[5] L. P. Kaelbling, "Hierarchical learning in stochastic domains: Preliminary results," in *Proc. Int. Conf. Mach. Learning*, 1993, pp. 167–173.
[6] T. Lane and L. P. Kaelbling, "Nearly deterministic abstractions of Markov decision processes," in *Proc. 18th Nat. Conf. Artif. Intell.*, 2002, pp. 260–266.
[7] Y. Takahashi, K. Edazawa, and M. Asada, "Multi-controller fusion in multi-layered reinforcement learning," in *Proc. Int. Conf. Multisensor Fusion Integr. Intell. Syst.*, 2001, pp. 7–12.
[8] E. L. Lawler, J. K. Lenstra, A. H. G. Rinooy Kan, and D. B. Shumoys, *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. New York: Wiley, 1985.
[9] S. Kambhampati and L. S. Davis, "Multiresolution path planning for mobile robots," *IEEE J. Robot. Autom.*, vol. RA-2, no. 3, pp. 135–145, Jun. 1986.
[10] H. Asama, M. Sato, L. Bogoni, H. Kaetu, A. Matsumoto, and I. Endo, "Development of an omni-directional mobile robot with 3DOF decoupling drive mechanism," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1995, pp. 1925–1930.
[11] Y. Fukazawa, C. Trevai, J. Ota, H. Yuasa, T. Arai, and H. Asama, "Region exploration path planning for a mobile robot expressing working environment by grid points," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2003, pp. 2448–2454.

Nonholonomic Distance to Polygonal Obstacles for a Car-Like Robot of Polygonal Shape

Paolo Robuffo Giordano, M. Vendittelli, Jean-Paul Laumond, and Philippe Souères

***Abstract*—This paper shows how to compute the *nonholonomic* distance between a polygonal car-like robot and polygonal obstacles. The solution extends previous work of Reeds and Shepp by finding the shortest path to a manifold (rather than to a point) in configuration space. Based on optimal control theory, the proposed approach yields an analytic solution to the problem.**

***Index Terms*—Car-like robots, nonholonomic distance, optimal control theory, shortest paths.**

I. Introduction

Distance computation plays a crucial role in robot motion planning. Numerous motion-planning algorithms rely on obstacle distance computation, e.g., skeletonization and potential field methods [1]. The distance from a robot configuration to an obstacle is the length of the shortest feasible path bringing one point on the robot boundary in contact with the obstacle. Car-like robots being nonholonomic systems, any path in configuration space is not necessarily feasible. As a consequence, the length of the shortest feasible path induces a special metric, the so-called nonholonomic metric, which is not a Euclidean metric [2].

The search for a shortest path between a polygonal robot and a polygonal obstacle in physical space can be easily reformulated into the configuration space $\mathcal{C}$, i.e., representing the robot as a point and mapping the obstacles in their $\mathcal{C}$-obstacle counterparts. The original problem is then transformed in finding the shortest path to the manifold defining the $\mathcal{C}$-obstacle.

Adopting an optimal control point of view, the proposed approach makes use of transversality conditions on the final state of the robot, which make the problem square everywhere (i.e., same number of unknowns and equations), and provide deeper insight of the solution. Moreover, simple continuity arguments allow restricting the search for the optimal path to a subset of the Reeds and Shepp (RS) families.

Manuscript received July 20, 2005; revised January 5, 2006. This paper was recommended for publication by Associate Editor J. Wen and Editor H. Arai upon evaluation of the reviewers' comments.

P. Robuffo Giordano and M. Vendittelli are with the Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza," 00184 Rome, Italy (e-mail: robuffo@dis.uniroma1.it; venditt@dis.uniroma1.it).

J.-P. Laumond and P. Souères are with LAAS-CNRS, 31077 Toulouse Cedex 4, France (e-mail: jpl@laas.fr; soueres@laas.fr).

Digital Object Identifier 10.1109/TRO.2006.878956

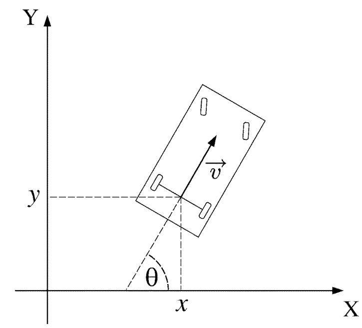

Fig. 1. Car-like robot.

II. Car-Like Robots and the Shortest Path Problem

The configuration of a car-like robot, sketched in Fig. 1, at the instant t is completely defined in $\mathcal{C} = \mathbb{R}^2 \times \mathcal{S}^1$, by the position $(x(t), y(t))$ of the reference point and the heading direction $\theta(t)$ of the robot. The model of the car to which we will refer in this paper is described by the control system

$$\begin{cases} \dot{x}(t) = \cos\theta(t) \cdot u_1(t) \\ \dot{y}(t) = \sin\theta(t) \cdot u_1(t) \\ \dot{\theta}(t) = u_2(t) \end{cases} \tag{1}$$

where $|u_1(t)| = 1$ and $|u_2(t)| \leq 1$ are, respectively, the linear and angular velocities of the car. This model is referred to as the RS car.

A. Shortest Paths Without Obstacles

The study of the shortest paths between any two configurations in the absence of obstacles has been addressed first by Dubins [3], who proved the existence of a sufficient set of optimal paths when $u_1 \equiv 1$ (the robot moves only forward).

Reeds and Shepp [4] extended this result to the forward/backward case ($u_1 = \pm 1$), showing that there always exists a shortest path composed of straight lines and turns of minimal radius with at most two cusps, among a family of 48 different paths; moreover, every path is specified by three parameters representing the length of the basic components (arcs, lines). The problem has been revisited from a control-theory point of view by Boissonnat *et al.* [5] and by Sussman and Tang [6], who reduced the sufficient family to 46 paths.

Souères and Laumond, using these results, computed a synthesis of the shortest paths [7], i.e., a partition of the manifold $\mathbb{R}^2 \times \mathcal{S}^1$ into cells defined by the type of optimal paths (among the 46 candidates) that reach their points. They also showed the metric nature of the length of the shortest path between two configurations [2].

B. Shortest Paths With Obstacles

The problem of computing the shortest paths for a car-like robot in the presence of obstacles is a very difficult one. A shortest path for an

RS car may not exist [8]. The problem for Dubins' car has been proved to be NP-hard [9]. In [10], Fortune and Wilfong propose an algorithm running in exponential time and space to decide if a path exists, but the algorithm does not generate a solution. Mirtich and Canny [11] propose a skeletonization of $\mathcal{C}$ which takes into account the nonholonomic constraint. The skeleton is then used for planning feasible trajectories. The algorithm, however, requires the discretization of the robot's configuration space $\mathcal{C}$ and of the images of the obstacles in $\mathcal{C}$.

Strictly related to our work is [12], where Moutarlier *et al.* explored an analytic tool to compute shortest paths for a polygonal RS car to some manifold in $\mathcal{C}$ by minimizing a distance function of three variables (the three RS parameters) with equality constraints (the equation of the manifold). The proposed approach leads to an analytic solution for "optimally crashing" a car-like robot against obstacles of, in principle, any shape. However, the implementation requires managing the original 46 paths, and every possible combination of subpaths, by recursively minimizing the distance function along the target manifold, its boundaries, and any set of singular and nonregular points [12]. Vendittelli *et al.* [13] developed a geometric method to compute obstacle distance for a pointwise RS car and Dubin's car to polygonal obstacles. The algorithm has complexity $O(n)$ for a polygonal environment with n vertices. In [14], Vendittelli *et al.* extended their previous work by considering a polygonal car-like robot; using an optimal control approach, they reduced the problem to the minimization of a function of one variable, namely, the robot's final orientation.

C. Contribution of the Paper

Our work naturally extends [13] by computing the nonholonomic distance for a polygonal RS car in a polygonal obstacle environment. The approach adopted, however, differs from [12]–[14], since it takes advantage of the combination of tools from optimal control theory and geometric constructions, allowing reducing the optimal paths from 46 to 26, and solving the problem without resorting to numerical optimization techniques. To find the shortest path to a manifold, it is sufficient to solve the problem for each of the 26 RS paths and then to choose the shortest solution. Moreover, transversality conditions from optimal control tools provide a deeper insight into the general structure of the shortest paths.

The paper is organized as follows. In Section III, we briefly summarize the general structure and properties of the RS paths. In Section IV, we attack the main problem addressed in this paper by decomposing it into three subcases handled separately. In Section V, we reduce the original families of RS paths by showing that some paths can never be optimal, and in Section VI, we focus on the smoothness of the defined nonholonomic distance.

III. Shortest Paths for the RS Car

This section summarizes the results presented in [5]–[7]. In accordance with the notation proposed in [6], we will use C and S to denote, respectively, an arc of circle of minimum radius and a straight-line segment, while the symbol $|$ denotes a cusp at the junction of two arcs of a circle. To specify the direction of motion along the path, l and r will denote, respectively, a counterclockwise or clockwise sense of rotation of the direction vector $\overrightarrow{v}$, while s will mean motion along a straight segment. The superscript $+(-)$ will denote forward (backward) motion. 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 $C|CC$ may be specified as $l_a^+ l_b^- r_e^-$, that is, forward left turn of length a, backward left turn of length b, and backward right turn of length e.

Letting $\xi(t) = (x(t), y(t), \theta(t))^T$, $g_1(\xi) = (\cos\theta \;\; \sin\theta \;\; 0)^T$, $g_2(\xi) = (0 \;\; 0 \;\; 1)^T$, and expressing (1) in the form $\dot{\xi} = f(\xi, u) = g_1(\xi)u_1 + g_2(\xi)u_2$, we want to minimize the time to travel from $\xi(t_i)$ to $\xi(t_f)$. The starting configuration $\xi(t_i)$ is assumed, without loss of generality (wlog), to be the origin of the configuration space $(0 \;\; 0 \;\; 0)^T$. For the RS car, this is equivalent to minimizing the length of the path linking $\xi(t_i)$ to $\xi(t_f)$. In this case, the Hamiltonian is expressed by $H = \langle\psi, f\rangle = \psi_1 \cos\theta u_1 + \psi_2 \sin\theta u_1 + \psi_3 u_2 = \phi_1 u_1 + \phi_2 u_2$, where ψ is the costate satisfying the adjoint equation

$$\dot{\psi}(t) = -\frac{\partial H}{\partial \xi}\left(\psi(t), \xi(t), u(t)\right) = -\psi(t)\left[u_1 \frac{\mathrm{d}g_1}{\mathrm{d}\xi} + u_2 \frac{\mathrm{d}g_2}{\mathrm{d}\xi}\right] \quad (2)$$

for almost all t, and $\phi_1 = \langle\psi, g_1\rangle$, $\phi_2 = \langle\psi, g_2\rangle$ represent the switching functions. 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 [15].

Results from [5] and [6] allow restricting the search of optimal paths for the RS car to a sufficient family of paths consisting of concatenations of at most five pieces, that are either arcs of circle of minimum radius (C) or straight-line segments (S). These paths are of two types:

- A paths $C|C|C$, along which $\phi_1 \equiv 0$ and either $u_2 \equiv 1$ or $u_2 \equiv -1$;
- B paths lying between two parallel lines $\mathcal{D}_+$ and $\mathcal{D}_-$, and such that straight-line segments and points of inflection are on the median line $\mathcal{D}_0$ of both lines, the cusps lie on $\mathcal{D}_+$ and $\mathcal{D}_-$, and at each cusp, the heading direction is perpendicular to the common direction of the lines (see Fig. 2).

$\mathcal{D}_+$, $\mathcal{D}_-$, and $\mathcal{D}_0$ are defined as the lines

$$\begin{aligned}
\mathcal{D}_0 &= \left\{(x, y) \in \mathbb{R}^2 | \psi_1 y(t) - \psi_2 x(t) + \psi_3(t_0) = 0\right\} \\
\mathcal{D}_+ &= \left\{(x, y) \in \mathbb{R}^2 | \psi_1 y(t) - \psi_2 x(t) + \psi_3(t_0) + \psi_0 = 0\right\} \\
\mathcal{D}_- &= \left\{(x, y) \in \mathbb{R}^2 | \psi_1 y(t) - \psi_2 x(t) + \psi_3(t_0) - \psi_0 = 0\right\}
\end{aligned}$$

where ψ_0 is a negative constant given by the condition of maximization of the Hamiltonian. For almost all $t \in [t_0, t_1]$, the following equality holds:

$$\begin{aligned}
-\psi_0 &= \langle\psi(t), g_1(\xi(t))\rangle u_1 + \langle\psi(t), g_2(\xi(t))\rangle u_2 \\
&= \max_{(v_1, v_2) \in U} \left(\langle\psi(t), g_1(\xi(t))\rangle v_1 + \langle\psi(t), g_2(\xi(t))\rangle v_2\right). \quad (3)
\end{aligned}$$

From (2) and (3), it can be deduced that:

1) ψ_1 and ψ_2 are constants, and the ratio ψ_2/ψ_1 gives the common direction of $\mathcal{D}_+$, $\mathcal{D}_-$, and $\mathcal{D}_0$;
2) the line $\mathcal{D}_0$ corresponds to the equation

$$\psi_3(t) = \psi_1 y(t) - \psi_2 x(t) + \psi_3(t_0) = 0. \quad (4)$$

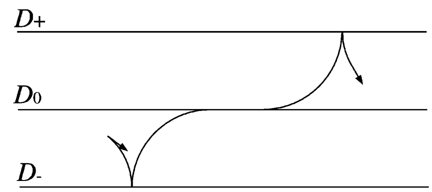

Fig. 2. Example of optimal path Type-B.

The sufficient family of optimal paths can be partitioned into nine path types, as described in (5). The first path type (I) represents Type-A trajectories, whereas the remaining types, (II)–(IX), represent Type-B trajectories

$$\begin{array}{lll} (I) & C_a|C_b|C_e & a \geq 0, b \geq 0, e \geq 0 a+b+e \leq \pi \\ (II) & C_a|C_bC_e & 0 \leq a \leq b, 0 \leq e \leq b 0 \leq b \leq \pi/2 \\ (III) & C_aC_b|C_e & 0 \leq a \leq b, 0 \leq e \leq b 0 \leq b \leq \pi/2 \\ (IV) & C_aC_b|C_bC_e & 0 \leq a \leq b, 0 \leq e \leq b 0 \leq b \leq \pi/2 \\ (V) & C_a|C_bC_b|C_e & 0 \leq a \leq b, 0 \leq e \leq b 0 \leq b \leq \pi/2 \\ (VI) & C_a|C_{\pi/2}S_eC_{\pi/2}|C_b & 0 \leq a \leq \pi/2, 0 \leq b \leq \pi/2 e \geq 0 \\ (VII) & C_a|C_{\pi/2}S_eC_b & 0 \leq a \leq \pi, 0 \leq b \leq \pi/2 e \geq 0 \\ (VIII) & C_aS_eC_{\pi/2}|C_b & 0 \leq a \leq \pi/2, 0 \leq b \leq \pi e \geq 0 \\ (IX) & C_aS_eC_b & 0 \leq a \leq \pi/2, 0 \leq b \leq \pi/2 e \geq 0. \end{array} \tag{5}$$

Some remarks are in order.

1) Every RS path maps smoothly the parameter space into the configuration space [12], i.e., for each path p_i, one can define a function $W_i : \mathbb{R}^3 \longrightarrow \mathcal{C}$ associating the final configuration in $\mathcal{C}$ with the parameters (a, b, e)

$$\begin{pmatrix} x_i \\ y_i \\ \theta_i \end{pmatrix} = W_i(a,b,e) = \begin{pmatrix} X_i(a,b,e) \\ Y_i(a,b,e) \\ \Theta_i(a,b,e) \end{pmatrix} \tag{6}$$

where X_i, Y_i, and Θ_i are smooth.

2) Denoting with $L_p(a, b, e)$ the length of the path p defined by the parameters (a, b, e), the following hold:
— Type-A trajectories: $L_p = |\theta(t_f)|$
— Type-B trajectories: $L_p > |\theta(t_f)|$.

3) Lines $\mathcal{D}_+, \mathcal{D}_-, \mathcal{D}_0$, and transversality conditions are defined only for paths of Type-B.

IV. Distance Function

The aim of this paper is to find the length d of the shortest path bringing any point on the boundary of a RS polygonal robot in contact with any point on the boundary of any polygonal obstacle in physical space. The searched path will link the robot's starting configuration to the configuration in contact with one of the obstacles in the environment; it will, therefore, belong to one of the families of RS-optimal paths (5) linking any couple of robot configurations. For this reason, the search will be restricted to these families. As shown by [2], the length of the RS shortest paths induces a metric on the configuration space, and this metric is equivalent to the sub-Riemannian distance [16] associated with the control system representing the RS car model.

The length d is, therefore, a distance to the closest obstacle in the configuration space and is a function of the robot's current state and of the shapes of the robot and the obstacles.

In this paper, we will assume, wlog, that the robot starting configuration is the origin $(0, 0, 0)^T$. It is useful to partition the distance computation problem into the three subproblems of bringing into contact (see Fig. 3):

(i) one vertex q_i of the robot with one vertex o_j of the obstacle;
(ii) one vertex q_i of the robot with the line v_j supporting one edge o_jo_{j+1} of the obstacle;
(iii) the line w_i supporting one edge q_iq_{i+1} of the robot with one vertex of the obstacle o_j.

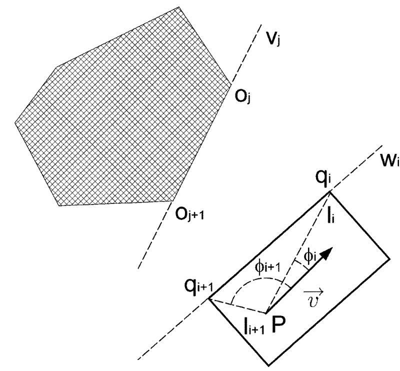

Fig. 3. Partitioning the distance computation.

If one is able to solve these subtasks, it is possible to find the shortest path to an obstacle by iterating the three steps over all the robot/obstacle vertex/edge combinations, and by choosing the minimum from among the obtained path lengths.[1] The three problems (i)–(iii) can be associated with the following three functions:

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

where L^{VV}, L^{VE}, and L^{EV} will be defined in the following sections. With this notation, the distance is defined as

$$d() : \mathbb{R}^8 \rightarrow \mathbb{R}$$
$$d() = \min\left\{\min_{i,j} L^{VV}(q_i, o_j), \min_{i,j} L^{VE}(q_i, v_j), \min_{i,j} L^{EV}(w_i, o_j)\right\}. \tag{7}$$

In the next sections, we will describe the approach adopted to solve each specific subproblem. The set of all 46 RS paths will be denoted by $\{OP\}$, while $p_k \in \{OP\}$, $k = 1, \ldots, 46$ will denote an optimal path of the sufficient family, with the associated RS path parameters (a_k, b_k, e_k) determining its length $L_{p_k}(a_k, b_k, e_k)$.

A. Vertex–Vertex Distance

The solution to problem (i), using the specific path p_k, is provided by the map

$$VV_{p_k}: \quad \mathbb{R}^4 \quad \rightarrow \mathbb{R}^3 \quad (q_i, o_j) \rightarrow (a_k, b_k, e_k).$$

The function $L^{VV}(q_i, o_j)$ is then defined as

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

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

- the solution does not exist, i.e., at least one RS path parameter is complex;

[1] Note that in a polygonal environment, the problem of bringing into contact one edge of the robot with the line supporting one edge of the obstacle is already covered by points (ii) and (iii) [12].

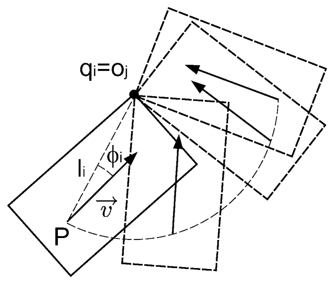

Fig. 4. Projection on the plane (dotted arc) of the contact manifold.

- the solution exists but it is not valid, i.e., at least one RS path parameter is outside its range of validity;
- a valid solution exists.

In the first two cases, $L_{p_k}(VV_{p_k}(q_i, o_j))$ is discarded by setting it equal to ∞.

1) Handling Type-B Paths: Preliminary remark: throughout the following sections, time dependency is omitted when no confusion is possible.

In order to find the solution of $VV_{p_k}(q_i, o_j)$ for each path p_k of Type-B, we will use a property derived from the transversality conditions on the final state. Let (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 $\overrightarrow{v}$ (Fig. 3). The coordinates (q_{i_x}, q_{i_y}) of the robot vertex are

$$\begin{cases} x + l_i\cos(\theta+\phi_i) = q_{i_x} \\ y + l_i\sin(\theta+\phi_i) = q_{i_y}. \end{cases} \tag{9}$$

Denoted by (o_{j_x}, o_{j_y}) the Cartesian coordinates of the target vertex o_j of the obstacle, we define the 1-D contact manifold $C_{VV}^{ij}(\xi)=\{\xi | q_i=o_j\}$, which can be represented by

$$\chi_{VV}^{ij}(\xi) = \begin{pmatrix} x - o_{j_x} + l_i\cos(\theta+\phi_i) \\ y - o_{j_y} + l_i\sin(\theta+\phi_i) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{10}$$

Equation (10) describes a vertical helix centered on o_j (Fig. 4) and will be used for finding the solution path, i.e., for determining the three RS path parameters (a, b, e). An additional constraint is, however, necessary to make the problem "square" (three parameters and three equations), and it will be derived from transversality conditions, as shown below.

Lemma 1: A necessary condition for a path of Type-B to be optimal for problem (i) is that the line $\mathcal{D}_0$ passes through the point o_j.

Proof: Let $\xi_f = (x(t_f), y(t_f), \theta(t_f))$ be the final robot configuration. The constraint $\xi_f \in C_{VV}^{ij}$ on the robot final state, expressed as $\chi_{VV}^{ij}(\xi_f) = 0$, can be used to derive the transversality condition $\psi_f = M^T\zeta$, 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 $\zeta = (\zeta_1\ \zeta_2)^T$. We get the system

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

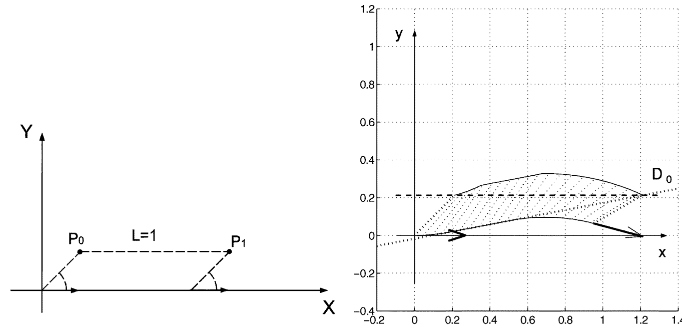

Fig. 5. Straight path (left) versus the shortest path (right) satisfying *Lemma 1* condition.

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

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

Using the definition of $\psi_3(t)$, we can get

$$\begin{aligned} \psi_3(t_0) &= -\,\psi_1\left(y(t_f) + l_i\sin(\theta(t_f)+\phi_i)\right) \\ &\quad + \psi_2\left(x(t_f) + l_i\cos(\theta(t_f)+\phi_i)\right) \\ &= -\,\psi_1 o_{j_y} + \psi_2 o_{j_x}. \end{aligned}$$

Thus, the line $\mathcal{D}_0$ has equation

$$\psi_3(t) = \psi_1\left(y(t) - o_{j_y}\right) - \psi_2\left(x(t) - o_{j_x}\right) = 0 \tag{11}$$

which implies our thesis. □

Recalling from (6) that every path p_k is associated with a smooth map $(x_k, y_k, \theta_k)^T = W_k(a, b, e)$, and denoting by $\overline{Y}_k$ and $\overline{X}_k$ the final positions computed via (6), associated with the subpath of p_k which brings the robot on the line $\mathcal{D}_0$ (where $\psi_3(t) = 0$),[2] we get from (10) and (11) a square system of equations for every VV_{p_k} directly projected into the RS parameter space

$$\begin{cases} \chi_{VV}^{ij}\left(W_k(a,b,e)\right) = 0 \\ \psi_1 \cdot \left(\overline{Y}_k(a,b,e) - o_{j_y}\right) - \psi_2 \cdot \left(\overline{X}_k(a,b,e) - o_{j_x}\right) = 0. \end{cases} \tag{12}$$

The solution of (12) yields the candidate optimal path.

The condition stated in *Lemma 1* deserves some additional considerations. Suppose that we want to bring the point located on the robot at $(l = 0.3, \phi = \pi/4)$ from the home position $P_0(l\cos(\phi), l\sin(\phi))$ to the goal position $P_1(1 + l\cos(\phi), l\sin(\phi))$, i.e., we want to shift it by $L = 1$ to the right [Fig. 5 (left)]. A trivial solution would be to travel along the x axis for a distance exactly equal to L. This kind of path, however, does not satisfy *Lemma 1*, since the line $\mathcal{D}_0$ (the x axis, in this case) does not pass through P_1; hence, there exists a shorter path of some other shape. The shortest path, shown in Fig. 5 (right), is of type $l_a^+ S_e^+ r_b^+$ with $a = 0.188, b = 0.449, e = 0.327$, and total length $\hat{L} = a + b + e = 0.964 < L$.

This strange phenomenon can be explained by noting that although we plan a path for the generic robot point q_i, such a path must minimize a cost associated with the control point P. On straight segments, both points cover the same distance, but this is not true for arcs where P moves on circles of radius 1, and q_i on circles of radius amplified by its distance from P. Hence, the shortest path shall avoid straight lines

[2]For instance, if p_k is $l_a^+ l_b^- r_e^-$, $\mathcal{D}_0$ is reached through the subpath $l_a^+ l_b^-$.

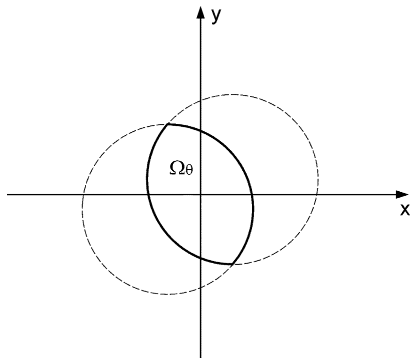

Fig. 6. Region Ω_θ is the intersection of two discs.

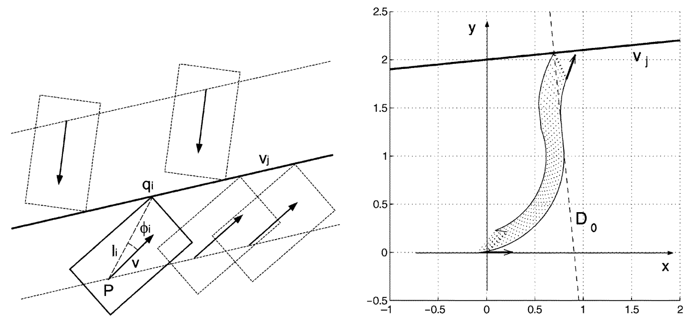

Fig. 7. Vertex–Edge case. Projection on the plane (dotted lines) of the contact manifold for two values of θ (left); an example of shortest path to a line (right).

as much as possible, since arcs can be cheaper in terms of traveled distance of P with respect to q_i.

2) Handling Type-A Paths: The solution of problem (i) when p_k is a trajectory of Type-A cannot rely upon transversality conditions; a direct geometric approach will be adopted instead. The method is the same for all three L^{VV}, L^{VE}, and L^{EV} distances, though in this specific case, it can be applied straightforwardly (the other cases will need some more work). We will focus only on $\theta \in [0, \pi]$, the other case being symmetric; according to [7], define Ω_θ as the region of the plane[3] P_θ associated with the family $C|C|C$. Every Ω_θ is a bounded and closed region centered at the origin composed of the intersection of two discs (see Fig. 6); moreover, the family $\{\Omega_\theta\}$, $\theta \in [0, \pi]$ is a monotonic succession of sets such that:

$$\begin{cases} \Omega_0 = (0,0) \\ \Omega_{\theta_1} \subset \Omega_{\theta_2}, \quad \text{if } \theta_1 < \theta_2. \end{cases}$$

Recalling that for Type-A paths, $L = \theta$, we can consider the regions Ω_θ as level sets of the path length L evaluated for different values of θ. It is clear, then, that tangency between the $C|C|C$ domain and the contact manifold can occur only at the boundaries of regions Ω_θ, where only $C|C$ paths are defined, i.e., one of the RS parameters is always equal to zero. Thus, the equation

$$\chi_{VV}^{ij}\left(W_{p_k}(0, b, e)\right) = 0 \tag{13}$$

is square, and can be solved analytically.

B. Vertex–Edge Distance

In this section, we will show the method proposed to solve problem (ii); some preliminary remarks may be useful to better understand our approach. In this case, the contact manifold being 2-D, two more constraints are needed in order to get a square equation system. As usual, these additional constraints will be derived from transversality conditions on the final robot state. Moreover, since we assume the target edge of the obstacle to be an unbounded line, there could be solutions returning a contact point outside the edge boundaries (o_j, o_{j+1}); in this case, the solution is discarded.

Using the same notation of the previous section, we define the map

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

[3]Planes whose points are reached with constant final orientation θ.

which solves problem (ii) for a given RS path p_k returning the three RS parameters and the contact point on the edge. The main $L^{VE}()$ function can be expressed as

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

with $L_{p_k}(VE_{p_k}(q_i, v_j)) = \infty$ if the contact point on the edge lies outside the edge boundaries, i.e., the minimization in (14) is performed over the set (which can also be empty) of RS paths landing inside the edge. All the remarks stated for (8) hold also in this case.

1) Handling Type-B Paths: Let $y = m_j x + n_j$ be the equation of the target edge v_j; by using (9), the contact manifold between q_i and v_j is defined by $C_{VE}^{ij}(\xi) = \{\xi | q_i \in v_j\}$, and is represented as

$$\chi_{VE}^{ij}(\xi) = y - m_j x - n_j - l_i m_j \cos(\theta + \phi_i) + l_i \sin(\theta + \phi_i) = 0$$

which represents a 2-D surface whose projection on the plane xy for a given θ is a line parallel to v_j [Fig. 7 (left)].

Lemma 2: If a Type-B path is optimal for 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

$$M = (-m_j \;\; 1 \;\; l_i m_j \sin(\theta(t_f) + \phi_i) + l_i \cos(\theta(t_f) + \phi_i)).$$

Thus, we get the system

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

from which we get the following relations:

$$\frac{\psi_2}{\psi_1} = -\frac{1}{m_j} \tag{15}$$

$$\psi_3(t_f) = -\psi_1 l_i \sin\left(\theta(t_f) + \phi_i\right) + \psi_2 l_i \cos\left(\theta(t_f) + \phi_i\right). \tag{16}$$

Point 1 of *Lemma 2* is proved by (15); by using (4), (9), and (16), we can compute the constant $\psi_3(t_0) = -\psi_1 q_{i_y}(t_f) + \psi_2 q_{i_y}(t_f)$ that yields $\psi_3(t) = \psi_1(y(t) - q_{i_y}(t_f))) - \psi_2(x(t) - q_{i_x}(t_f))$, which implies that the point q_i at the end of the path must lie on the line $\mathcal{D}_0$. Thus, combined with the contact condition between $q_i(t_f)$ and the target line v_j, we prove point 2. $\square$

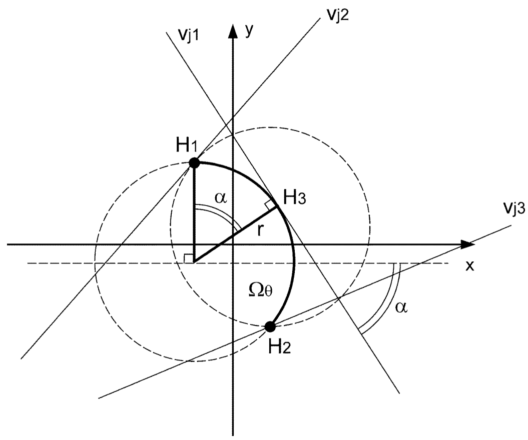

Fig. 8. Geometric construction for handling Type-A paths.

Putting together the contact manifold constraint on the final state and the two transversality conditions, we get again a square system of equations for each path p_k of Type-B.

As an example, in Fig. 7 (right), we show the solution for the line of equation $y = 0.1x + 2$ and the pair $(l_i = 0.3, \phi_i = \pi/4)$; the shortest path is of type $l_a^- l_{\pi/2}^+ s_e^+ r_b^+$, with $a = 0.099$, $b = 0.449$, $e = 0.268$, and total length $L = a + b + e + \pi/2 = 2.386$.

2) Handling Type-A Paths: Although the conditions derived in Section IV-A.2 are still valid for this case, they are no longer sufficient, since

$$\chi_{VE}^{ij}\left(W_{p_k}(0, b, e)\right) = 0 \tag{17}$$

is underspecified (one equation and two parameters). The missing information is recovered with the following geometric reasoning (see Fig. 8): tangency between Ω_θ and C_{VE}^{ij} may occur on points H_1, H_2 (the intersection of the two circles), or on one of the two arcs bounding Ω_θ; in Fig. 8, these three tangency conditions are represented by the three generic lines $v_{j_1}, v_{j_2}, v_{j_3}$. At points H_1 and H_2, the path $C|C|C$ reduces to a single arc C, thus, the only free parameter can be computed using (17). On the upper (lower) arc of circle, the angular coefficient of v_{j_1} matches the angle α between the vertical radius vector through $H_1 (H_2)$ and the radius vector $\overrightarrow{r}$ through the tangency point H_3. On the other hand, α fixes one of the two parameters, i.e., $b = \alpha$ on the upper circle and $e = \alpha$ on the lower circle [7], then (17) can be used to obtain the second parameter.

C. Edge–Vertex Distance

The approach adopted to solve problem (iii) is very similar to the method outlined in the last section. In particular, 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

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

as the map which solves problem (iii) for a specific path p_k, the $L^{EV}()$ function is

$$L^{EV}(w_i, o_j) = \min_{p_k \in \{OP\}} L_{p_k}\left(EV_{p_k}(w_i, o_j)\right) \tag{18}$$

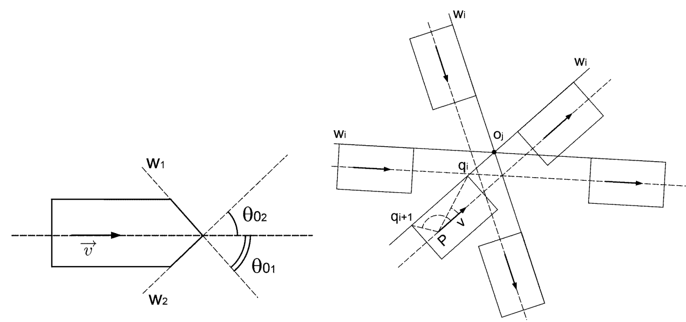

Fig. 9. Edge–Vertex case. Angles θ_{0_i} (left). Projection on the plane (dotted lines) of the contact manifold (right).

with $L_p(L^{EV}(w_i, o_j)) = \infty$ if the contact point lies outside the edge boundaries.

1) Handling Type-B Paths: Let (q_i, q_{i+1}) be two adjacent robot vertices. From (9), the line w_i can be expressed as $y = m_i(\theta)x + n_i(\xi)$, where

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

and θ_{0_i} is the angle made by the edge w_i and the direction vector $\overrightarrow{v}$. Fig. 9 (left) shows, for instance, the angles $\theta_{0_1}, \theta_{0_2}$ relative to two edges w_1, w_2 for a generic polygonal robot.

The coordinates of the target point o_j being (o_x, o_y), the manifold representing $o_j \in w_i$ is $C_{EV}^{ij}(\xi) = \{\xi | o_j \in w_i\}$ and is expressed by

$$\begin{aligned} \chi_{EV}^{ij}(\xi) &= o_y - m_i(\theta) o_x - n_i(\xi) \\ &= (-o_x + x + l_i \cos(\theta + \phi_i)) \sin(\theta + \theta_{0_i}) \\ &\quad + (o_y - y - l_i \sin(\theta + \phi_i)) \cos(\theta + \theta_{0_i}) = 0 \end{aligned}$$

which describes, as θ varies, a 2-D surface whose projection on the xy plane is made of straight lines rotating at a fixed distance from o_j [Fig. 9 (right)].

Lemma 3: If a Type-B path is optimal for problem (iii), then:

1) line $\mathcal{D}_0$ must be perpendicular to the edge w_i at the end of the path;
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

$$\begin{aligned} M &= \frac{\partial \chi_{EV}^{ij}(\xi_f)}{\partial \xi_f} \\ &= [\sin(\theta(t_f) + \theta_{0_i}) - \cos(\theta(t_f) + \theta_{0_i}) \; (-o_y + y(t_f)) \\ &\qquad \sin(\theta(t_f) + \theta_{0_i}) + (-o_x + x(t_f)) \cos(\theta(t_f) + \theta_{0_i})]. \end{aligned}$$

Hence, we get

$$\begin{cases} \psi_1 = \sin(\theta(t_f) + \theta_{0_i}) \zeta \\ \psi_2 = -\cos(\theta(t_f) + \theta_{0_i}) \zeta \\ \psi_3(t_f) = ((-o_y + y(t_f)) \sin(\theta(t_f) + \theta_{0_i}) \\ \quad + (-o_x + x(t_f)) \cos(\theta(t_f) + \theta_{0_i})) \zeta \end{cases}$$

which yields

$$\frac{\psi_2}{\psi_1} = -\frac{1}{\tan\left(\theta(t_f)+\theta_{0_i}\right)} = -\frac{1}{m_i\left(\theta(t_f)\right)} \tag{19}$$

$$\psi_3(t_f) = \psi_1\left(y(t_f)-o_y\right) - \psi_2\left(x(t_f)-o_x\right). \tag{20}$$

Equation (19) proves point 1 of *Lemma 1*; from (20), we have $\psi_3(t_0) = -\psi_1 o_y + \psi_2 o_x$, which yields $\psi_3(t) = \psi_1(y(t)-o_y) - \psi_2(x(t)-o_x)$. This relation, together with the contact constraint between o_j and the line w_i, proves point 2. □

2) Handling Type-A Paths: The solution for Type-A paths follows the same approach adopted in the previous section for the L^{VE} case. Referring to Fig. 8, if C^{ij}_{EV} is tangent in H_1 or H_2, the solution degenerates to a single arc C (lines v_{j_2}, v_{j_3}). If C^{ij}_{EV} is tangent to one of the two arcs of circle (e.g., line v_{j_1}), one of the two parameters (e,b) can be computed from the angular coefficient of v_{j_1}. For instance, if tangency occurs on the upper circle, then $e=0, b=-(\theta+\theta_0), a=\theta_0$, $\tan(\theta+\theta_0)$ being the angular coefficient of v_{j_1}.

V. Reduction of the Sufficient Family

In this section, we will reduce the set of optimal RS paths (5) by showing that four of these families are never optimal, and can be excluded *a priori* from the computations. The proof takes advantage of the continuity of the RS parameters (a,b,e) with respect to the parameter l_i of the robot. Some preliminary steps are required.

Lemma 4: For any robot/obstacle vertex/edge q_i/o_j, w_i/v_j, as introduced in the beginning of Section IV, we have the following.

1) If $L^* = L^{VE}(q_i, v_j)$ is the optimal solution of problem (ii) and P^*_l is 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 is the associated contact point, then $L^* = L^{VV}(P^*_l, o_j)$.

Proof: The proof is given for point 1; point 2 can be proved using the same approach. If $L^* \neq L^{VV}(P^*_l, q_i)$, then there exists a shorter path $\hat{p} \neq p^*$ bringing the vertex q_i to P^*_l. But then $\hat{p}$ would be also the optimal solution for $L^{VE}(v_j, q_i)$, thus contradicting the hypothesis. □

Lemma 4 allows focusing only on the L^{VV} distance, since the other two functions can always be reduced to this case. The solution of every instance $L^{VV}(q_i, o_j)$ is found by solving (12) for Type-B paths or (13) for Type-A paths. We have also the following.

Lemma 5: ψ_1 and ψ_2 are smooth functions of the RS path parameters (a,b,e).

Proof: It is convenient to split Type-B paths into two sets:

S_1: paths without a straight segment, i.e., families (II)–(V);

S_2: paths with a straight segment, i.e., families (VI)–(IX).

For S_1 paths, the direction of $\mathcal{D}_0$ is perpendicular to the robot orientation θ_c at a cusp, hence, $\psi_2/\psi_1 = -1/\tan(\theta_c)$. For S_2 paths, the direction of $\mathcal{D}_0$ coincides with the constant robot orientation θ_s on the straight segment, hence $\psi_2/\psi_1 = \tan(\theta_s)$. Both θ_s and θ_c are the orientations achieved after at most two basic RS path components (two arcs of circle), and can be expressed as $\kappa_1 a + \kappa_2 b$, with $\kappa_1, \kappa_2 = \{0, 1\}$ depending on the RS path. □

Hence, (12) and (13) are smooth with respect to the unknowns (a,b,e) via the W_i maps, and with respect to (q_i, o_j) by construction of the contact manifold C^{ij}_{VV} and the transversality condition (11). The RS parameters appear either algebraically (when relative to straight segments) or inside trigonometric functions (when relative to arcs of circle); a classic change of variable $\alpha = \tan(\beta/2)$ applied to each trigonometric function can transform these equations into a fully algebraic set whose solution is smooth with respect to the coefficients, which, in turn, are smooth functions of the l_i.

From [17], we know that when $l_i = 0$ (a pointwise robot), the optimal paths reduce to the following three families, where each path is specified by only two parameters:

$$\begin{aligned} \mathcal{F}_1: &\quad C_a|C_b \\ \mathcal{F}_2: &\quad C_a|C_{\pi/2}S_e \\ \mathcal{F}_3: &\quad C_aS_e. \end{aligned} \tag{21}$$

Thus, the continuity of the three RS parameters with respect to l_i implies that the nine families described in (5) must converge continuously towards (21), i.e., one parameter must go to zero, and the resulting path must be compatible with (21). We can then state the following.

Lemma 6: Path families (IV)–(VI) and (VIII) cannot be optimal for problems (i)–(iii).

Proof:

- Families (IV) and (V): b cannot vanish, since the inequalities $a \le b, e \le b$ would yield a zero-length path. If a or e vanish, the resulting path does not belong to (5), e.g., $C_b|C_bC_e \notin \{\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3\}$.
- Families (VI) and (VIII): when one of the three parameters vanish, the resulting path never belongs to (5), e.g., $S_eC_{\pi/2}|C_b \notin \{\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3\}$. □

Thus, the search for the optimal path can be restricted to the 26 paths described by the families (I)–(III), (VII), and (IX). Formally, by defining $\{OP^*\}$, the set of these 26 optimal paths, the functions (8), (14), and (18) can be reformulated as

$$\begin{aligned} L^{VV}(q_i, o_j) &= \min_{p_k \in \{OP^*\}} L_{p_k}\left(VV_{p_k}(q_i, o_j)\right) \\ L^{VE}(q_i, v_j) &= \min_{p_k \in \{OP^*\}} L_{p_k}\left(VE_{p_k}(q_i, v_j)\right) \\ L^{EV}(o_j, w_i) &= \min_{p_k \in \{OP^*\}} L_{p_k}\left(EV_{p_k}(o_j, w_i)\right). \end{aligned}$$

VI. Smoothness

Here, we study the smoothness of the distance function (7) with respect to its arguments, i.e., the actual robot state and its shape. Intuitively, we would like to have a smooth measure of how far the obstacles are while a robot of any shape moves through the environment. Actually, such desired feature can be guaranteed almost everywhere for (7).

Lemma 7: The distance function (7) is piecewise smooth, and the not-derivable points are located at the switches between the L^{VV}, L^{VE}, and L^{EV} functions.

Proof: The distances L^{VV}, L^{VE}, L^{EV} being smooth on their own, as shown in Section VII, a problem may arise when one has to switch among them. Such switches can occur at those configurations where:

A distances between two distinct robot vertices and an edge coincide, i.e., there exist two different but equivalent shortest paths linking the two vertices to the edge;

B the contact point P_l, computed with the L^{VE} or L^{EV} function, coincides with a vertex of the obstacle or of the robot, respectively.

Proof for case 1) is trivial; for case 2), *Lemma 4* guarantees that for such configurations, the three distances match exactly. Thus, when the L^{VV}, L^{VE}, and L^{EV} functions are taken together, as in (7), they actually define a piecewise smooth function, since the switching points are continuously connected. □

As an example, we computed the distance from a bounded segment P_1P_2 of a rectangular robot against its initial orientation

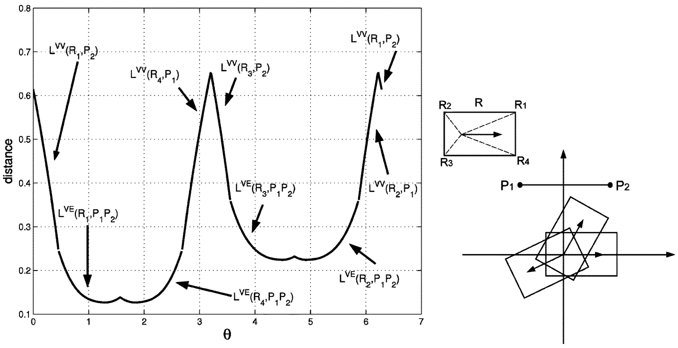

Fig. 10. Distance (7) versus initial orientation (left) of robot R from the segment P_1P_2 (right).

[Fig. 10 (right)]; as expected, while the robot turns, the distance is smooth almost everywhere [Fig. 10 (left)]. Values for the robot and the segment are

$$\begin{cases} R_1 : l_1 = 0.28 & \phi_1 = \pi/6 \\ R_2 : l_2 = 0.2 & \phi_2 = 3\pi/4 \\ R_3 : l_3 = 0.2 & \phi_3 = 5\pi/4 \\ R_4 : l_4 = 0.28 & \phi_4 = 11\pi/6 \\ P_1 : (-0.3, 0.4) & \\ P_2 : (0.3, 0.4). & \end{cases}$$

VII. Conclusion

In this paper, we presented an analytical method to compute the nonholonomic distance of a polygonal RS car from a polygonal obstacle. By extending the original RS work, we computed the shortest distance to a manifold (the C-obstacle), rather than to a point. In particular, we were able to reduce this problem to that of finding the solution of a set of algebraic equations by using geometric and optimal control arguments, which also provided deeper understanding of the underlying structure of the shortest paths. Moreover, the distance $d(\xi)$ being a piecewise smooth function of the robot state ξ, it is easy to compute analytically the gradient $\overrightarrow{\nabla}_\xi d(\xi)$ almost everywhere; thus, it is possible to build an artificial potential field with $d(\xi)$.

The computation of a candidate optimal path is performed in constant time. For a robot and an environment with m and n vertices, respectively, the complexity is $O(3 * m * n * 26)$, where 3 accounts for the three subproblems that must be solved to compute the distance, and 26 accounts for the number of candidate paths whose length will determine the distance value.

References

[1] J. Latombe, *Robot Motion Planning*. Boston, MA: Kluwer, 1991.

[2] J.-P. Laumond and P. Souères, "Metric induced by the shortest paths for a car-like mobile robot," in *Proc. Int. Conf. Intell. Robots Syst.*, 1993, pp. 1299–1303.

[3] 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.

[4] J. Reeds and R. Shepp, "Optimal paths for a car that goes both forward and backwards," *Pacific J. Math.*, vol. 2, p. 145, 1990.

[5] J. Boissonnat, A. Cerezo, and J. Leblond, "Shortest paths of bounded curvature in the plane," in *Proc. IEEE Int. Conf. Robot. Autom.*, Nice, France, May 1992, pp. 2315–2320.

[6] H. J. Sussmann and W. Tang, Shortest paths for the Reeds–Shepp car: A worked-out example of the use of geometric techniques in nonlinear optimal control, Rutgers Univ., New Brunswick, NJ, Rep. SYCON-91-10, 1991.

[7] P. Souères and J.-P. Laumond, "Shortest path synthesis for a car-like robot," *IEEE Trans. Autom. Control*, vol. 41, no. 5, pp. 672–688, May 1996.

[8] G. Desaulniers, "On shortest paths for a car-like robot maneuvering around obstacles," *Robot. Auton. Syst.*, vol. 17, pp. 139–148, 1996.

[9] J. Reif and H. Wang, "The complexity of the two dimensional curvature-constrained shortest-path problem," in *Robotics: The Algorithmic Perspective*, P. K. Agarwal, L. E. Kavraki, and M. T. Mason, Eds. Natick, MA: A.K. Peters, 1998.

[10] S. Fortune and G. Wilfong, "Planning constrained motions," in *Proc. ACM STOCS*, 1988, pp. 445–459.

[11] B. Mirtich and J. Canny, "Using skeletons for nonholonomic path planning among obstacles," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1992, vol. 3, pp. 2533–2540.

[12] 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.

[13] M. Vendittelli, J.-P. Laumond, and C. Nissoux, "Obstacle distance for car-like robots," *IEEE Trans. Robot. Autom.*, vol. 15, no. 4, pp. 678–691, Aug. 1999.

[14] M. Vendittelli, J.-P. Laumond, and P. Souères, "Shortest paths to obstacles for a polygon car-like robot," in *Proc. Conf. Decision Control*, 1999, pp. 17–22.

[15] L. Pontryagin, V. Boltyansky, R. Gamkrelidze, and E. Mischenko, *The Mathematical Theory of Optimal Processes*. New York: Wiley, 1962.

[16] A. Bellaïche, "The tangent space in sub-Riemannian geometry," in *Sub-Riemannian Geometry*, A. Bellaïche and J.-J. Risler, Eds. Cambridge, MA: Birkhäuser, 1996, pp. 1–78.

[17] 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.

Metric-Based Iterative Closest Point Scan Matching for Sensor Displacement Estimation

Javier Minguez, Luis Montesano, and Florent Lamiraux

***Abstract*—This paper addresses the scan matching problem for mobile robot displacement estimation. The contribution is a new metric distance and all the tools necessary to be used within the *iterative closest point* framework. The metric distance is defined in the configuration space of the sensor, and takes into account both translation and rotation error of the sensor. The new scan matching technique ameliorates previous methods in terms of robustness, precision, convergence, and computational load. Furthermore, it has been extensively tested to validate and compare this technique with existing methods.**

***Index Terms*—Mobile robots, scan matching, sensor displacement estimation.**

I. Introduction

A key issue in autonomous mobile robots is to keep track of the vehicle position. One strategy is to estimate the robot displacement using

Manuscript received July 14, 2005; revised February 2, 2006. This paper was recommended for publication by Associate Editor D. Fox and Editor L. Parker upon evaluation of the reviewers' comments. This work was supported in part by MCYT DPI2003-7986, in part by DGA2004T04, and in part by the Caja de Ahorros de la Inmaculada de Aragóon. This paper was presented in part at the IEEE International Conference on Robotics and Automation, Barcelona, Spain, April 2005.

J. Minguez and L. Montesano are with the Departamento de Informática e Ingeniería de Sistemas, Universidad de Zaragoza, 50015 Zaragoza, Spain (e-mail: jminguez@unizar.es; montesano@unizar.es).

F. Lamiraux is with LAAS-CNRS, 31077 Toulouse, France (e-mail: florent@laas.fr).

Digital Object Identifier 10.1109/TRO.2006.878961