Heuristic algorithms for the robust traveling salesman problem with interval data

R. Montemanni J. Barta M. Mastrolilli L.M. Gambardella

Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA)
Galleria 2, CH-6928 Manno-Lugano, Switzerland
{roberto, monaldo, luca}@idsia.ch, janos.barta@supsi.ch

Abstract

The traveling salesman problem is one of the most famous combinatorial optimization problems, and has been intensively studied in the last decades. Many extensions to the basic problem have been also proposed, with the aim of making the resulting mathematical models as much realistic as possible.

We study an extension to the basic problem where travel times are specified as a range of possible values. This model reflects the intrinsic difficulties to estimate travel times exactly in reality. The robust deviation criterion is adopted to drive optimization over the interval data problem so obtained. A negative result about the approximability of the problem is presented together with fast and easy-to-implement heuristic approaches.

Computational experiments are presented and discussed.

1 Introduction and problem description

A variation of the classic symmetric traveling salesman problem (TSP) is studied in this paper. The variation is motivated by the observation that estimating travel times exactly is often a difficult task, since they depend on many factors that are difficult to predict. Uncertainty about data should be consequently taken into account. We treat the case where the only information available is represented by a set of equally possible values for each travel time. We optimize the resulting problem according to the *robust deviation criterion* (see Kouvelis and Yu [6]). A *robust tour* is, intuitively, a tour which minimizes the maximum deviation from the optimal tour over all realizations of edge costs.

The robust TSP with interval data is defined on an undirected graph $G = \{V, E\}$, where V is a set of vertices, with vertex 0 associated with the depot, and vertices $1, \ldots, |V|$ representing the cities to be visited, and E is the set of edges of the graph. An interval $[l_{ij}, u_{ij}]$, with $0 \leq l_{ij} \leq u_{ij}$, is associated with each edge $\{i, j\} \in E$, and represents the possible travel times. The objective of the optimization is to find a Hamiltonian cycle (*tour*) with the minimum cost, according to the cost function associated with the notion of robust deviation. In order to formally describe the *robust TSP*, we need the following definitions. A **scenario** R is a realization of the edge costs, i.e. a cost $c_{ij}^R \in [l_{ij}, u_{ij}]$ is chosen for each edge of the graph. The **robust deviation** of a tour t in scenario R is the difference between the cost of t in scenario R and the cost of a shortest tour in R. A tour t is said to be a **robust tour** if it has the smallest (among all possible tours) maximum (among all possible scenarios) robust deviation.

2 MILP formulation

Theorem 1 (Montemanni et al. [9]) *Given a tour t, the scenario R that maximizes the robust deviation for t is the one where all the edges of tour t have the highest possible cost, and* *the costs of the remaining edges are at their lowest possible value, i.e.* $c^R_{ij} = u_{ij}\ \forall\{i,j\} \in t$ *and* $c^R_{ij} = l_{ij}\ \forall\{i,j\} \notin t$.

We are ready to give a mathematical formulation of the robust traveling salesman problem. Variables x are $\{0,1\}$ variables that identify the edges of the robust deviation tour: $x_{ij} = 1$ if edge $\{i,j\}$ is on the robust tour; 0 otherwise. Variables y define the shortest tour of the scenario induced by the tour defined by x variables: $y_{ij} = 1$ if edge $\{i,j\}$ is on the shortest tour; 0 otherwise. In the formulation we will also refer to the set of all possible tours on graph G as $\mathcal{TSP}$.

The problem can be formalized in a compact way as follows:

$$\min_{x\in\mathcal{TSP}} \left\{ \max_{s\in S} \left\{ \sum_{\{i,j\}\in E} c^s_{ij} x_{ij} - \min_{y\in\mathcal{TSP}} \left\{ \sum_{\{i,j\}\in E} c^s_{ij} y_{ij} \right\} \right\} \right\} \tag{1}$$

Using the result of Theorem 1, (1) can be rewritten as follows:

$$\min_{x\in\mathcal{TSP}} \left\{ \sum_{\{i,j\}\in E} u_{ij} x_{ij} - \min_{y\in\mathcal{TSP}} \{(l_{ij} + (u_{ij} - l_{ij})x_{ij})y_{ij}\} \right\} \tag{2}$$

The result of Theorem 1 has been used to give a closed form definition of the shortest tour in the scenario defined by x variables.

Formulation (2) can be reduced to a mixed integer linear program by adding a free variable r, that will contain the regret term now expressed through the nested min operator, and by inserting a new set of linear constraints. What we obtain, after a reorganization of terms, is the following formulation:

$$(RTSP) \quad \min \sum_{\{i,j\}\in E} u_{ij} x_{ij} - r \tag{3}$$

$$\text{s.t. } r \leq \sum_{\{i,j\}\in E} y_{ij} l_{ij} + \sum_{\{i,j\}\in E} y_{ij}(u_{ij} - l_{ij}) x_{ij} \qquad \forall\, y \in \mathcal{TSP} \tag{4}$$

$$x \in \mathcal{TSP} \tag{5}$$

Constraint (5) states that x variables must be constrained to form a Hamiltonian cycle. Any feasible mixed integer programming formulation for the classic traveling salesman problem can be used for this purpose. A survey on these formulations can be found, for example, in Langevin et al [7]). Constraints (4) substitute the nested min operator and incorporate the result of Theorem 1 for the calculation of the maximum robust deviation of the tour defined by x variables. Notice that in these inequalities, y plays the role of a constant vector representing a tour.

3 Approximation algorithms

When a problem is NP-hard, this does not prevent the case that it can be approximated quite well, i.e. it admits an *r-approximation algorithm* that always returns in polynomial time a solution that is within r times the optimal value, for any fixed *approximation ratio* r (see e.g. Hochbaum [3] for an excellent glimpse on the subject). However, we start this section by observing that in the worst-case, when robustness comes into play, then obtaining "good" solutions is as hard as computing optimal solutions. More precisely, unless P=NP, there does not exist any r-approximation algorithm for the robust version of the problem, for any arbitrarily large $r \geq 1$. This inapproximability result holds not only for the traveling salesman problem, but for any NP-hard classic combinatorial optimization problem. This gives a measure of the additional complexity when the robustness criterion we consider is taken into account with an NP-hard problem.

Theorem 2 (Montemanni et al. [9]) *For any NP-hard classic combinatorial optimization problem, no polynomial time r-approximation algorithm (for any $r \geq 1$) can exist for the robust deviation/interval data version of the problem, unless P=NP.*

Theorem 2 suggests that even when NP-hard special cases of the classic TSP can be approximated arbitrarily close to the optimum (see e.g. Arora [1]), the corresponding robust counterpart cannot be approximated in polynomial time within any arbitrarily large approximation ratio, unless $P = NP$.

Besides this negative result, we mention that Kasperski and Zieliński [5] provided a polynomial time approximation algorithm with a performance ratio of 2 for the minmax regret versions of classical combinatorial optimization problems that are polynomially solvable.

Taking inspiration by the approach used in Kasperski and Zieliński [5] and by using well-known heuristic algorithms for the classic traveling salesman problem, we suggest the following heuristics for the addressed problem.

4 Heuristic Algorithms HU and HM

The method we propose is based on the construction of an ad-hoc scenario and on the solution of the related classic traveling salesman problem. The resulting tour will also be a feasible solution for the robust problem. We consider in particular two scenarios, that according to Kasperski and Zieliński [5] are the most promising ones: scenario U (all the costs at the highest possible value) and scenario M, defined as follows: $c_{ij}^M = \frac{(u_{ij}+l_{ij})}{2}$ $\forall\{i,j\} \in E$. According to Kasperski and Zieliński [5], the optimal solution of the classic TSP on scenario M, if available, would guarantee a 2-approximation for the optimal solution of the robust TSP itself, while scenario U has been proven to perform well (often better than M) for the robust counterpart of some combinatorial optimization problems (see Kasperski and Zieliński [4], Montemanni and Gambardella [10]). It is important to stress that we cannot guarantee a "good" approximate solution in polynomial time for our algorithm, without solving to optimality the classic problem on scenario M.

In the reminder of this paper we will refer to the heuristic algorithm based on scenario U as HU, and to that based on scenario M as HM.

Note that it may be prohibitive to set up more complex heuristic methods, like metaheuristics, where the evaluation of costs has to be carried out many times. This happens because the evaluation of the robustness cost of a tour is already an NP-hard problem (a classic traveling salesman problem has to be solved on the scenario induced by the tour under evaluation).

4.1 Computational experiments

4.1.1 Implementation details

We coded the algorithms in *ANSI C*. The callable library version of heuristic algorithm *LKH 1.3* for the classic traveling salesman problem is embedded in our implementation (exact method *Concorde 03.12.19* is used to calculate robustness costs). The total computational complexity of the overall heuristic algorithm is as follows: the time required to construct scenarios M and U is bounded by $O(|V|^2)$, since each edge of the graph has to be processed. Producing an upper bound for the (classic) cost of the optimal tour in a given scenario has, in our implementation, an experimental approximate running time of $O(|V|^{2.2})$ (see Helsgaun [2]). This last bound dominates the other.

4.1.2 Results

All the tests have been carried out on an Intel Pentium 4 1.5 GHz / 256 MB machine. We used two families of benchmarks: pure random problems (indicated as *R-**) and problems derived from TSPLIB instances[1] (indicated as *name-**, where *name* is the name of the original TSPLIB instance). A detailed description of the problems considered is given in Montemanni et al. [9]. Ten instances are considered for each problem and information about the optimality gap are summarized.

On the reference machine the computation times are negligible (the time required to compute the robustness cost of the solution is not counted) for all the benchmarks considered. For this reason computation times are not reported.

In Tables 1 and 2 we report, for each problem and for each algorithm considered, average and standard deviation for the following indicator:

$$100 \cdot \frac{UB(a) - LB_{ex}}{UB(a)} \tag{6}$$

where $UB(a)$ indicates the robustness cost of the solution provided by the heuristic algorithm a under investigation, and LB_{ex} is the best lower bound for the cost of the optimal robustness solution available (see Montemanni et al. [9]). Indicator (6) provides then an upper bound for the gap between the cost of the heuristic solution and that of the optimal one.

In the second column of Tables 1 and 2 it is indicated whether an optimal solution is available for all the instances of each problem considered. Rows with entry *No* in the second column may therefore overestimate the gap between the cost of the heuristic and that of an optimal one.

Notwithstanding the very short computation times required by the algorithms, the solutions provided by methods HU and HM are of good quality. In particular, we conjecture that the performance decay for the biggest problems most depends on the poor quality of the lower bounds. If this is true, the quality of the heuristic solutions does not seem to be strictly related to the dimension of the problems.

We can observe also that HU seems to perform better than HM, especially on the biggest problems, for which the dominance is clear.

4.1.3 2-opt and 3-opt improving heuristics

Notwithstanding the difficulties connected with a repeate evaluation of robustness costs (that might slow down the computation too much), it is interesting to measure how the solutions returned by the heuristic methods described in Section 4 can be improved by applying classic local search methods. We here consider 2-opt and 3-opt algorithms (see Lawler at al. [8]).

Starting from a feasible solution, 2-opt (3-opt) sequentially considers each possible pair (triple) of nodes and evaluate the solution obtained by swapping some parts of the tour according to the two (tree) nodes selected. Only improving swaps are accepted. The procedure is repeated until a full iteration, during which all possible pairs (triples) are evaluated, is completed without any improvement in the solution quality.

As seen before, the repeated evaluation of robustness costs can be a problem, since a single robustness cost evaluation is already an NP-hard problem. For this reason, it is crucial to limit

[1] http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95.

Table 1: Heuristic algorithms. Random instances. Percentage deviations from best lower bounds (6). Averages over 10 instances.

Problem	Optimal	Algorithm HM		Algorithm HU	
		Avg	StDev	Avg	StDev
R-10-100	Yes	4.16	8.97	2.38	3.00
R-10-1000	Yes	6.96	7.61	4.28	5.57
R-20-100	Yes	3.51	5.14	1.48	2.37
R-20-1000	Yes	2.44	3.13	1.03	1.51
R-30-100	Yes	2.30	3.50	1.66	1.98
R-30-1000	Yes	3.06	2.95	1.03	1.35
R-40-100	Yes	3.99	3.32	1.02	0.98
R-40-1000	Yes	3.84	3.51	1.08	1.22
R-50-100	No	5.55	4.78	2.18	3.87
R-50-1000	Yes	4.87	2.22	1.11	1.19
R-60-100	No	6.33	4.26	3.41	2.82
R-60-1000	Yes	3.86	1.36	0.69	0.69
R-80-1000	No	3.77	1.46	1.39	1.44
R-30-10	Yes	3.37	2.75	0.49	1.03
R-30-10000	Yes	2.23	1.87	1.79	2.61
R-120-100	No	11.30	1.61	8.30	1.72
R-120-1000	No	12.54	2.93	9.71	2.07
R-240-100	No	13.21	1.92	9.32	1.49
R-240-1000	No	16.39	6.82	12.53	6.98
R-360-100	No	9.71	4.18	8.72	3.94
R-360-1000	No	15.68	1.45	12.74	1.16

the number of evaluations as much as possible. We therefore implemented optimized strategy as described in Montemanni et al. [9].

Computational results (not reported here, but available in Montemanni et al. [9]) suggest that the local search algorithms rarely improve the solutions provided by heuristic algorithm HU, and even when this happens, the improvement is extremely marginal (less than 1% on average). It is interesting to observe that the computation time required by the local search algorithm is strictly related to the dimension of the problems (as expected), and is absolutely not negligible for the largest problems considered. These results suggest that applying 2-opt and 3-opt methods to the solutions provided by HU and HM might not be convenient.

5 Conclusion

Heuristic methods for the robust traveling salesman problem with interval data, where uncertainty about edge costs is taken into account, have been studied and discussed in this paper.

The heuristic approaches proved to be extremely fast and to guarantee good approximations of the optimal solutions.

A negative theoretical results about the approximability of the problem has been finally reported.

References

[1] S. Arora. Polynomial time approximation schemes for euclidean traveling salesman and other geometric problems. *Journal of the ACM*, 45(5):753–782, 1998.

Table 2: Heuristic algorithms. TSPLIB instances. Percentage deviations from best lower bounds (6). Averages over 10 instances.

Problem	Optimal	Algorithm HM		Algorithm HU	
		Avg	StDev	Avg	StDev
gr17-0.25	Yes	4.13	5.42	4.44	4.20
gr17-0.50	Yes	3.37	3.67	3.82	3.86
gr21-0.25	Yes	5.53	8.01	7.98	10.04
gr21-0.50	Yes	0.87	2.64	1.02	1.53
gr24-0.25	Yes	5.01	4.45	4.13	4.48
gr24-0.50	Yes	2.10	1.94	0.88	1.11
fri26-0.25	Yes	2.11	2.15	10.38	10.78
fri26-0.50	Yes	2.20	2.34	0.82	0.99
swiss42-0.25	Yes	6.05	4.06	6.26	4.21
swiss42-0.50	No	2.10	2.09	2.12	1.42
dantzig42-0.25	Yes	3.22	3.08	3.33	3.30
dantzig42-0.50	Yes	2.99	2.87	2.28	1.49
gr48-0.25	Yes	8.93	5.33	10.69	4.88
gr48-0.50	No	11.13	5.51	14.61	4.45
hk48-0.25	Yes	3.85	4.10	5.66	6.02
hk48-0.50	No	4.92	3.83	6.63	4.86
brazil58-0.25	Yes	6.05	7.37	3.74	1.68
brazil58-0.50	No	7.99	4.74	4.27	1.52
gr120-0.25	No	15.45	4.34	15.30	4.36
gr120-0.50	No	33.09	4.21	30.61	4.60
si175-0.25	No	41.38	2.09	42.03	1.81
si175-0.50	No	38.56	2.04	37.57	2.09
brg180-0.25	No	30.03	2.03	24.85	2.18
brg180-0.50	No	31.76	1.53	27.73	1.53

[2] K. Helsgaun. An effective implementation of the lin-kernighan traveling salesman heuristic. *European Journal of Operational Research*, 126(1):106–130, 2000.

[3] D.S. Hochbaum, editor. *Approximation Algorithms for NP-hard Problems.* PWS Publishing Company, Boston, 1995.

[4] A. Kasperski and P. Zieliński. The shortest path problem with interval data. Submitted for publication.

[5] A. Kasperski and P. Zieliński. An approximation algorithm for interval data minmax regret combinatorial optimization problems. *Information Processing Letters*, 97:171–180, 2006.

[6] P. Kouvelis and G. Yu. *Robust Discrete Optimization and its applications.* Kluwer Academic Publishers, 1997.

[7] A. Langevin, F. Soumis, and J. Desrosiers. Classification of travelling salesman formulations. *Operations Research Letters*, 9:127–132, 1990.

[8] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B. Shmoys. *The Traveling Salesman Problem.* Wiley, Chichester, 1985.

[9] R. Montemanni, J. Barta, M. Mastrolilli, and L.M. Gambardella. The robust traveling salesman problem with interval data. *Submitted for publication*, 2006.

[10] R. Montemanni and L.M. Gambardella. An exact algorithm for the robust shortest path problem with interval data. *Computers and Operations Research*, 31(10):1667–1680, 2004.