
5	Statistical analysis	40
5.1	Experimental design	40
5.2	Initial examination of data.....	41
5.3	Analysis of variance	41
5.4	Estimation	43
5.5	Correlation and regression analysis.....	43
5.6	Testing of hypotheses	44
5.6.1	Parametric tests	45
5.6.1.1	One-sample t-test	45
5.6.1.2	Two-sample t-test.....	46
5.6.2	Non-parametric tests	47
5.6.2.1	Sign test for matched pairs	47
5.6.2.2	Wilcoxon matched pairs signed rank test	48
5.6.2.3	Friedman test	50
5.6.3	Tabular overview of commonly used statistic tests.....	51
6	Assessment of different performance criteria	54
6.1	Static ranking	54
6.2	Utility functions.....	58

Part II

1	Introduction.....	61
2	Information.....	63
2.1	Types of information	64
2.2	Relevance of information	66
2.3	Administration of information	67
2.4	Verification of informational benefit	67
2.5	Incorporation of information	69
2.5.1	Information as a factor for the solution initialization.....	69
2.5.2	Information as a factor for operational selection rules.....	70
2.5.2.1	Connection of information within function SELECT ()	71
2.5.2.2	Connection of information within function PRIORITIZE ()	72
2.6	Weighting of information	73
2.7	Diversification of information.....	74
2.8	Rivalry and conflicts among informational criteria.....	75
2.9	Technical considerations	75
3	The problems under consideration.....	77
3.1	Formal description.....	77
3.1.1	Quadratic assignment problem	77
3.1.2	Steiner tree problem in graphs	77
3.1.3	Traveling salesman problem	78
3.1.4	Vehicle routing problem	78
3.2	Data sets	79
3.2.1	Quadratic assignment problem	79
3.2.2	Steiner tree problem in graphs	81

3.2.3	Traveling salesman problem.....	82
3.2.4	Vehicle routing problem	85
3.3	Experimental results.....	87
3.3.1	Quadratic assignment problem	87
3.3.2	Steiner tree problem in graphs	88
3.3.3	Traveling salesman problem.....	88
3.3.4	Vehicle routing problem	89
4	Test runs with cooperative agent systems – an introduction.....	90
4.1	Preparing the experiment.....	90
4.1.1	Questions.....	90
4.1.2	Measurements	91
4.1.3	Factors.....	92
4.1.3.1	Algorithmic factors	92
4.1.3.2	Problem factors	93
4.1.3.3	Test factors.....	93
4.1.4	Test log book and statistic log book	93
4.1.5	Definition of a utility function	93
4.2	Performing the experiment.....	94
4.3	Presenting the experiment.....	94
5	Tabu search.....	96
5.1	Static tabu search	96
5.2	Reverse elimination method	96
5.2.1	REM for the QAP.....	98
5.2.2	REM for the TSP	99
5.3	Alternate tabu search approach.....	100
6	Communication - ant systems	101
6.1	Applications of ant systems for different combinatorial optimization problems	101
6.1.1	Traveling salesman problem	101
6.1.2	Quadratic assignment problem	103
6.1.3	Steiner tree problem in graphs	103
6.1.4	Vehicle routing problem	104
6.2	Fundamental criticism of primitive agent systems as the ant system	104
6.3	Extensions and modifications of the basic ant system	105
6.3.1	Verification of informational benefit.....	106
6.3.2	Weighting of information	106
6.3.3	Rivalry and conflicts between information	107
6.4	Tests with the ant system	107
6.4.1	Preparing the experiment.....	107
6.4.1.1	Questions.....	107
6.4.1.2	Measurements	108
6.4.1.3	Factors.....	108
6.4.1.3.1	Algorithmic factors	108
6.4.1.3.2	Problem factors	110
6.4.1.4	Test log book	110
6.4.1.5	Statistic log book	110
6.4.2	Performing the experiment.....	112
6.4.2.1	Preliminary test run A.1 (RANDOM).....	112
6.4.2.2	Preliminary test run A.2 (SOL).....	113

6.4.2.3	Preliminary test run A.3 (ANT).....	115
6.4.2.4	Test run A.4 (RANDOM).....	117
6.4.2.5	Test run A.5 (SOL).....	119
6.4.2.6	Test run A.6 (ANT).....	121
6.4.3	Presenting the experiment.....	122
6.4.3.1	Preliminary analysis.....	122
6.4.3.2	Analysis of variance.....	124
6.4.3.3	Internal comparative analysis.....	125
6.4.3.3.1	Box-plot analysis.....	125
6.4.3.3.2	Matrix comparison.....	126
6.4.3.4	Analysis of system-independent computational effort.....	127
6.4.3.5	Utility function.....	128
6.4.4	Conclusions and open questions.....	129
7	Cooperation - multi-chain approach.....	130
7.1	A cooperative agent system for the traveling salesman problem.....	130
7.1.1	Information-based coordination by solution initialization.....	131
7.1.2	Information-based cooperation and coordination by edge trails.....	132
7.2	A cooperative agent system for the vehicle routing problem.....	132
7.3	Tests with the multi-chain approach.....	133
7.3.1	Preparing the experiment.....	133
7.3.1.1	Questions.....	133
7.3.1.2	Measurements.....	133
7.3.1.3	Factors.....	133
7.3.1.3.1	Algorithmic factors.....	133
7.3.1.3.2	Problem factors.....	135
7.3.1.4	Test log book.....	135
7.3.1.5	Statistic log book.....	135
7.3.2	Performing the experiment.....	137
7.3.2.1	Computational results for the TSP.....	137
7.3.2.2	Computational results for the VRP.....	140
7.3.3	Presenting the experiment.....	143
7.3.3.1	Analysis of variance.....	143
7.3.3.2	Internal comparative analysis.....	144
7.3.3.2.1	Box-plot analysis.....	144
7.3.3.2.2	High-low diagram analysis.....	145
7.3.3.2.3	Matrix comparison.....	146
7.3.3.3	Time series analysis.....	147
7.3.3.4	Analysis of system-independent computational effort.....	149
7.3.3.5	Statistical analysis.....	150
7.3.3.5.1	Friedman test.....	150
7.3.3.5.2	Wilcoxon matched pairs signed rank test.....	152
7.3.3.6	Utility function.....	155
7.3.4	Conclusions and open questions.....	155
8	Coordination - multiple star-shaped approach.....	157
8.1	Star-shaped approach.....	158
8.1.1	Star-shaped diversification approach using distance measurements.....	158
8.1.2	Star-shaped diversification approach using the reverse elimination method.....	159
8.2	Multi-level star-shaped approach.....	161

8.3	Tests with the multiple star-shaped approach	162
8.3.1	Preparing the experiment	162
8.3.1.1	Questions	162
8.3.1.2	Measurements	163
8.3.1.3	Factors	163
8.3.1.3.1	Algorithmic factors	163
8.3.1.3.2	Problem factors	163
8.3.1.4	Test log book	164
8.3.1.5	Statistic log book	164
8.3.2	Performing the experiment	166
8.3.2.1	Computational results for the QAP	166
8.3.2.1.1	Test run D.1 (non-probabilistic versions)	166
8.3.2.1.2	Test run D.2 (probabilistic versions)	167
8.3.2.2	Computational results for the STPG	168
8.3.2.2.1	Test run E.1 (non-probabilistic versions)	168
8.3.2.2.2	Test run E.2 (probabilistic versions)	169
8.3.3	Presenting the experiment	170
8.3.3.1	Analysis of variance	170
8.3.3.2	Internal comparative analysis	172
8.3.3.2.1	Box-plot analysis	172
8.3.3.2.2	High-low diagram analysis	174
8.3.3.2.3	Matrix comparison	175
8.3.3.3	Analysis of system-independent computational effort	176
8.3.3.4	Statistical analysis	177
8.3.3.4.1	Friedman test	177
8.3.3.4.2	Wilcoxon matched pairs signed rank test	181
8.3.3.5	Utility function	183
8.3.4	Conclusions and open questions	184
9	Long runs / runs for large-sized problems	185
9.1	Quadratic assignment problem	185
9.2	Traveling salesman problem	185
9.3	Vehicle routing problem	186
	Conclusion	187
	Outlook	190
<u>Appendix</u>		
1	Meta-framework for iterative heuristics	192
1.1	Iterative (meta-)heuristics	192
1.2	Syntax of the heuristic	193
1.2.1	Problem	193
1.2.2	Population / solutions	194
1.2.3	Operations	198
1.2.4	Strategy	202
1.2.5	Information	204
1.3	Explanation of the most important framework sequences	207
1.4	Generating a meta-strategy and a meta-meta-strategy	210

1.5	Examples for embedding well-known meta-heuristics in the framework.....	210
1.5.1	Threshold accepting.....	210
1.5.2	Tabu thresholding.....	211
1.5.3	Simulated annealing.....	212
1.5.4	Genetic algorithms.....	212
1.6	Implemented strategies.....	215
1.6.1	Traveling salesman problem.....	215
1.6.2	Quadratic assignment problem.....	218
1.6.3	Steiner tree problem in graphs.....	219
1.6.4	Vehicle routing problem.....	219
	Literature.....	222
	Index.....	232

List of symbols

Statistics and statistic tests

$(\text{error})_{ij}$	residual error
$(\tilde{p}\tilde{t})_{ij}$	combined contribution (interaction) of a level of the problem factor and a level of the treatment factor.
α	significance level
$cv(\dots)$,	
$cv^{\text{lower}}(\dots)$,	
$cv^{\text{upper}}(\dots)$	critical values
d_i	difference value between a matched pair
f_{ij}, f_{ijk}	response variables
$F(\dots)$	distribution function
H_0	null hypothesis
H_a	alternate hypothesis
μ	sample mean
$N()$	normal distribution
P	probability
ρ	correlation coefficient
R_i	rank value
$R +, R -$	sum of ranks
s, s_1^2, s_2^2	standard deviations
σ	variance
t	test statistic
t_{n-1}	t-distribution with $n - 1$ degrees of freedom
v_1, v_2	degrees of freedom (DF)
W	test statistic
X, Y	(random) samples
x_1, \dots, x_n	sample of observations
\bar{x}	sample mean
\hat{x}	median
Z	test statistic

Algorithms (general)

$F(S)$	objective function
F^{BC}	best-case solution value

F^{LB}	lower bound
F^{UB}	upper bound
F^{WC}	worst-case solution value
FOP	set of feasible operations
$h(\rho, P)$	vectorized results of an algorithmic instance applied on a problem instance
nb_ iterations	number of iterations
nb_ levels	number of levels
$N(S)$	neighborhood of a solution S
op	heuristic operation
OP	set of all heuristic operations
$\Omega(P)$	solution space
(P)	optimization problem
$P \in (P)$	problem instance
r_i	performance criterion rank
ρ	algorithmic instance
R	set of real values
S, S'	solutions
S^*	optimal solution
S^{CURR}	current solution
S^{START}	start solution
td	tabu tenure
w_i	performance criterion weight
w_{ij}	relative performance criterion value
$X \subseteq \Omega$	feasible part of a solution space
$z = (P, \rho, h(\rho, P))$	vector representing an algorithmic application
Z	set of all $z = (P, \rho, h(\rho, P))$

Ant systems

α, β	weighting factors for the prioritizing rule
L_k	tour length of ant k
m	number of ants
Q	constant trail factor
ρ	evaporation parameter
t	number of levels

Algorithmic evaluation¹

$\text{dev}^{\text{AV}}(\dots)$	average deviation
$\text{dev}^{\text{B}}(\dots)$	best deviation
$\text{dev}^{\text{W}}(\dots)$	worst deviation
$f^{\text{AV}}(\dots)$	average solution
$f^{\text{B}}(\dots)$	best solution
$f^{\text{W}}(\dots)$	worst solution
$i = 1..I$	number of problem instances p_i
$j = 1..J$	number of heuristic treatments t_j
μ, μ_X, μ_Y	mean values
P_i	problem instance
$r = 1..R$	number of computation runs
$\text{rank}^{\text{AV}}(\dots)$	average rank
$\text{rank}^{\text{B}}(\dots)$	best rank
$\text{rank}^{\text{W}}(\dots)$	worst rank
$t(\dots)$	run time
t_j	heuristic treatment

Optimization problems

a) Knapsack problem

c_i	capacity units of item i
C	capacity
$i = 1..I$	number of items
v_i	value of item i
x_i	number of item i

b) Quadratic assignment problem

$C = [c_{ij}]_{n \times n}$	distance matrix
$N = \{1, \dots, n\}$	set of objects
π	permutation
$T = [t_{ij}]_{n \times n}$	flow matrix

¹ For the evaluation of numerical results a variety of different figures is used that cannot be listed here completely.

c) Steiner tree problem in graphs

$G = (V, E)$	graph
p^*	vertex
Q	set of basic vertices
S	set of Steiner vertices
v^*	vertex
V	set of vertices
w	start vertex

d) Traveling salesman problem

$C = [c_{ij}]_{n \times n}$	distance matrix
$G = (V, E)$	graph
e	edge in a graph
E	set of edges
$N = \{1, \dots, n\}$	set of cities