

Technical Report

ETSP on a Polygon in the Presence of a Polygonal Obstacle

Jeff Abrahamson
Ali Shokoufandeh

September 2004

ETSP on a Polygon in the Presence of a Polygonal Obstacle

September 2004

Jeff Abrahamson, Dept. of Computer Science, Drexel University, Philadelphia, PA, USA,
jeffa@cs.drexel.edu.

Ali Shokoufandeh, Dept. of Computer Science, Drexel University, Philadelphia, PA, USA,
ashokouf@cs.drexel.edu.

Abstract

We present a polynomial-time algorithm for a variant of the Euclidean traveling salesman tour where n vertices are on the boundary of a polygon P and m vertices form the boundary of a polygonal obstacle Q completely contained within P . In the worst case the algorithm needs $O(n^2m + nm^2 + m^3)$ time and $O(n^2 + m^2)$ space.

1 Introduction

The Euclidean TSP (ETSP) is the problem of finding a tour of minimum length through a given set of points in d -dimensional Euclidean space. In this paper, we will address a variant of the ETSP in which the points are the vertices of a polygon P and of a polygonal obstacle Q located completely in the interior of P . We assume the exterior of P and the interior of Q are forbidden regions through which the tour may not pass. We give a polynomial time algorithm for finding the shortest tour T through the vertices of P and Q while completely avoiding the interior of Q and the exterior of P .

Deineko et al. [4] considered a related variant of the ETSP with a convex polygon P and a set of points on a line segment Q inside P . They referred to this problem as the *convex-hull-and-line* ETSP, and gave an $O(m^2 + mn)$ time and $O(m + n)$ space algorithm, where n and m are the number of P and Q vertices, respectively. Cutler [3] gave an $O(n^3)$ time and $O(n^2)$ space dynamic programming algorithm for solving the *3-line* ETSP where all points lie on three distinct parallel lines in the plane. Rote [6] extended this result to m -line ETSP by giving a polynomial dynamic programming algorithm for a fixed number of lines m . Deineko et al. [5] addressed the problem of recognizing instances of the ETSP for which there is a permutation of points such that the underlying distance matrix fulfills so called Demidenko, Kalmanson and Supnick properties. It is known that if a distance matrix fulfills one of these properties, the ETSP is solvable in polynomial time. We previously considered a simpler version of this problem [1] in which P and Q are both convex. We gave an $O(m^3 + m^2n \log m)$ time and $O(m^3n)$ space exact solution to that problem.

2 Basic Definitions

In the following we assume a distance function that obeys the triangle inequality and is additive: $d(a, c) = d(a, b) + d(b, c)$ for all co-linear points a , b , and c with b between a and c . Let P denote a polygon in the plane and Q an obstacle that is defined to be completely in the interior of P , both in general position (taken together). Let p_1, \dots, p_n denote the vertices of P and q_1, \dots, q_m be the vertices of Q , with $m, n \geq 3$. Throughout this paper we will assume that vertices are numbered in the clockwise direction and that subscripts are interpreted modulo n for P and m for Q . We will also denote two consecutive P or Q vertices as v and v^+ .

An *edge* uv that connects a pair of points u and v is a straight line segment between u and v of length $\|uv\|$. If the edge between any points u and v avoids the forbidden regions then the two points are said to be *visible* to each other. A *path* is a sequence u_1, u_2, \dots, u_k of vertices and the interconnecting edges $u_1u_2, u_2u_3, \dots, u_{k-1}u_k$. We will often write the path $\pi = u_1 \rightsquigarrow u_k = u_1u_2 \oplus u_2u_3 \oplus \dots \oplus u_{k-1}u_k$, where \oplus denotes edge concatenation.

Consider two paths π_1 and π_2 that share a common vertex v . We say that v is a *vertex intersection* if the two paths cross each other at v . If they touch at v but do not cross, we call v a *touch point*. If some edges $e_1 \in \pi_1$ and $e_2 \in \pi_2$ intersect, we say that π_1 and π_2 have an *edge intersection*. Two paths intersect if they have either a vertex or an edge intersection.

If $\pi = u_1 \rightsquigarrow u_k$ is a path and $u_1 = u_k$, then the path is called a *tour*. A *simple* tour T has no duplicate vertices except the necessary first and last. A tour T is *weakly-simple* if it has no intersections except possibly for backtracking ($\dots \oplus vv' \oplus v'v \oplus \dots$). Thus, both simple and weakly-simple tours have well-defined interiors. Finally, in the context of the problem addressed in this paper, a weakly-simple tour with no P vertices is said to be *degenerate*.

A tour through all vertices of P and Q involves three types of edges: *exterior edges* connecting P vertices, *interior edges* connecting consecutive Q vertices, and *cross-over edges* connecting P vertices with Q vertices or non-consecutive P or Q vertices. (We will see later that non-consecutive P or Q vertices can not be adjacent in a shortest tour.) We define $\|q_i \rightsquigarrow q_j\|$ to be the length of the polygonal path $q_i \rightsquigarrow q_j = q_iq_{i+1} \oplus \dots \oplus q_{j-1}q_j$. Note that the path $q_i \rightsquigarrow q_j$ is clockwise, and that $(q_i \rightsquigarrow q_j) \oplus (q_j \rightsquigarrow q_i) = Q$. We similarly define $p_i \rightsquigarrow p_j$ and its length $\|p_i \rightsquigarrow p_j\|$.

Consider a polygon edge $p_k p_k^+$. We define a clockwise *detour* (cf. Figure 1 (**modify**)) $d_{i,j}^k$ of $p_k p_k^+$, for any pair of (not necessarily distinct) Q vertices q_i and q_j to be the path $d_{i,j}^k = \mathcal{P}(p_k, q_i) \oplus (q_i \rightsquigarrow q_j) \oplus \mathcal{P}(q_j, p_k^+)$, where $\mathcal{P}(u, v)$ denotes a shortest obstacle-avoiding path from u to v .

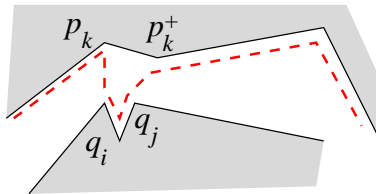


Figure 1: The detour $d_{i,j}^k$ from p_k to p_k^+ through $q_i \rightsquigarrow q_j$.

Note that the path $\mathcal{P}(u, v)$ may traverse points of P and Q and even result, when added to $q_i \rightsquigarrow q_j$ or to the tour sections before p_k or after p_k^+ , in retracing vertices. Two detours are *disjoint* if the sets of their P

and Q vertices are disjoint. The *incremental cost* $c_{i,j}^k$ of the detour $d_{i,j}^k$ is

$$c_{i,j}^k = \|\mathcal{P}(p_k, q_i)\| + \|q_i \rightsquigarrow q_j\| + \|\mathcal{P}(q_j, p_k^+)\| - \|p_k \rightsquigarrow p_k'\|.$$

Let $d_{i,j}$ denote a cheapest clockwise detour through $q_i \rightsquigarrow q_j$ taken over all polygon sections $p_k \rightsquigarrow p_{k'}$. That is

$$d_{i,j} = \arg \min_{p_k} \{c_{i,j}^k\} \quad \text{and} \quad c_{i,j} = \min_{p_k} \{c_{i,j}^k\}$$

Let A be a polygon and denote by $\text{Convex}(A)$ its convex hull. Let a_i and a_j be two consecutive points on $\text{Convex}(A)$. We call the points a_i, a_{i+1}, \dots, a_j together with the segments connecting them a *bay* of A . We say that the segment $a_i a_j$ of the convex hull *crosses the bay*. Cf. Figure 4.

3 Structural Properties

In this section we provide a structural characterization of a shortest tour T through the vertices of P and Q . It will facilitate a transformation of the original problem to $O(m)$ shortest paths problems in an appropriately defined digraph.

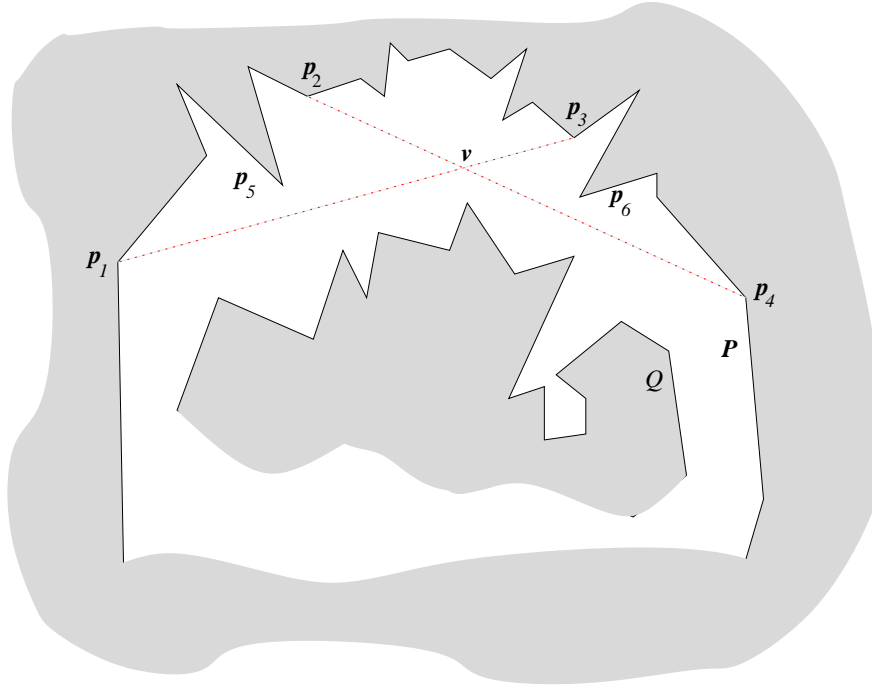


Figure 2: Lemma 1: Replacing p_1p_3 and p_2p_4 with p_1vp_2 and p_3vp_4 results in a tour of the same length. Relaxing to $p_1p_5p_2$ and $p_3p_6p_4$ results in a shorter tour.

Lemma 1 *An optimal tour T has intersections only on P and Q vertices.*

The proof follows similar reasoning to the arguments given in [1] with the addition of intermediate points when points are not visible.

Proof: Assume to the contrary that the optimal tour T contains an intersection in the region between P and Q . Suppose without loss of generality that all four points are on P and are called $p_1, p_2, p_3,$ and p_4 and call their point of intersection v . Suppose the shortest tour contains segments p_1p_3 and p_2p_4 , as shown in Figure 2.

Then replacing p_1p_3 and p_2p_4 with either p_1vp_2 and p_3vp_4 or with p_1vp_4 and p_2vp_3 results in a tour of the same length, one of which must still be connected. Relaxing the tour away from v to the convex hull of the points in the triangles (in the diagram to p_5 and p_6) results in a shorter tour.

The cases that any or all of the points are on Q proceed by identical reasoning, although the diagram and the labeling of points changes to reflect where the points lie. \square

It would appear that for an optimal solution to the problem of finding a tour through the vertices of polygons P and Q the tour must visit each vertex exactly once. In [1] we showed that in the case of P and Q convex, some tours that visit some vertices of Q more than once may be shorter than tours that visit each Q vertex precisely once. In the case of arbitrary P and Q , moreover, even P vertices may be visited more than once in a shortest tour, as Figure 3 shows.

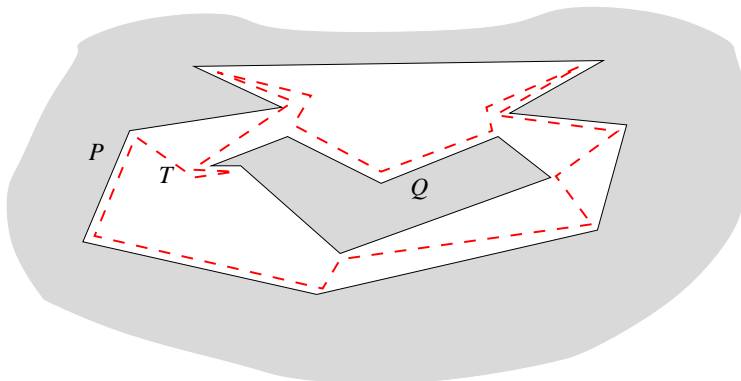


Figure 3: A shortest tour might visit some P vertices more than once.

When P is convex, optimal solutions can be characterized as consisting of detours to Q vertices between appropriately chosen pairs of consecutive P vertices. As Figure 3 shows, this is no longer the case when P and Q are not convex. A similar result is still true, however:

Corollary 2 *There exists a shortest tour T on which P vertices preserve their cyclic order after all sub-tours of the form $\dots pp'p \dots$ are removed.*

That is, if we remove backtracking on P , the order of vertices on P is strictly cyclic.

Proof: Assume the contrary. Then the tour must intersect itself in the space between P and Q , which is forbidden by Lemma 1. \square

Detours in which the orientation on Q differs from the orientation on P are still special by a variant on the same proof as in [1]:

Lemma 3 *If a shortest path, ignoring backtracking, is clockwise on P but counterclockwise on Q , then T has precisely one detour from P to Q .*

Proof: By the Jordan curve theorem and Lemma 2, the detour on Q either covers all of Q or else it leaves vertices on Q which can not be reached by T without crossing the given detour. Since T visits all vertices of P and Q , the single detour covers all of Q . \square

When P is convex, we can partially compensate for the concavity of Q by computing the shortest tour on $P \cup \text{Convex}(Q)$ and repairing the bay crossings.

Lemma 4 *Let P be a convex polygon and Q a simple polygon contained within P . Let T be a shortest tour on $P \cup \text{Convex}(Q)$. Suppose that some segment of T crosses a bay $B \subset Q$. Then there exists a shortest tour T' on $P \cup Q$ that follows the points of B .*

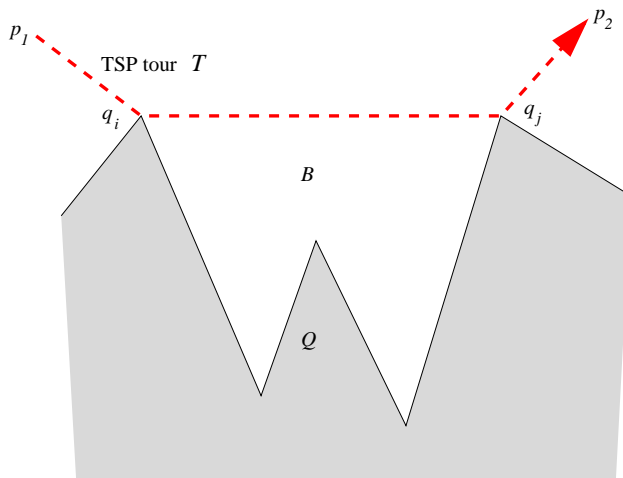


Figure 4: In the special case that P is convex, the case that the optimal tour of $P \cup \text{Convex}(Q)$ crosses a bay.

Proof: Suppose without loss of generality that the bay B and the tour T are labeled as in Figure 4. The structural lemmas of [1] tell us that p_1 and p_2 must be adjacent on P . Suppose that some tour \mathcal{T} on $P \cup Q$ is shorter than any tour that follows $p_1 q_i q_{i+1} \dots q_j p_2$. Suppose that \mathcal{T} enters B at some point other than q_k . Then $q_i \rightsquigarrow q_{k-1}$ must return to P , and straightforward application of the triangle inequality leads to a contradiction. \square

Unfortunately, the related structural lemma we would need to extend our results directly—that is, by fixing each bay—does not hold, even in the case that P is convex. Suppose T is a shortest tour on $P \cup \text{Convex}(Q)$ and that some segment of T returns to P rather than cross a bay $B \subset Q$ (Figure 5). Fixing B is not in general any easier than solving the general channel ETSP, as Figure 6 shows. Citation for channel ETSP hardness?

Because repairing bay crossings doesn't work in general, we do not consider the special case of convex P further.

4 Algorithm

Let P and Q be arbitrary simple polygons with Q entirely in the interior of P . In [1] we computed minimum detours $d_{i,j}$ from some p_k to q_i to q_j returning to p_k^+ . We were able to ignore the cost of the implicit

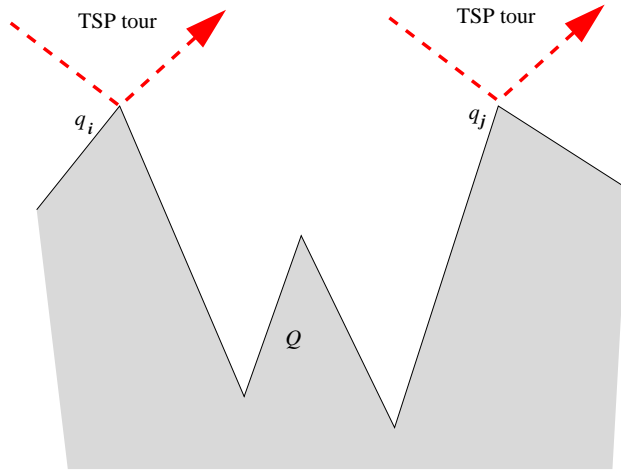


Figure 5: In the special case that P is convex, the case that the optimal tour of $P \cup \text{Convex}(Q)$ does not cross some bay.

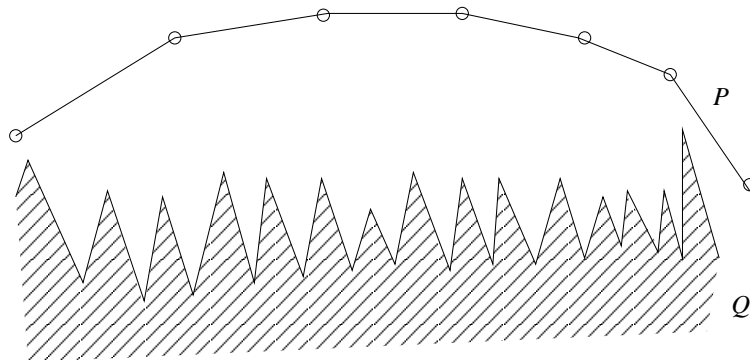


Figure 6: A bay with $O(m)$ peaks and $O(n)$ vertices in P above..

minimization in the $\mathcal{P}(p, q)$ terms, since the convexity of P and Q guaranteed that the furthest visible point in each direction moved monotonically. Without convexity, we must compute the set of visible points anew for each vertex $p \in P$. Indeed, some points of P may be visible from no point of Q and vice versa.

In practice we can be clever by computing the convex hull of Q and noting that no point inside (meaning not protruding from) a bay of Q both of whose endpoints are not visible to some p is itself visible to p . Nonetheless, in the worst case (cf. Figure 6), $O(m)$ vertices of Q may be visible to $O(n)$ points of P .

By Lemma 3 we must consider two cases: the shortest tour with same orientation on P and Q and the shortest tour with opposite orientations. Suppose first that the tour has the same orientation on P and Q .

Let $G = (V, E)$ be a directed weighted graph with $V = \{x_1, \dots, x_m, y_1, \dots, y_m\}$ conceptually equal to the vertices of Q repeated twice, with q_i represented at x_i and again at y_i . The edges (v_i, v_j) have weight $w(i, j)$, where $w(x_i, y_j) = c_{i,j}$ and $w(y_i, x_j) = 0$ if $j = i$ or $j = i + 1$ (and ∞ otherwise). As noted earlier, subscripts are interpreted modulo m and $c_{i,j}$ is the cost of the cheapest detour $d_{i,j}$, as defined in Section 2. Then one of the m shortest paths from x_i to y_{i-1} corresponds to a shortest obstacle avoiding tour of P and Q :

Theorem 5 *For each $1 \leq h \leq m$ let π_h be the shortest path from x_h to y_{h-1} . Let π be the π_h with shortest length. Then π corresponds to a shortest same-orientation obstacle avoiding tour through all P and Q vertices.*

Proof: By Lemma 1, the path T corresponding to π is a weakly-simple tour. If a tour of shorter length on the vertices of P and Q existed, it would have a representation as a shortest path between some $x_{h'}$ and $y_{h'-1}$ in G . By construction, therefore, T is the shortest same-orientation tour as claimed. \square

Suppose, on the other hand, that the tour on Q has opposite orientation than on P . By Lemma 3, a shortest tour T will omit precisely one edge of P and one edge of Q . We need, therefore, only consider the shortest detour for each candidate edge of Q .

The shortest tour, then, is the shorter of the shortest same-orientation tour and the shortest opposite-orientation tour.

5 Complexity

The algorithm in Section 4 leads us down two paths: finding the shortest same-orientation tour and finding the shortest opposite orientation tour. In the former case, we first find the least cost detours for each pair of Q vertices, then we find m shortest paths in a graph on $2m$ vertices. In the latter case we find and compare m shortest detours.

Theorem 6 *Finding the shortest tour of P and Q vertices requires $O(n^2m + nm^2 + m^3)$ time and $O(n^2 + m^2)$ space.*

Proof: The cost of computing the set of visible points for each $p \in P$ is $O(m(n+m))$ [2], and so $O(nm(n+m))$ for all p . Computing the points of P visible to vertices of Q is trivial, on the other hand, once we have computed Q 's visibility from P , since the visibility graph is undirected. Computing visibility thus costs $O(nm(n+m))$. The space requirements are $O((n+m)^2)$ to store the edges of the graph.

Since we must consider $\Theta(nm^2)$ detours, the total cost of computing detours is $\Theta(nm^2)$. We use $O(m^2)$ space to store the values, since we minimize $c_{i,j}^k$ over k and so only store the $c_{i,j}$ and $d_{i,j}$.

Finding the shortest same-orientation detours requires $O(m^3)$ time and $O(m^2)$ space to find the m shortest paths (using, for example, the Floyd-Warshall algorithm).

The total complexity is thus $O(nm^2 + m^3 + nm(n+m)) = O(n^2m + nm^2 + m^3)$ time and $O((n+m)^2 + m^2) = O(n^2 + m^2)$ space for the same-orientation shortest-tour.

To find the shortest opposite-orientation tour we compute the visibility graph and then compare m detours to Q for each of n adjacent vertices of P , using $O(nm)$ time and constant space (since we only need store the shortest). The total cost is thus $O(nm(n+m) + m^3) = O(n^2m + nm^2 + m^3)$ time and $O((n+m)^2) = O(n^2 + m^2)$ space. \square

Acknowledgments

The authors thank Pawel Winter of the University of Copenhagen for suggesting the initial idea for this problem. We also thank M. Fatih Demirci of Drexel University for his helpful suggestions. This work was funded in part by a grant from the Office of Naval Research (ONR-N000140410363).

References

- [1] J. Abrahamson, A. Shokoufandeh, P. Winter, Touring Vertices of a 2-Dimensional Closed Convex Channel, to appear, hopefully.
- [2] M. de-Berg et al., Computational Geometry: Algorithms and Applications, Springer, Berlin, 1998.
- [3] M. Cutler, Efficient special case algorithms for the n -line planar traveling salesman problem, *Networks* **10** (1980) 183-195.
- [4] V.G. Deineko, R. van Dal and G. Rote, The convex-hull-and-line traveling salesman problem: a solvable case, *Inf. Proc. Lett.* **51** (1994) 141-148.
- [5] V.G. Deineko, R. Rudolf, J.A.A. van der Veen and G.J. Woeginger, Three easy special cases of the Euclidean traveling salesman problem, Tech. Report 17, Technische Univ., Graz, Austria (1995).
- [6] G. Rote, The N -line traveling salesman problem, *Networks* **22** (1992) 91-108.