Approximating Maximum Flow in Polygonal Domains using Spanners

Joondong Kim * Joseph S.B. Mitchell * Jingyu Zou *

Abstract

We study a maximum flow problem in a polygonal domain P: Determine the maximum number of disjoint "thick" paths (of specified width w) through P from a source edge to a sink edge of P.

We show that Euclidean spanners offer a means of computing approximately optimal solutions. For a polygonal domain with n vertices and h point holes, we give a 1/2-approximation algorithm that runs in time $O(n + h\log(nh))$; this is to be contrasted with the known exact methods that take time $O(nh + n\log n)$. Further, we show experimentally that using a spanner (e.g., Delaunay graph) yields approximation ratios very close to one.

1 Introduction

We consider the problem of routing multiple disjoint "thick" paths through a polygonal domain in the plane. The problem arises in various application domains, including VLSI wiring, robotics, sensor networks, and air traffic management (ATM). Our motivation comes from an ATM application in which the goal is to compute the "capacity" of an airspace: find the maximum number of disjoint "air lanes" avoiding hazardous weather and other "constraints" (obstacles) within an airspace of interest (e.g., a "flow-constrained area") [7, 8]. The goal is to provide computer-automated decision support tools to perform "capacity estimation" on an airspace to determine its maximum throughput, which measures how constrained the airspace is.

Problem Formulation. The input to our problem is a polygonal domain P, consisting of an outer polygon and a set H of h holes. Let n denote the total number of vertices of P. In this paper we focus on the special case in which H consists of a set of *point holes*; thus, the outer boundary of P is a simple polygon with $n - h$ vertices. This is the special case that arises in our ATM application, since the weather data is typically given as a set of points (pixels) at which there is hazardous weather predicted,

Two edges, Γ_s and Γ_t of ∂P are designated as the *source* and the *sink*. A *w-thick path*, or *lane of width w*, is the Minkowski sum of a usual ("thin") source-to-sink path and a disk of radius $w/2$ centered at the origin. (Refer to [10] for more definitions and background.) We consider the parameter w to be fixed and refer to a w-thick path simply as a *thick path*. Our goal is to compute (approximately) the maximum number of pariwise-disjoint thick paths within P from Γ_s to Γ_t.

Related Work. Algorithms for computing maximum flows and minimum cuts in the continuum (2D geometric domains) were first studied in [9]. Recent results have examined the problems of minimizing path lengths for multiple thick paths (minimum-cost flow) [10] and routing thick paths in dynamic environments (e.g., moving weather systems) [1]. See also [11]. The application of max-flow techniques in ATM is addressed, e.g., in [7, 8].

Summary of Results. Using the propagation algorithm of [9], appropriately modified to handle discrete thick paths (versus continuous flow fields), our optimization problem can be solved exactly in time $O(nh+n\log n)$ (see [1, Thm. 2.1]), for a polygonal domain with h polygonal holes, having a total of n vertices. In this paper we propose a simple 1/2-approximation algorithm for the case that P is a polygonal domain with point holes. Our algorithm searches a Euclidean spanner graph for an approximate min-cut, in time $O(n+h\log(nh))$. We show that this results in an approximation for the problem of maximizing the number of disjoint thick paths. We also conduct experiments, using the Delaunay graph as spanner, to validate the effectiveness of the approximation in practice on both randomly generated data and actual weather data.

*Applied Math and Statistics, Stony Brook University. jdkim,jsbm,jasonzou@ams.sunysb.edu

2 An Approximation Bound

Let B (resp., T) be that portion of ∂P counterclockwise between Γ_s and Γ_t (resp., between Γ_t and Γ_s). We define $G = (V, E)$ where $V = H \cup \{T\} \cup \{B\}$, $E = \{(i,j) | i, j \in V, i \neq j\}$. The weight of edge (i, j) is $c(i,j) = \lfloor d(i,j)/w \rfloor$, where $d(i,j)$ is the (minimum) Euclidean distance between i and j, and w is the path width (thickness). This graph G is called the *thresholded critical graph.* Let G_H be the subgraph of G induced by nodes H. Finally, we let G_H^ε denote the subgraph of G_H whose edge set consists of the union of the set of Delaunay edges of H and the edges of a Euclidean $(1+\varepsilon)$-spanner of H. (Edge weights remain as in G.)

Lemma 1 *For $\varepsilon \leq 0.5$, G_H^ε is a 2-spanner for G_H.*

Proof. It suffices to show that for any edge (u, v) in G_H, there is a path from u to v in G_H^ε of length at most $2 \cdot c(u,v)$. There are two cases to consider,

Case 1: $c(u,v) = \lfloor d(u,v)/w \rfloor = 0$. Since G_H^ε includes Delaunay edges, we know from [3] and [5] that there exists a path π_1 in G_H^ε with vertices $u = v_1, v_2, \ldots, v_k = v$ such that $\forall i, d(v_i, v_{i+1}) \leq d(u,v)$. Since $c(u,v) = 0$ implies that $d(u,v) < w$, we have that $\forall i, c(v_i, v_{i+1}) = 0$, so the path π_1 has zero length in G_H^ε and the claimed stretch factor automatically holds.

Case 2: $c(u,v) > 0$. Since G_H^ε contains an Euclidean $(1+\varepsilon)$-spanner, there exists a path π_2 in G_H^ε such that $\sum_{i=1}^{k-1} d(v_i, v_{i+1}) \leq (1+\varepsilon) d(u,v)$. Thus, the stretch factor of G_H^ε is at most

$$\begin{aligned}\frac{\sum_{i=1}^{k-1} c(v_i, v_{i+1})}{c(u,v)} &= \frac{\sum_{i=1}^{k-1} \lfloor d(v_i, v_{i+1})/w \rfloor}{\lfloor d(u,v)/w \rfloor} \\ &\leq \frac{\lfloor \sum_{i=1}^{k-1} d(v_i, v_{i+1})/w \rfloor}{\lfloor d(u,v)/w \rfloor} \\ &\leq \frac{\lfloor (1+\varepsilon) d(u,v)/w \rfloor}{\lfloor d(u,v)/w \rfloor}.\end{aligned}$$

We show in the appendix (Proposition 3) that if $f(x) = \frac{\lfloor (1+\varepsilon)x \rfloor}{\lfloor x \rfloor}$ and $\varepsilon \leq 0.5$, then $f(x) \leq 2$, for $x > 0$. □

Theorem 2 *The maximum number of disjoint w-thick paths in a polygonal domain with n vertices and h point holes can be $\frac{1}{2}$-approximated in time $O(n + h \log(nh))$*

Proof. G_H^ε is constructed in time $O(h \log h)$, the time needed to build the Delaunay graph or a $(1+\varepsilon)$-spanner (with $\varepsilon = 0.5$ and $O(h)$ edges) for h points. We then construct a graph G^ε from G_H^ε by adding nodes for B and T, and linking these nodes to each point of H. We compute the distance from each point of H to the polygonal chains B and T in time $O(\log n)$ per point of H (after spending time $O(n)$ to construct the Voronoi diagrams, and a corresponding point location data structure, of the simple chains B, T [4]). This augmented graph G^ε has $O(h)$ edges and is a 2-spanner for the critical graph G.

By the continuous max-flow min-cut theorem in [1, 9, 12], we know that the maximum number, OPT, of thick paths from source to sink is equal to the length, $|\pi_G|$, of a shortest path from B to T in G. Since G^ε is a 2-spanner for G, we know that the length, $|\pi_{G^\varepsilon}|$, of a shortest B-to-T path in G^ε is at most $2|\pi_G|$: $OPT \leq |\pi_{G^\varepsilon}| \leq 2 \cdot OPT$, i.e., $(1/2) \cdot OPT \leq (1/2)|\pi_{G^\varepsilon}| \leq OPT$. Thus, $(1/2)|\pi_{G^\varepsilon}|$ is a $\frac{1}{2}$-approximation to the maximum number of source-to-sink thick paths.

Our algorithm takes time $O(h \log h)$ to build G_H^ε, $O(n + h \log n)$ to build G^ε from G_H^ε, and another $O(h \log h)$ to search for a shortest path in G^ε. Altogether, the time bound is $O(n + h \log(nh))$. □

3 Experiments

We did experiments based on computing a specific spanner – the Delaunay graph of the points H. The Delaunay graph is a Euclidean spanner, with stretch factor known to be between 1.581 ($> \pi/2$ [2]) and $\frac{4\pi}{3\sqrt{3}} \leq 2.42$ [6]. Theorem 2 tells us that if we use a spanner with stretch factor at most 1.5 ($\varepsilon = 0.5$), then our approach gives a $\frac{1}{2}$-approximation. Since the Delaunay graph does not have the required property, we do not have a theoretical guarantee that the Delaunay-based results give a $\frac{1}{2}$-aproximation; however, we will see that, in practice, the Delaunay performs very well. In the experiments here, we report only our experience with the Delaunay spanner; further experiments are underway with other spanners.

We use a unit square box as the outer boundary of the polygonal domain P, and use two types of input data for the point holes H: (1) uniformly generated points, and (2) real weather data, scaled to the unit square. For both sets of input data, we examine the relationship between the stretch factor (ratio $|\pi_{G^\epsilon}|/|\pi_G|$) as a function of the "average density" of the point set H. We define the *average density* of the points H to be the average edge length in the nearest neighbor graph of H; the smaller this average edge length is, the denser the point set is.

In the experiments with random point sets, we vary the number, h, of points and the lane width, w.

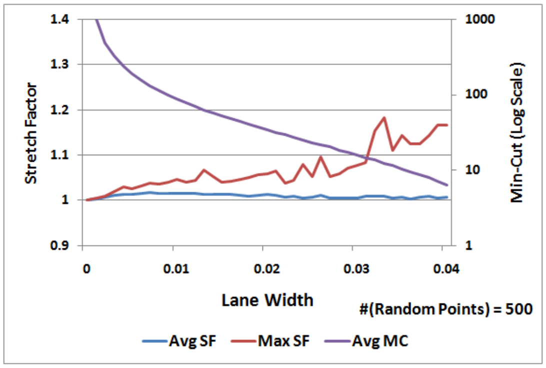

Figure 1: Average and maximum stretch factor (SF) and length of min-cut (MC) as a function of lane width w for random points.

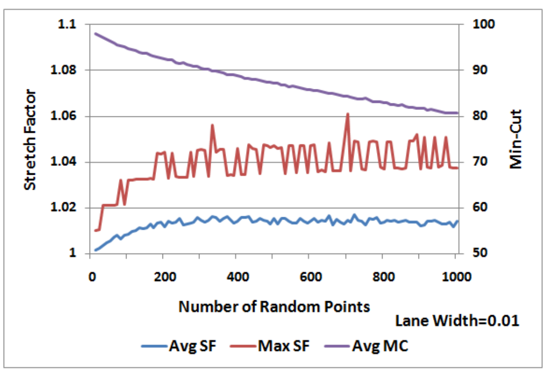

Figure 2: Average and maximum stretch factor (SF) and length of min-cut (MC) as a function of the number of random points.

For the experiments with varying lane width w, we fix the number, $h = 500$, of random points and vary w from 0 to 0.4. For each w, we generated 100 random instances and compiled simple statistics – the average and the maximum stretch factor. Figure 1 shows that the stretch factor is usually very close to 1. Even the maximum stretch factor (over all 100 instances) is low when the min-cut length is large. For example, if the min-cut length is greater than 20, then the maximum stretch factor is always less than 1.1. This means that the difference between exact min-cut length and our approximation is less than 2 even in the worst instance.

For the experiments with varying number h of points, we fix the width, $w = 0.01$, and vary h from 0 to 1000 in increments of 10. For each h, we generated 100 random instances and recorded the average and maximum stretch factor. Figure 2 shows that stretch factor is close to 1 in most cases. The stretch factor is close to 1 even when w is relatively small (i.e., the min-cut is large).

	h	min-cut	avg dist	stretch factor
Set1	576	179	0.0039	1.0056
Set2	502	182	0.0039	1.0110
Set3	430	184	0.0048	1.0163
Set4	752	177	0.0038	1.0113
Set5	820	180	0.0028	1.0000

Table 1: Results for real weather data with $w = 0.005$. Here h is the number of point obstacles and "avg dist" is the average nearest neighbor edge length.

Table 1 shows the stretch factor data for real weather data. Since real weather tends to have clusters of weather points (pixels), the average nearest neighbor distance is much smaller than for random point sets; accordingly, we set the lane width to be very small, $w = 0.005$. (For $w = 0.05$, we found that the stretch factor is always 1.) The results show that the stretch factor is very close to 1, even if the average nearest neighbor distance is comparable to w.

4 Conclusion

Our goal has been to explore the use of spanners in computing approximations for maximizing the number of pairwise disjoint thick paths that can be routed through a polygonal domain in the plane. The advantage of using a spanner (e.g., Delaunay graph) for computing minimum cut values approximately is that it gives a linear space and near-linear time simple algorithm in place of the far more complex exact $O(nh + n \log n))$ algorithm, or the naive $O(n^2)$ algorithm that is easiest to implement. We have seen experimentally that the Delaunay graph does very well in most cases; thus, the Delaunay-based approximation is likely an effective and practical means of doing capacity estimation for ATM.

The exact min-cut problem is a shortest path problem in the plane that may be solvable in $O(n \log n)$ exactly, e.g., by a variant of the continuous Dijkstra paradigm that solves obstacle-avoiding shortest paths in $O(n \log n)$ time. Also, we are examining possible improved approximations possible using spanner techniques, and we are now developing algorithms to produce a set of disjoint thick paths that achieve the capacity determined by our approximation algorithms. Finally, we are doing further experimentation with other Euclidean spanner graphs, with stretch factors approaching 1.

Acknowledgements. The data used for the experiments was provided by Metron Aviation. We thank Valentin Polishchuk and Girishkumar Sabhnani for helpful discussions. This work was partially supported by NSF (CCF-0528209, CCF-0729019), NASA Ames, and Metron Aviation.

References

[1] E. M. Arkin, J. S. B. Mitchell, and V. Polishchuk. Maximum thick paths in static and dynamic environments. In *Proceedings 24th ACM Symposium on Computational Geometry*, pages 20–27, 2008.

[2] P. Bose, L. Devroye, M. Löffler, J. Snoeyink, and V. Verma. The spanning ratio of the Delaunay triangulation is greater than $\pi/2$. In *Proc. 21th Canadian Conference on Computational Geometry*, 2009.

[3] P. Bose, A. Maheshwari, G. Narasimhan, M. Smid, and N. Zeh. Approximating geometric bottleneck shortest paths. *Computational Geometry*, 29(3):233–249, Nov. 2004.

[4] F. Y. Chin, J. Snoeyink, and C. A. Wang. Finding the medial axis of a simple polygon in linear time. In *ISAAC '95: Proceedings of the 6th International Symposium on Algorithms and Computation*, pages 382–391, London, UK, 1995. Springer-Verlag.

[5] D. P. Dobkin, S. J. Friedman, and K. J. Supowit. Delaunay graphs are almost as good as complete graphs. *Discrete Comput. Geom.*, 5(4):399–407, 1990.

[6] J. M. Keil and C. A. Gutwin. Classes of graphs which approximate the complete euclidean graph. *Discrete Comput. Geom.*, 7(1):13–28, 1992.

[7] J. Krozel, J. S. B. Mitchell, V. Polishchuk, and J. Prete. Airspace capacity estimation with convective weather constraints. In *AIAA Guidance, Navigation, and Control Conference*, Aug 2007.

[8] J. Krozel, J. Prete, J. S. B. Mitchell, J. Kim, and J. Zou. Capacity estimation for superdense operations. In *AIAA Guidance, Navigation, and Control Conf.*, Aug 2008.

[9] J. S. B. Mitchell. On maximum flows in polyhedral domains. *J. Comput. Syst. Sci.*, 40:88–123, 1990.

[10] J. S. B. Mitchell and V. Polishchuk. Thick non-crossing paths and minimum-cost flows in polygonal domains. In *Proceedings 23rd ACM Symposium on Computational Geometry*, pages 56–65, 2007.

[11] V. Polishchuk. *Thick Non-Crossing Paths and Minimum-Cost Continuous Flows in Geometric Domains*. PhD thesis, Stony Brook University, Aug 2007. Available at `http://cs.helsinki.fi/~polishch/pages/thesis.pdf`.

[12] G. Strang. Maximal flow through a domain. *Math. Program.*, 26:123–143, 1983.

Appendix

Proposition 3 *The function $f(x) = \frac{\lfloor (1+\varepsilon)x \rfloor}{\lfloor x \rfloor}$, $x > 0$ is bounded above by 2 when $\varepsilon \leq 0.5$.*

Proof. Note that $1 + \varepsilon \leq 3/2$ and that $f(x)$ is a step function whose value changes only when $x = n$ or $x = \frac{n}{1+\varepsilon}$, for $n \in \mathbb{Z}^+$.

When $x = n$, $n \in \mathbb{Z}^+$, $f(x) \leq \frac{(1+\varepsilon)n}{n} = 1 + \varepsilon \leq 3/2$.

When $x = \frac{n}{1+\varepsilon}$, $n \in \mathbb{Z}^+$, $f(x) = \frac{\lfloor n \rfloor}{\lfloor n/(1+\varepsilon) \rfloor} \leq \frac{\lfloor n \rfloor}{\lfloor 2n/3 \rfloor}$. Let $g(n) = \frac{\lfloor n \rfloor}{\lfloor 2n/3 \rfloor}$. If $n = 3k$, $g(n) = \frac{3}{2}$. If $n = 3k-1$, $g(n) = \frac{3k-1}{2k-1}$ and it achieves its maximum of 2 at $k = 1$. If $n = 3k+1$, $g(n) = \frac{3k+1}{2k}$ which also achieves its maximum of 2 at $k = 1$. Thus $f(x) \leq 2$ in this case. $\square$