

Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

840

Advisory Board: W. Brauer D. Gries J. Stoer



Gerhard Reinelt

The Traveling Salesman

Computational Solutions
for TSP Applications

Springer-Verlag

Berlin Heidelberg New York

London Paris Tokyo

Hong Kong Barcelona

Budapest

Series Editors

Gerhard Goos
Universität Karlsruhe
Postfach 69 80
Vincenz-Priessnitz-Straße 1
D-76131 Karlsruhe, Germany

Juris Hartmanis
Cornell University
Department of Computer Science
4130 Upson Hall
Ithaca, NY 14853, USA

Author

Gerhard Reinelt
Institut für Angewandte Mathematik, Universität Heidelberg
Im Neuenheimer Feld 294, D-69120 Heidelberg, Germany

CR Subject Classification (1991): G.2, I.3.5, G.4, I.2.8, J.1-2

ISBN 3-540-58334-3 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-58334-3 Springer-Verlag New York Berlin Heidelberg

CIP data applied for

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

© Springer-Verlag Berlin Heidelberg 1994
Printed in Germany

Typesetting: Camera-ready by author
SPIN: 10475419 45/3140-543210 - Printed on acid-free paper

Preface to the Online Edition

Still today I am receiving requests for reprints of the book, but unfortunately it is out of print. Therefore, since the book still seems to receive some attention, I proposed to Springer Verlag to provide a free online edition. I am very happy that Springer agreed. Except for the correction of some typographical errors, the online edition is just a copy of the printed version, no updates have been made. In particular, Table 13.1 gives the status of TSPLIB at the time of publishing the book. For accessing TSPLIB the link <http://www.iwr.uni-heidelberg.de/iwr/comopt/software/TSPLIB95/> should be used instead of following the procedure described in Chapter 13.

*Heidelberg, January 2001
Gerhard Reinelt*

Preface

More than fifteen years ago, I was faced with the following problem in an assignment for a class in computer science. A brewery had to deliver beer to five stores, and the task was to write a computer program for determining the shortest route for the truck driver to visit all stores and return to the brewery. All my attempts to find a reasonable algorithm failed, I could not help enumerating all possible routes and then select the best one.

Frustrated at that point, I learnt later that there was no fast algorithm for solving this problem. Moreover, I found that this problem was well known as the traveling salesman problem and that there existed a host of published work on finding solutions. Though no efficient algorithm was developed, there was a tremendous progress in designing fast approximate solutions and even in solving ever larger problem instances to optimality. I started some work on the traveling salesman problem several years ago, first just writing demos for student classes, but then trying to find good and better solutions more effectively. I experienced the fascination of problem solving that, I think, everyone studying the traveling salesman problem will experience. In addition, I found that the problem has relevance in practice and that there is need for fast algorithms.

The present monograph documents my experiments with algorithms for finding good approximate solutions to practical traveling salesman problems. The work presented here profited from discussions and meetings with several people, among them Thomas Christof, Meinrad Funke, Martin Grötschel, Michael Jünger, Manfred Padberg, Giovanni Rinaldi, and Stefan Thienel, not naming dozens of further international researchers.

It is the aim of this text to serve as a guide for practitioners, but also to show that the work on the traveling salesman problem is not at all finished. The TSP will stimulate further efforts and continue to serve as the classical benchmark problem for algorithmic ideas.

Heidelberg, June 1994
Gerhard Reinelt

Contents

1	Introduction	1
2	Basic Concepts	4
2.1	Graph Theory	4
2.2	Complexity Theory	7
2.3	Linear and Integer Programming	12
2.4	Data Structures	14
2.5	Some Fundamental Algorithms	25
3	Related Problems and Applications	31
3.1	Some Related Problems	31
3.2	Practical Applications of the TSP	35
3.3	The Test Problem Instances	40
4	Geometric Concepts	42
4.1	Voronoi Diagrams	42
4.2	Delaunay Triangulations	48
4.3	Convex Hulls	54
5	Candidate Sets	64
5.1	Nearest Neighbors	64
5.2	Candidates Based on the Delaunay Graph	67
5.3	Other Candidate Sets	70
6	Construction Heuristics	73
6.1	Nearest Neighbor Heuristics	73
6.2	Insertion Heuristics	82
6.3	Heuristics Using Spanning Trees	89
6.4	Savings Methods and Greedy Algorithm	94
6.5	Comparison of Construction Heuristics	97

7	Improving Solutions	100
7.1	Node and Edge Insertion	100
7.2	2-Opt Exchange	105
7.3	Crossing Elimination	115
7.4	The 3-Opt Heuristic and Variants	117
7.5	Lin-Kernighan Type Heuristics	123
7.6	Comparison of Improvement Heuristics	130
8	Fast Heuristics for Large Geometric Problems	133
8.1	Space Filling Curves	133
8.2	Strip Heuristics	137
8.3	Partial Representation	139
8.4	Decomposition Approaches	145
9	Further Heuristic Approaches	153
9.1	Simulated Annealing	153
9.2	Evolutionary Strategies and Genetic Algorithms	157
9.3	Tabu Search	158
9.4	Neural Networks	159
10	Lower Bounds	161
10.1	Bounds from Linear Programming	161
10.2	Simple Lower Bounds	163
10.3	Lagrangian Relaxation	172
10.4	Comparison of Lower Bounds	184
11	A Case Study: TSPs in PCB Production	187
11.1	Drilling of Printed Circuit Boards	187
11.2	Plotting of PCB Production Masks	193
12	Practical TSP Solving	200
12.1	Determining Optimal Solutions	200
12.2	An Implementation Concept	204
12.3	Interdependence of Algorithms	207
	Appendix: TSPLIB	211
	References	214
	Index	222