Covering Tours with Turn Cost: Variants, Approximation and Practical Solution

Sándor P. Fekete* Dominik Krupke*

Abstract

For a given set P of points in the plane, the Angular-Metric Traveling Salesman Problem (AM-TSP) asks for a tour on P that minimizes the total turn along the tour. While there exists a PTAS for the Euclidean TSP, for the AM-TSP only a $O(\log n)$ approximation algorithm is known. We introduce a number of generalizations and provide approximation algorithms whose performance depend on the angular resolution. We also develop exact methods for computing provably optimal solutions, and present an array of experimental results.

1 Introduction

Consider an outdoor setting with a number of obstacles. Swarms of mosquitos populate the area, with a number of known hotspots. How can we lower the danger of diseases by zapping the mosquitos with a flying drone, such as the one shown in Figure 1?

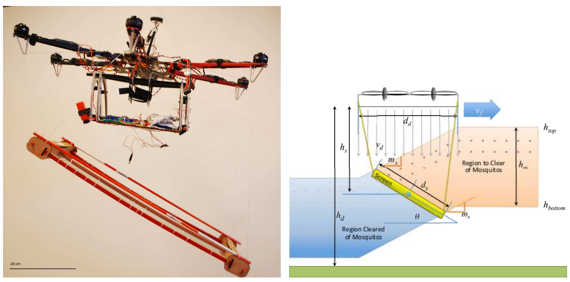

Figure 1: A drone equipped with an electrical lattice to hunt mosquitoes. Images by Aaron Becker.

Visiting a set of points by an optimal tour is a natural and important problem, both in theory and practice. If we are only concerned with minimizing the total distance traveled for visiting all points this is the classic Traveling Salesman Problem (TSP). However, for path planning by a flying robot, we also incur a cost for changing direction; this is related to the *Angular-Metric TSP* (AM-TSP), in which the objective is to minimize the total turn. In addition, we may want to focus on a subset of the points in order to provide better coverage, incurring a penalty for the uncovered ones.

*Department of Computer Science, TU Braunschweig, Germany. s.fekete@tu-bs.de, krupke@ibr.cs.tu-bs.de

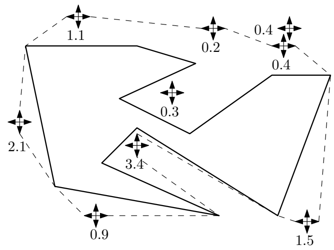

Figure 2: An example instance with obstacles and individual penalty costs for the points in P.

Related Work. Angle-restriced tour problems were studied by Fekete and Woeginger [4]. Touring points in the plane with minimal turn cost was considered by Aggarwal et al. [1], who provide a $O(\log n)$ approximation algorithm, but also show that already the cycle cover version is NP-hard. Arkin et al. [3] consider different grid-based versions of covering with turn cost, and provide a spectrum of approximation algorithms. Minimizing the total turn cost can be modeled as a special case of the quadratic TSP, which has received a fair amount of attention; see [5, 6, 9, 8] for research in optimal solutions and heuristics.

Our Results. We consider a number of variants for AM-TSP, motivated by practical applications. In particular, we consider the setting in which the set of possible headings at visited points is discretized; this also allows it to easily add polygonal obstacles into the environment. We also provide approximation algorithms for the generalization in which a penalty can be paid instead of covering a point. In addition, we present computational results.

2 Preliminaries and Problems

Preliminaries. We are given a set of points $P \subset \mathbb{R}^2$ in a polygonal environment with the obstacles $\mathcal{O}$. For every point $p \in P$ a set of ω angles $\delta(p) \in [0, \pi)^\omega$ that describe possible headings when covering p; each angle corresponds to two possible, opposite headings. A *pose* consists of a position and a heading. The respective set of poses results in an undirected weighted graph. There is weighted edge between any two poses that represents the cheapest collision-free polygonal path connecting them. The travel cost is a linear combination of the sum of turn angles (with coefficient τ) and the length (with coefficient κ).

Problems. The *Full Cycle Cover Problem* (FCCP) asks for a set of non-trivial cycles of minimum total cost, such that every point is covered. The *Full Tour Problem* (FTP) asks for a single cycle of minimum total cost that covers all points. The *Penalty Cycle Cover Problem* (PCCP) and the *Penalty Tour Problem* (PTP) are defined analogously, but points may be left uncovered by paying an individual penalty $\rho(p) \in \mathbb{R}_0^+$ for every omitted point $p \in P$.

All problem variants are NP-hard. The hardness of the Full Tour Problem is implied by the hardness of the Euclidean TSP; the Full Cycle Cover Problem is NP-hard with $\omega \geq 2$ by a straightforward adaption of the NP-hardness proof for the angular metric cycle cover problem by Aggarwal et al. [1]. Clearly, this implies hardness for the penalty variants. For $\omega = 1$, the problem can be solved using a minimum weight perfect matching on an appropriate auxiliary graph.

A subproblem is to calculate the cheapest transition between two poses around polygonal obstacles. If there are only distance costs, this problem equals the Euclidean shortest path, for which only the visibilty graph needs to be considered. It can easily be shown that this is also true with turn costs and the graph can easily be transformed such that the turn costs are integrated in edge weights resulting in a complexity of $O(|V_{\mathcal{O}}|^2 * \log |V_{\mathcal{O}}|)$ for the calculation of a cheapest transition, where $V_{\mathcal{O}}$ are the vertices of the obstacles.

3 Approximation Algorithms

We now propose approximation techniques for all four problems. Due to limited space, we only outline the main ideas; details are left to the full paper.

Note that for all the proposed approximation algorithms, the approximation factor and the runtime both depend linearly on the maximum number of orientations ω, which is assumed to be constant.

3.1 Full Coverage

Theorem 1 *For a fixed ω, there is a $2 * \omega$ approximation algorithm for the FCCP. In case $\omega = 1$, the solution is optimal.*

Proof. Solve the LP-relaxation of the integer program (IP); select for each point the orientation of highest variable value. Do a minimum weight perfect matching on the vertices associated with these points. Both takes polynomial time, with the LP-relaxation being the dominant part.

Because every point has at most ω orientations, at least one orientation of each point is used with a fractional weight of at least $1/\omega$. Multiplying the solution by ω and applying some local modifications that do not increase the cost (like skipping a point) yields a half-integral solution. This can be multiplied by two to obtain an integral solution. By further local modifications that do not increase the cost (possibly even decrease it), we obtain a perfect matching with at most $2 * \omega$ times the cost of the LP-relaxation. This (not minimal) perfect matching is an upper bound. ☐

To connect the cycles provided by Theorem 1, we simply use a minimum spanning tree (MST). Doubling the edges results in cycles with u-turns on the original cycles, which can be connected with no additional cost (but the u-turns from the doubling involve an extra cost).

Theorem 2 *For a fixed ω, there is a $4 * \omega + 2$ approximation algorithm for the FTP. In case of $\omega = 1$, there is a 4-approximation algorithm.*

Proof. We compute a $2 * \omega$-approximation of the cycle cover using Theorem 1, then connect these cycles via an MST. The MST provides $m - 1$ edges for a cycle cover with m cycles. An edge between two cycles corresponds to the cheapest connection between two of it points ($\in P$), ignoring the headings at the end. These $m - 1$ edges are doubled to create a valid tour. The minimum spanning tree is a lower bound on the optimal value. Connecting the edges to the cycles involves additional turn costs (360° for each doubled edge), but these can be charged to the cycles, because every cycle has a turn angle sum of at least 360°. ☐

3.2 Penalty Coverage

The adaption of the approximation algorithms to the penalty variants is surprisingly simple.

Theorem 3 *For a fixed ω, there is a $2 * (\omega + 1)$ approximation algorithm for a PCCP.*

Proof. We proceed as in Theorem 1, but we add an artificial orientation that allows a single artificial cycle with the cost of the penalty. ☐

From this penalty cycle cover we use the prize-collecting Steiner tree to select and connect good cycles. The connecting of the selected cycles via the edges of the tree is identical to full coverage. Only the analysis is slightly more difficult.

Theorem 4 *For a fixed ω, there is a $4 * (\omega + 1) + 4$ approximation algorithm for the PTP.*

Proof. We first compute a penalty cycle cover approximation with a factor of $2 * (\omega + 1)$, using Theorem 3. We remove all points for which the penalty in the cycle cover has been paid. Next we compute a 2-approximation of the prize-collecting Steiner tree, using the approximation algorithm of Goemans and Williamson [7] that has a time complexity of $O(n^2 \log n)$. This is done on a graph that contains all remaining points, with edge costs corresponding to shortest paths with turn costs but arbitrary start and goal headings. For two points that are in the same cycle, we set the cost to zero. We remove all cycles for which no point is in the resulting prize-collecting Steiner tree. All other components we connect by selecting edges from the tree (i.e., $m-1$ edges for m cycles). This can be done by iterating over all edges in the tree, adding an edge if it connects two different components. Clearly, the cost of an optimal prize-collecting Steiner tree is a lower bound for the tour. Due to the 2-approximation, the sum of all edge weights and penalties is at most twice the cost of the optimal penalty tour. The selected edges are transformed to cycles by doubling them and adding 180° turns at the ends. We can merge the cycles with no additional cost, as in Theorem 2. This results in at most four times the cost of the optimal tour plus $2 * (m-1) \times 180°$ turns for m cycles in the cycle cover. As every cycle in the cycle cover has also at least 360°, we can charge the 180° turns to the cycles, which leads to $2*2*(\omega+1)*\mathrm{OPT}$. Combined, this results in $2*2*(\omega+1)*\mathrm{OPT}+2*2*\mathrm{OPT} = (4*(\omega+1)+4)*\mathrm{OPT}$. □

4 Integer Programming

We work on an auxiliary graph $G(V,E)$: For every point $p \in P$ with orientations $\delta(p)$, we create the vertices $V(p) = \bigcup_{\alpha\in\delta(p)}\{v_{p,\alpha}, v_{p,\alpha+\pi}\}$ representing the two poses by which a point can be left/entered through one of its orientations. Furthermore, there is an edge $e=\{v,v'\}$ between any two $v=v_{p,\alpha} \in V(p)$ and $v'=v_{p',\alpha'} \in V(p')$ with the cost $c(e)$ representing the minimum cost path from the pose of being at p and heading α to the pose of being at p' and heading $\alpha'+\pi$ (this cost is symmetric). Entering on the vertex for the pose of being at p' and heading $\alpha'+\pi$ implies leaving through the vertex for the pose of being at p' with the heading α'.

For the cycle cover variant, there is also an edge $e=\{v_{p,\alpha}, v_{p,\alpha+\pi}\}$ for every $p \in P$ with the cost of the cheapest cycle (covering it and at least one other point). cycle cover, points can be in two unconnected cycles even in an optimal solution. The additional edge is used to implicitly represent this kind of cycle.

The integer programming formulation for cycle cover can be given as follows:

$$\min \quad \sum_{e\in E} c(e) * x_e \tag{1}$$

$$\text{s.t.} \quad \sum_{e\in E(v_{p,\alpha})} x_e = \sum_{e\in E(v_{p,\alpha+\pi})} x_e \quad \begin{array}{l}\forall p\in P\\ \forall \alpha\in\delta(v)\end{array} \tag{2}$$

$$\sum_{\alpha\in\delta(p)}\sum_{e\in E(v_{p,\alpha})} x_e = 1 \quad \forall p \in P \tag{3}$$

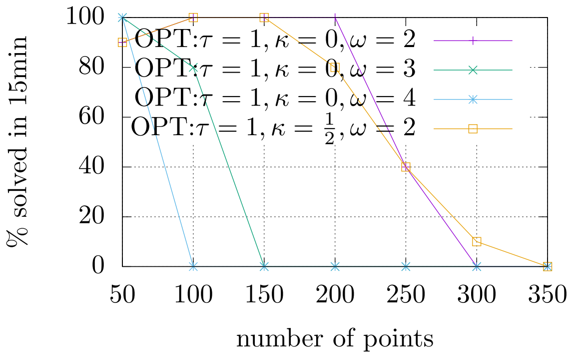

Figure 3: Percentage of tour instances solved to optimum within 15 min. 10 instances for each size $50, 100, \ldots, 350$. The cycle cover variant is only slightly better. i

$$x_e \in \{0,1\} \quad \forall e \in E \tag{4}$$

Eq. 2 states that if and only if there is an incoming edge on one side, there has to be an outgoing edge on the opposite site. Eq. 3 states that there have to be exactly two edges entering/leaving (using the symmetry induced by the previous equation). $E(v)$ represents the set of edges incident to v.

The subcycle elimination constraints for obtaining a tour can be adapted directly from the TSP.

$$\sum_{e\in E(V(C),V(P\setminus C))} x_e \geq 2 \quad \forall C \subsetneq P, C\neq\emptyset \tag{5}$$

Hence, the IP for tours is given by adding Eq. 5 to the cycle cover formulation. The penalty versions are omitted due to space constraints.

5 Experiments

For the full coverage problem variants, we implemented the integer programs and the approximation algorithms for tour and cycle cover without obstacles. In this section we discuss the experimental results for these implementations. Experiments were executed on modern desktop computers equipped with an *Intel(R) Core(TM) i7-6700K CPU @* 4.00 GHz and 64 GB of RAM. The used CPLEX version was 12.5.0.0 with the parameters `EpInt`=0, `EpGap`=0, `EpOpt`=1×10^{-9}, and `EpAGap`=0. No further optimizations were performed. As cost parameters we used $\tau=1, \kappa=0$, with one additional run with $\omega=2, \tau=1, \kappa=0.5$. Experiments were run for 10 random instances in the unit square per size $50, 100, \ldots, 350$. As passing orientations $\{i*\frac{\pi}{\omega} | i = 0,\ldots,\omega-1\}$ were chosen for all points equally.

Fig. 3 shows the percentage of instances solved to optimum within 15 min via the integer program for tours. The cycle cover variant is only slightly better. It can be seen that we can solve 50% of the instance of

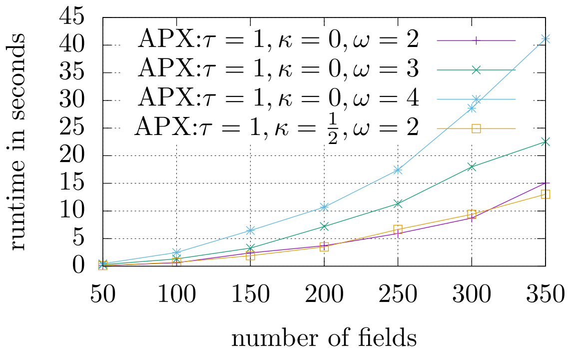

Figure 4: Average runtime of the approximation algorithm for cycle cover and tour (with nearly identical runtime). For each size $50, 100, \ldots, 350$,10 instances were tested.

size up to 200 points for $\omega = 2$. Already for $\omega = 3$ the performance drops strongly such that the maximum instance size becomes 100. For $\omega = 4$ only the smallest instances (50 points) have been solved in time. Hence, with our formulation only for $\omega = 2$ there is a serious advantage compared with the angular metric cycle cover and traveling salesman problem (using the work of Aichholzer et al. [2] as reference point).

The average runtime of the approximation algorithm is shown in Fig. 4. Here we do not differentiate between cycle cover and tour because they have nearly the same runtime. It can be seen that ω also has a lot of influence on the runtime of the approximation algorithm and the runtime for $\omega = 4$ grows significantly stronger than for $\omega = 2$. However, the runtime is much shorter than for the integer programs. The run with $\omega = 2, \tau = 1, \kappa = 0.5$ is nearly identical to the run with $\omega = 2, \tau = 1, \kappa = 0$

The objective value of the approximation algorithm for cycle cover differs only slightly from the optimum. For $\omega = 2$ the difference is on average less than 5%, with a maximum difference of 7.3%. The differences are higher for $\omega = 3$ and $\omega = 4$, with a maximum difference of 17.8%, but there are too few instances solved for a reliable statement. The difference is higher for the tour version, but still close to the optimum and far better than the proven bounds. The maximum difference found is 1.485 times the optimal value for $\omega = 4$. It has to be noted that the implementation does not do a local optimization of the cycle connections, i.e., the points of the connection are visited multiple times. A removal of the redundant visits could further improve the ratio.

We further considered random instances with $\omega = 2$, $\tau = 1$, and $\kappa = 0$ up to a size of 2000 points. For instances with 1700 points and more, the memory consumption of the linear program becomes problematic. The average runtime at this point is roughly 8 min.

6 Conclusion

The assumption that the discretization makes the angular metric traveling salesman problem simpler is only partially true. Surprisingly, the integer program becomes very slow already for low resolutions, which is also true for the cycle cover variant. Experiments for the penalty variants are still to be performed. The approximation algorithm is only practical for very low resolutions. The experimental quality of the solutions of the approximation algorithm is considerably better than the worst-case ratio, but this evaluation is only based on few and relatively small instances. It would also be interesting to evaluate the quality of the solutions compared to the optimal solutions of the AM-TSP. The bottleneck of the approximation algorithm is solving the linear program, but often there are possibilities of replacing the blackbox LP-solver by a more direct approach.

References

[1] A. Aggarwal, D. Coppersmith, S. Khanna, R. Motwani, and B. Schieber. The angular-metric traveling salesman problem. *SIAM Journal on Computing*, 29(3):697–711, 2000.

[2] O. Aichholzer, A. Fischer, F. Fischer, J. F. Meier, U. Pferschy, A. Pilz, and R. Stanek. Minimization and maximization versions of the quadratic traveling salesman problem. Preprint of the Institute for Numerical and Applied Mathematics of the University of Goettingen, Mar. 2016.

[3] E. M. Arkin, M. A. Bender, E. D. Demaine, S. P. Fekete, J. S. Mitchell, and S. Sethia. Optimal covering tours with turn costs. *SIAM Journal on Computing*, 35(3):531–566, 2005.

[4] S. P. Fekete and G. J. Woeginger. Angle-restricted tours in the plane. *Computational Geometry*, 8(4):195–218, 1997.

[5] A. Fischer, F. Fischer, G. Jäger, J. Keilwagen, P. Molitor, and I. Grosse. Exact algorithms and heuristics for the quadratic traveling salesman problem with an application in bioinformatics. *Discrete Applied Mathematics*, 166:97–114, 2014.

[6] A. Fischer and C. Helmberg. The symmetric quadratic traveling salesman problem. *Mathematical Programming*, 142(1-2):205–254, 2013.

[7] M. X. Goemans and D. P. Williamson. A general approximation technique for constrained forest problems. *SIAM Journal on Computing*, 24(2):296–317, 1995.

[8] G. Jäger and P. Molitor. Algorithms and experimental study for the traveling salesman problem of second order. In *Combinatorial Optimization and Applications*, pages 211–224. Springer, 2008.

[9] B. Rostami, F. Malucelli, P. Belotti, and S. Gualandi. Quadratic TSP: A lower bounding procedure and a column generation approach. In *Computer Science and Information Systems (FedCSIS), 2013 Federated Conference on*, pages 377–384. IEEE, 2013.