

Preliminary Solutions for Autonomous Mobile Robot Docking Subject to Uniform Braking Safety Constraints

Elon Rimon and Gil Manor

Dept. of Mechanical Engineering, Technion, Israel

Abstract *This paper considers the synthesis of time optimal paths for a mobile robot docking at targets located on the boundary of a polygonal obstacle in \mathbb{R}^2 . The robot must avoid the obstacle as well as satisfy velocity dependent braking safety constraints during the docking process. The classical Brachistochrone problem studies the time optimal path of a particle moving in an obstacle free environment subject to a constant force field. By encoding the braking safety constraint as a force field surrounding the obstacle, the paper generalizes the Brachistochrone problem into synthesis of time optimal paths for a mobile robot attempting to dock against an obstacle boundary. To solve the time optimal docking problem, the safe travel time functional, a path dependent function, is formulated. Convexity properties of this functional allow computation of the time optimal docking path as a convex optimization problem in $O(n^3 \log(1/\epsilon))$ time, where n is the number of obstacle vertices and ϵ is the desired solution accuracy.*

1 Introduction

As autonomous mobile robots are deployed in ever more demanding tasks, they strive to complete their tasks while minimizing overall travel time. This requirement is achieved by maximizing the robot’s speed while maintaining velocity dependent safety constraints throughout the navigation process. Examples are self-driving cars [12] and traffic assistance systems [1] that will eventually allow vehicles to be piloted autonomously at speeds up to 50 km/h [15]. High speed mobile robots can also improve cargo delivery operations in large warehouses [18], and may become important first-responders in search and rescue missions. In all of these applications collision safety is modeled by velocity dependent constraints, which require a consideration of the robot’s full position-velocity state space in order to synthesize safe time optimal paths for high speed mobile robots.

This paper considers the synthesis of safe time optimal paths for a mobile robot docking at targets located on the boundary of a polygonal obstacle. To ensure collision safety, the robot must approach the target along the obstacle’s perpendicular direction without colliding with the obstacle. The robot must additionally limit its speed to a level which maintains braking safety throughout the docking process. Braking safety is encoded as a state (position and velocity) dependent constraint, which allows formulation of the *safe* time optimal docking problem using calculus of variations. The safe time optimal path minimizes a travel time functional, $T(\alpha)$, defined over all collision free paths, $\alpha : [0, 1] \rightarrow \mathbb{R}^2$, connecting the start and target. The paper analyzes the properties of $T(\alpha)$ such as existence and uniqueness of a safe time optimal path in each path homotopy class of the environment. Convexity properties of $T(\alpha)$ are then used to compute the safe time optimal path within each candidate homotopy class as a convex optimization problem. Importantly, the entire process takes place in the robot’s full state space, thus treating the geometric path planning and the speed profiling in a single framework.

The classical time optimal motion planners operate locally in discrete time steps, whereby at each time step the mobile robot selects a local time optimal maneuver that avoids nearby obstacles. A notable example is the *velocity obstacles* method [4, 16], which has been extended to motion in the presence of other moving agents such as cars and pedestrians [9, 17]. Another example is the *dynamic window* approach, which plans the robot’s path within a finite time horizon whose size is determined by the robot’s current state [5]. Some dynamic window implementations explicitly account for safe braking constraints [2, 5]. All of these local planners are useful in dynamically evolving environments. This paper considers a static environment, where motion along the *global* time optimal path provides the best overall system performance.

An important global motion planning approach searches the robot’s discretized state space for safe time optimal paths. Karvaki [3] and LaValle and Kuffner [10] introduced the RRT method, or rapidly-exploring random trees, computed in the robot’s discretized state space. At each time step until the target is reached, a randomly chosen point is connected to the current tree with a feasible maneuver that obeys predefined

velocity dependent constraints and minimizes a cost function such as path length or travel time. Karman and Frazzoli [8] introduced the RRT* method that additionally updates the connections between the nodes in the current search tree, in a way that ensures convergence to the global time optimal path as the number of iterations reaches infinity. This paper offers a more analytic approach for computing time optimal paths subject to velocity dependent safety constraints.

The paper is structured as follows. Section 2 formulates the time optimal docking problem under uniform braking safety constraints. Section 3 describes analytic solutions for the time optimal docking path reaching an obstacle edge and an obstacle vertex. Section 4 describes basic properties of the safe travel time functional associated with the docking problem. Section 5 describes a convex optimization scheme for computing the exact time optimal docking path in each path homotopy class of the environment. Section 6 describes computation examples of time optimal docking paths. The concluding section discusses extension to safe time optimal docking in more demanding scenarios.

2 Problem Description and Preliminaries

This paper considers a mobile robot, modeled as point or a disc, which has to dock with zero speed at a target located along the boundary of a convex polygonal obstacle in \mathbb{R}^2 . The robot is assumed to have ideal on-board position and local obstacle detection sensors. The robot moves freely in \mathbb{R}^2 and its *state*, denoted $(p, \nu) \in \mathbb{R}^2 \times \mathbb{R}^+$, consists of the robot’s *position*, $p = (x, y)$, and *speed* $\nu = \|\dot{p}\|$. To ensure collision safety, the robot must approach the target along the obstacle’s perpendicular direction. While the robot navigates with high speed toward a docking target, emergency events may occur at any time along the obstacle’s boundary. To ensure motion safety, the robot must brake and reach a *full stop* without hitting the obstacle whenever such an event is detected. Braking safety is ensured when the robot maintains the following distance from the obstacle’s boundary.

Definition 1. *The robot’s safe braking distance, d , is the length of the shortest kinematically feasible braking path leading from the robot’s current state (p, ν) to a full stop at $\nu = 0$.*

The shortest braking path may be any kinematically feasible path along which the robot applies maximal deceleration (a system dependent parameter), eventually reaching a full stop without hitting the obstacle. Note that mobile robots are typically deployed in environments containing multiple static as well as dynamic obstacles. The current paper assumes a *single static obstacle*, with the intention of extending the results to multiple obstacles in future work.

To derive an expression for the safe braking distance as a function of the robot’s speed, first consider the robot’s maximal deceleration, denoted a_{max} . Braking on the verge of sliding determines the robot’s maximal deceleration. The net reaction force acting at the robot’s wheels during maximal deceleration is given by μmg , where μ is the coefficient of friction at the ground contacts, m is the robot’s mass, and g is the gravitational acceleration. The robot’s dynamics during maximal deceleration is given by the equation $ma_{max} = \mu mg$, which determines the robot’s maximal deceleration, $a_{max} = \mu g$.

The safe braking distance is computed using energy balance. When the robot moves along a kinematically feasible braking path, the reaction forces at the ground contacts are aligned with the friction cone edge opposing the direction of motion (note that the wheels are rolling without sliding during this braking phase). A complete stop of the robot requires that all its initial kinetic energy be absorbed by the braking force. This energy balance leads to the expression (see e.g. [6]),

$$d(\nu) = \frac{1}{2a_{max}}\nu^2 \quad (1)$$

where a_{max} is the robot’s maximal deceleration, and ν is the robot’s initial speed. The robot is required to maintain a uniform braking safety distance while approaching the obstacle’s boundary, as stated in the following definition.

Definition 2. *When a mobile robot travels with speed ν , **uniform braking safety** is ensured when the robot is kept at least $d(\nu)$ from the obstacle’s boundary at every state (p, ν) during the docking process.*

Uniform braking safety is clearly a *conservative* safety criterion. To satisfy this requirement, the robot must maintain a circular safety zone of radius $d(\nu)$ as to allow safe deceleration to a full stop without hitting the obstacle. Since $d(\nu)$ is monotonically increasing in ν , the circular safety zone increases in size when the robot navigates with higher speeds, thus limiting the robot’s maximal speed at every point in the environment.

Other velocity dependent safety concerns: This paper does not explicitly enforce other velocity dependent safety concerns, most notably sideways sliding and vehicle tipovers during high speed turns. The safe

time optimal path generated under the uniform braking safety constraint has a sufficiently flat curvature which prevents sideways sliding at the allowed speeds along this path [11][App. D]. Tipover can be prevented by careful design of the mobile robot’s mass distribution.

3 The Safe Time Optimal Path Near a Single Obstacle Feature

This section describes analytic solutions for the safe time optimal path near a single obstacle edge or vertex.

3.1 The Travel Time Functional

The robot’s path is modeled as a piecewise smooth curve, $\alpha(s) : [0, 1] \rightarrow \mathbb{R}^2$, with endpoints $p_s = \alpha(0)$ and $p_T = \alpha(1)$. When the geometric parameter s is parameterized by time, $s(t)$, the robot’s *speed* along α is given by

$$\nu(s(t)) = \left\| \frac{d}{dt} \alpha(s(t)) \right\| = \left\| \alpha'(s) \right\| \frac{ds(t)}{dt}$$

where $\alpha'(s) = \frac{d}{ds} \alpha(s)$. Solving for dt gives:

$$dt = \frac{\|\alpha'(s)\|}{\nu(s)} ds \quad (2)$$

Integrating both sides of (2) gives the *travel time functional*:

$$T(\alpha) = \int_0^1 \frac{\|\alpha'(s)\|}{\nu(s)} ds \quad (3)$$

Note that $T(\alpha)$ is a path dependent function. The collection of piecewise smooth paths in \mathbb{R}^2 forms a *metric space* under the d_1 metric: $d_1(\alpha_1, \alpha_2) = \max_{s \in [0, 1]} \{\|\alpha_1(s) - \alpha_2(s)\|\} + \max_{s \in [0, 1]} \{\|\alpha'_1(s) - \alpha'_2(s)\|\}$. Moreover, $T(\alpha)$ is a continuous function of α with respect to this metric [7]. The integrand of $T(\alpha)$ in (3), F , is given by

$$F(\alpha(s), \alpha'(s), s) = \frac{\|\alpha'(s)\|}{\nu(s)}.$$

When F is differentiable with respect to its arguments, any *extremal path* of $T(\alpha)$ over all piecewise smooth paths connecting p_s and p_T must satisfy the Euler-Lagrange equation [7] along the path’s smooth segments:

$$\frac{\partial F}{\partial \alpha} - \frac{\partial}{\partial s} \left(\frac{\partial F}{\partial \alpha'} \right) = 0 \quad (4)$$

where α and α' are to be treated as formal variables. The safe time optimal paths near individual obstacle features are next computed.

3.2 The Safe Time Optimal Path Near an Obstacle Edge

The case of a point robot traveling near an obstacle edge, or a wall segment, is a direct extension of the classical *Brachistochrone* problem.¹ Consider a *wall segment* aligned with the x axis as shown in Figure 1(a). Denote by $p_s = (x_s, y_s)$ and $p_T = (x_T, y_T)$ the start and target, which are assumed to lie in the upper half of the infinite strip spanned by the wall segment. The robot’s minimal distance from the wall is given by its y coordinate. Along a safe time optimal path, the robot must keep a distance of at least $d(\nu)$ from the wall segment: $y \geq d(\nu)$. Substituting $d(\nu) = \nu^2 / 2a_{max}$ while requiring maximal robot speed for travel time minimization gives:

$$\nu(y) = \sqrt{2a_{max} \cdot y} \quad y \geq 0 \quad (5)$$

Let the wall segment’s x coordinate serve as the path parameter, s , so that the robot path is parameterized by $\alpha(x) = (x, y(x))$ for $x \in [x_s, x_T]$. The term $\|\alpha'(s)\| ds$ in (3) becomes $\|\alpha'(x)\| dx = \sqrt{1 + (y'(x))^2} dx$, and the *safe* travel time functional (3) is given by $T(\alpha) = \int_{x_s}^{x_T} F(y(x), y'(x)) dx$, where

$$F(y, y') = \sqrt{\frac{1 + y'^2}{2a_{max} \cdot y}} \quad y \geq 0 \quad (6)$$

with end conditions $y(x_s) = y_s$ and $y(x_T) = y_T$. Any extremal path of $T(\alpha)$ must satisfy the Euler-Lagrange equation. Substituting for $F(y, y')$ in (4), then integrating with respect to x , gives the implicit scalar differential equation:

$$1 = c^2 (1 + (y'(x))^2) y(x) \quad (7)$$

where c is yet to be determined. The extremal path solving (7) is a *cycloid*, $\alpha(\varphi) = (x(\varphi), y(\varphi))$, where φ represents the angle that a point on a rolling circle spans with respect to the vertical direction (Figure 1(a)).

¹The Brachistochrone problem computes the shape of a wire allowing minimum travel time for a particle sliding between two endpoints under gravity.

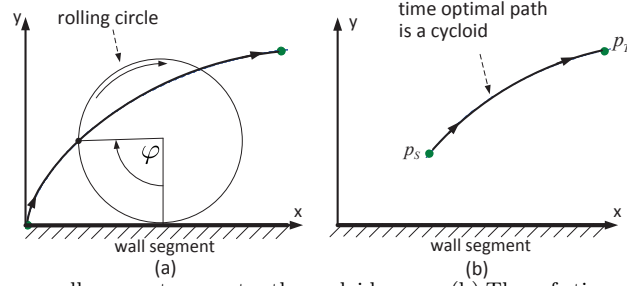


Figure 1: (a) A circle rolling on a wall segment generates the cycloid curve. (b) The safe time optimal path near a wall segment.

Lemma 3.1 ([7]). *The safe extremal path of a point robot moving in a planar environment near a wall segment aligned with the x axis is the cycloid curve:*

$$\alpha(\varphi) = \begin{pmatrix} x_0 \\ 0 \end{pmatrix} + \frac{1}{2c^2} \begin{pmatrix} \varphi - \sin \varphi \\ 1 - \cos \varphi \end{pmatrix} \quad \varphi \in [\varphi_s, \varphi_T] \quad (8)$$

where $\varphi_s, \varphi_T, x_0, c$ are determined by the endpoints p_s and p_T . Moreover, $\alpha(\varphi)$ is a **local minimum** of $T(\alpha)$.

The time optimal path of a point robot traveling near a wall segment from p_s to p_T is shown Figure 1(b). The path is a cycloid that bends away from the wall segment, and reaches the x axis along the *normal* direction to this axis (Figure 1(a)).

3.3 The Safe Time Optimal Path Near an Obstacle Vertex

Next consider a point robot traveling near an obstacle vertex, represented by a point obstacle located at the origin of the (x, y) plane. Using polar coordinates centered at the origin, (r, θ) , the robot’s position is given by $(x, y) = (r \cos \theta, r \sin \theta)$. The robot’s distance from the point obstacle is given by its r coordinate. Braking safety requires that the robot keep a distance of at least $d(\nu)$ from the point obstacle: $r \geq d(\nu)$. Substituting $d(\nu) = \nu^2 / 2a_{max}$ while maximizing the robot’s speed for travel time minimization gives:

$$\nu(r) = \sqrt{2a_{max} \cdot r} \quad r \geq 0$$

Let the robot path be parametrized by the angle θ , so that $\alpha(\theta) = (r(\theta) \cos \theta, r(\theta) \sin \theta)$ for $\theta \in [\theta_s, \theta_T]$. Then $\|\alpha'(\theta)\| d\theta = \sqrt{r^2(\theta) + (r'(\theta))^2} d\theta$, and the travel time functional is given by $T(\alpha) = \int_{\theta_s}^{\theta_T} F(r(\theta), r'(\theta)) d\theta$, where

$$F(r, r') = \sqrt{\frac{r^2 + r'^2}{2a_{max} \cdot r}} \quad r \geq 0 \quad (9)$$

with end conditions $r(\theta_s) = r_s$ and $r(\theta_T) = r_T$. Substituting for $F(r, r')$ in the Euler-Lagrange equation, then integrating with respect to θ , gives the implicit differential equation:

$$c^2 (r^2(\theta) + (r'(\theta))^2) = r^3(\theta) \quad (10)$$

where c is yet to be determined. The solution of (10) is a parabolic curve specified in the following lemma.

Lemma 3.2 ([11]). *The safe extremal path of a point robot moving in a planar environment near a point obstacle is the parabolic curve $\alpha(\theta) = (r(\theta) \cos \theta, r(\theta) \sin \theta)$, where $r(\theta)$ is*

$$r(\theta) = 2c^2 \frac{1 + \sin(\theta - \theta_0)}{\cos^2(\theta - \theta_0)} \quad \theta \in [\theta_s, \theta_T] \quad (11)$$

such that θ_s, θ_T, c , and θ_0 are determined by the endpoints p_s and p_T . Moreover, $\alpha(\theta)$ is a **local minimum** of $T(\alpha)$.

The safe time optimal path can be graphically constructed as follows. Let $p_s = (r_s, \theta_s)$ and $p_T = (r_T, \theta_T)$. Construct the line l tangent to the circles centered at p_s and p_T with radii r_s and r_T , as shown in Figure 2(a). The safe time optimal path is the *parabolic arc* equidistant from the point obstacle and l . This path is uniquely determined by p_s and p_T , except when the path’s endpoints lie on opposite sides of the point obstacle, as illustrated in Figure 2(b). The safe time optimal path of a disc robot traveling near an obstacle vertex is considered in [11][Section 3].

4 Basic Properties of the Safe Time Optimal Path

This section formulates the variational problem associated with safe time optimal docking then discusses basic properties of the safe travel time functional, $T(\alpha)$, in this environment.

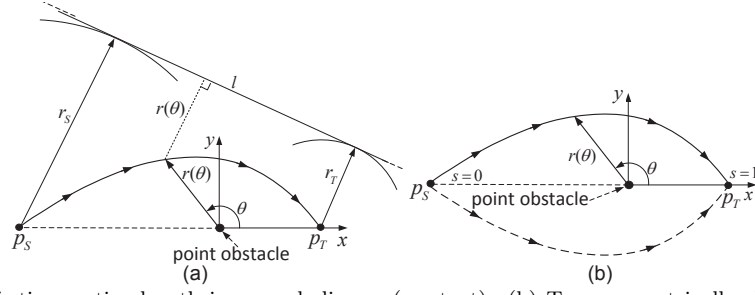


Figure 2: (a) The safe time optimal path is a parabolic arc (see text). (b) Two symmetrically shaped time optimal paths from p_s to p_T .

4.1 The Safe Time Optimal Variational Problem

When a mobile robot is attempting to dock against an obstacle boundary, the uniform braking safety constraint limits the robot’s maximal speed according to its distance from the obstacle. Let \mathcal{O} denote the polygonal obstacle and let \mathcal{F} denote the obstacle-free portion of the environment together with the obstacle’s boundary. The robot’s path can be any piecewise smooth curve, $\alpha(s): [0, 1] \rightarrow \mathcal{F}$, such that $\alpha(0) = p_s$ and $\alpha(1) = p_T$. Braking safety requires that the robot keep a distance of at least $d(\nu)$ from the obstacle:

$$\text{dst}(\alpha(s), \mathcal{O}) \geq d(\nu(s)) \quad s \in [0, 1] \quad (12)$$

where $\text{dst}(\alpha(s), \mathcal{O})$ is the minimal distance between the robot positioned at $\alpha(s)$ and the obstacle \mathcal{O} . Travel time minimization requires maximal speed. Substituting the robot’s maximal allowed speed, $d(\nu) = \nu^2/2a_{max}$, into (12) gives the maximal *safe* speed: $\nu(s) = \sqrt{2a_{max} \cdot \text{dst}(\alpha(s), \mathcal{O})}$ for $s \in [0, 1]$. The safe travel time functional to be minimized is thus:

$$T(\alpha) = \int_0^1 F(\alpha(s), \alpha'(s)) ds,$$

where $F(\alpha, \alpha')$ is given by

$$F(\alpha, \alpha') = \frac{\|\alpha'\|}{\sqrt{2a_{max} \cdot \text{dst}(\alpha, \mathcal{O})}} \quad (13)$$

The minimization of $T(\alpha)$ over all piecewise smooth paths connecting the states $(p_s, 0)$ with $(p_T, 0)$ in \mathcal{F} defines the *safe time optimal docking problem*.

4.2 Basic Properties of the Safe Travel Time Functional

This section describes basic properties of the safe travel time functional, $T(\alpha)$, starting with the notion of homotopic paths.

Definition 3. Two continuous paths with endpoints p_s and p_T , $\alpha(s): [0, 1] \rightarrow \mathcal{F}$ and $\beta(s): [0, 1] \rightarrow \mathcal{F}$, belong to the same **homotopy class** if there exists a continuous mapping, $f(s, t): [0, 1] \times [0, 1] \rightarrow \mathcal{F}$, such that $f(s, 0) = \alpha(s)$, $f(s, 1) = \beta(s)$ for $s \in [0, 1]$, with $f(0, t) = p_s$, $f(1, t) = p_T$ for $t \in [0, 1]$.

In order to compute time optimal paths in \mathcal{F} , it suffices to consider the path homotopy classes circumnavigating the obstacle from its left and right sides. For instance, the two paths depicted in Figure 2(b) belong to the two distinct homotopy classes, since one path cannot be continuously deformed into the other without crossing the obstacle. When two paths belong to the same path homotopy class, they can be thought of as “points” connected by a continuous curve (the path homotopy), in the metric space of piecewise smooth paths in \mathbb{R}^2 . Under this interpretation, every path homotopy class forms a connected set in the metric space. The analysis of $T(\alpha)$ will use the following notion of *proximity cells* induced by an obstacle’s geometric features.

Definition 4. The geometric features of the obstacle (polygon edges and vertices) define **proximity cells**, each consisting of the points in \mathcal{F} closest to the given obstacle feature.

The following lemma establishes the *existence* of a safe time optimal path in every path homotopy class connecting p_s and p_T in \mathcal{F} .

Lemma 4.1 (Optimal Path Existence). The safe travel time functional, $T(\alpha)$, attains a **global minimum** in each path homotopy class connecting p_s and p_T in \mathcal{F} . Moreover, every local minimum path of $T(\alpha)$ lies in the interior of \mathcal{F} , except at its endpoints that can lie on the obstacle’s boundary.

Proof sketch: First consider the existence of a global minimum path. The functional $T(\alpha)$ is a continuous function of α . This property ensures that $T(\alpha)$ attains a global minimum in each path homotopy class of \mathcal{F} [11][Section 4]. Let us next show that every local minimum path of $T(\alpha)$ lies in the interior of \mathcal{F} . Braking safety requires that the robot travel with zero speed along any obstacles edge, thus giving $T(\alpha) = \infty$ along

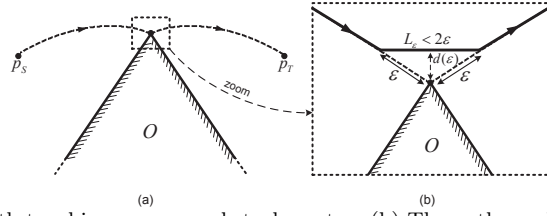


Figure 3: (a) A time optimal path touching a convex obstacle vertex. (b) The path can be moved away from the vertex, with a new segment of length $L_\epsilon < 2\epsilon$.

any such edge. It follows that a local minimum path of $T(\alpha)$ can only touch *isolated* obstacle boundary points. If such an isolated point lies in the interior of a polygon edge or at a concave polygon vertex, the path can be locally shortened and moved away from the obstacle, resulting in a reduced total travel time along the modified path. A local minimum path of $T(\alpha)$ can therefore touch only isolated *convex* obstacle vertices. Suppose the path touches a convex obstacle vertex as depicted in Figure 3(a). The path locally consists of two cycloids that reach the vertex along the respective edge normals. Such a path can be moved away from the vertex as depicted in Figure 3(b), resulting in a *shorter* path that maintains larger clearance from the obstacle. The modified path has a reduced total travel time, implying that any local minimum path of $T(\alpha)$ must lie in the interior of \mathcal{F} , except possibly at its endpoints. \square

Calculus of variations is next used to establish the $C^{(1)}$ smoothness of the safe time optimal paths in \mathcal{F} .

Proposition 4.2 (Optimal Path Smoothness). *Every local minimum path of the safe travel time functional, $T(\alpha)$, is a $C^{(1)}$ curve having a continuous tangent.*

Proof: Let α_{opt} be a local minimum of $T(\alpha)$, connecting p_s and p_r in \mathcal{F} . The path α_{opt} lies in the interior of \mathcal{F} according to Lemma 4.1. Since α_{opt} satisfies the Euler-Lagrange equation in each proximity cell of \mathcal{F} , it forms a $C^{(1)}$ curve in each cell. Let two segments of α_{opt} meet at a *corner point*, $p_0 = \alpha_{opt}(s_0)$, located on the common boundary line of two proximity cells. The term $\partial F(\alpha, \alpha') / \partial \alpha'$ appearing in the Euler-Lagrange equation is a continuous function of the path parameter, s , along any extremal path of $T(\alpha)$ [7]. Using the expression for $F(\alpha, \alpha')$ specified in (13), $\partial F(\alpha, \alpha') / \partial \alpha' = \alpha'(s) / L(\alpha(s)) \cdot \|\alpha'(s)\|^{3/2}$, where $L(\alpha(s)) = \sqrt{2a_{max} \cdot \text{dst}(\alpha(s), O)}$. Since L is a continuous function of s , continuity of $\partial F(\alpha, \alpha') / \partial \alpha'$ with respect to s implies that $\alpha'_{opt}(s_0^-) = \alpha'_{opt}(s_0^+)$. Hence α_{opt} is a $C^{(1)}$ curve. \square

The next theorem establishes the *minimality* of the extremal paths of $T(\alpha)$.

Theorem 1 (Optimal Path Minimality). *Every extremal path of the safe travel time functional, $T(\alpha)$, which lies in the interior of \mathcal{F} is a **local minimum** of $T(\alpha)$.*

Proof sketch: Let α^* be an extremal path of $T(\alpha)$ which lies in the interior of \mathcal{F} . If α^* lies in a single proximity cell, it forms a local minimum of $T(\alpha)$ according to Lemmas 3.1–3.2. Let us therefore assume that α^* passes through $m + 1 \geq 2$ proximity cells, while crossing the proximity cell boundary lines l_1, \dots, l_m . We will show that α^* is a local minimum of $T(\alpha)$ with respect to the *piecewise time optimal paths*, consisting of time optimal segments within each proximity cell (Figure ??).

The piecewise time optimal paths, $\alpha(s) : [0, 1] \rightarrow \mathcal{F}$ such that $\alpha(0) = p_s$ and $\alpha(1) = p_r$, will be parametrized by a fixed division of the unit interval into sub-intervals $[0, s_1], [s_1, s_2], \dots, [s_m, 1]$, such that s_1, \dots, s_m correspond to the crossing points of α with the lines l_1, \dots, l_m . A *proximity cell path variation* is a continuous mapping $H(s, t) : [0, 1] \times [0, 1] \rightarrow \mathcal{F}$, such that $H(s, 0) = \alpha^*(s)$ for $s \in [0, 1]$, with fixed endpoints $H(0, t) = p_s$ and $H(1, t) = p_r$ for $t \in [0, 1]$. Each path of this variation is denoted $\alpha_t(s)$. When $T(\alpha)$ is evaluated along the paths $\alpha_t(s)$, it becomes a function of the parameter t , $T(t) = \int_0^1 F(\alpha_t(s), \alpha'_t(s)) ds$. These paths cross the proximity cells boundary lines at $s = s_1, \dots, s_m$, hence $T(t)$ can be equivalently written as the sum: $T(t) = \sum_{i=0}^m \int_{s_i}^{s_{i+1}} F(\alpha_t(s), \alpha'_t(s)) ds$. Using Leibnitz rule, the derivative of $T(t)$ with respect to t is:

$$\frac{d}{dt} T(t) = \sum_{i=0}^m \int_{s_i}^{s_{i+1}} \left(\frac{\partial F(\alpha, \alpha')}{\partial \alpha} - \frac{d}{ds} \frac{\partial F(\alpha, \alpha')}{\partial \alpha'} \right) \cdot \frac{\partial H(s, t)}{\partial t} ds + \sum_{i=1}^m \left(\frac{\partial F(\alpha, \alpha')}{\partial \alpha'} \Big|_{s_i^-} - \frac{\partial F(\alpha, \alpha')}{\partial \alpha'} \Big|_{s_i^+} \right) \cdot \frac{\partial H(s_i, t)}{\partial t} \quad (14)$$

where we used the fact that $H(s, t)$ is a fixed endpoints variation. Each of the variant paths, $\alpha_t(s)$, consists of time optimal segments satisfying the Euler-Lagrange equation (4) in each proximity cell. Hence the integrals in (14) vanish for all $t \in [0, 1]$, and the derivative of $T(t)$ with respect to t becomes:

$$\frac{d}{dt} T(t) = \sum_{i=1}^m \left(\frac{\partial F(\alpha, \alpha')}{\partial \alpha'} \Big|_{s_i^-} - \frac{\partial F(\alpha, \alpha')}{\partial \alpha'} \Big|_{s_i^+} \right) \cdot \frac{\partial H(s_i, t)}{\partial t}. \quad (15)$$

According to (13), $F(\alpha, \alpha') = \|\alpha'(s)\| / \nu(s)$, where $\nu(s)$ is the robot’s maximal speed at $\alpha(s)$. The expression for $\partial F(\alpha, \alpha') / \partial \alpha'$ is thus: $\partial F(\alpha, \alpha') / \partial \alpha' = \frac{1}{\nu(s)} (\alpha'(s) / \|\alpha'(s)\|) = \frac{1}{\nu(s)} \hat{v}(s)$, where $\hat{v} = \alpha' / \|\alpha'\|$ is the unit magnitude

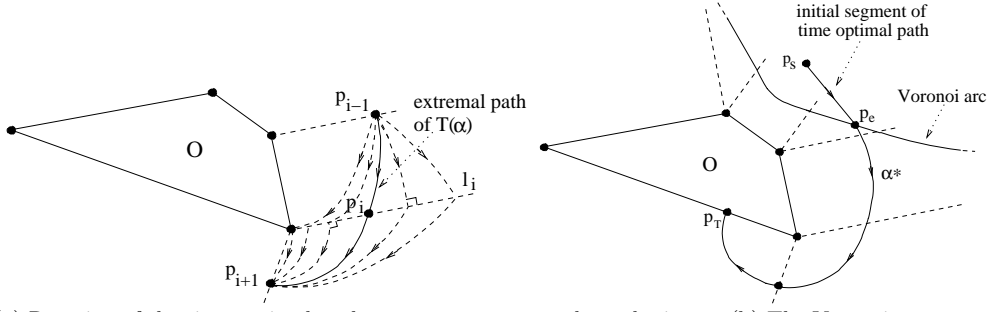


Figure 4: (a) Rotation of the time optimal path segment tangent at the endpoint p_i . (b) The Voronoi arc separating the start point p_s from the obstacle O , with an initial straight line segment along the time optimal path α^* .

tangent vector of α . Substituting for $\partial F(\alpha, \alpha')/\partial \alpha'$ in (15), using the subscript t to denote terms associated with α_t , gives:

$$\frac{d}{dt}T(t) = \sum_{i=1}^m \frac{1}{\nu_t(s_i)} (\hat{v}_t(s_i^-) - \hat{v}_t(s_i^+)) \cdot \frac{\partial}{\partial t} H(s_i, t). \quad (16)$$

Note that $\frac{d}{dt}T(0) = 0$, since α^* is a $C^{(1)}$ curve according to Proposition 4.2, hence $\hat{v}_t(s_i^-) = \hat{v}_t(s_i^+)$ for $i = 1 \dots m$ at $t=0$. Let us next determine the sign of the second derivative of $T(t)$ at $t=0$. Consider the two segments of α^* meeting at a common endpoint p_i on the line l_i . Let the segments' other endpoints, p_{i-1} and p_{i+1} , be held fixed. As p_i varies along l_i , the travel time along the time optimal segments having fixed endpoints at p_{i-1} and p_{i+1} form two *unimodal* functions. Hence, the *tangent* to each time optimal segment at p_i is collinear with l_i when p_i is located at the obstacle's vertex, rotates monotonically until it becomes orthogonal to l_i at the minimum travel time location of p_i along l_i , then continues to rotate monotonically until becoming collinear again with l_i as p_i moves arbitrarily far from O along l_i (Figure 4(a)). Based on this geometric insight, the *change* in the tangent of each time optimal path segment at p_i satisfies the relation: $\frac{d}{dt}(\hat{v}_t(s_i^-) \cdot \frac{\partial}{\partial t} H(s_i, t)) > 0$ and $\frac{d}{dt}(\hat{v}_t(s_i^+) \cdot \frac{\partial}{\partial t} H(s_i, t)) < 0$. Summation of the two inequalities gives:

$$\frac{d}{dt} \left((\hat{v}_t(s_i^-) - \hat{v}_t(s_i^+)) \cdot \frac{\partial}{\partial t} H(s_i, t) \right) > 0 \quad t \in [0, 1].$$

Hence, the second derivative of $T(t)$ at $t=0$ is:

$$\frac{d^2}{dt^2}T(0) = \sum_{j=1}^m \frac{1}{\nu(s_j)} \frac{d}{dt} \left((\hat{v}(s_j^-) - \hat{v}(s_j^+)) \cdot \frac{\partial}{\partial t} H(s_j, 0) \right) > 0$$

where we used the fact that $\hat{v}(s_i^-) = \hat{v}(s_i^+)$ at $t=0$. Since $\frac{d}{dt}T(0) = 0$ and $\frac{d^2}{dt^2}T(0) > 0$, $T(t)$ has a local minimum at $t=0$, which corresponds to the extremal path α^* . \square

The last property concerns the *uniqueness* of the safe time optimal path in each path homotopy class.

Theorem 2 (Optimal Path Uniqueness). *The safe travel time functional, $T(\alpha)$, possesses a unique time optimal path in each path homotopy class connecting p_s with p_T in \mathcal{F} .*

Proof sketch: According to Lemma 4.1, $T(\alpha)$ attains a global minimum in every path homotopy class connecting p_s with p_T in \mathcal{F} . Suppose one of the path homotopy classes contains *two* local minimum paths of $T(\alpha)$, α_{opt} and β_{opt} . Since α_{opt} and β_{opt} belong to the *same* path homotopy class, the two paths can be joined by a path homotopy, $f(s, t) : [0, 1] \times [0, 1] \rightarrow \mathcal{F}$, such that $f(s, 0) = \alpha_{opt}(s)$ and $f(s, 1) = \beta_{opt}(s)$ for $s \in [0, 1]$. The path homotopy, $f(s, t)$, can be interpreted as a continuous path connecting α_{opt} with β_{opt} in the metric space of piecewise smooth paths in \mathbb{R}^2 .

The functional $T(\alpha)$ is a continuous function of α . According to the mountain pass theorem [14], the collection of all continuous paths (each being a path homotopy) connecting α_{opt} and β_{opt} contains a distinguished path along which the maximal value attained by $T(\alpha)$ is *minimized*. Moreover, the maximal value is attained at an extremal path, γ , which is necessarily a *saddle point* of $T(\alpha)$. If γ touches an obstacle vertex, a new path homotopy between α_{opt} and β_{opt} can be obtained by locally pulling the path homotopy containing γ away from the vertex (Figure 3). The travel time along the modified path homotopy is reduced, hence the maximal value attained by $T(\alpha)$ along the modified path homotopy is *lower* than $T(\gamma)$. The path γ must therefore lie in the interior of \mathcal{F} . This leads to a contradiction, since every extremal path of $T(\alpha)$ in the interior of \mathcal{F} is a local minimum according to Theorem 1. Hence $T(\alpha)$ attains a unique minimum in each path homotopy class connecting p_s with p_T in \mathcal{F} . \square

5 Computation of the Time Optimal Docking Path

This section describes a convex optimization scheme for computing safe time optimal docking paths in \mathcal{F} . The computation scheme will use the following *Voronoi arc* separating the start point, p_s , from the obstacle.

Definition 5. *When p_s lies in the interior of \mathcal{F} , the Voronoi arc separating p_s from the obstacle O , denoted e , is the collection of points equidistant from p_s and O (Figure 4(b)).*

The following lemma specifies the shape of the safe time optimal path between p_s and the Voronoi arc.

Lemma 5.1. *Starting with zero speed at p_s , the safe time optimal path forms a **straight line** between p_s and the Voronoi arc separating p_s from the obstacle \mathcal{O} .*

Proof sketch: Consider the safe time optimal path α^* starting with zero speed at p_s . Let p_e be the crossing point of α^* with the Voronoi arc e . Since α^* is a local minimum of $T(\alpha)$, the portion of α^* between p_s and p_e is also a local minimum of $T(\alpha)$, with respect to paths which start with zero speed at p_s and reach p_e . The robot’s initial braking safety distance at $(p_s, 0)$ satisfies $d(\nu) = \nu^2/2a_{max} = 0$. Hence, if p_s is not located on the obstacle’s boundary, $d(\nu) < \text{dst}(p_s, \mathcal{O})$, and the uniform braking safety constraint is initially *inactive*. The robot can therefore safely use its maximal acceleration, a_{max} , in order to gain as much speed as possible. Accelerating with a_{max} along the straight line connecting p_s and p_e ensures maximal attainable speed along the shortest path connecting p_s with p_e . The straight line segment is therefore time optimal, and hence forms the portion of α^* between p_s and p_e . \square

The computation scheme focuses on the *piecewise time optimal paths* connecting p_s and p_T . These paths consist of a straight line segment leading from p_s to the Voronoi arc, followed by time optimal path segments in each proximity cell (Section 3). The crossing points of these paths with the Voronoi arc and proximity cell boundary lines will serve as the optimization variables. The computation starts with a selection of a particular path homotopy class connecting p_s and p_T in \mathcal{F} , associated with a sequence of $m + 1$ proximity cells with boundary lines l_1, \dots, l_m . The *crossing points* along e and l_1, \dots, l_m are parametrized by $p(u_0) : [0, 1] \rightarrow e$ and $p(u_i) : [0, 1] \rightarrow l_i$ for $i = 1 \dots m$ (Figure 5(a)). Since each time optimal path segment is uniquely determined by its two endpoints, the safe travel time functional, $T(\alpha)$, becomes a function of the crossing point parameters u_0, u_1, \dots, u_m . Computation of the safe time optimal path connecting p_s and p_T is next performed in the set $(u_0, \dots, u_m) \in [0, 1] \times \dots \times [0, 1]$, based on the following monotonicity property of $T(\alpha)$.

Proposition 5.2 ([11]). *Let $\alpha(u_0, \dots, u_m)$ parametrize the piecewise time optimal paths connecting p_s and p_T in \mathcal{F} , while crossing the Voronoi arc e and the proximity cell boundary lines l_1, \dots, l_m . Let $\mathbf{u}^* = (u_0^*, \dots, u_m^*)$ be the parameter values of the optimal crossing points. The functional $T(\alpha(u_0, \dots, u_m))$ is **monotonically increasing** along any ray emanating from \mathbf{u}^* in the set $(u_0, \dots, u_m) \in [0, 1] \times \dots \times [0, 1]$.*

Since $T(\alpha(u_0, \dots, u_m))$ is monotonically increasing along rays emanating from (u_0^*, \dots, u_m^*) , its level sets form *star shaped sets* in the set $(u_0, \dots, u_m) \in [0, 1] \times \dots \times [0, 1]$.² This property is illustrated in Figure 5, which plots the contours of $T(\alpha(u_0, u_1))$, for the piecewise time optimal paths passing through two crossing points, $p(u_0)$ and $p(u_1)$. The contours form convex sets (and hence star shaped sets) in the (u_0, u_1) plane, with center point at the optimal parameter values (u_0^*, u_1^*) .

Convex optimization algorithms require that the level sets of $T(\alpha(u_0, \dots, u_m))$ be convex. However, the same algorithms can be applied when the level sets are star shaped with a common center point at the optimal parameter values \mathbf{u}^* . This observation is based on the relation $\nabla T(\alpha(\mathbf{u})) \cdot (\mathbf{u} - \mathbf{u}^*) \geq 0$ satisfied by star shaped level sets, which allows construction of a *barrier function* (or a bounding ellipsoid) in each iteration of the convex optimization process. Standard convex optimization algorithms run in $O(m^3 \log(R/\epsilon))$ time [13], where m is the number of crossing points, R is the travel time difference between the initial path and the exact solution, and ϵ is the desired solution accuracy. The total number of crossing points is upper bounded by the number of obstacle features, n , and the convex optimization therefore runs in $O(n^3 \log(R/\epsilon))$ time.

6 Computation Example

This section describes a safe time optimal docking path example, depicted in Figure 6(a). The mobile robot is modeled a point moving with $a_{max} = 0.5 \text{ m/sec}^2$, which is typical for autonomous robots operating in warehouse environments. The example describes docking maneuvers around a rectangle obstacle corner. The obstacle is of size 15×30 meters, with vertices at $(0, 0)$, $(30, 0)$, $(30, -15)$, and $(0, -15)$ in the (x, y) plane. The start point is located at $p_s = (10, 10)$. Figure 6(a) depicts three time optimal docking paths, α_1 , α_2 , and α_3 , associated with targets located on the obstacle’s boundary at $p_{T_1} = (5, 0)$, $p_{T_2} = (0, -7.5)$, and $p_{T_3} = (10, -15)$. The initial path in each computation started with a straight line segment connecting p_s with a point on the Voronoi arc, followed by time optimal segments connecting the obstacle vertices leading to the respective target. The computation resulted in the time optimal docking paths depicted in Figure 6(a), with $T(\alpha_1) = 8.2$, $T(\alpha_2) = 17.1$, and $T(\alpha_3) = 27.8$ seconds.

Importantly, the time optimal solutions specify the geometric shape of the docking paths as well as the

²A set S in \mathbb{R}^m is *star shaped* when there exists a center point, $p_0 \in S$, such that every point $p \in S$ lies on a unique ray emanating from p_0 .

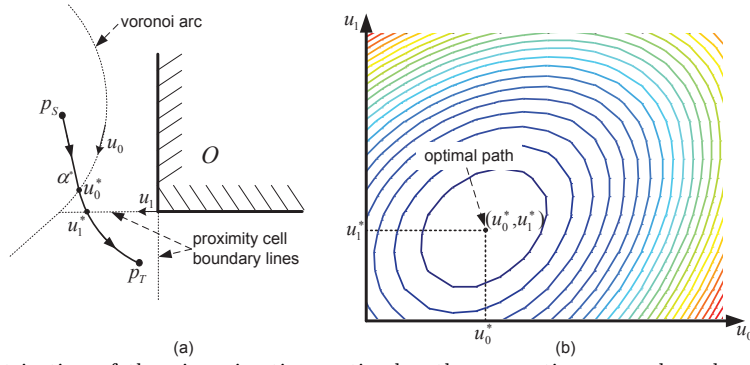


Figure 5: (a) Parametrization of the piecewise time optimal paths connecting p_s and p_T by (u_0, u_1) . (b) Contours of $T(\alpha(u_0, u_1))$ in the (u_0, u_1) plane.

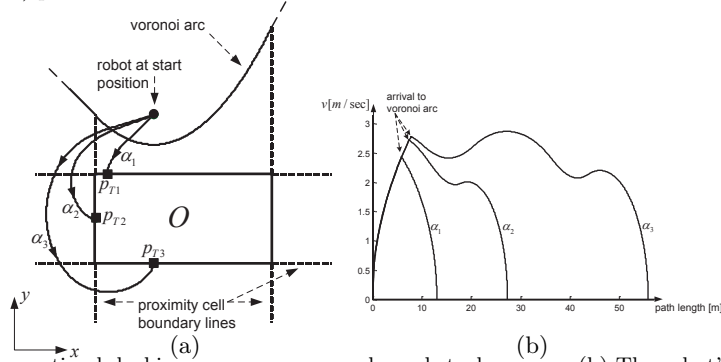


Figure 6: (a) Three time optimal docking maneuvers around an obstacle corner. (b) The robot’s speed plotted as a function of the path length parameter s .

speed profile along these paths. The robot’s speed along each path, $\nu(s)$, is plotted as a function of the path length parameter, s , in Figure 6(b). Note that each path starts with maximal acceleration along the straight line segment leading to the Voronoi arc. The robot next gradually decreases its speed until docking at the target, with maximal deceleration attained in the final approach to the obstacle’s boundary.

7 Conclusion

The paper considered the synthesise of *safe* time optimal paths for a mobile robot docking at targets located on a polygonal obstacle boundary. Starting with zero speed, the robot is required to approach a docking target with zero final speed, while maintaining a speed dependent circular safety zone centered at the robot’s current position. The safe travel time functional, $T(\alpha)$, was formulated and its basic properties were described. The paper next established that time optimal docking paths always start with a straight line segment leading to the Voronoi arc separating between the start point and the obstacle. Monotonicity of $T(\alpha)$ over the piecewise time optimal paths in each path homotopy class of the environment lead to an efficient convex optimization scheme executed over the piecewise time optimal paths, with optimization variables consisting of the paths’ crossing points locations on the Voronoi arc and boundary lines between proximity cells induced by the obstacle’s geometric features.

As demonstrated in simulation examples, the solutions specify the geometric shape of the time optimal docking paths, as well as the *speed profile* along these paths. This feature provides a significant progress with respect to classical configuration-space path planning approaches, which only generate collision avoiding paths without any consideration of the robot’s speed during the path planning process. Moreover, the geometric shape of the safe time optimal docking paths is *invariant* with respect to the robot’s maximal deceleration, a_{max} . Hence, while the actual travel time along the safe time optimal paths is a system dependent parameter, the geometric shape of the safe time optimal docking paths depends only on the location of the start and target relative to the obstacle.

An important extension concerns the generalization of the circular braking safety zone into a heading dependent safety zone. For instance, when a mobile robot travels with high speed in a long narrow corridor, a *non-circular* safety zone aligned with the corridor’s axis will allow the robot significantly higher speeds than the conservative circular safety zone. This extension would require planning in the mobile robot’s four-dimensional position and velocity space, and we hope to report initial results in the near future.

References

- [1] Traffic jam assistance. *Volvo Car Group*. www.media.volvocars.com/global, 2013.
- [2] O. Brock and O. Khatib. High-speed navigation using the global dynamic window approach. In *IEEE Int. Conf. on Robotics and Automation*, pages 341–346, 1999.
- [3] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun. *Principles of Robot Motion*. MIT Press, 2005.
- [4] P. Fiorini and Z. Shiller. Motion planning in dynamic environments using velocity obstacles. *Int. J. of Robotics Research*, 17(7):760–772, 1998.
- [5] D. Fox, W. Burgard, and S. Thrun. The dynamic window approach to collision avoidance. *Robotics Automation Magazine, IEEE*, 4(1):23–33, 1997.
- [6] T. Fraichard. A short paper about motion safety. In *IEEE Int. Conf. on Robotics and Automation*, pages 1140–1145, 2007.
- [7] I. M. Gelfand and S.V. Formin. *Calculus of Variations*. Prentice-Hall, 1963.
- [8] S. Karaman and E. Frazzoli. Sampling-based algorithms for optimal motion planning. *The Int. J. of Robotics Research*, 30(7):846–894, 2011.
- [9] F. Large, D. Vasquez, T. Fraichard, and C. Laugier. Avoiding cars and pedestrians using velocity obstacles and motion prediction. In *IEEE Intelligent Vehicle Symp.*, pages 375–379, 2004.
- [10] S. M. Lavalle. Rapidly-exploring random trees: A new tool for path planning. Technical report, Dept. of Computer Science, Iowa State University, 1998.
- [11] G. Manor. *Autonomous Mobile Robot Navigation With Velocity Constraints*. PhD thesis, Technion Israel Inst. of Technology, Haifa, Israel, <http://robots.technion.ac.il/publications>, 2014.
- [12] J. Markoff. Google cars drive themselves in traffic. *New York Times*, October 2010.
- [13] Y. E. Nesterov and A. S. Nemirovsky. *Interior Point Polynomial Methods in Convex Programming: Theory and Applications*. Springer Verlag, New York, 1992.
- [14] L. Nirenberg. Variational and topological methods in nonlinear problems. *Bulletin of the AMS*, 4(3): 267–302, 1981.
- [15] I. Sherr and M. Ramsey. Toyota and Audi move closer to driverless cars, January 2013.
- [16] Z. Shiller, S. Sharma, I. Stern, and A. Stern. Online obstacle avoidance at high speeds. *Int. J. of Robotics Research*, 32:1030–1047, 2013.
- [17] J. Snape, J. van den Berg, and S. J. Guy. The hybrid reciprocal velocity obstacle. *IEEE Trans. on Robotics*, 27(4):696–706, 2011.
- [18] P. R. Wurman, R. D’Andrea, and M. Mountz. Coordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI Magazine*, 29:9–19, 2008.