Autonomous Robots manuscript No.
(will be inserted by the editor)

Motion Planning for a Pair of Tethered Robots

Reza H. Teshnizi · **Dylan A. Shell**

Received: date / Accepted: date

Abstract Considering an environment containing polygonal obstacles, we address the problem of planning motions for a pair of planar robots connected to one another via a cable of limited length. Much like prior problems with a single robot connected via a cable to a fixed base, straight line-of-sight visibility plays an important role. The present paper shows how the reduced visibility graph provides a natural discretization and captures the essential topological considerations very effectively for the two robot case as well. Unlike the single robot case, however, the bounded cable length introduces considerations around coordination (or equivalently, when viewed from the point of view of a centralized planner, relative timing) that complicates the matter. Indeed, the paper has to introduce a rather more involved formalization than prior single-robot work in order to establish the core theoretical result—a theorem permitting the problem to be cast as one of finding paths rather than trajectories. Once affirmed, the planning problem reduces to a straightforward graph search with an elegant representation of the connecting cable, demanding only a few extra ancillary checks that ensure sufficiency of cable to guarantee feasibility of the solution. We describe our implementation of A* search, and report some limited experimental results.

Keywords Motion Planning · Tethered Robots · Multi-Robot Coordination · A* Search

R.H. Teshnizi · D.A. Shell
Distributed AI and Robotics Laboratory,
Department of Computer Science and Engineering,
Texas A&M University, College Station, TX 77843, USA.
E-mail: reza.teshnizi@gmail.com
E-mail: dshell@tamu.edu

Fig. 1: Rope team at a ridge nearing a summit. (source: Wikimedia Commons)

1 Introduction

In recent years, a variety of techniques have been developed to plan motions for a tethered mobile robot (**???**). A tether can be useful as a conduit for power or communication but the main motivating application for robotic tethers is in navigation of rovers in extreme terrain, where the tether can help provide physical security. Examples of robotic rovers equipped in this way include TRESSA (**?**), Axel and DuAxel (**?**), vScout (**?**), and TReX (**?**), among others. Humans deal with extreme terrain too. A common practice among mountaineers, as a measure of protection against falling, is to form a group that can move together while the members are roped to one another. This forms what is referred to as a *rope team* (**?**) (Fig. 1). From the perspective of motion planning, one might interpret a rope team as a practical scenario in which a tethered robot's base is itself subject to motion. A prominent robotic example of comparable operation is DuAxel, where two Axel rovers are connected to a central module. DuAxel is designed to work as a *mother-daughter ship*, having one Axel remain stationary with the central module while the other explores the terrain. Indeed, enabling both rovers to be deployed simultaneously may benefit both agents and improve the versatility of the design.

One key to efficient solution of planning problems in finding a suitable representation, ideally one that expresses constraints and is amenable to adaptation and generalization to various requirements. Much like our earlier work (**?**), we take advantage of the properties of reduced visibility graph (**?**). Namely, we show certain characteristics of straight line motions enable us to find solutions to tethered pair problem with minimal book keeping. Unlike that work, however, we will not be examining the structure of configuration space (c-space). We will show in Section 3 that the c-space of this problem is indeed 4 dimensional and rather complicated. The theoretical foundation provided in this work, shows that we shall not be concerned with the intermediate state of the cable: as long as we can achieve a goal configuration that is permitted by the length of the available cable, one can provide a planner to execute the motions that will transform the initial cable configuration to its final configuration. On this basis, we introduce a tree data structure that represent different cable configurations up to homotopy. Each branch in the tree down to a certain node is a representation of the shortest path required for the tethered pair to arrive at that node's cable configuration. We have implemented A* search to expand the search tree and find an optimal solution while keeping track of the cable's configuration up to homotopy. Lastly, we suggest an optimal controller for the motions produced by the planner that can minimize the execution time for the given path pair.

2 Related Work

We are interested in what is perhaps the most natural motion planning question for a tethered pair of robots, namely finding paths to take a pair of tethered robots from some initial configuration to a goal one, never violating a bound on the tether's length throughout the motion. Although motion planning for a single tethered robot has been extensively studied (**??????**), the literature reports comparatively little work on motion planning problems involving pairs of robots tethered to one another.

A notable exception is that pairs of conjoined robots have been studied for purposes of object manipulation. **?** studied object separation using a pair of robots connected by a cable. Though superficially similar to our problem, as it involves the motion of a mutually connected pair of robots, the separation problem imposes quite a different set of constraints to a shortest path planning problem. For object separation the solution is only required to satisfy a homotopy requirement, allowing the robots to choose any arbitrary goal in the workspace that can satisfy such constraint. Consequently, in that work, the two robots move to the boundaries of the workspace. (This assumption also helps distinguish between separating versus non-separating configurations elegantly and concisely.) As **?** are addressing a problem where the goal is specified topologically and they are not concerned with a cable of finite length, several of the complications we tackle do not arise in their setting.

More recently, **?** demonstrated a physical multi-robot system in which robots can dynamically make or break tether connections; a planner exploits this capability to find ways in which a robot team can manipulate objects efficiently, either with single robots operating concurrently, or as coupled pairs, as called for by the particular problem instance. A part of that problem is combinatorial and the work uses a sampling-based method, the present work being distinguished from that work on both fronts.

Our own prior work on finding short paths for a pair of tethered robots (**?**), attempted to use the solution to a single robot problem as an algorithmic building block. That approach was devised primarily as a means to build intuition for the topology of the 4D configuration space; the algorithm is inadequate, being neither a solution to the complete problem nor one that always yields optimal paths.

RETURN TO ME Reviewing the prior work, we have identified the following two core traditions in approaching the tethered robot motion planning problems.

– **Sampling-based vs. visibility-based planning:** to represent the c-space of a given problem one has to create a graph that is complete, *i.e.*, it must represent all possible configurations. Historically, efficient sampling-based planning algorithms such as PRM (**?**) and RRT (**?**) have proven to be beneficial in high dimensional c-spaces. However, in visibility-based techniques (**?**) reduce the dimensions of the problem by using minimal representations which only keep *critical* information about the c-space. That is, rather than creating many nodes that comparatively do not add significant value during the search, they

represent the c-space with as few nodes as possible that compared to one another have large entropy in the information gained. It is important to note that the two techniques are not meant as substitutes for one another: it would be wasteful to use sampling based techniques in low dimensional spaces, whereas producing the visibility graph for high dimensional spaces is computationally impractical.
- **word construction vs. event tracking:** tethered robots require a technique that can distinguish between the different cable configurations to be topologically sound. One way to maintain the cable's configuration is by constructing a *word* using sensor beams that uniquely represents its homotopy class (?). That is, after execution of each portion of a given path for a robot, one has to update the word according to the visited node in the search graph. Another way to maintain the cable's configuration is to keep track of critical events that signify changes in its configuration (?). One can then uniquely identify the cable's configuration up to homotopy by presenting the shortest representative of its homotopy class.

While each tradition provides its own value in an appropriate setting when solving motion planning related problems for tethered robots, we strongly believe that decomposition-based planning and visibility go hand-in-hand. It is infeasible/impractical to use visibility to retain cable configurations in a sampling-based method. Additionally, using h-signatures with decomposition-based planning would not fully take advantage of the decomposed c-space and would lead to redundancy in the information.

3 Why is the tethered robot pair problem difficult?

Our earlier work (?) defined and examined a useful way of decomposing the c-space of a single tethered robot. Namely, into a tree structure that relates to an atlas representation of the manifold[1], the associated charts of which are the largest satisfying the following two main properties:

1. planning a motion between the tether's base and any configuration inside the same chart is a constant time problem, and
2. if a path between two configurations is contained within a given chart, it is guaranteed that the homotopy class of the cable's trajectory remains unchanged after the execution of the path.

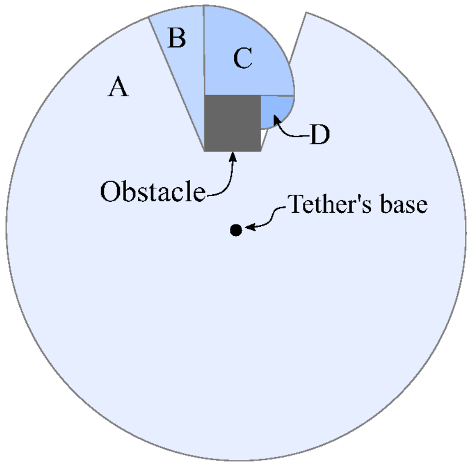


Fig. 2: Some of the charts of the atlas representing the c-space of a tethered robot. A, B, C, and D are four charts in the atlas. Red dashed line illustrates boundary of charts.

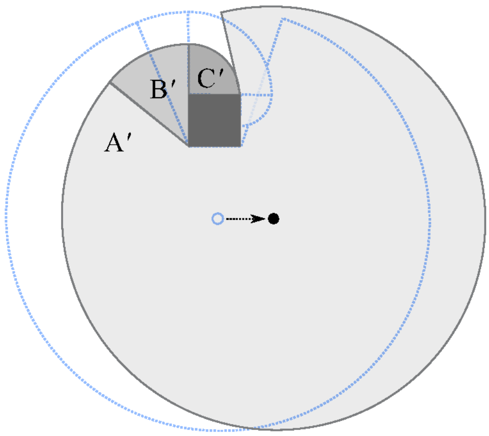
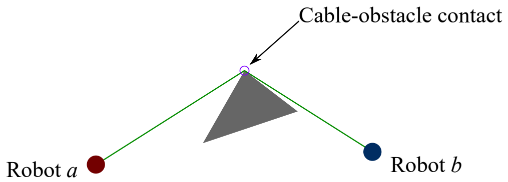


Fig. 3: Moving the base of the tether leads to a changes in the boundaries of the charts. A′, B′, C′ are the new charts. The charts prior to moving the base are illustrated in light blue for comparison.

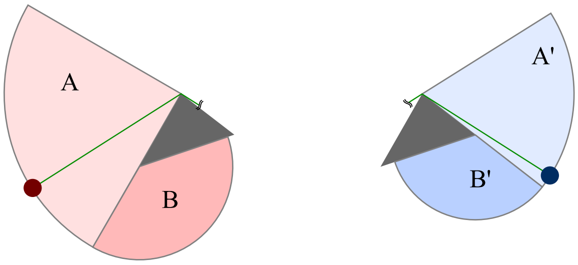
Given the inputs to the single robot problem, one visualizes its c-space most easily by sketching the charts and showing where they touch (see Fig. 2). For the single tethered robot, this is a 2 dimensional object, and consequently easy to see. This structure allows for the convenient and simultaneous representation of two different pieces of data: the motion of the robot through space, and the homotopy class of the cable.

The single tethered robot problem is a special case of tethered pairs where one robot remains stationary: requiring only 2 degrees of freedom. For a general teth-

[1] Technically one must consider a manifold with a boundary, and the notion of 'chart' here may be closed set; for simplicity throughout we will ignore these nuances.



(a) The configuration of the robots and their cable.



(b) The visibility charts for robot a.

(c) The visibility charts for robot b.

Fig. 4: A snapshot of the conjoined atlases for a tethered robot pair after the first contact is made between the cable and an obstacle.

ered pair, a comparable figure would require four dimensions, which impedes easy visualization.

In Fig. 3 we have moved the base of the tether slightly to the right to show the changes in the charts. The boundaries of these charts are 2D surfaces in a 4D space. However, once the first contact is made between the cable and an obstacle, it is again possible to visualize snapshots of the boundaries of the charts (see Fig. 4).

We believe exploiting the two above mentioned properties is the key to understanding the structure of the complex c-space of this problem.

(note to self) Although compatibility does not show the charts exactly, it is revealing some structured-ness in this problem. Although we will not be breaking the c-space into charts, we divide the problem based on properties of reduced visibility graph such that low level details are abstracted away from the planning. We believe the decomposition technique introduced in this body of work is strongly related to the four-dimensional charts in the c-space.

4 Problem statement

Let $O = \{o_1, o_2, \ldots, o_n\}$ be a (possibly empty) set of pairwise disjoint polygonal obstacles with vertices verts(O) in $\mathbb{R}^2$, with boundary curves $\delta o_1, \delta o_2, \ldots, \delta o_n$, respectively. Let the free space $\mathscr{W} = (\mathbb{R}^2 \setminus \bigcup_{i=1}^n o_i) \cup (\bigcup_{i=1}^n \delta o_i)$. Further, let robots a and b be two unoriented points in $\mathscr{W}$ that are connected to one another via a cable of finite length. Let $\mathbb{I} = [0, 1]$ be the unit interval.

Definition 1 (TRPMPP) The tethered robot pair motion planning problem (TRPMPP) is a tuple, $(\mathscr{W}, O, \mathbf{r_a}, \mathbf{r_b}, \mathbf{d_a}, \mathbf{d_b}, \ell, c_0)$, wherein:

- $\mathscr{W}$ is the free space,
- O is the set of obstacles,
- $\mathbf{r_a} \in \mathscr{W}$ is the initial position of a,
- $\mathbf{r_b} \in \mathscr{W}$ is the initial position of b,
- $\mathbf{d_a} \in \mathscr{W}$ is the goal or destination of a,
- $\mathbf{d_b} \in \mathscr{W}$ is the goal or destination of b,
- $\ell \in \mathbb{R}^+$ is the length of the tether,
- $c_0 : \mathbb{I} \to \mathscr{W}$ is the initial arrangement of the cable in $\mathscr{W}$, where $c_0(0) = \mathbf{r_a}$, and $c_0(1) = \mathbf{r_b}$, and[2]

$$\mathfrak{L}[c_0(s)] := \int_{\mathbb{I}} c_0(s)\, ds \leq \ell.$$

A solution to TRPMPP $(\mathscr{W}, O, \mathbf{r_a}, \mathbf{r_b}, \mathbf{d_a}, \mathbf{d_b}, \ell, c_0)$ is a pair of paths (τ_a, τ_b) in which:

- $\tau_a : \mathbb{I} \to \mathscr{W}$ where $\tau_a(0) = \mathbf{r_a}$ and $\tau_a(1) = \mathbf{d_a}$, and
- $\tau_b : \mathbb{I} \to \mathscr{W}$ where $\tau_b(0) = \mathbf{r_b}$ and $\tau_b(1) = \mathbf{d_b}$, and
- for τ_a and τ_b, there exists a $\mathscr{W}$-feasible motion for an ℓ-length cable, as formalized next.

Definition 2 ($\mathscr{W}$-feasible motion for an ℓ-length cable) In the closed subset of the plane $\mathscr{W} \subseteq \mathbb{R}^2$, for two paths $\tau_a, \tau_b : \mathbb{I} \to \mathscr{W}$ and length $\ell \in \mathbb{R}^+$, a function $c : \mathbb{I} \times \mathbb{I} \to \mathscr{W}$ is called a *$\mathscr{W}$-feasible motion for an ℓ-length cable* if and only if the following three conditions hold:

C1. (The cable connects the robots)

$$\forall s \in \mathbb{I} : c(s, 0) = \tau_a(s) \text{ and } c(s, 1) = \tau_b(s),$$

C2. (Cable has bounded length)

$$\forall s \in \mathbb{I} : \mathfrak{L}[c_s(x)] \leq \ell, \text{ where } c_s(x) := c(s, x),$$

C3. (Continuity) c is continuous with respect to the induced topologies.

Definition 3 (Distance optimality) A solution $(\tau_a^\star, \tau_b^\star)$ for TRPMPP $(\mathscr{W}, O, \mathbf{r_a}, \mathbf{r_b}, \mathbf{d_a}, \mathbf{d_b}, \ell, c_0)$ is called *distance optimal* if all other solutions (p, q) have

$$\max\left(\mathfrak{L}[\tau_a^\star], \mathfrak{L}[\tau_b^\star]\right) \leq \max\left(\mathfrak{L}[p], \mathfrak{L}[q]\right).$$

We have chosen this particular optimality metric in order to ensure minimum energy consumption between the two robots. We present a method that finds a distance optimal solution to TRPMPP. In Section **??** we take advantage of this metric to define an optimal controller given a distance optimal solution.

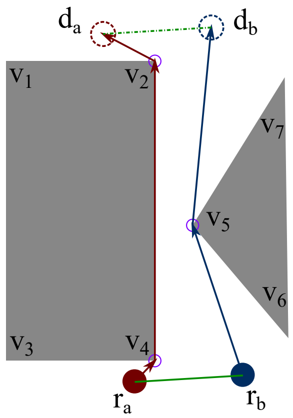


Fig. 5: The green line segment is the initial cable configuration and the green dashed line segment is the final cable configuration. The arrows show the distance optimal path for each robot.

5 The solution concept

In this section we prove that it suffices to have robots a and b equipped with a motion primitive that moves the robot on a straight line from one vertex to another in the reduced visibility graph (RVG). The following theorem and corollary show that, for a single robot, the shortest path between two points in $\mathscr{W}$ is a concatenation of line segments that are edges of the RVG.

Theorem 1 *There exists a semi-free path between any two given points* $\mathbf{p}$ *and* $\mathbf{q}$ *if and only if there exists a simple polygonal line T lying in $\mathscr{W}$ whose endpoints are* $\mathbf{p}$ *and* $\mathbf{q}$*, and such that T's vertices are in* $\operatorname{verts}(O)$*. (??)*

More useful for us the statement below which, though only stated informally by **?**, follows directly the previous result.

Corollary 1 *The shortest path for a robot from one point to another in a subset of $\mathbb{R}^2$ can be found by searching the shortest path roadmap or reduced visibility graph.*

The list of RVG vertices $\pi_a = (v_a^0, v_a^1, \ldots, v_a^{n-1}, v_a^n)$ connecting initial and destination positions, *i.e.*, with $\mathbf{r_a} = v_a^0$ and $\mathbf{d_a} = v_a^n$, is easily turned into a curve, $\tau_a(s)$, by joining line segments connecting v_a^{i-1} to v_a^i sequentially, head to tail, and parameterizing appropriately via $\mathbb{I}$. Hence, a pair of sequences of RVG vertices for the two robots, (π_a, π_b), suffices to give a pair of paths (τ_a, τ_b).

[2] We find it convenient to use $\mathfrak{L}[f]$ to denote arc length of functions $f : \mathbb{I} \to \mathscr{W}$ throughout this work.

Although the preceding classical results hold for individual robots, when two robots are constrained such that the action of one limits the actions of the other (as in the case of a tether of finite length), then it is less clear that the discrete structure of the RVG encodes an optimal solution. One might conceive, in the two robot setting, one robot deviating from visibility edges in order to enable the other to move. One of the main results of paper, and the basis for the algorithm we present, is Theorem 2 establishing that no such deviations are necessary and, indeed, the RVG will still suffice to find optimal solutions. The theorem's proof comes after several definitions a useful lemma.

In what follows, we can think of an always taut tether. This conception is without loss of generality because: (1) tightening a tether is a continuous operation, so it preserves homotopy; (2) if the cable length constraint is satisfied for a taut tether, it must be for others as well. The reader should bear in mind that the taut tether is merely a special representative of the homotopy class of tethers. It is helpful to have an operator to give this representative:

Definition 4 (Tightening Operator) Given a path $\alpha : \mathbb{I} \to \mathscr{W}$, we define the operator $\hat{\cdot} : (\mathbb{I} \to \mathscr{W}) \to (\mathbb{I} \to \mathscr{W})$ such that $\hat{\alpha}$ is the (unique) shortest path in the homotopy class of α.

Practically, the classical algorithm of **?** is used to obtain the shortest path homotopic to a given path. Tautening will often (but not always) be used to compute the form the taut cable will take, so the following is needed.

Definition 5 (Cable concatenation) Given two paths $\tau_1 : \mathbb{I} \to \mathscr{W}$ and $\tau_2 : \mathbb{I} \to \mathscr{W}$, and a cable configuration $c : \mathbb{I} \to \mathscr{W}$, let the function that gives a concatenation of τ_1, τ_2, and c be

$$\operatorname{cat}(s; t, \tau_1, \tau_2, c) := \begin{cases} \tau_1(t - 3st) & 0 \le s \le \frac{1}{3} \\ c(3(s - \frac{1}{3})) & \frac{1}{3} < s < \frac{2}{3} \\ \tau_2(3(s - \frac{2}{3})t) & \frac{2}{3} \le s \end{cases} .$$

Definition 1, describing the planning problem, requires paths for which a feasible trajectory exists. The connection between paths and trajectories is, of course, a timing. Thus we need the following concept:

Definition 6 (Re-parameterization) A *re-parameterization* is a monotonically increasing, continuous function $r : \mathbb{I} \to \mathbb{I}$ with $r(0) = 0$ and $r(1) = 1$. A pair (r_1, r_2) is a *re-parameterization pair* if both functions, r_1 and r_2, are re-parameterizations.

With the preceding scaffolding, we can next give a definition that is valuable in helping to identify the existence of a feasible trajectory given paths.

Definition 7 Given two paths $\tau_1 : \mathbb{I} \to \mathscr{W}$ and $\tau_2 : \mathbb{I} \to \mathscr{W}$, and a cable configuration $c : \mathbb{I} \to \mathscr{W}$, let

$$\mathbb{C}^\star(\tau_1, \tau_2, c) := \min_{\substack{(r_1,r_2) \text{ over all} \\ \text{re-parameterization pairs}}} \left(\max_{t\in\mathbb{I}} \mathfrak{L}\left[\widehat{\text{cat}}(\cdot; t, \tau_1 \circ r_1, \tau_2 \circ r_2, c)\right]\right).$$

Informally, $\mathbb{C}^\star$ gives the shortest cable that permits one to execute τ_1 and τ_2. The intuition which connects us to main result is that any solution (τ_a, τ_b) to a given TRPMPP must have a $\mathbb{C}^\star(\tau_a, \tau_b, c_0) \leq \ell$. We would like to establish that any optimal solution to a given TRPMPP can be related to some solution on the RVG (i.e, in $\Pi_a \times \Pi_b$) that, while being distance optimal, also abides by the conditions imposed by cable. Corollary 2 states this formally. The proof of the theorem makes use of the following lemmas and theorem.

Lemma 1 *Path pair (τ_a, τ_b) is a solution to a given* TRPMPP*, iff $\mathbb{C}^\star(\tau_a, \tau_b, c_0) \leq \ell$.*

Proof. For a contradiction assume $\mathbb{C}^\star(\tau_a, \tau_b, c_0) > \ell$ and reach a contradiction with C2 in Definition 2. □

Lemma 2 *Let $\widehat{\tau_a}$ and $\widehat{\tau_b}$, being the shortest paths in their respective homotopy classes, be the prescribed paths for robots a and b. Then the length of the consumed cable is a convex function, if $\widehat{\tau_a}$ and $\widehat{\tau_b}$ use the same curve parameter.*

Proof. Because both $\widehat{\tau_a}$ and $\widehat{\tau_b}$ are the shortest paths in their respective homotopy classes, we can assert that they comprise sequences of pairwise connected straight line motions which lie on the RVG edges (following Corollary 1). Moreover, each straight line motion is of one of the following type:

F: the motion is on the same edge as the cable and is [F]ollowing the cable, or
L: the motion is on the same edge as the cable and is [L]eading the cable, or
O: is any [O]ther straight line motion.

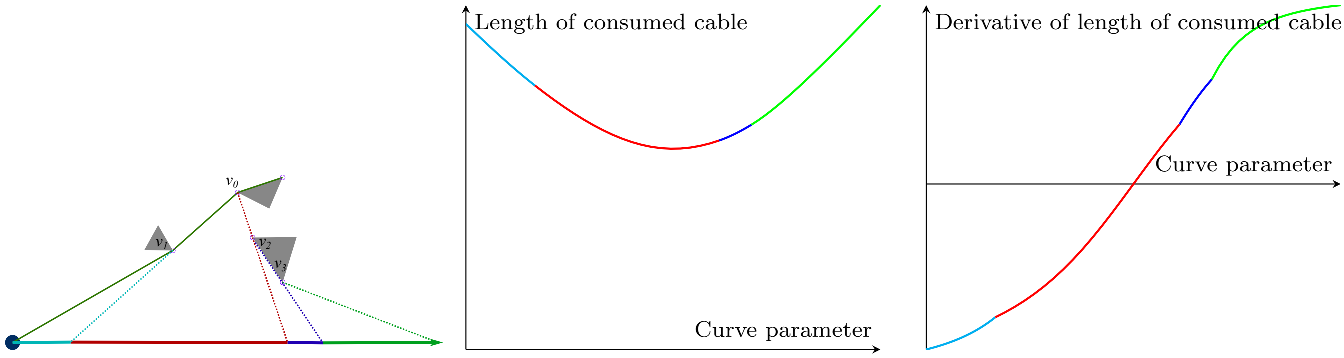
We will argue that the length of cable consumed as a robot moves along such trajectories is a convex function by showing that the gradient of the function is monotonically increasing. A technical difficulty with cable consumption is that it is a continuous but only piecewise differentiable function. In circumstances where the derivative is undefined, we take the value of the derivative from the right. These circumstances arise when contacts between the cable and obstacles are made or broken. The cable consumption is a function of two curve parameters: so by derivative we are referring to the partials with respect to each parameter—one for each robot.

When the robot is moving on an **F** (or **L**) segment the derivative is -1 (or 1, respectively). For individual **O** segments the function might have multiple pieces. For each piece of each such segment, the pieces are regions where the points that the cable contacts are unchanged. Then, with a single robot moving, only the cable between the robot and the closest contact point contributes to the change in cable consumption. Writing the consumption as a function of the curve parameter and simply taking its derivative results in a monotonically increasing function whose value is in the $(-1, 1)$ interval. When unwinding around an object the derivative is negative but increasing; when breaking contact, the radius to the closest object increases, but the derivative continues to increase. When making or maintaining contact, the derivative is positive, but still increasing. It is, thus, always increasing. (Fig. 6 illustrates this fact using an example.)

Next, we argue that $\widehat{\tau_a}$ and $\widehat{\tau_b}$ have a specific structure: each trajectory is of the form $\mathbf{F}^*\mathbf{O}^*\mathbf{L}^*$ (where we have used regular expression notation with Kleene stars). Showing the following suffices:

(a) no **O** segment is followed by an **F** segment, and
(b) no **L** segment is followed by an **O** segment, and
(c) no **L** segment is followed by an **F** segment.

All three cases are established via proofs that are similar: assume the contrary, then reach a contradiction owing to a taut cable and shortest motions failing to agree. Taken together, this proves that $\widehat{\tau_a}$ and $\widehat{\tau_b}$ are of the form $\mathbf{F}^*\mathbf{O}^*\mathbf{L}^*$. Hence, global monotonicity of the partial derivative of the cable consumption function holds if we can show that monotonicity is preserved between two consecutive **O** segments. To show no violation occurs at the transition between segments, monotonicity in a small open interval around the transition is sufficient. If the two **O**-segments, g_1 and g_2, are collinear, then they could be treated as a single segment and the prior argument for a single **O**-segment holds. Hence, there must be a 'turn' from segment g_1 to g_2. Both segments are on the RVG so that turn occurs at a vertex v_o of some obstacle. Presume that we extend and continue along g_1 an extra $\epsilon > 0$; then the derivative of the cable consumption continues to increase (as the single segment argument holds). Since g_2 is not along this little extension, it falls to one side. If that side is away from the cable-obstacle contact, then the additional motion away consumes extra cable, so the derivative only increases faster. Otherwise, when turning towards the contact, monotonicity may indeed



(a) The scenario for which the cable consumption function is derived. Originally the cable is in contact with $v0$ and v_1. As the robot travels towards its destination (to the right), it will release contact with v_1. It will then make contact with v_2 and v_3, respectively. The dotted lines represent places in which cable events occur.

(b) Length of consumed cabled as a function of distance traveled.

(c) Derivative of length of consumed cabled per change in distance traveled.

Fig. 6: A scenario where a robot executes a motion on a straight line. As the robot makes its way towards the destination, the cable will make or release contact with verts(O). These events change the the derivative of the consumed cable. The motion is color coded throughout all the above figures to differentiate each time a cable event occurs. Fig. 6a shows the scenario for which the function is derived. Fig. 6b and 6c show the function and its derivative respectively.

fail. However, such a turn leads to a contradiction; two cases are possible: the obstacle to which v_o belongs is on the inside of the turn, or it is on the outside. If it is on the inside, then the cable itself wraps around v_o, in which case g_2 is not an **O** segment (but an **L** one). If the obstacle is on the outside, then simply shaving off a small corner at the turn is feasible. But that is shorter, contradicting the supposition that g_1 and g_2 result from shortest motions in their homotopy class.

Hence monotonicity of the partial derivative of the cable consumption holds across the entire $\mathbb{I}$. Now consider the concurrent motion of both robots: we feed them the same curve parameter and the derivative of the cable consumption simply becomes the total derivative. Being the sum of two partials, each of which is monotone, gives a monotone function. Thus, the total cable consumption is a convex upward function of the curve parameter. □

The preceding is enough to do the heavy lifting needed for proof of our theorem.

Theorem 2 *If* (τ_a, τ_b) *is a solution for a given* TRPMPP, *then* $\exists(\tau_a^{VG}, \tau_b^{VG}) \in \mathit{\Pi}_a \times \mathit{\Pi}_b : \mathbb{C}^\star(\tau_a^{VG}, \tau_b^{VG}, c_0) \leq \ell$.

Proof. Let $\widehat{\tau_a}$ and $\widehat{\tau_b}$ be the shortest paths homotopic to τ_a and τ_b, respectively. Hence, $(\widehat{\tau_a}, \widehat{\tau_b}) \in \mathit{\Pi}_a \times \mathit{\Pi}_b$ from Corollary 1. Then to prove this theorem it suffices to show that the following inequality holds:

$$\mathbb{C}^\star(\widehat{\tau_a}, \widehat{\tau_b}, c_0) \leq \ell.$$

Since $\widehat{\cdot}$ is homotopy preserving, the following conditions hold:

$$\mathfrak{L}\left[\widehat{\mathrm{cat}}(\cdot; 0, \widehat{\tau_a}, \widehat{\tau_b}, c_0)\right] = \mathfrak{L}\left[\widehat{\mathrm{cat}}(\cdot; 0, \tau_a, \tau_b, c_0)\right] = \mathfrak{L}\left[\widehat{c_0}\right],$$
$$\mathfrak{L}\left[\widehat{\mathrm{cat}}(\cdot; 1, \widehat{\tau_a}, \widehat{\tau_b}, c_0)\right] = \mathfrak{L}\left[\widehat{\mathrm{cat}}(\cdot; 1, \tau_a, \tau_b, c_0)\right].$$

Moreover, because (τ_a, τ_b) is a solution we have $\mathfrak{L}\left[\widehat{\mathrm{cat}}(\cdot; 0, \widehat{\tau_a}, \widehat{\tau_b}, c_0)\right] \leq \ell$, and $\mathfrak{L}\left[\widehat{\mathrm{cat}}(\cdot; 1, \widehat{\tau_a}, \widehat{\tau_b}, c_0)\right] \leq \ell$.

Following Lemma 2, feeding the same curve parameter to both $\widehat{\tau_a}$ and $\widehat{\tau_b}$, gives a convex upward cable consumption. Therefore,

$$\max_{t \in \mathbb{I}} \mathfrak{L}\left[\widehat{\mathrm{cat}}(\cdot; t, \widehat{\tau_a}, \widehat{\tau_b}, c)\right] = \max_{t \in \{0,1\}} \mathfrak{L}\left[\widehat{\mathrm{cat}}(\cdot; t, \widehat{\tau_a}, \widehat{\tau_b}, c)\right] \leq \ell.$$

Because the above holds for the trivial identity reparameterization, the minimum over the set of all reparameterization pairs must be less than or equal to the above. This fact combined with Lemma 1 completes the proof. □

Corollary 2 (Distance Optimality) *Let* $(\tau_a^\star, \tau_b^\star)$ *be a distance optimal solution to a given* TRPMPP. *Then the RVG also contains a distance optimal solution,* $(\tau_a^{\star VG}, \tau_b^{\star VG})$.

Proof. A pair $(\tau_a^{\star VG}, \tau_b^{\star VG}) \in \Pi_a \times \Pi_b$ is constructed from $(\tau_a^\star, \tau_b^\star)$. Simply take $(\tau_a^{\star VG}, \tau_b^{\star VG}) = (\widehat{\tau_a^\star}, \widehat{\tau_b^\star})$ and observe the three following facts:

- $\widehat{\tau_a^\star}$ and $\widehat{\tau_b^\star}$ is on the RVG,
- the traveled, thus, distance along the two path for each robot is no worse following Corollary 1,
- $\mathbb{C}^\star(\widehat{\tau_a^\star}, \widehat{\tau_b^\star}, c_0) \leq \ell$ following Theorem 2.

□

6 The Efficient Planning Algorithm

In this section we present an A* (**?**) implementation to construct and explore the search tree for the solution to this problem. Let $(\mathscr{W}, O, \mathbf{r_a}, \mathbf{r_b}, \mathbf{d_a}, \mathbf{d_b}, \ell, c_0)$ be a TRPMPP. The search tree's nodes represent cable configurations. An edge between two nodes represents a motion for each robot along an RVG edge that abides by the cable requirements. To do so, we define a data-structure with the following fields:

- *taut cable*: a list of RVG vertices each of which is visible from the previous ones describing a valid cable configuration,
- *cost*: a pair of costs indicating the distance traveled by each robot to arrive at this cable configuration from their initial poses,
- a reference to a parent node.

Fig. **??** and Fig. **??** visualize the structure of the search tree for an example scenario. Using A* search, Algorithm **??** explores the search tree—which gives a structured representation to the set $\Pi_a \times \Pi_b$—for the optimal solution to a given TRPMPP. A* uses an estimated cost function which is a sum of a cost function and a heuristic function. For the heuristic, we use the Euclidean distance between the vertex and the given robot's destination, which can be computed in constant time. The cost is calculated cumulatively with a constant time operation when creating/updating a node. Since the heuristic function is admissible, we can safely terminate the search in a branch whose estimated cost is higher than the best solution found (**?**).

Given any node in the tree, we can obtain a pair of paths — one for each robot — that continuously transforms the original cable configuration to the configuration stored in the node. To do so, we can traverse the tree from the root down to the node. Let π_a and π_b be to empty sequences. While traversing the tree down to the node, for each node we take the first and last element of the taut tether and append it to the end of π_a and π_b, respectively. Notice that the sequences π_a and π_b will have equal cardinality.

Algorithm 1: The A* Search Algorithm

1: **Search**$(\mathscr{W}, O, \mathbf{r_a}, \mathbf{r_b}, \mathbf{d_a}, \mathbf{d_b}, \ell, c_0)$
2: Build RVG, g, with vertices $\{\text{verts}(O), \mathbf{r_a}, \mathbf{r_b}, \mathbf{d_a}, \mathbf{d_b}\}$
3: root = Node($\widehat{c_0}$, $\varnothing$)
4: priorityQ = { root }
5: **while** priorityQ is not empty **do**
6: n = priorityQ.dequeue()
7: **if** n.cable.first = $\mathbf{d_a}$ n.cable.last = $\mathbf{d_b}$ **then**
8: **if** $\mathfrak{L}$ [node.cable] $\leq \ell$ **then**
9: **return** constructed path by following parent references from n up to the root
10: **else**
11: **continue**
12: **end if**
13: **end if**
14: $V_a = g$.visibleVerts(n.cable.first)
15: $V_b = g$.visibleVerts(n.cable.last)
16: **for all** $(v_a, v_b) \in V_a \times V_b$ **do**
17: $c_1 = [v_a]+$ n.cable $+[v_b]$
18: child = Node($\widehat{c_1}$, n)
19: priorityQ.enqueue(child)
20: **end for**
21: **end while**

Theorem 3 (Soundness) *A pair of paths, (τ_a, τ_b), generated by traversing the above mentioned search tree from the root to a leaf is a solution to the corresponding* TRPMPP, *if the leaf contains a taut tether configuration whose end points are $\mathbf{d_a}$ and $\mathbf{d_b}$.*

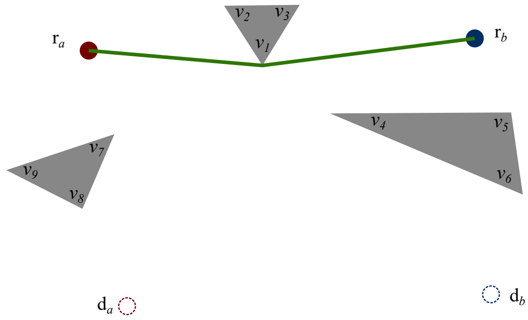
Proof. The root contains the taut tether $\hat{c_0}$, which by definition has end points $\mathbf{r_a}$ and $\mathbf{r_b}$. Because the algorithm checks whether a leaf contains a taut tether configuration whose end points are $\mathbf{d_a}$ and $\mathbf{d_b}$ on line **??**, the requirements $\tau_a(0) = \mathbf{r_a}$, $\tau_a(1) = \mathbf{d_a}$, $\tau_a(0) = \mathbf{r_a}$, and $\tau_a(1) = \mathbf{d_a}$ is satisfied. Using the result of Theorem 2, the check on line **??** ensures (τ_a, τ_b) satisfies the final requirement for a solution. □

Theorem 4 *Algorithm* **??** *is complete.*

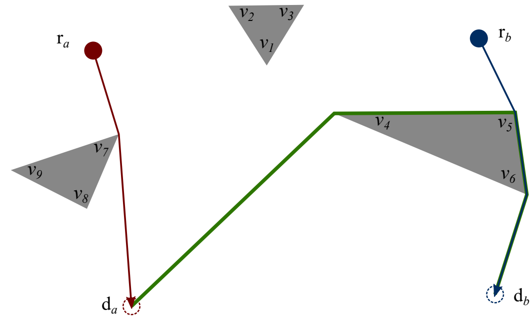
Proof. Completeness is shown by considering completeness of A* search and Corollary 2. □

7 The Optimal Controller

In Section 4 we only provided a definition for a *distance optimal* solution. As the last piece in the body of work, we would like to provide an *optimal controller* for distance optimal solutions. Informally, a controller for a tethered pair of robots is a function that maps a TRPMPP solution (*i.e.*, a pair of paths) to a pair of trajectories (as defined by **?**), one for each robot. In what follows we assume robots a and b are homogeneous and have a maximum speed of mv. To make the discussion easier, we assume that the robots can achieve maximum

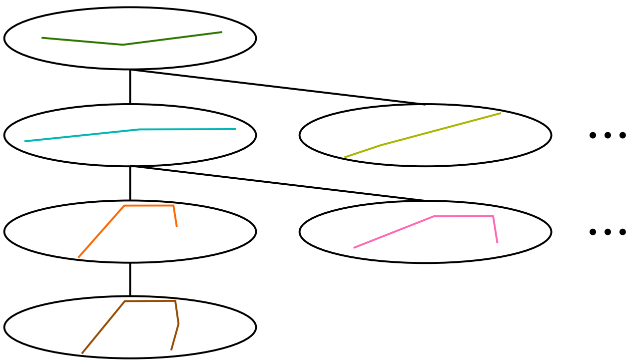


(a) The original configuration of the cable.

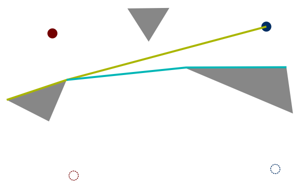


(b) The optimal solution to the given TRPMPP which is found by searching the search tree.

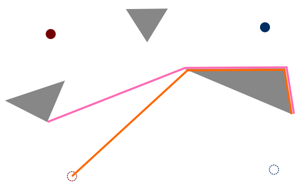
Fig. 7: An example scenario for which we illustrate part of the search tree in Fig. **??**



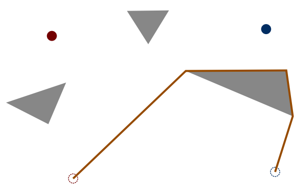
(a) A visual representation of the search tree depicting the relationship between the cable configurations shown on the right.



(b) Two of the configurations that are compatible with the configuration stored in the root node.



(c) Both cable configurations shown here are compatible with the cyan cable from the previous snapshot. However, the orange cable configuration should is preferred as it reduces the distance traveled by robot a.



(d) The final configuration to arrive at the destination node.

Fig. 8: A small portion of the search tree for the TRPMPP in Fig. **??**. The optimal solution to the given TRPMPP in this figure can be obtained by traversing the node containing the brown cable configuration up to the root.

speed instantaneously and that turning does not take time[3].

Definition 8 (Controller) Let (τ_a, τ_b) be a plan (*i.e.* solution) for a given TRPMPP. A controller is a function M such that $M(\tau_a, \tau_b) = (t_a, t_b)$ where

$$t_a : [0, t_1] \to \mathscr{W}; t_1 \in \mathbb{R}^+ \text{ and } 0 < t_1 \text{ and } t_1 \geq \frac{\mathfrak{L}[\tau_a]}{mv}, \text{ and}$$

$$t_b : [0, t_2] \to \mathscr{W}; t_2 \in \mathbb{R}^+ \text{ and } 0 < t_2 \text{ and } t_2 \geq \frac{\mathfrak{L}[\tau_b]}{mv}, \text{ and}$$

the trajectories respect the cable constraints[4].

[3] In practice, one can easily adjust the definitions/theorem in this section to include a more accurate model of the robots by considering other physical characteristics such as acceleration rate, maximum angular speed, etc.

[4] The reader can easily adjust Definition 2 to use the intervals mentioned here.

Next, we provide a metric to compare controllers.

Definition 9 (Execution Time of Controller) Let M be a controller. We define the execution time of the plan (τ_a, τ_b) for M to be

$$\mathscr{T}\left[M(\tau_a, \tau_b)\right] = \max\left\{t_1, t_2\right\}.$$

We can now define optimality over the set of all controllers. Informally, an optimal controller is one that reduces the overall execution time for a plan.

Definition 10 (Optimal Controller) given an optimal plan (τ_a, τ_b), a controller $M^\star$ is optimal if for all other controllers M

$$\mathscr{T}\left[M^\star(\tau_a, \tau_b)\right] \leq \mathscr{T}\left[M(\tau_a, \tau_b)\right].$$

Theorem 5 *Given a distance optimal solution (τ_a, τ_b) returned by Algorithm **??**, a controller $M^\star$ is optimal if $M^\star(\tau_a, \tau_b) = (t_a, t_b)$ such that $t_a : [0, t_1] \to \mathscr{W}$ and $t_b : [0, t_1] \to \mathscr{W}$ provided that $t_1 = \max\left\{\frac{\mathfrak{L}[\tau_a]}{mv}, \frac{\mathfrak{L}[\tau_b]}{mv}\right\}$.*

Proof. Algorithm **??** produces solutions such that $(\tau_a, \tau_b) = (\widehat{\tau_a}, \widehat{\tau_b})$. Then Lemma 2 provides that the trajectories (t_a, t_b) abide by the cable constraints. Because our optimality metric searches for minimum of the maximum distance traveled by either robots, and because the $t_1 = \max\left\{\frac{\mathfrak{L}[\tau_a]}{mv}, \frac{\mathfrak{L}[\tau_b]}{mv}\right\}$, $M^\star$ satisfies Definition **??**. □

8 Discussion of the Method

To demonstrate this algorithm we implemented it in Python (v3). Our implementation makes heavy use of **?** for computational geometry algorithms, namely convex hull and triangulation algorithms. Fig. **??** shows some screenshots of the user interface.

8.1 Limited Cable Length

As mentioned in Section 3, the existence a cable of limited length changes the nature of planning motions for the tethered pair. Although having a bounded cable length reduces the size of the search space, it induces its own complexity on the problem. Consider the Fig. **??** where for both cable of length 200 and 300 units the optimal solution is the same. The algorithm visits the same number of nodes for $\ell = 300$ and $\ell = 200$. The reason for this is that the algorithm will enumerate all solutions from most cable consumed to the least cable consumed. Interestingly, as the search reaches solutions that require a shorter cable, the average branching factor in the search tree becomes smaller. This apparent by looking at Fig. **??**. This implies that the complexity of the search space is affected by a combination of the cable length and the topology of the environment.

8.2 Effect of Informed Search

As we discussed in the previous section, the topology and cable length can increase both the number of visited and expanded nodes in the search space. One way to manage this is to use an informed search algorithm. As mentioned in Section **??**, we have implemented A* search for the optimal solution with a simple Euclidean distance heuristic. Even though this heuristic is a very simple metric, it improves the efficiency of the search substantially. For comparison, we implemented Dijkstra's algorithm which performs a greedy search. Fig. **??** shows the significant impact of an informed versus greedy search. Indeed, a more intelligent heuristic, rather than a simple Euclidean distance, can reduce the search time even more.

9 Conclusion

This work presents an algorithm for planning distance optimal motions f

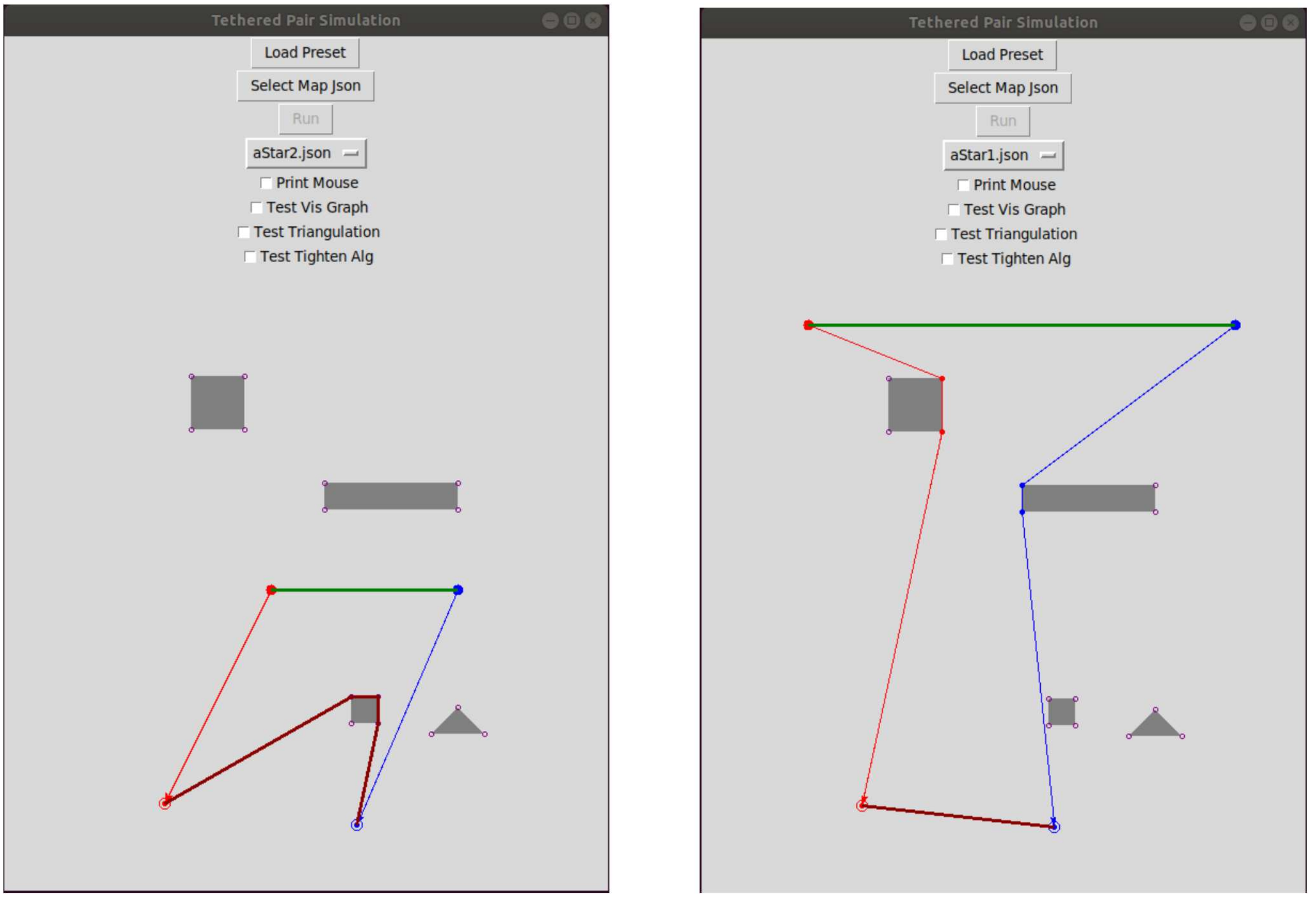
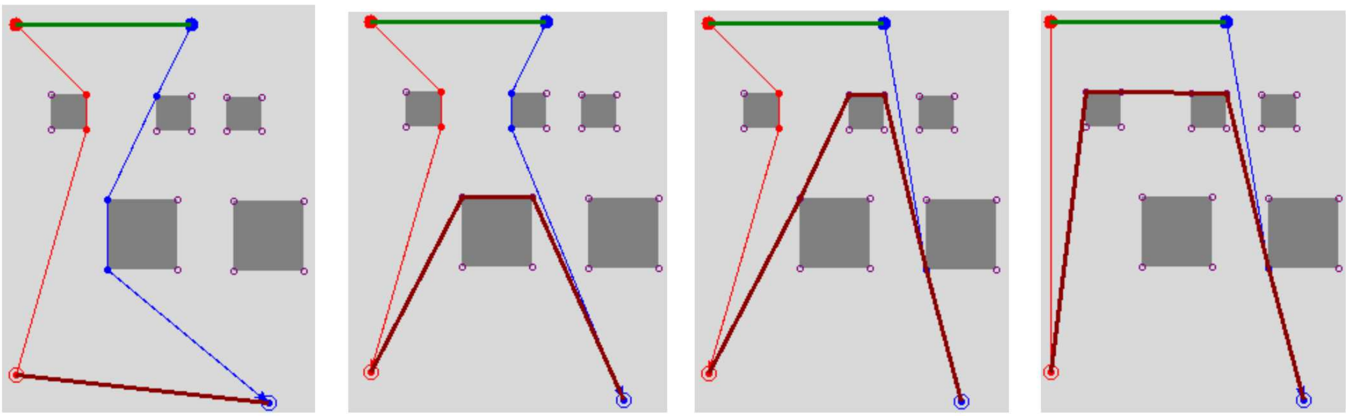
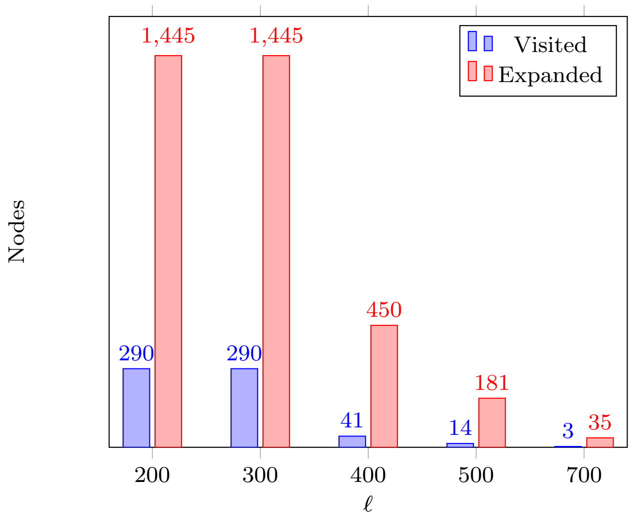


Fig. 9: Screenshots of user interface of our implementation. The red and blue circle filled circles represent the robots. Obstacles are in grey. The initial and final configuration of the cable are shown in green and dark red. The prescribed path for each robot is shown with lines of the same color. The timing and RVG vertices where for each robot is shown with small dots along the path.

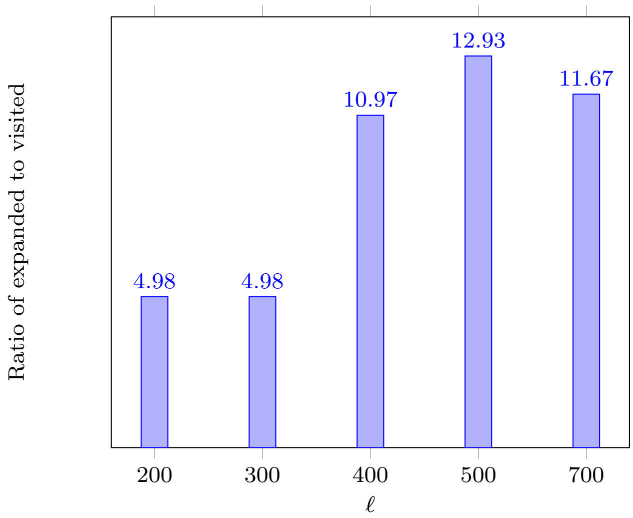


(a) $\ell = 200$ and $\ell = 300$ (b) $\ell = 400$ (c) $\ell = 500$ (d) $\ell = 700$

Fig. 10: Solving the same scenario with different cable length.



(a) A comparison of the number of visited vs expanded nodes.



(b) The ratio of expanded to the visited nodes gives us some insight about the branching factor in the search tree.

Fig. 11: Some data about the structure of the search tree for the scenarios in Fig. **??**

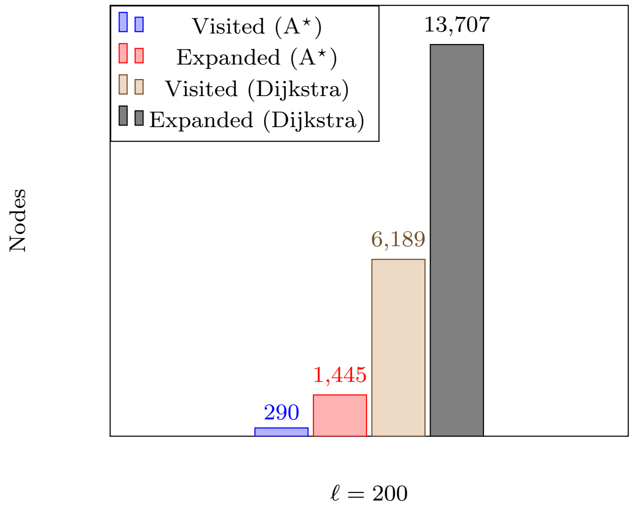


Fig. 12: The impact of informed search on the number of nodes visited and expanded.