

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/224378743>

Shortest Paths to Obstacles for a Polygonal Dubins Car

Article in IEEE Transactions on Robotics · November 2009

DOI: 10.1109/TRO.2008.2011421 · Source: IEEE Xplore

CITATIONS

26

READS

156

2 authors:



Paolo Robuffo Giordano

Max Planck Institute for Biological Cybernetics

138 PUBLICATIONS 3,519 CITATIONS

[SEE PROFILE](#)



Marilena Vendittelli

Sapienza University of Rome

71 PUBLICATIONS 2,362 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Vision-based active estimation [View project](#)



Toward MAGMaS; Multiple Aerial-Ground Manipulator Systems [View project](#)

- [16] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1465–1476, Sep. 2004.
- [17] Z. Lin, B. A. Francis, and M. Maggiore, "Necessary and sufficient graphical conditions for formation control of unicycles," *IEEE Trans. Autom. Control*, vol. 50, no. 1, pp. 121–127, Jan. 2005.
- [18] T. I. Fossen, *Guidance and Control of Ocean Vehicles*. New York: Wiley, 1994.
- [19] D. B. Edwards, T. A. Bean, D. L. Odell, and M. J. Anderson, "A leader-follower algorithm for multiple AUV formations," in *Proc. IEEE/OES Auton. Underwater Vehicles*, 2004, pp. 40–46.
- [20] T. J. Koo and S. M. Shahruz, "Formation of a group of unmanned aerial vehicles (UAVs)," in *Proc. Amer. Control Conf.*, 2001, pp. 69–74.
- [21] L. E. Buzogany, M. Pachter, and J. J. D'Azzo, "Automated control of aircraft in formation flight," in *Proc. AIAA Guid., Navigat., Control Conf.*, 1993, pp. 1349–1370.
- [22] H. Schaub, S. R. Vadali, J. L. Junkins, and K. T. Alfriend, "Spacecraft formation flying control using mean orbit elements," *J. Astronaut. Sci.*, vol. 48, no. 1, pp. 69–87, 2000.
- [23] R. Burns, C. A. McLaughlin, J. Leitner, and M. Martin, "TechSat21: Formation design, control and simulation," in *Proc. IEEE Aerosp. Conf.*, 2000, vol. 7, pp. 19–25.
- [24] A. K. Das, R. Fierro, V. Kumar, J. P. Ostrowsky, J. Spletzer, and C. Taylor, "A vision-based formation control framework," *IEEE Trans. Robot. Autom.*, vol. 18, no. 5, pp. 813–825, Oct. 2002.
- [25] H. G. Tanner, G. J. Pappas, and V. Kumar, "Leader-to-formation stability," *IEEE Trans. Robot. Autom.*, vol. 20, no. 3, pp. 443–455, Jun. 2004.
- [26] F. Morbidi, G. L. Mariottini, and D. Prattichizzo, "Vision-based range estimation via Immersion and Invariance for robot formation control," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 504–509.
- [27] L. Consolini, F. Morbidi, D. Prattichizzo, and M. Tosques, "Leader-follower formation control of nonholonomic mobile robots with input constraints," *Automatica*, vol. 44, no. 5, pp. 1343–1349, 2008.

Shortest Paths to Obstacles for a Polygonal Dubins Car

Paolo Robuffo Giordano and Marilena Vendittelli

Abstract—In this paper, we characterize the time-optimal trajectories leading a Dubins car in collision with the obstacles in its workspace. Due to the constant velocity constraint characterizing the Dubins car model, these trajectories form a sufficient set of shortest paths between any robot configuration and the obstacles in the environment. Based on these paths, we define and give the algorithm for computing a distance function that takes into account the nonholonomic constraints and captures the nonsymmetric nature of the system. The developments presented here assume that the obstacles and the robot are polygons although the methodology can be applied to different shapes.

Index Terms—Dubins car, Pontryagin's maximum principle, shortest paths.

I. INTRODUCTION

In his pioneering work [1], Dubins determined the continuously differentiable curves of minimal length between any two points in the plane with assigned initial and final tangent and subject to curvature

Manuscript received August 30, 2008; revised December 4, 2008. First published February 2, 2009; current version published October 9, 2009. This paper was recommended for publication by Associate Editor E. Papadopoulos and Editor J.-P. Laumond upon evaluation of the reviewers' comments.

The authors are with the Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", 00185 Rome, Italy (e-mail: robuffo@dis.uniroma1.it; venditt@dis.uniroma1.it).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TRO.2008.2011421

constraints. These curves are time-optimal trajectories for a wheeled mobile robot with unitary speed and bounded steering velocity. For this reason, a vehicle with constant driving velocity is known in the literature as the Dubins car. The interest arisen by this system is due to both the theoretical challenges that it provides and its practical relevance in modeling the kinematics of road vehicles, aircrafts cruising at constant altitude, or sea vessels.

The difficulty of planning for a wheeled mobile robot derives from the pure rolling constraint of the wheels that prevents the vehicle to move instantaneously toward certain directions. As a consequence, not all the paths in the configuration space are feasible for this robot, and the Euclidean metric is not appropriate for determining the distance to obstacles populating the robot environment. In a previous work [2], we have defined and showed how to compute a distance between a pointwise robot and the obstacles in its environment by taking into account the nonholonomic constraints acting on the vehicle. In this paper, we consider both the nonholonomic constraint and the shape of the robot. Specifically, we define and compute a distance in the robot configuration space that takes into account the nonholonomic constraints of the vehicle. The computation does not require the explicit representation of the configuration space obstacles and is applicable to robots and obstacles that can be modeled as planar polygons. This distance function is particularly relevant in the context of motion planners relying on obstacle distance computation, like skeletonization, potential field methods, and even sampling-based motion planners. See, e.g., [3] for an interesting discussion on the use of the obstacle distance information in planning algorithms. In particular, the correct metric information provided by the distance developed in this paper can improve the efficiency of sampling-based methods: by computing the distance to the obstacles in the environment, it is possible to adjust the resolution at which configurations along local paths are checked for collision, in the line of [4]. This avoids the difficult resolution tuning step and provides an exact solution to the collision check test for local paths.

Our work relies on Dubins' results but adopts the optimal control approach proposed in [5] and [6]. In particular, the study of transversality conditions allows selecting a sufficient family of time-optimal trajectories whose length will determine the distance to the obstacles. A preliminary version of this paper has been presented in [7]. Here, we present a revised work extended with detailed proofs and formal equations for distance computation. The paper is organized as follows. In Section II, we summarize the basic results on time-optimal trajectories for the Dubins car. Due to the constant velocity constraint, these trajectories also minimize the path length. This is exactly the property that we exploit in defining the distance function in Section III where we also give the necessary conditions satisfied by the shortest paths bringing the robot in collision with obstacles. The distance function is based on the length of the shortest paths belonging to a sufficient family. In Section IV, we show how to reduce the cardinality of this set of paths to lower the distance computation complexity, and in Section V, we provide the formal system of equations needed for implementation. These are used in Section VI to obtain the isodistance curves on the xy plane for two robots with different shapes.

II. BACKGROUND MATERIAL

The kinematic model of the Dubins car considered in this paper is

$$\dot{\xi}(t) = f(\xi(t), u(t)) = g_1(\xi(t)) + g_2(\xi(t))u(t) \quad (1)$$

where $\xi \in R^2 \times S^1$ denotes the configuration of the vehicle, given by the position (x_r, y_r) of a reference point P , chosen as the mid point of the rear wheels' axle, and by the orientation θ of the vehicle w.r.t. the abscissa axis of a fixed reference frame. The drift and the input

vector fields are, respectively, given by $g_1(\xi) = (\cos \theta, \sin \theta, 0)^T$ and $g_2(\xi) = (0, 0, 1)^T$. In this model, the module of the driving velocity is assumed, without loss of generality (w.l.o.g.), to be 1 and the steering velocity $u(t)$ is such that $|u(t)| \leq 1$. The paths in the family obtained by Dubins in [1] are time-optimal solutions of system (1). Later, Sussmann and Tang [5] and Boissonnat *et al.* [6] obtained the same result by adopting an optimal control point of view. Exploiting this approach, we determined [2] the shortest paths between any given robot configuration and the obstacles populating its workspace both for the Reeds and Shepp car and for the Dubins car. In particular, we defined the distance between a point-wise robot and the polygonal obstacles in the environment as the length of the shortest path between the robot and the obstacles. Due to the one-to-one correspondence between the paths in the configuration space and the corresponding paths in the plane, this also represents a distance in configuration space. Recently [8], we have removed the hypothesis of point robot (i.e., we take into account the shape of the robot in determining the distance to the obstacles) for a Reeds and Shepp car. In this paper, we also extend the result in [8] to the Dubins car. In particular, using transversality conditions on the final state of the robot, complemented with continuity arguments, we show that, in this case, the search of the optimal solution can be restricted to a set of six paths. Our study is based on the results reported in the following section.

A. Shortest Paths Between Configurations

This section summarizes the results presented in [5], [6], and [9], leading to a synthesis of the shortest paths for a Dubins car. In accordance with the notation proposed in [5], we will use C and S to denote, respectively, an arc of circle of minimal radius and a straight line segment. To specify the direction of motion along the path, letters L and R will denote, respectively, a counterclockwise or clockwise sense of rotation of the driving velocity vector \vec{v} , while S will indicate motion along a straight segment. Subscripts are positive real numbers giving the length of each elementary path composing an optimal path and they will be referred to as *path parameters* (a, b, e) . For example, a path of type CSC may be specified as $L_a S_b R_e$, i.e., forward left turn of length a , straight motion of length b , and forward right turn of length e . The starting configuration of each path is assumed, w.l.o.g., to be the configuration space origin $(0, 0, 0)^T$.

Considering system (1), we want to minimize the time to travel from $\xi_i = \xi(t_i)$ to $\xi_f = \xi(t_f)$. For system (1), this is equivalent to minimizing the path length. The Hamiltonian [10] is

$$H = \langle \psi, f \rangle = \psi_1 \cos \theta + \psi_2 \sin \theta + \psi_3 u$$

where the costate ψ satisfies the adjoint equation

$$\dot{\psi} = \frac{\partial H}{\partial \xi}(\psi, \xi, u) = -\psi \left[\frac{dg_1}{d\xi} + u \frac{dg_2}{d\xi} \right] \quad (2)$$

for almost all t . Minimization of the Hamiltonian gives necessary conditions for a trajectory to be optimal. If a constraint on the final state $\chi(\xi_f) = 0$ of dimension σ_f is present, it is possible to derive a set of transversality conditions $\psi_f = M^T \zeta$, where $M = \partial \chi / \partial \xi_f$ is a $\sigma_f \times 3$ matrix and ζ is an auxiliary vector of dimension σ_f .

Results from [5] and [6], based on the Pontryagin maximum principle complemented by geometric arguments, allow to restrict the search of time-optimal paths between ξ_i and ξ_f to a sufficient set of paths consisting in concatenations of arcs of circle of minimum radius (C) or line segments (S). The line segments and the points of inflection lie on a line \mathcal{D}_0 defined by the equation

$$\mathcal{D}_0 : \psi_3 = \psi_1 y_r - \psi_2 x_r + \psi_3(t_0) = 0 \quad (3)$$

where ψ_1 and ψ_2 are constants and the ratio ψ_2/ψ_1 gives the direction of \mathcal{D}_0 , (see, e.g., [6, Lemma 7]).

The sufficient set of optimal paths can be partitioned into two families

$$\begin{aligned} \text{(I)} & C_a C_b C_e, \quad 0 \leq a \leq b, \quad \pi < b < 2\pi, \quad 0 \leq e \leq b, \quad \min\{a, e\} < b - \pi \\ \text{(II)} & C_a S_b C_e, \quad 0 \leq a < 2\pi, \quad 0 \leq e \leq 2\pi, \quad b \geq 0 \end{aligned} \quad (4)$$

where

- 1) family (I) includes two types: LRL and RLR ;
- 2) family (II) includes four types: LSL , LSR , RSL , and RSR .

These paths induce a partition of the configuration space into a finite number of cells, each corresponding to a path type optimal for linking a point in the cell to the origin [9]. Every Dubins path smoothly maps the parameter space into the robot configuration space \mathcal{C} , i.e., for each path p_i , one can define a function $W_i : \mathbb{R}^3 \rightarrow \mathcal{C}$ associating the final configuration in \mathcal{C} to the parameters (a, b, e)

$$\begin{pmatrix} x_r \\ y_r \\ \theta \end{pmatrix} = W_i(a, b, e) = \begin{pmatrix} X_i(a, b, e) \\ Y_i(a, b, e) \\ \Theta_i(a, b, e) \end{pmatrix} \quad (5)$$

where X_i, Y_i, Θ_i are smooth [11]. This property will be used in Section IV for reducing the family of candidate optimal paths.

III. DISTANCE FUNCTION

In deriving the distance, we take the following assumptions.

- 1) The robot \mathcal{R} is a Dubins car with kinematic model (1).
- 2) The robot moves in a planar workspace \mathcal{W} populated by a finite number m of static obstacles \mathcal{O}_j ($j = 1, \dots, m$).
- 3) $\mathcal{R}(\xi)$, the region occupied by the robot at configuration ξ , and the obstacles \mathcal{O}_j are compact subsets of \mathcal{W} . In particular, they are polygons.

Note that assumption 2) is very common in motion planning for nonholonomic robots. The compactness property required by assumption 3) is necessary for guaranteeing the existence of the distance to be defined. Finally, the methodology used in this paper can also be applied to nonpolygonal shapes but the developments described herein are valid only for robots and obstacles with polygonal shape.

Given these assumptions, we solve the following problem.

Problem 1: Determine the length d of the shortest path p^* from a given configuration ξ_i to a configuration ξ_f (called *optimal contact configuration*) such that the boundary of $\mathcal{R}(\xi_f)$ is in contact with the boundary of any of the \mathcal{O}_j 's.

Lemma 1: The search of the path p^* solving Problem 1 can be restricted to the families of Dubins optimal paths (4).

Proof: Assume that a solution p^* to Problem 1 has been found that does not belong to the families (4). Then, its length d is necessarily greater or equal to the length of the corresponding shortest path in (4) from ξ_i to ξ_f . If this was not true, it would contradict the fact that families (4) provide a sufficient set of optimal paths for the Dubins car, i.e., they contain the shortest path between *any* two given configurations (thus, including ξ_i and ξ_f). Since we are only interested in the length d of the path solving Problem 1, not in its particular shape, the search can be restricted to the families (4). ■

Note that the set of all possible contact configurations \mathcal{CO} can be constructed by "sliding" the robot along the boundaries of the obstacles and by letting the orientation θ varying in the interval $[-\pi, \pi]$. Finding a solution to Problem 1 corresponds to minimizing the length of the paths in families (4) over the set \mathcal{CO} encoding the constraint that the final configuration ξ_f should be in contact with any of the obstacles in the environment. By virtue of assumption 3), the set \mathcal{CO} is compact.

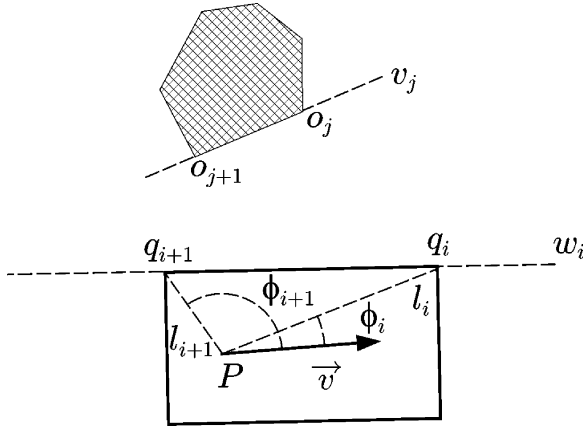


Fig. 1. Partitioning the distance computation.

On the other hand, the paths in families (4) induce a partition of the configuration space into a finite number of domains. In each domain, the length of the path from any point ξ to the origin is a continuous function of ξ [12]. Hence, a candidate solution to Problem 1 exists inside each region of the partition such that the intersection with \mathcal{CO} is not empty.¹ The necessary conditions to be satisfied by paths in each region, obtained through the application of transversality conditions, allow to determine the length of the candidate optimal path in each domain. The solution to Problem 1 is then obtained by minimizing the length of the path over the partition.

The one-to-one correspondence between paths in configuration space and the corresponding paths in the plane allows to choose the length d of p^* as the distance between the robot and the obstacles. Due to the nonsymmetric nature of system (1), the length of the shortest paths induces a quasi-metric because the symmetry property is not verified. Symmetry can always be recovered by properly defining the distance, even though the use of a nonsymmetric distance function in planning collision-free motion for the Dubins car would naturally take into account the nonsymmetric nature of this system. The implication of the presence of discontinuities in the distance function is, however, beyond the scope of this paper.

In a polygonal environment, Problem 1 can be solved considering the three subproblems of optimally bringing in contact (see Fig. 1):

- (i) a robot vertex q_i with an obstacle vertex o_j ;
- (ii) a robot vertex q_i with the line v_j through an obstacle;
- (iii) the line w_i through a robot edge with an obstacle vertex o_j .

Once solved these subtasks, the shortest path to an obstacle is found by iterating the three steps over all the robot/obstacle vertex/edge combinations and by choosing the minimum among the obtained path lengths. Problems (i)–(iii) can be associated to three functions

1. $L^{VV} : \mathbb{R}^4 \rightarrow \mathbb{R}$
 $(q_i, o_j) \rightarrow L^{VV}(q_i, o_j)$
2. $L^{VE} : \mathbb{R}^4 \rightarrow \mathbb{R}$
 $(q_i, v_j) \rightarrow L^{VE}(q_i, v_j)$
3. $L^{EV} : \mathbb{R}^4 \rightarrow \mathbb{R}$
 $(w_i, o_j) \rightarrow L^{EV}(w_i, o_j)$

¹To recover the correct structure of configuration space submanifold for \mathcal{CO} , it is sufficient to identify the configurations $(x_r, y_r, -\pi)$ and (x_r, y_r, π) .

where L^{VV} , L^{VE} , and L^{EV} are specified in the following sections. The distance d is defined as

$$d() : \mathbb{R}^8 \rightarrow \mathbb{R}$$

$$d() = \min \left\{ \min_{i,j} L^{VV}, \min_{i,j} L^{VE}, \min_{i,j} L^{EV} \right\}. \quad (6)$$

In the next sections, we will detail the procedure adopted to solve each specific subproblem. Following the idea in [11], we associate a *contact manifold* to each subproblem, expressing the constraint that the final state of the robot should be such that the corresponding type of contact (i.e., VV , VE , or EV) will occur between the robot and the obstacle. In addition, a transversality condition will express the constraint that the contact should occur in minimum time. These constraints will provide a system of three nonlinear equations in the three unknowns (a, b, e) . The search for the optimal path solving Problem 1 can be limited to family (II), as shown in Section IV.

In the following, we will denote by p a specific Dubins path and by OP the sufficient set of optimal paths (4).

A. Vertex–Vertex Distance

Let $p_k \in OP$ and (a_k, b_k, e_k) be the associated path parameters; we define the function

$$VV_{p_k} : \mathbb{R}^4 \rightarrow \mathbb{R}^3$$

$$(q_i, o_j) \rightarrow (a_k, b_k, e_k) \quad (7)$$

where $VV_{p_k}(q_i, o_j)$ is the map solving problem (i) with the path p_k . The function $L^{VV}(q_i, o_j)$ is then defined as

$$L^{VV}(q_i, o_j) = \min_{p_k \in OP} L_{p_k}(VV_{p_k}(q_i, o_j)) \quad (8)$$

where L_{p_k} is the length of p_k in terms of the parameters (a_k, b_k, e_k) .

Remarks: When solving for each p_k , three scenarios may arise:

- 1) the solution does not exist, i.e., at least one path parameter is complex;
- 2) the solution exists but it is not valid, i.e., at least one path parameter is outside its range of validity;
- 3) a valid solution exists.

In the first two cases, p_k is discarded.

With reference to Fig. 1, let P be the robot reference point and (l_i, ϕ_i) be the pair representing the length of the segment $\overline{Pq_i}$ and the angle between the vectors $\overrightarrow{Pq_i}$ and \vec{v} . The cartesian coordinates (q_{ix}, q_{iy}) of the robot vertex q_i are given by

$$\begin{cases} x_r + l_i \cos(\theta + \phi_i) = q_{ix} \\ y_r + l_i \sin(\theta + \phi_i) = q_{iy} \end{cases} \quad (9)$$

Denoted by (o_{jx}, o_{jy}) the cartesian coordinates of the target vertex o_j of the obstacle, the 1-D contact manifold is defined as

$$C_{VV}^{ij}(x_r, y_r, \theta) = \{(x_r, y_r, \theta) | q_i \equiv o_j\} \quad (10)$$

and can be represented by the following equation:

$$\chi_{VV}^{ij}(x_r, y_r, \theta) = \begin{pmatrix} x_r - o_{jx} + l_i \cos(\theta + \phi_i) \\ y_r - o_{jy} + l_i \sin(\theta + \phi_i) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \quad (11)$$

Equation (11) describes a vertical helix centered on o_j and will be used for finding the solution path, i.e., for determining the path parameters (a_k, b_k, e_k) associated to p_k . The additional constraint, necessary to make the problem “square” (three parameters and three equations), is derived from transversality conditions, as shown later.

Lemma 2: If a path $p \in OP$ is optimal for problem (i), then the line \mathcal{D}_0 associated to p passes through the point o_j .

Proof: The constraint on the robot final state $\chi_{VV}^{ij}(\xi_f) = 0$ can be used to derive the transversality condition (necessary for optimality) $\psi_f = M^T \zeta$, where

$$M = \frac{\partial \chi_{VV}^{ij}(\xi_f)}{\partial \xi_f} = \begin{pmatrix} 1 & 0 & -l_i \sin(\theta(t_f) + \phi_i) \\ 0 & 1 & l_i \cos(\theta(t_f) + \phi_i) \end{pmatrix}$$

and $\zeta = (\zeta_1 \quad \zeta_2)^T$. We then get the system

$$\begin{cases} \psi_1 = \zeta_1 \\ \psi_2 = \zeta_2 \\ \psi_3(t_f) = -l_i \sin(\theta(t_f) + \phi_i) \zeta_1 + l_i \cos(\theta(t_f) + \phi_i) \zeta_2. \end{cases}$$

By substituting ψ_1 and ψ_2 in the third equation, we obtain

$$\psi_3(t_f) = -l_i \sin(\theta(t_f) + \phi_i) \psi_1 + l_i \cos(\theta(t_f) + \phi_i) \psi_2. \quad (12)$$

From the definition (3) of ψ_3 , we get $\psi_3(t_0) = -\psi_1 o_{jy} + \psi_2 o_{jx}$. Thus, the line \mathcal{D}_0 has equation

$$\psi_3 = \psi_1(y_r - o_{jy}) - \psi_2(x_r - o_{jx}) = 0 \quad (13)$$

which implies our thesis. ■

B. Vertex–Edge Distance

In this section, we will show how to solve problem (ii). In this case, the contact manifold is 2-D. Two additional constraints will be derived from transversality conditions. Since we assume that the target edge of the obstacle is an unbounded line, there could be solutions returning a contact point outside the edge boundaries; in this case, the solution is discarded.

Analogously to the previous section, the function

$$VE_{p_k}(): \mathbb{R}^4 \rightarrow \mathbb{R}^3 \\ (q_i, v_j) \rightarrow (a_k, b_k, e_k)$$

describes the map that solves the problem (ii) for a path $p_k \in OP$ returning the three path parameters and—as a byproduct—the contact point on the edge. The $L^{VE}()$ function can be expressed as

$$L^{VE}(q_i, v_j) = \min_{p_k \in OP} L_{p_k}(VE_{p_k}(q_i, v_j)). \quad (14)$$

The remarks stated for (8) also hold in this case.

Let $y = m_j x + n_j$ be the equation of the target edge v_j ; using (9), the contact manifold $C_{VE}^{ij}(x_r, y_r, \theta) = \{(x_r, y_r, \theta) | q_i \in v_j\}$ is described by

$$\chi_{VE}^{ij}(x_r, y_r, \theta) \\ = y_r - m_j x_r - n_j - l_i m_j \cos(\theta + \phi_i) + l_i \sin(\theta + \phi_i) = 0. \quad (15)$$

Equation (15) represents a 2-D surface whose projection on the plane xy , for a given θ , is a line parallel to v_j .

Lemma 3: If a path in OP is optimal for the problem (ii), then:

- 1) the line \mathcal{D}_0 is perpendicular to the line v_j ;
- 2) the contact point lies at the intersection of \mathcal{D}_0 and v_j .

Proof: The constraint $\xi_f \in C_{VE}^{ij}$, expressed as $\chi_{VE}^{ij}(\xi_f) = 0$, yields $\psi_f = M^T \zeta$ where $M = (\partial \chi_{VE}^{ij}(\xi_f)) / \partial \xi_f$ is given by

$$M = \begin{pmatrix} -m_j & 1 & l_i m_j \sin(\theta(t_f) + \phi_i) + l_i \cos(\theta(t_f) + \phi_i) \end{pmatrix}.$$

Thus, we get the system

$$\begin{cases} \psi_1 = -m_j \zeta \\ \psi_2 = \zeta \\ \psi_3(t_f) = (l_i m_j \sin(\theta(t_f) + \phi_i) + l_i \cos(\theta(t_f) + \phi_i)) \zeta \end{cases}$$

from which we get the following relations:

$$\frac{\psi_2}{\psi_1} = -\frac{1}{m_j} \quad (16)$$

$$\psi_3(t_f) = -l_i \sin(\theta(t_f) + \phi_i) \psi_1 + l_i \cos(\theta(t_f) + \phi_i) \psi_2. \quad (17)$$

Point 1) is proved by (16); by using (3), (9), and (17), we can compute the constant $\psi_3(t_0) = -\psi_1 q_{iy}(t_f) + \psi_2 q_{ix}(t_f)$ that yields

$$\psi_3(t) = \psi_1(y_r - q_{iy}(t_f)) - \psi_2(x_r - q_{ix}(t_f)) = 0 \quad (18)$$

which implies that the point q_i at the end of the path must lie on the line \mathcal{D}_0 ; thus, combining with the contact condition between $q_i(t_f)$ and the target line v_j , we prove point 2). ■

C. Edge–Vertex Distance

The approach adopted to solve problem (iii) is analogous to the method outlined in the last section: we assume an unbounded robot edge and we discard any solution yielding a contact point outside the edge boundaries (q_i, q_{i+1}) . Defining

$$EV_{p_k}(): \mathbb{R}^4 \rightarrow \mathbb{R}^3 \\ (w_i, o_j) \rightarrow (a_k, b_k, e_k)$$

as the map solving problem (iii) for p_k , the $L^{EV}()$ function is

$$L^{EV}(w_i, o_j) = \min_{p_k \in OP} L_{p_k}(EV_{p_k}(w_i, o_j)). \quad (19)$$

Let (q_i, q_{i+1}) be two adjacent robot vertices. Using (9), the line w_i in Fig. 1 can be expressed as $y = m_i(\theta)x + n_i(x_r, y_r, \theta)$, where

$$\begin{cases} m_i(\theta) = \tan(\theta + \theta_{0i}) \\ n_i(\xi) = q_{iy} - m_i(\theta)q_{ix} \\ \theta_{0i} = \arctan \frac{l_i \sin(\phi_i) - l_{i+1} \sin(\phi_{i+1})}{l_i \cos(\phi_i) - l_{i+1} \cos(\phi_{i+1})} \end{cases}$$

and θ_{0i} is the angle between w_i and the robot heading direction \vec{v} .

Denoting by $o_j = (o_{jx}, o_{jy})$ the target point, the contact manifold

$$C_{EV}^{ij}(x_r, y_r, \theta) = \{(x_r, y_r, \theta) | o_j \in w_i\} \quad (20)$$

can be represented as

$$\chi_{EV}^{ij}(\xi) = o_{jy} - m_i(\theta)o_{jx} - n_i(\xi) \quad (21)$$

and describes, as θ varies, a 2-D surface whose projection on the plane is made of lines rotating at a fixed distance from o_j .

Lemma 4: If a path in OP is optimal for problem (iii), then:

- 1) the line \mathcal{D}_0 must be perpendicular to the edge w_i at $t = t_f$;
- 2) the contact point lies at the intersection of \mathcal{D}_0 and w_i .

Proof: The constraint $\xi_f \in C_{EV}^{ij}$, expressed as $\chi_{EV}^{ij}(\xi_f) = 0$, yields the transversality conditions $\psi_f = M^T \zeta$, where

$$M = \frac{\partial \chi_{EV}^{ij}(\xi_f)}{\partial \xi_f} \\ = [s_f \quad -c_f \quad (-o_{jy} + y_r(t_f))s_f + (-o_{jx} + x_r(t_f))c_f]$$

with $s_f = \sin(\theta(t_f) + \theta_{0_i})$ and $c_f = \cos(\theta(t_f) + \theta_{0_i})$. Hence, we get the system

$$\begin{cases} \psi_1 = s_f \zeta \\ \psi_2 = -c_f \zeta \\ \psi_3(t_f) = ((-o_{j_y} + y_r(t_f))s_f + (-o_{j_x} + x_r(t_f))c_f)\zeta \end{cases}$$

which yields

$$\frac{\psi_2}{\psi_1} = -\frac{1}{\tan(\theta(t_f) + \theta_{0_i})} = -\frac{1}{m_i(\theta(t_f))} \quad (22)$$

$$\psi_3(t_f) = \psi_1(y_r(t_f) - o_{j_y}) - \psi_2(x_r(t_f) - o_{j_x}). \quad (23)$$

Equation (22) proves point 1) of the lemma; from (23), we have $\psi_3(t_0) = -\psi_1 o_{j_y} + \psi_2 o_{j_x}$, which yields

$$\psi_3(t) = \psi_1(y_r(t) - o_{j_y}) - \psi_2(x_r(t) - o_{j_x}) = 0. \quad (24)$$

This relation, together with the contact constraint between o_j and the line w_i , proves point 2). ■

Remark: For the *EV* case, condition (23) on the final value $\psi_3(t_f)$ is formally different from the corresponding conditions in the *VV* and *VE* cases [see (12) and (17)]. That form can be recovered as follows: assume that the *EV* problem has a valid solution, and let $q_c : \{l_c, \phi_c\}$ be the resulting contact point on the edge w_j . Then, by exploiting (9) and the fact that, by construction, $q_c \equiv o_j$ at $t = t_f$, we can rewrite (23) as

$$\psi_3(t_f) = -l_c \sin(\theta(t_f) + \phi_c)\psi_1 + l_c \cos(\theta(t_f) + \phi_c)\psi_2 \quad (25)$$

which is formally identical to (12) and (17).

Note that conditions (12), (17), and (25) depend linearly on the parameters l_i and l_c that tend to 0 when the robot shape tends to a point. Thus, these conditions smoothly degenerate in the unique condition $\psi_3(t_f) = 0$ given in [13] for a point robot.

IV. FAMILY REDUCTION

In this section, we show that a path *CCC* in family (I) is never optimal. The proof takes advantage of the continuity of the path parameters (a, b, e) w.r.t. the parameter l_i of the robot. Some preliminary results are needed.

Lemma 5: For any robot/obstacle vertex/edge $q_i/o_j, w_i/v_j$:

- 1) if $L^* = L^{VE}(q_i, v_j)$ is the optimal solution of problem (ii) and P_l^* the associated contact point, then $L^* = L^{VV}(q_i, P_l^*)$;
- 2) if $L^* = L^{EV}(w_i, o_j)$ is the optimal solution of problem (iii) and P_l^* the associated contact point, then $L^* = L^{VV}(P_l^*, o_j)$.

Proof: See [8, Lemma 4]. ■

Lemma 6: ψ_1 and ψ_2 are smooth functions of the Dubins path parameters (a, b, e) .

Proof: Given a Dubins path p , the associated line \mathcal{D}_0 passes through all the inflection points and all the points on the straight segment S (if present in p). Let F be any of these points, and $p|_F$ the subpath of p that goes from the origin to F . Since $p|_F$ is still a Dubins path [5], the coordinates of F are smooth functions of (a, b, e) via the maps (5). From Lemma 2, we also know that line \mathcal{D}_0 must pass through the target point o_j . Hence, slope of \mathcal{D}_0 is $\psi_2/\psi_1 = (F_y - o_{j_y})/(F_x - o_{j_x})$, or equivalently, $\psi_2 = (F_y - o_{j_y})v$, $\psi_1 = (F_x - o_{j_x})v$ for any $v \neq 0$, which proves the thesis. ■

Theorem 1: A path *CCC* in family (I) is not optimal for solving problems (i)–(iii).

Proof: Lemma 5 allows to focus only on the vertex–vertex distance L^{VV} , since the other two functions can always be reduced to this case. The equations to be solved for computing the L^{VV} distance are given

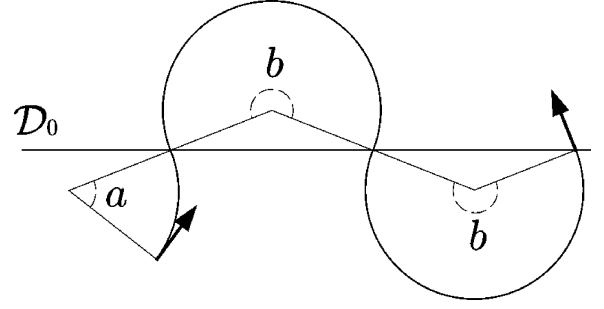


Fig. 2. Path $C_a C_b C_b$ that ends on the line \mathcal{D}_0 .

by the 2-D contact condition (11) and the necessary condition (13) for any pair (x_r, y_r) such that the robot reference point lies on the line \mathcal{D}_0 . By direct inspection, these equations are smooth w.r.t. (l_i, ϕ_i, o_j) and w.r.t. the triplet (x_r, y_r, θ) , which, in turn, is a smooth function of the Dubins parameters (a, b, e) due to the maps W_i in (5). In particular, the Dubins parameters appear either algebraically (when relative to straight segments) or as arguments of trigonometric functions (when relative to arcs of circle); the change of variable $\alpha = \tan(\beta/2)$ applied to each trigonometric function can then transform the original conditions into a set of three polynomial equations in the unknowns (a, b, e) , with coefficients being smooth functions of the quantities (l_i, ϕ_i, o_j) . At this point, we exploit the well-known fact that the roots of a polynomial depend continuously on its coefficients [14]. Hence, the roots (a, b, e) will be continuous functions of the polynomial coefficients, which, in turn, are smooth functions of (l_i, ϕ_i, o_j) .

From [15], we know that when $l_i = 0$ (point-wise robot), the optimal paths must satisfy the condition $\psi_3(t_f) = 0$, i.e., the car must stop exactly on the line \mathcal{D}_0 . A path *CCC* meets this condition only if it is of type $C_a C_b C_b$, as depicted in Fig. 2. The continuity of the parameters (a, b, e) w.r.t. l_i implies that a path *CCC* must converge continuously toward $C_a C_b C_b$ when $l_i \rightarrow 0$. The authors in [5] proved that a path $C_a C_b C_b$ is never optimal. Thus, by continuity, a path *CCC* is not optimal for solving problems (i)–(iii). ■

Note that Theorem 1 does not completely exclude family (I) from the set of candidate paths, since all subpaths of type $\{C_a C_b, C_b C_e, C_b\}$ are still valid candidates for the distance computation. However, for our goals, these subpaths can also be thought as degenerate cases of a path *CSC* in family (II) when the segment S (and possibly an arc C) is not present. In other words, the formal system of equations needed to determine a subpath *CC* does not depend on its parent family, so that any subinstance of family (I) can be actually treated as an equivalent subinstance of family (II). Still, when solving problems (i)–(iii), a distinction between full *CSC* paths and subinstances *CC* is necessary because the line \mathcal{D}_0 behaves differently in these two cases. This fact, and its implications, will be addressed in the next section.

V. PATH LENGTH COMPUTATION

In this section, we give the formal expression of a square system of equations in the three unknowns (a, b, e) to be solved for computing the length L of the shortest path for each of the three contact types (i.e., *VV*, *VE*, and *EV*) defined in Section III (i.e., the three subproblems in which we have decomposed the problem of distance computation). Once determined the path parameters (a_k, b_k, e_k) for a candidate path p_k , the associated length L_{p_k} is computed as $L_{p_k} = a_k + b_k + e_k$.

Due to Theorem 1, a path *CCC* is never optimal for bringing in contact the robot with the obstacles in the environment. The search

of the shortest path to collision can therefore be restricted to the set $OP^* = OP_{C_a S_b C_e} \sqcup OP_{C_a C_b}$, where:

- 1) $OP_{C_a S_b C_e} = \{L_a S_b L_e, L_a S_b R_e, R_a S_b L_e, R_a S_b R_e\}$ is the set of paths belonging to family (II) for which D_0 coincides with the segment S or, if S is not present, D_0 is still tangent to the path at the inflection point;
- 2) $OP_{C_a C_b} = \{L_a R_b, R_a L_b\}$ is the set of (degenerate) paths belonging to family (II) for which the segment S is not present, and D_0 is not tangent to the path at the inflection point. Note that this set also includes all the degenerate cases of family (I) (see end of the previous section).

This distinction cannot be avoided since the different properties of the slope of D_0 give rise to different systems of equations for the paths in the two sets. Therefore, in implementing the algorithm for the distance computation, a total of six cases (the cardinality of OP^*) must be considered.

The following facts are used to determine the systems of equations to be solved:

- 1) for every $p_k \in OP^*$, the robot configuration along p_k is associated to a smooth map $(x_r, y_r, \theta)^T = W_k(a, b, e)$ [see (5)];
- 2) for every path $p_k \in OP^*$, the robot configuration at the end of the arc C_a , computed via (5), is $(\bar{X}_k(a), \bar{Y}_k(a), \bar{\Theta}_k(a)) = (\sin(a), \pm 1 \mp \cos(a), \pm a)$, where the sign $+$ ($-$) is associated to left (right) turn;
- 3) the orientation of the robot at the end of a path $p_k \in OP_{C_a S_b C_e}$ is $\theta(t_f) = \pm a_k \pm e_k$, while for a path $p_k \in OP_{C_a C_b}$, it is $\theta(t_f) = \pm a_k \pm b_k$;
- 4) for a path $p_k \in OP_{C_a S_b C_e}$, the slope of D_0 is

$$\frac{\psi_2}{\psi_1} = \pm \tan(a). \quad (26)$$

Case I-A: $VV_{p_k}(q_i, o_j)$, $p_k \in OP_{C_a S_b C_e}$

The contact constraint (11) and the necessary condition (13), combined with (26), lead to the following system of three equations in the three unknowns (a, b, e) :

$$\begin{cases} \chi_{VV}^{ij}(W_k(a, b, e)) = 0 \\ \cos(a)(\bar{Y}_k(a) - o_{jy}) \mp \sin(a)(\bar{X}_k(a) - o_{jx}) = 0. \end{cases} \quad (27)$$

For the sake of illustration, consider the following situation: a robot with a vertex q_i , represented by the pair (l_i, ϕ_i) [Fig. 3(b)], must reach the point o_j along an $L_a S_b L_e$ path [Fig. 3(a)]. In this case, system (27) takes the expression

$$\begin{cases} b \cos(a) + \sin(a + e) + l_i \cos(a + e + \phi_i) = o_{jx} \\ 1 + b \sin(a) - \cos(a + e) + l_i \sin(a + e + \phi_i) = o_{jy} \\ o_{jx} \sin(a) + (1 - o_{jy}) \cos(a) - 1 = 0. \end{cases} \quad (28)$$

The last equation is of the type $A \sin(a) + B \cos(a) + C = 0$ and admits the two solutions

$$a = \arctan \frac{-CA \mp B\sqrt{B^2 + A^2 - C^2}}{-CB \pm A\sqrt{B^2 + A^2 - C^2}}. \quad (29)$$

If both solutions are not valid, e.g., they are outside the validity range, or result in complex numbers, the path can be discarded. By solving the first two equations in (28) for b , one gets

$$\begin{cases} b = -\frac{l_i \cos(a + e + \phi_i) + \sin(a + e) - o_{jx}}{\cos(a)} \\ b = -\frac{1 + l_i \sin(a + e + \phi_i) - \cos(a + e) - o_{jy}}{\sin(a)} \end{cases} \quad (30)$$

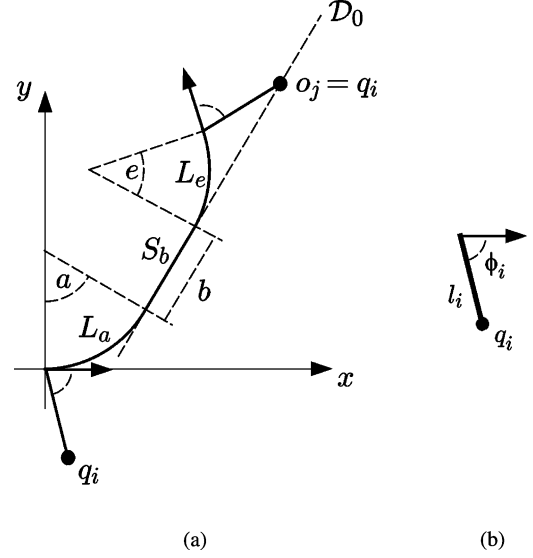


Fig. 3. Example of the VV computation for an $L_a S_b L_e$ path.

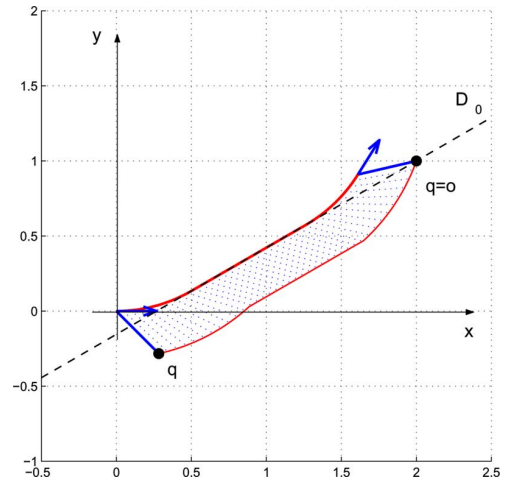


Fig. 4. Distance between the robot vertex q and the point o is the length of the robot shortest path bringing the vertex q in o .

which yields

$$l_i \cos(\phi_i) \sin(e) + (l_i \sin(\phi_i) - 1) \cos(e) + o_{jx} \sin(a) + (1 - o_{jy}) \cos(a) = 0.$$

This is an equation of the form $A \sin(e) + B \cos(e) + C = 0$ and can be solved for e as in (29). Finally, the parameter b is obtained from the two (equivalent) equations in (30), by selecting the nonsingular one according to the value of a . In principle, system (28) can have four admissible solutions, i.e., the path parameters are in the range of validity of LSL paths as given in (4). The solution is found by comparing the length of the candidate paths. It is easy to verify that similar developments also allow to solve system (27) for the other types of paths belonging to family (II).

As an example, Fig. 4 shows the optimal path for a robot with a vertex q_i in $(l_i, \phi_i) = (0.4, -\pi/4)$ to the point $o_j = (2, 1)$. The shortest path type is $L_a S_b L_e$, with $(a, b, e) = (0.5236, 0.8802, 0.4886)$ and total length $L = 1.8924$.

Case I-B: $VV_{p_k}(q_i, o_j)$, $p_k \in OP_{C_a C_b}$

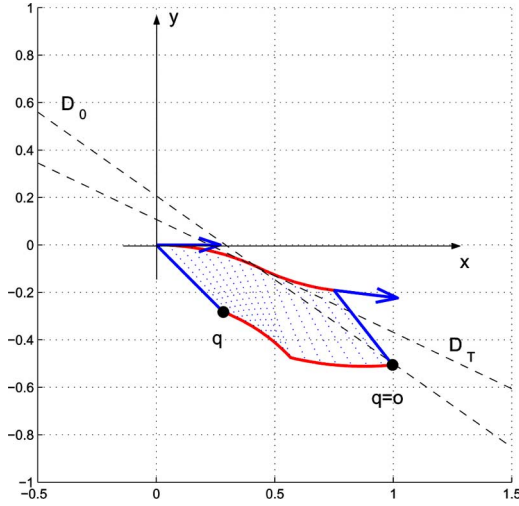


Fig. 5. Shortest path bringing the robot vertex q in o is of the type $R_a L_b$ belonging to the set of paths OP_{CC} for which D_0 is not tangent to the path at the inflection point. The actual tangent D_T is also shown to emphasize this fact.

In this case, condition (13) is not informative because (26) does not hold. However, the two equations (11) of the contact constraint $\chi_{VV}^{ij}(W_k(a, b, 0)) = 0$ are sufficient to make the problem square (two parameters for two equations). For illustration, consider the same situation of Fig. 3, but along a path $R_a L_b$. Conditions (11) become

$$\begin{cases} 2\sin(a) + l_i \cos(a - b - \phi_i) - \sin(a - b) = o_{jx} \\ 2\cos(a) - 1 - l_i \sin(a - b - \phi_i) - \cos(a - b) = o_{jy} \end{cases} \quad (31)$$

The change of variables $\alpha = \tan(a/2)$ and $\beta = \tan(b/2)$ transforms (31) in an algebraic system that yields the following second-order equation for α :

$$(2l_i \sin(\phi_i) + o_{jy}^2 + 8 + o_{jx}^2 + 6o_{jy} - l_i^2)\alpha^2 - 8o_{jx}\alpha + o_{jx}^2 + o_{jy}^2 + 2l_i \sin(\phi_i) - 2o_{jy} - l_i^2 = 0. \quad (32)$$

Hence, $a = 2 \arctan(\alpha)$ can be determined by solving (32). As before, in general, this results in two distinct solutions, i.e., the two roots of (32). Having determined a , value of b can be obtained by solving one of the two equations in (31), which, again, can be put in the form $A \sin(b) + B \cos(b) + C = 0$. As in the previous case, system (31) may have up to four different solutions, and the final one is found by comparing the resulting path lengths.

Fig. 5 depicts the shortest path for a robot with a vertex q_i in $(l_i, \phi_i) = (0.4, -\pi/4)$ to the point $o_j = (1, -0.5)$. In this case, the shortest path is of type $R_a L_b$ with $(a, b) = (0.444, 0.334)$ and total length $L = 0.7785$. As expected, line D_0 , which passes through the inflection point and o_j , is not tangent to the path—the actual tangent D_T is reported in the plot for the reader's convenience.

Case 2-A: $VE_{p_k}(q_i, v_j)$, $p_k \in OP_{C_a S_b C_e}$

In this case, the contact constraint is given by (15) while the necessary conditions given by Lemma 3 are expressed by (16) and (18). By combining these equations together with (26), we can write the following system of three equations in the three unknowns (a, b, e) :

$$\begin{cases} \chi_{VE}^{ij}(W_k(a, b, e)) = 0 \\ \pm m_j \sin(a) + \cos(a) = 0 \\ \cos(a)(\bar{Y}_k(a) - q_{jy}(a, b, e)) \mp \sin(a)(\bar{X}_k(a) - q_{jx}(a, b, e)) = 0 \end{cases} \quad (33)$$

which can be solved similarly to the previous cases.

Case 2-B: $VE_{p_k}(q_i, v_j)$, $p_k \in OP_{C_a C_b}$

Like in case 1-A, the condition $\psi_2/\psi_1 = \pm \tan(a)$ is not valid, but only the two unknowns (a, b) have to be determined. By plugging (16) into (18), and by using the contact constraint (15), we get the following system of two equations in the unknowns (a, b) :

$$\begin{cases} \chi_{VE}^{ij}(W_k(a, b, 0)) = 0 \\ -m_j(\bar{Y}_k(a) - q_{jy}(a, b, 0)) - (\bar{X}_k(a) - q_{jx}(a, b, 0)) = 0 \end{cases} \quad (34)$$

which can be solved similarly to the previous cases.

Case 3-A: $EV_{p_k}(w_i, o_j)$, $p_k \in OP_{C_a S_b C_e}$

The contact conditions (21), the necessary conditions of Lemma 4, expressed by (22) and (24), together with (26), allow writing the following system of three equations in the three unknowns (a, b, e) :

$$\begin{cases} \chi_{EV}^{ij}(W_k(a, b, e)) = 0 \\ \pm \sin(a) \sin(\theta(t_f + \theta_{0_i})) + \cos(a) \cos(\theta(t_f) + \theta_{0_i}) = 0 \\ \cos(a)(\bar{Y}_k(a) - o_{jy}) \mp \sin(a)(\bar{X}_k(a) - o_{jx}) = 0 \end{cases} \quad (35)$$

which can be solved following the approach outlined in the VV cases.

Case 3-B: $EV_{p_k}(w_i, o_j)$, $p_k \in OP_{C_a C_b}$

Condition (26) does not hold. Nevertheless, the two unknowns (a, b) can be determined by plugging (22) into (24), and combining with the contact constraint (21), to obtain the square system

$$\begin{cases} \chi_{EV}^{ij}(W_k(a, b, 0)) = 0 \\ -s_f(\bar{Y}_k(a) - o_{jy}) + c_f(\bar{X}_k(a) - o_{jx}) = 0 \end{cases} \quad (36)$$

where $s_f = \sin(\theta(t_f) + \theta_{0_i})$ and $c_f = \cos(\theta(t_f) + \theta_{0_i})$. The solution of system (36) can be found as in the other cases.

Recalling the definition of the distance (6) and the definitions (8), (14), and (19), it is easy to show that for a robot with m vertices moving in an environment populated by polygonal obstacles with n vertices, the complexity of the distance computation is $O(3 \cdot m \cdot n \cdot 6)$ where 3 accounts for the three subproblems to be solved to determine the distance and 6 is the number of candidate shortest paths providing the value of the distance function.

VI. ISODISTANCE CURVES

This last section presents the isodistance curves for a robot at the origin of the configuration space. We computed the nonholonomic distance to the points in the xy plane, and assigned the same gray-scale level to those points sharing the same distance value. In order to obtain a wavefront effect, a modulus operation resets the gray-scale level when the distance is greater than a predetermined threshold.

Fig. 6(a) shows the result for a point robot. The small arrow at the origin of the xy reference frame represents the robot configuration. In this case, the distance encodes only the nonholonomic constraint and the nonsymmetric nature of the system since the robot is a point. Operatively, this means that the nonholonomic distance is computed by considering only the VV distance between the robot reference point and the points in the plane. The difference with the Euclidean distance is therefore easy to visualize in Fig. 6(b): the dashed circle of radius 2, superimposed to the level curves between 0 and 2 of the nonholonomic distance, encloses points at a Euclidean distance smaller than 2 from the origin of the xy plane. Clearly, using the Euclidean distance points “behind” the robot would appear erroneously nearer than they actually are (i.e., considering the system kinematics). To give an idea of the error induced by the use of the Euclidean distance, consider that the nonholonomic distance of the point $(-2, 0)$ on the x -axis is 6.0689, i.e.,

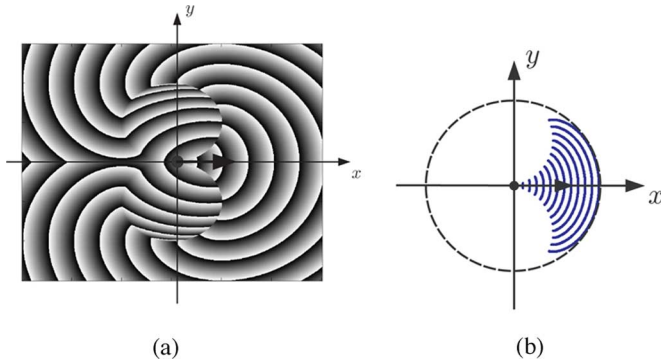


Fig. 6. (a) Isodistance curves on the xy plane for a point robot. (b) (Dashed) Circle of radius 2, enclosing points at a Euclidean distance smaller than 2 from the origin, and the level curves of the nonholonomic distance for values between 0 and 2.

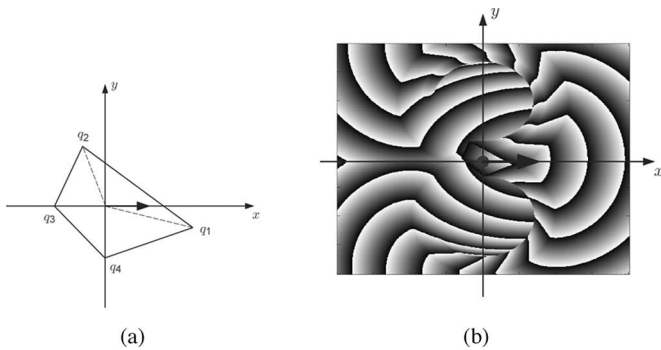


Fig. 7. Level curves for a robot with an irregular shape.

approximately three times further than what indicated by the Euclidean distance.

Fig. 7 illustrates the result for a robot with an irregular shape. Note the deformation of the level curves, w.r.t. the previous case, induced by the shape of the robot.

VII. CONCLUSION

We have presented an analytical method to compute a distance to polygonal obstacles for a Dubins car that takes into account the non-holonomic constraints and the nonsymmetry of the system. By extending the Dubins' work, we computed the shortest paths from a robot configuration to manifolds (the \mathcal{C} -obstacle), rather than to points, in configuration space. The length of these paths represents a distance in configuration space but its computation does not require the explicit representation of the configuration space obstacles. In particular, using optimal control tools complemented by continuity arguments, we reduce the distance computation problem to that of finding the solution of a set of algebraic equations. Due to the generality of the adopted methodology, the presented developments can also be extended to more general planning problems, like, e.g., optimally reaching a given path, or applied to second-order models of vehicles. This last case, however, is expected to be quite complex since even the characterization of the optimal paths in the absence of obstacles for these systems is, in general, incomplete [12]. We are working on the integration of the developed distance function in planning methods based on artificial potentials. Future work includes the investigation of the use of this function in collision checking for sampling-based planners, in the line of [4], and in path smoothing.

REFERENCES

- [1] L. Dubins, "On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents," *Amer. J. Math.*, vol. 79, pp. 497–516, 1957.
- [2] M. Vendittelli, J. Laumond, and C. Nissoux, "Obstacle distance for car-like robots," *IEEE Trans. Robot. Autom.*, vol. 15, no. 4, pp. 678–691, Aug. 1999.
- [3] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2006.
- [4] F. Schwarzer, M. Saha, and J.-C. Latombe, "Adaptive dynamic collision checking for single and multiple articulated robots in complex environments," *IEEE Trans. Robot.*, vol. 21, no. 3, pp. 338–353, Jun. 2005.
- [5] H. J. Sussmann and W. Tang, "Shortest paths for the reeds-sheep car: A worked out example of the use of geometric techniques in nonlinear optimal control," Rutgers Univ., Piscataway, NJ Rep. SYCON-91-10, 1991.
- [6] J. Boissonnat, A. Cerezo, and J. Leblond, "Shortest paths of bounded curvature in the plane," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1992, pp. 2315–2320.
- [7] P. Robuffo-Giordano and M. Vendittelli, "The minimum-time crashing problem for the Dubins' car," presented at the 8th Int. IFAC Symp. Robot Control, Bologna, Italy, Sep. 2006.
- [8] P. Robuffo-Giordano, M. Vendittelli, J.-P. Laumond, and P. Souères, "Non-holonomic distance to polygonal obstacles for a car-like robot of polygonal shape," *IEEE Trans. Robot.*, vol. 22, no. 5, pp. 1040–1047, Oct. 2006.
- [9] X.-N. Bui, P. Souères, J.-D. Boissonnat, and J.-P. Laumond, "Shortest path synthesis for Dubins nonholonomic robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, pp. 2–7, 1994.
- [10] L. Pontryagin, V. Boltyansky, R. Gamkrelidze, and E. Mischenko, *The Mathematical Theory of Optimal Processes*. New York: Wiley, 1962.
- [11] P. Moutarlier, B. Mirtich, and J. Canny, "Shortest paths for a car-like robot to manifolds in configuration space," *Int. J. Robot. Res.*, vol. 15, no. 1, pp. 36–60, Feb. 1996.
- [12] P. Souères and J.-D. Boissonnat, "Optimal trajectories for nonholonomic mobile robots," in *Robot Motion Planning and Control*, J.-P. Laumond, Ed. New York: Springer-Verlag, 1998, pp. 93–169.
- [13] J. D. Boissonnat and X. N. Bui, "Accessibility region for a car that only move forward along optimal paths," Nat. Inst. Res. Comput. Sci. Control (INRIA), Sophia-Antipolis, France, Res. Rep. 2181, 1994.
- [14] E. E. Tyrtshnikov, *A Brief Introduction to Numerical Analysis*. Boston: Birkhauser Verlag, 1997.
- [15] P. Souères, J.-Y. Fourquet, and J.-P. Laumond, "Set of reachable positions for a car," *IEEE Trans. Autom. Control*, vol. 39, no. 8, pp. 1626–1630, Aug. 1994.