The Minimum Length Separating Cycle Problem

Esther M. Arkin* Jie Gao† Adam Hesterberg‡ Joseph S. B. Mitchell*
Jiemin Zeng†

1. INTRODUCTION

Given a set S of pairs of points in the plane, we seek a minimum-length separating cycle T having exactly one point of each pair within T. Such a tour T is said to be a *separating cycle*. By minimality, T will be polygonal. The minimum-length separating cycle (MLSC) problem is clearly NP-hard (from TSP). We prove hardness of approximation and provide approximation algorithms for various cases.

We note that if the pairs of points are *colored*, one red and one blue, and the goal is to find a minimum-length red-blue separator, the problem is very different and techniques of Arora and Mitchell apply directly to yield a PTAS [1, 2, 6]. The challenge in the MLSC problem is that we do not know the coloring.

MLSC is also related to the traveling salesperson problem with neighborhoods which has a PTAS for fat regions with bounded depth [3, 5, 7]. Another related problem is the group (or one-of-a-set) traveling salesperson problem where given a set of points in the plane and regions covering subsets of points, the aim is to construct a minimum length path that visits at least one point in each region [4].

2. HARDNESS OF APPROXIMATION

We show that the MLSC with unit length segments is APX-hard by a reduction from MAX NAE 2-SAT. When there is no restriction on the distance between pairs of points, the MLSC problem is NP-hard to approximate within a factor of ≈ 1.012987. We omit the proofs in this abstract.

3. APPROXIMATION ALGORITHMS FOR SPECIAL CASES

We describe our constant approximation algorithm for the problem when the input segments are unit length horizontal or vertical segments.

3.1 Board size $2 - \varepsilon$

Board Size	Unit Length Horizontal	Unit Length Horizontal and Vertical	Unit Length Arbitrary Orientation
$2-\varepsilon$	in P	4-approx	NP-hard
$M = O(1)$	$O(1)$-approx	(M^2+1)-approx	NP-hard to approx
n	$O(1)$-approx	$O(1)$-approx	NP-hard to approx

Table 1: Hardness and Approximation algorithm results for different settings.

We will first start with the scenario in which all n segments are inside a square board of size $2 - \varepsilon$ for some $\varepsilon > 0$. Without loss of generality we assume all the endpoints of input segments do not share a common x-coordinate or y-coordinate. This can be done by perturbing the input slightly.

In our algorithm we first find all unit length boxes which contain *at least* one endpoint of each segment. In fact, we can test this in polynomial time by checking all possible combinatorial configurations of unit length boxes. The total number of combinatorial types of such squares is $O(n^2)$, since we can assume without loss of generality that the square always has at least two input endpoints on its boundary. Each such box actually contains *exactly* one endpoint of each input segment – as the two endpoints of one segment cannot both fit inside the box. Then we consider the convex hull of all endpoints inside the box to be one candidate separating cycle. We can enumerate all such boxes to find the separating cycle T with minimum length.

Now consider the optimal solution T^*, there can be two possibilities – (i) T^* fits inside a unit length box, or (ii) it does not. If T^* fits inside a box say B, this means any segment has at least one endpoint inside B and B will surely be discovered by our algorithm. Further B has exactly one endpoint and the convex hull of the points inside B is no longer than T^* and is optimal. So we are done.

In the following we will focus on case (ii) when T^* cannot fit inside any unit length box. This means T^* has length at least 2.

Now we divide the board into four squares each of size $1-\varepsilon/2$ and color them in a checkerboard pattern. We name these squares S_1, S_2, S_3, S_4 in a counter clockwise manner. Take the two squares along the diagonal (named S_1 and S_3, see Figure 1 (ii)) and take a tour along the perimeter of their

*Department of Applied Mathematics and Statistics, Stony Brook University, Stony Brook, NY 11794. Partially supported by NSF (CCF-1526406). Email: esther.arkin@stonybrook.edu, jsbm@ams.stonybrook.edu.
†Department of Computer Science, Stony Brook University, Stony Brook, NY 11794. Email: {jgao, jiezeng}@cs.stonybrook.edu.
‡Department of Mathematics, Massachusetts Institute of Technology, Boston, Massachusetts. Email: achester@mit.edu

union. This generates a curve T' of length $8 - 4\varepsilon$. We now argue that T' is a valid separating cycle.

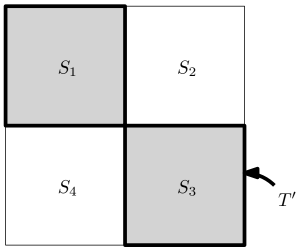

Figure 1: $2 - \varepsilon$ square board with constant approximation curve.

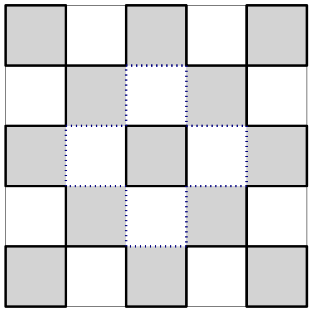

Figure 2: Constant sized square board with cycles along the perimeter of the dark squares.

Since each segment has unit length, the two endpoints of each segment cannot fit inside any single square of size $1 - \varepsilon/2$ and thus cannot both lie inside $S_1 \cup S_3$ nor inside $S_2 \cup S_4$. Therefore each segment must have exactly one endpoint inside $S_1 \cup S_3$. This proves that the cycle T is a valid separating cycle.

Now we want to analyze the approximation factor. T' has length $8 - 4\varepsilon$ and the optimal tour has length at least 2. Thus T' is a 4-approximation of the optimal solution.

3.2 Constant Board Size

We can extend the above argument for any constant board size M, where M is an integer. Then we use essentially the same algorithm as above. The only modification is that we will use a bigger checkerboard of $M \times M$ unit squares. The squares are colored in a checkerboard pattern, and partitioned into white squares or dark squares. To make the tiling a perfect partition of the plane, we consider each square pixel to include its top edge, except for the NE corner, and to include its left edge, except for the SW corner; then, the pixels partition the plane. Again any unit length vertical or horizontal segment has two endpoints in different colored squares. Thus if we take a tour T' that separates the white squares from the dark squares, T' would be a valid separating cycle.

Again either we can find a unit length box containing at least one endpoint of each segment (in which case we find the optimal), or the optimal tour has length at least 2. In the second case we will take the tour T' along the perimeter of the union of the dark squares. To see that one can always find such a tour we can first take a cycle along the boundary of the outermost ring of the dark squares, and iterate towards the center. All the tours can be joined into a single tour of the same length. See Figure 2 for an illustration.

T' has length $4 \cdot M^2/2$ if M is even, and $4 \cdot (M^2+1)/2$ if M is odd. Thus the approximation factor is roughly (M^2+1).

3.3 Any Board Size – A Better Algorithm

If M is a large number then the approximation ratio is not good. Here we use a different algorithm and provide a constant approximation bound for any size board.

We refer to dark squares that have a point (from a pair) in them as "occupied". Let S be the occupied squares. Let S' be the 3-by-3 squares centered on the squares of S. In the following we assume without loss of generality that $|S| \geq 5$. The case $|S| \leq 4$ can have an arbitrarily small optimal value; but this constant-size case can be easily handled.

Now we consider the shortest TSP with Neighborhoods tour on the set S' of enlarged squares and name the length as $TSPN(S')$.The following claim is now straightforward since the optimal solution OPT must visit each of the enlarged squares S'.

Claim 1: $OPT = \Omega(TSPN(S'))$.

Now we have

Claim 2: $OPT = \Omega(|S|)$, assuming $|S| \geq 5$.

The proof of Claim 1 follows from Claim 2, since, assuming no single point stabs all squares of S' (i.e., assuming $|S| > 4$), the standard packing argument shows that TSPN on a set of nearly disjoint (i.e., constant depth of overlap), equal-sized squares requires length proportional to the number of squares times the side length of the squares.

Our algorithm is simply this: Run a TSPN algorithm on S' (for which there is a PTAS [3]), and traverse the boundaries of the red pixels S, enclosing them, while excluding the white region.

The length of this tour is then at most $(1+\epsilon)TSPN(S') + O(|S|)$, which is $O(OPT)$.

4. REFERENCES

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

[2] S. Arora and K. Chang. Approximation schemes for degree-restricted mst and redblue separation problems. *Algorithmica*, 40(3):189–210, 2004.

[3] A. Dumitrescu and J. S. Mitchell. Approximation algorithms for {TSP} with neighborhoods in the plane. *Journal of Algorithms*, 48(1):135 – 159, 2003. Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms.

[4] K. Elbassioni, A. V. Fishkin, N. H. Mustafa, and R. Sitters. Approximation algorithms for euclidean group tsp. In *Automata, Languages and Programming*, pages 1115–1126. Springer, 2005.

[5] C. Feremans and A. Grigoriev. Approximation schemes for the generalized geometric problems with geographic clustering. In *EWCG 2005*. Citeseer, 2005.

[6] J. S. B. Mitchell. Guillotine subdivisions approximate polygonal subdivisions: A simple polynomial-time approximation scheme for geometric tsp, k-mst, and related problems. *SIAM Journal on Computing*, 28(4):1298–1309, 1999.

[7] J. S. B. Mitchell. A ptas for tsp with neighborhoods among fat regions in the plane. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, pages 11–18, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.