Shortest Path Problem among Imprecise Obstacles

Zahra Gholami*   Mansoor Davoodi†

Abstract

In this paper, we consider the shortest path problem among a set of imprecise segments as obstacles. In the precise context, each segment is defined by its endpoints. Here, we assume an uncertainty region for the endpoints of the segments and study the problem of arranging the obstacles by placing a point inside each uncertainty region in such a way that maximizes the possible shortest path. We prove the NP-completeness of the maximum shortest path problem for both the Euclidean and Manhattan metrics and propose a $\frac{1}{2}$-approximation algorithm for it when the uncertainty regions are modelled as independent disjoint disks. If the regions are dependent on separation factor k, we obtain the approximation factor of $1 - \frac{2}{k+4}$.

1 Introduction

Regarding the widespread applications of shortest path problems in motion planning, VLSI design and network wire-routing, the necessity of investigation over shortest path problems has currently become evident. It is unsurprising, therefore, that finding a shortest path for robots or points within a workspace with obstacles has been under investigation as an intriguingly applicable problem. Taking some constraints and properties of obstacles or workspaces into consideration, various approaches like visibility graph [8] have been suggested for the shortest path problem. These approaches assume the workspace, data processing, and motions completely in a precise manner, however, this assumption is not realistic. For instance, in the robot motion planning, many uncertainties such as robot's sensing and acting are inevitable. In addition to the mechanical constraints of robots, a raft of data is inaccessible due to the different sources of error such as collecting real data about the world and its dynamical properties. Clearly, these uncertainties make these approaches inefficient under uncertainty and, consequently, imprecision consideration will draw a more complete and accurate picture of finding shortest paths.

Region-based models [3, 12], are popular approaches to model the imprecision. In this models, the precise point may appear anywhere in the region with a uniform probability. The goal of the region-based models for handling imprecision is to find the critical point for each geometric region in order to minimize or maximize specific values. For example, Löffler and van Kreveld discussed the convex hull of imprecise points in various types of regions which maximize or minimize area/perimeter of the convex hull [12]. For each variant, they either provide an NP-hardness proof or a polynomial-time algorithm. As another example, the problem of finding Minimum Spanning Tree (MST) for imprecise points, turn out to be the problems of finding the Minimum and Maximum-weight MST. This problem has been studied by Dorrigiv et al. [6] under disk-shaped uncertainty regions.

Regarding to the applications of the shortest path problem, it has been considerably studied under different parameters close to uncertainty conditions. For a sequence of simple polygons and two points s and e, the Touring Polygons Problem (TPP) is looking for a tour from s to e so that all polygons are visited in the given order. A more general form of TPP is the Shortest Path Problem (hereafter: SPP) for imprecise points. In this problem, a graph of polygons is given instead of an order of polygons. In a directed graph, traversal between vertices is only allowed through the edges. The aim of SPP is to find a placement of the vertices which minimizes the shortest distance between s and e. The maximum variant of SPP has been studied which searched for such a placement that maximizing the shortest path length.

For this problem in the case of significant uncertainty of the arc lengths, Yu et al. [14] presented exact and heuristic solutions. Dror et al. in [7] showed that for convex and disjoint polygons TPP is solvable in polynomial time. NP-hardness of such a problem has been proved for any metric, L_p, $p \geq 1$ in case of non-convex polygons (i.e. they are disjoint [1] or overlapping polygons [7]). Moreover, some approximation algorithms are given for TPP in cases for which the polygons are non-convex [13]. Also, the maximum variant of TPP is explored by Disser et al. in [5] that provided a polynomial time algorithm for computing a maximum placement.

In general, SPP is NP-hard for any metric L_p, $p \geq 1$. Disser et al. in [4] showed that for axis-aligned rectilinear polygons (not necessarily convex) under Manhattan metric, proposing a polynomial time algorithm is feasible. Their study in [5] shows that the problem is hard to

*School of Computer Science and Software Engineering, The University of Western Australia, eli.gholami@uwa.edu.au

†Department of Computer Science and Information Technology, Institute for Advanced Studies in Basic Sciences, Zanjan, mdmonfared@iasbs.ac.ir

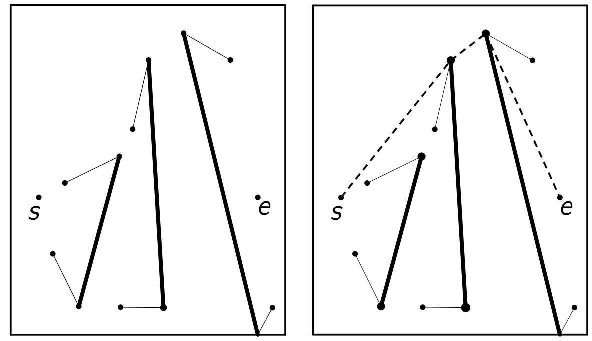
approximate for any approximation factor $(1-\epsilon)$ with $\epsilon < 1/4$, even when the polygons consist of only vertically aligned segments.

In this paper, we define an imprecise segment as an obstacle whose endpoints are some regions instead of points. We study the problem of maximum possible shortest paths' length. Our goal in Maximum Shortest Path Problem (hereafter: Max-SPP) is placing a point inside each region (a placement) in order to arrange the obstacles such that the shortest path from s to e becomes maximum. In other words, Max-SPP is SPP in continuous space instead of the graph.

This paper makes the following contributions:

1. We prove NP-completeness for the decision version of *Max-SPP* when the uncertainty regions are modelled as segments in the region-based models.

2. When the uncertainty regions have been modelled as disjoint disks, we propose a $\frac{1}{2}$-approximation algorithm for Max-SPP. Also in cases where the regions are *k-separable* disks (see Definition 4) we show that the approximation factor of the algorithm is $1 - \frac{2}{k+4}$.



(a) (b)

Figure 1: (a) A workspace containing start and endpoint of shortest path, segments as obstacles and their uncertain endpoints which are modeled as a line segments. (b) Placing a point for each region of uncertainty in order to maximize the distance between s and e.

2 Problem Formulation

Free Space: In this work, we assume points of s and e to be located within the free space. In the precise manner, the free space is introduced as all points in the workspace that do not belong to any obstacles. However, in the imprecise manner, the free space refers to all points that do not belong to any obstacles for all possible placements. So, there are no placements for which obstacles contain points of s and e. In other words, s and e are not allowed to be located in the obstacles for any possible placement.

For a robot, we consider a workspace containing start point s, endpoint e in free space and a set of imprecise segments as obstacles. We define the imprecise points or regions as a set

$\mathcal{R} = \{R_1, R_2, R_3, ..., R_n\}; R_i \subset \mathbb{R}^2, 1 \leqslant i \leqslant n$. Where n is the number of obstacles' endpoints in the workspace. Suppose $\mathcal{I}$ to be a set of points that we achieve by placing a point or *instance* inside each region of $\mathcal{R}$, like the placement

$$\mathcal{I} = \{I_1, I_2, I_3, \ldots, I_n\}; I_i \in R_i, 1 \leqslant i \leqslant n. \qquad (1)$$

If $\mathcal{L}(\mathcal{I})$ refers to the length of the shortest path from s to e for the placement $\mathcal{I}$, then in the **Max-SPP**, the goal is to maximize $\mathcal{L}(\mathcal{I})$ by setting a placement like

$$\mathcal{I}^{max} = \{I_1^{max}, I_2^{max}, I_3^{max}, \ldots, I_n^{max}\}; I_i^{max} \in R_i, 1 \leqslant i \leqslant n \qquad (2)$$

$(\mathcal{I})^{max}$ represents a placement which maximizes the shortest path length between s and e.

The Decision Version of Max-SPP:

Input: $\mathcal{R}$ as a set of imprecise points, points of s and e, and a length value of B.

Output: YES if there exists a placement like $\mathcal{I}$ that $\mathcal{L}(\mathcal{I}) \geqslant B$, NO otherwise.

The **Existence Path Problem** (hereafter: **EPP**) is a problem whose answer is YES if there exists a path from s to e for at least a set of $\mathcal{I} \subset \mathcal{R}$, NO otherwise.

3 Maximum Shortest Path Problem (Max-SPP)

In this section, with the help of reduction from the SAT problem to Max-SPP, we show the hardness results for Max-SPP. We assume a simple case of Max-SPP in which the imprecise regions are modelled by segments. Since in this case the approach for NP-hardness proof of the Largest Convex Hull problem in [12] is not applicable for Max-SPP, we add some crucial obstacles to the workspace.

For a given SAT instance (formula ψ), we construct a Max-SPP instance. For this, we setup $\mathcal{R}(\psi)$ including imprecise obstacles's endpoint. Then, we prove that the decision version of Max-SPP returns YES if and only if the SAT formula ψ is satisfiable.

As illustrated in Fig. 2(a), for converting the SAT formula to the Max-SPP instance, we divide a circle into $M = c + q$ arcs, where c and q are the numbers of clauses and variables in formula ψ, respectively. The c clauses and q variables of ψ are characterized as arcs in the Max-SPP instance. This unique circle contains one arc for each clause and one arc for each variable as well as two points s, e and M separator points that separate arcs from each other. We locate the points s and e by some sufficiently small $\epsilon > 0$ above and below the separator point of z (Fig. 2(a)). In addition, to insert some obstacles we draw segments from circle center at o to all separator points and a segment from point of z to the workspace boundary.

Variable Arcs Configuration: As Fig. 2(b) shows, for each variable in ψ like v, we have an arc that contains: a segment parallel to $\overline{lr}$ (shown as $\overline{tf}$), and two sets of points, P_v and Q_v, with the same number of elements equal to $3c$.

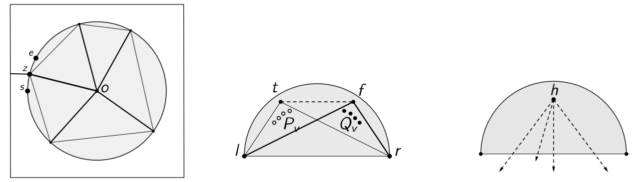
Notably, although the points in P_v corresponding to each variable like v are placed such that they are all on the convex hull of $\{l, r, f\}, P_v$ and Q_v, they are not on the convex hull of $\{l, r, t\}, P_v$ and Q_v. As shown in Fig. 2(a), points in Q_v are symmetrical with P_v.

Clause Arcs Configuration: As Fig. 2(c) shows, for each clause in ψ like c we have an arc containing a point h_c. If the variable v appears in clause c as a positive literal, we connect the point h_c to a member of P_v. If the variable v appears in clause c as a negative literal, we connect the point h_c to a member of Q_v. In this way, the obstacles as precise and imprecise regions would be produced.

- the connection between the workspace boundary and the separator point of z as a precise obstacle.
- the connection between the circle center at o and all separator points as precise obstacles.
- the connection between the circle center at o and the imprecise regions with an endpoint h_c and the other in P_v or Q_v sets as imprecise obstacles.
- the connection between the circle center at o and the imprecise regions with endpoints at t and f as imprecise obstacles.

Finally, in the workspace constructed by formula ψ, maximizing the shortest path between s and e is now equal to the sum of the maximized shortest paths between two separator points. So, in order to maximize the shortest path between two separator points (which locate on a single arc) for every variable arc like v, the selected endpoint should be either t together with all points in Q_v or f together with all points in P_v. Moreover, for the optimal placement of $\mathcal{I}^{max}$ point h_c should be selected in each clause arc such as c.

Theorem 1 *Suppose we are given a workspace containing a set of segment obstacles and a set of imprecise points as obstacles' endpoints. Obstacles are assumed to be arbitrary segments that can have common intersections only at their endpoints. For such a workspace with these imprecise obstacles, Max-SPP is NP-hard under the Euclidean metric and its decision version is NP-complete.*



(a) (b) (c)

Figure 2: (a) A circle is divided into arcs in which all the segments passing through o represent obstacles. The points s and e located in some sufficiently small $\epsilon > 0$ above and below the separator point of z. (b) A variable arc. (c) A clause arc.

4 Approximation Algorithm for Max-SPP

Regarding the NP-hardness of Max-SPP, the approximation algorithms could be used for estimating the solution. For approximating the optimal placement of Max-SPP, we focus on those workspaces which their obstacles are just segments and their imprecise endpoints are disjoint disks. Our approximation algorithm simply selects center of disks as placement $\mathcal{I}$ (i.e. as an approximate placement). Similar to [6], we have proved that the result of this simple algorithm, $\mathcal{L}(\mathcal{I})$, is not smaller than half of that in optimal placement.

Definition 1 *Let* $\boldsymbol{\mathcal{L}(\mathcal{I}^{center})}$ *and* $\boldsymbol{SP(\mathcal{I}^{center})}$ *denote respectively the solution value and the shortest path of the approximation algorithm that selects the disks' centers as the placement* $\mathcal{I}$*.*

Definition 2 *We define* $\boldsymbol{SP(\mathcal{I}^{max})}$ *and* $\boldsymbol{\mathcal{L}(\mathcal{I}^{max})}$ *as the shortest path and its length in the optimal placement for Max-SPP, respectively.*

Definition 3 *We suppose* $\boldsymbol{SP'(\mathcal{I}^{max})}$ *is the path from* s *to* e *with the same topology*[1] *as* $SP(\mathcal{I}^{center})$ *and with the point of* $(\mathcal{I}^{max})$*. Noticeably, this path is not necessarily the shortest path.*

Let $\mathcal{L}(\mathcal{I}^{max})$ and $\mathcal{L}(SP'(\mathcal{I}^{max}))$ stand for the length of paths from s to e for paths $SP(\mathcal{I}^{max})$ and $SP'(\mathcal{I}^{max})$, respectively. Then we have

$$\mathcal{L}(\mathcal{I}^{max}) \leqslant \mathcal{L}(SP'(\mathcal{I}^{max})) \quad (3)$$

Theorem 2 *Consider a workspace such that the imprecise obstacles' endpoints are disjoint disks. Now, in the approximation algorithm assuming the center of all disks as the placement* $\mathcal{I}^{center}$ *for Max-SPP, we have*

$$\frac{1}{2}\mathcal{L}(\mathcal{I}^{max}) \leqslant \mathcal{L}(\mathcal{I}^{center}) \quad (4)$$

[1] The sequence of obstacles and their endpoints throughout a path.

If the disks are sufficiently far from each other, the approximation factor of the algorithm in Theorem 2 will be improved to more accurate values. So, in the following, we prove that the larger the distances between disks, the better the approximation factor we get (i.e. closer to 1).

Definition 4 *As defined by [10], a given set of disks with the largest radius r_{max} are **k-separable** when for the maximum value of k the minimum distance between each pair of disks is at least $k.r_{max}$.*

Theorem 3 *Consider a workspace such that the imprecise obstacles' endpoints are k-separable disks with $k > 0$. Now, in the approximation algorithm assuming the center of all disks as the placement $\mathcal{I}^{center}$ for Max-SPP, we have*

$$(1 - \frac{2}{k+4})\mathcal{L}(SP(\mathcal{I}^{max})) \leqslant \mathcal{L}(SP(\mathcal{I}^{center})) \quad (5)$$

Proof. See the Appendix. □

Now, obviously, farther disks (i.e. larger value of k) leads the algorithm to more accurate approximation factors (i.e. closer to 1).

5 Conclusion

In this paper, we modelled the imprecise points by using some geometric approaches and proved that the Maximum Shortest Path Problem (Max-SPP) is NP-hard and its decision version is NP-complete. For this proof, we considered the obstacles to be segments and their endpoints to be imprecise points modelled as segments. Remarkably, the obstacles can only be intersected at their endpoints. In addition, we presented an approximation algorithm with approximation factors of $1/2$ and $1-\frac{2}{k+4}$ for disk and *k-separable* disk as imprecise points, respectively.

A possible future work includes the investigation of the hardness of Max-SPP for different shapes which imprecise points could be modelled with.

References

[1] Ahadi, A., Mozafari, A., Zarei, A.:Touring disjoint polygons problem is NP-hard. In Combinatorial Optimization and Applications (pp. 351-360). Springer International Publishing (2013)

[2] Choset, H., M., Ed.: Principles of robot motion: Theory, Algorithms, and Implementation MIT press (2005)

[3] Davoodi, M., Mohades, A.: Data imprecision under λ-geometry model: Range searching problem. Scientia Iranica, 20(3), pp. 663-669 (2013).

[4] Disser, Y., Mihalk, M., Montanari, S., Widmayer, P.: Rectilinear Shortest Path and Rectilinear Minimum Spanning Tree with Neighborhoods. In Combinatorial Optimization. Springer International Publishing, 208-220 (2014)

[5] Disser, Y., Mihalk, M., Montanari.: Max Shortest Path for Imprecise Points. In EuroCG, (2015)

[6] Dorrigiv, R., Fraser, R., He, M., Kamali, S., Kawamura, A., López-Ortiz, A., and Seco, D.: On Minimum and Maximum-weight Minimum Spanning Trees with Neighborhoods. In: Approximation and Online Algorithms. Springer Berlin Heidelberg, 93-106 (2013)

[7] Dror, M., Efrat, A., Lubiw, A., Mitchell, J. S.: Touring a sequence of polygons. In Proceedings of the thirty-fifth annual ACM symposium on Theory of computing, 473-482 (2003)

[8] Ghosh, S. K., Mount, D. M.: An output-sensitive algorithm for computing visibility graphs. SIAM Journal on Computing, 20(5), 888-910 (1991)

[9] LaValle, S., M.: Planning Algorithms. Cambridge university press (2006)

[10] Lichtenstein, D.: Planar Formulae and Their Uses. SIAM journal on computing, 11(2), 329-343 (1982)

[11] Löffler, M., van Kreveld, M.: Largest Bounding Box, Smallest Diameter, and Related Problems on Imprecise Points. In: Algorithms and Data Structures, Springer Berlin Heidelberg, pp. 446-457 (2007)

[12] Löffler, M., van Kreveld, M.: Largest and smallest convex hulls for imprecise points. Algorithmica, 56(2), 235-269 (2010)

[13] Pan, X., Li, F., Klette, R.: Approximate shortest path algorithms for sequences of pairwise disjoint simple polygons, 175-178 (2010)

[14] Yu, G., Yang, J.: On the robust shortest path problem. Computers and Operations Research, 25(6), 457-468 (1998)

[15] Surmann, H., Huser, J., Wehking, J.: Path Planning far a Fuzzy Controlled Autonomous Mobile Robot. In: Fifth IEEE International Conference on Fuzzy Systems, Vol. 3, pp. 1660-1665. IEEE (1996)