Sweeping Polygons with a Variable-Length Line Segment

Kien C. Huynh
Department of Computer Science
Stony Brook University
USA
kchuynh@cs.stonybrook.edu

Joseph S. B. Mitchell
Department of Applied Mathematics and Statistics
Stony Brook University
USA
joseph.mitchell@stonybrook.edu

Abstract

We study the problem of sweeping every point in a polygon using a variable-length line segment. The entire line segment is constrained to stay inside the polygon at all times. The endpoints can move with different (bounded) speeds. Our objective is to minimize makespan: find a pair of trajectories for the endpoints so that the sweeping can be concluded as soon as possible. If the polygons are simple, we present a polynomial time $(4 + \epsilon)$-approximation algorithm.

1 Introduction

Given a polygon P in the plane, we want to sweep every point in it using a line segment. A point is consider swept only if it is touched by the segment. One can consider there to be two agents moving within P that are required to remain covisible at all times; the agents are the endpoints of a sweeping line segment, which is required to sweep over every point of P. Initially, the agents are assumed to start at the same position, $s \in P$. Without the loss of generality, we can also assume that each endpoint can move with speed at least 0 (stationary) and at most 1. The general goal is to find a good sweeping solution that covers every point in P while minimizing some cost functions. These can be overall sweeping duration, the maximum distance traversed between the two agents, or total distance traveled. In this abstract, we are focusing on finding a short makespan, i.e. minimizing the sweeping duration.

This problem is motivated by the problem introduced in [1], which involves sweeping a polygon with a connected chain of guards. The main difference is, in that problem, the guards at the beginning and the end of the chain must always stay on the boundary of P at all times. With such a setup, it is possible for the chain to always catch any moving target inside the polygon, as the chain maintains a "cleaned" portion of the domain.

Our current problem is similar but with limited resource: we only have two agents (guards), but we still want to efficiently sweep the polygon, searching for a static target. A closely related problem is to minimize the sum of distances traveled by two endpoints, which can be solved in polynomial time [4]. The same authors later provided an $O(n^2)$ algorithm to minimize the maximum length of the line segment [5]. Another related problem is the watchman route problem, in which an agent moves in order to see every point in the domain; see, e.g., [2].

In this abstract, our main contribution is a polynomial time approximation algorithm for any simple polygon. We also present our ideas behind proving the hardness of the version with holes and finding an approximation solution for the case with multiple line segments.

2 Minimizing makespan in a simple polygon

In this version, we want to minimize the overall time that is needed for the agents to finish sweeping every point in P. This can be seen as minimizing the maximum time spent by each agent (endpoint of the segment) during the sweep schedule.

We begin with a straightforward observation:

Lemma 2.1. *To completely sweep P, it is necessary that all convex vertices must be visited by at least one endpoint.*

This also applies to polygon with holes. The same is not true for reflex vertices, as those can be swept by the interior of a sweeping segment (when the reflex vertex falls interior to the segment, and the segment becomes tangent at the vertex). Moreover, it is possible for the line segment to sweep an edge of P without requiring any endpoint to move along that edge. For example, the line segment can already be parallel to an edge, to sweep that edge each endpoint only needs to move to its closest vertex of that edge. However, in any optimal solution, the segment can only sweep an edge in such a way at most once, which leads to the following lemma:

Lemma 2.2. *Let T be the minimum geodesic Steiner tree spanning every convex vertex of P, then $OPT \geq \frac{1}{2}|T|$.*

Proof. Since the line segment starts at the same location at the beginning, the trajectories of the two endpoints would make a connected graph. Using Lemma 2.1, this graph must connect all convex vertices. It follows that $OPT \geq \frac{1}{2}|T|$ by the definition of T. □

Before introducing our approximation algorithm, the following definition is necessary:

Definition 2.1. *A polygon is called a* crescent *if it consists of one convex chain and one reflex chain, both chains connect at their two vertices at the ends.*

Such a polygon can be swept by one segment in time at most equal to the length of the convex chain, i.e. the longer chain. To sweep it, the segment will start at one of the shared vertices of the chains and will continue sweeping until it reaches the other vertex. In the entire motion, one endpoint of the segment will always stay on the convex chain and the other on the reflex chain. This type of polygon is employed in proving the following theorem:

Theorem 2.1. *There exists a $(4 + \epsilon)$-approximation algorithm that can be computed in polynomial time.*

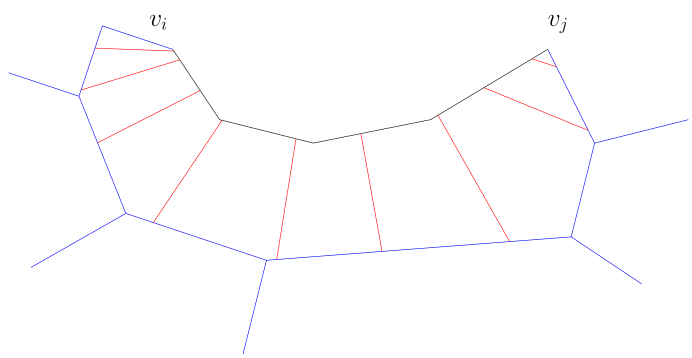

Figure 1: A crescent formed by the minimum geodesic Steiner tree T, colored in blue, and a reflex chain $v_i v_j$, colored in black. This type of subpolygon will always be sweepable by the line segment, depicted in red, with one endpoint moving along the path of T connecting v_i to v_j and the other along the corresponding path of the boundary.

Proof. Recall that T is the minimum geodesic Steiner tree that spans every convex vertices of P. Any two consecutive convex vertices v_i, v_j (with a possible reflex chain in between), along with the edges of T connecting v_i to v_j, will form a crescent. See Figure 1 for an illustration. The reason is that the path in T connecting the two convex vertices will always be left-turning (if we are traversing the polygon in clockwise order) since T is a minimum Steiner tree. The subpolygon made of this path of T, which is a convex chain, and the reflex chain from v_i to v_j will be a crescent.

Using the FPTAS from [3] we can compute a $(1 + \epsilon')$-approximate minimum Steiner tree spanning all convex vertices in any simple polygon. The running time is $O(n_{\epsilon'} k^2 (n_{\epsilon'} + k))$, where $n_{\epsilon'}$ is the number of vertices of the polygon plus the number of grid points used in the approximation and k is the number of convex vertices. In case a subpolygon is not a crescent, i.e. there exists a path from v_i to v_j that is not always left-turning, then the corresponding part of the tree is locally optimal. We can fix that by adjusting any right-turning vertex so that it is left-turning, or at least colinear with the two vertices before and after it.

Our sweep schedule will be as follows: starting from any convex vertex, the two agents will sweep from one convex vertex to the next using the corresponding walkable subpolygon generated by the above method. One agent will always follow the edges of T while the other agent will follow the boundary of the polygon. The total duration of this sweeping schedule will be the time it takes an agent, moving at full speed, to traverse twice the length of the tree: $2(1 + \epsilon')|T| \leq 4(1 + \epsilon')OPT$. If we set $\epsilon' = \epsilon/4$ then the theorem follows. □

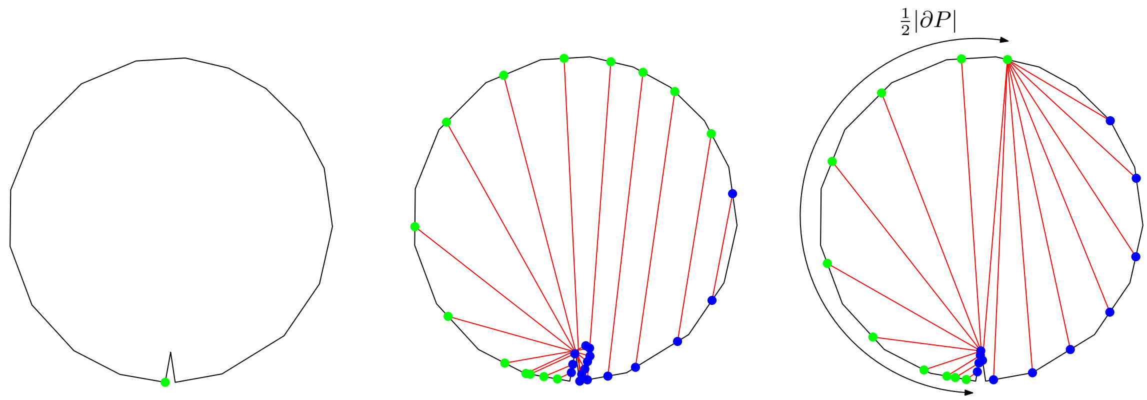

Figure 2: An example in which the optimal solutions for the makespan minimization, shown in the middle, and maximum distance minimization, shown on the right, are different. This assumes that both endpoints start at the same location, depicted as a green dot in the left figure. In the makespan minimization solution, the green endpoint will continue to move even if its traveled distance is already longer than $|\partial P|/2$. In contrast, if minimizing the maximum distance traveled by an agent, the solution has each agent traveling distance $|\partial P|/2$, while the overall duration of the sweep is greater than that of the minimum makespan solution.

It is interesting to note that minimizing the makespan of the sweeping motion is not the same as minimizing the maximum distance traveled by each endpoint. This can be seen in the example shown in Figure 2. The problem of min-maxing the distance traveled can be useful if we consider each endpoint as an agent with limited fuel, in that case we might not want one agent to travel much more than the other.

3 Sweeping a polygon with holes

The case of sweeping a polygon with holes is work in progress. In the talk and full paper, we expect to be able to show:

Claim 3.1. *The problem of minimizing the makespan to sweep a polygon with holes is NP-hard, even if the holes are convex.*

The (still incomplete) proof attempt uses a reduction from Hamiltonian path in a (triangular) grid graph.

Given the hardness of this case, we explore polynomial time approximation algorithms, based on lower bounds in terms of the total length of the boundary of the domain, and a minimum Steiner tree that spans all holes.

4 Sweeping with multiple line segments

In this problem, we are given k segments ($2k$ endpoints), each of which can start at any point on the boundary of P. We again seek to minimize the makespan of the sweeping schedule, i.e. minimizing the maximum time traveled by any endpoint of any segment. If we apply the reasoning similar to Lemma 2.2, then one lower bound of the optimal solution to this problem is $(|\partial P| - \sum_{e \in E_k} |e|)/(2k)$ where E_k is the set of k largest edges of P. This bound implies that we can find k largest edges that also partition the boundary of P into k parts of equal length. But that might not always be true, hence we believe that this bound could be improved.

Our approach to approximating OPT is to reuse the result in Theorem 2.1. We can partition the minimum geodesic Steiner tree T into k smaller trees of the same length. For each such tree, we can apply the same motion plan for each line segment and let them sweep from one crescent to another.

References

[1] Alon Efrat, Leonidas J Guibas, Sariel Har-Peled, David C Lin, Joseph SB Mitchell, and TM Murali. Sweeping simple polygons with a chain of guards. In *Proceedings of the eleventh annual ACM-SIAM symposium on Discrete algorithms*, pages 927–936, 2000.

[2] Joseph SB Mitchell. Approximating watchman routes. In *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*, pages 844–855. SIAM, 2013.

[3] J Scott Provan. An approximation scheme for finding steiner trees with obstacles. *SIAM Journal on Computing*, 17 (5):920–934, 1988.

[4] Xuehou Tan and Bo Jiang. Optimum sweeps of simple polygons with two guards. In *International Workshop on Frontiers in Algorithmics*, pages 304–315. Springer, 2010.

[5] Xuehou Tan and Bo Jiang. Minimization of the maximum distance between the two guards patrolling a polygonal region. In *Frontiers in Algorithmics and Algorithmic Aspects in Information and Management*, pages 47–57. Springer, 2012.