

Geometric Path Planning without Maneuvers for Non-Holonomic Parallel Orienting Robots

Patrick Grosch¹ and Federico Thomas¹

Abstract—Current geometric path planners for non-holonomic parallel orienting robots generate maneuvers consisting of a sequence of moves connected by zero-velocity points. The need for these maneuvers restrains the use of this kind of parallel robots to few applications. Based on a rather old result on linear time-varying systems, this paper shows that there are infinitely differentiable paths connecting two arbitrary points in $SO(3)$ such that the instantaneous axis of rotation along the path rest on a fixed plane. This theoretical result leads to a practical path planner for non-holonomic parallel orienting robots that generates single-move maneuvers. To present this result, we start with a path planner based on three-move maneuvers, and then we proceed by progressively reducing the number of moves to one, thus providing a unified treatment with respect to previous geometric path planners.

Index Terms—Motion and Path Planning, Parallel Robots, Nonholonomic Mechanisms and Systems, Nonholonomic Motion Planning.

I. INTRODUCTION

THIS paper addresses the problem of building a sequence of open-loop input commands that steers a parallel non-holonomic orienting robot to an arbitrary final orientation. Fig. 1 shows a simple implementation of such a robot (for a detailed account on the implementation of a more sophisticated version of this kind of robot see [1]). It consists of a passive non-holonomic spherical joint and two actuated prismatic joints connected to a fixed base and a moving platform through spherical joints. It can be thought as derived from a fully parallel spherical robot in which the motion of the central spherical joint is constrained by putting in contact with it a disk that freely rolls without slipping, and in which one actuated leg is also removed [2]. This robot belongs to a family of non-holonomic parallel spherical robots studied and developed by different authors (see, for example, [3], [4], [5], [6], [7], [8], [9]).

The interest of the studied non-holonomic robot is that it is able to attain an arbitrary orientation using only two actuators and hence its interest as a spherical wrist [1]. However, this reduction of actuation calls for sophisticated algorithms of path planning and control.

Manuscript received: August, 31, 2015; Revised November, 25, 2015; Accepted January, 28, 2016.

This paper was recommended for publication by Editor Antonio Bicchi upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the Spanish Ministry of Economy and Competitiveness through project DPI2014-57220-C2-2-P.

¹Authors are with the Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Llorens Artigas 4-6, 08028 Barcelona, Spain {pgrosch, fthomas}@iri.upc.edu

Digital Object Identifier (DOI): see top of this page.



Fig. 1. A simple implementation of the studied non-holonomic parallel orienting robot (see [6] for details).

Since the studied non-holonomic robot has two inputs (its leg lengths) and three generalized coordinates (its orientation parameters), its kinematics can be formulated in chained form and, as a consequence, its path planning problem can be solved using well-established procedures (see [12] and the references therein). Using one of these analytical methods, the motion planning problem for the studied robot is solved in [7] using control functions given by truncated trigonometric series. The algorithm stops when the Euclidean norm of the planning error becomes less than a given amount.

The tackled problem here can also be seen as a variation the rolling sphere problem, an example of an optimal control problem with nonholonomic constraints which have studied by a variety of authors (see [10], [11] and the references therein).

The use of the mentioned analytical procedures requires a good understanding of sophisticated methods in non-linear control. As an alternative, geometric path planners have been proposed, for example, in [6], [8]. The main advantage of this kind of path planners is that they are based on elementary spherical kinematics arguments. However, they generate ma-

neuers, that is, sequences of moves that guide the moving platform to the desired orientation. This paper is aimed at presenting a geometric path planner able to steer the robot to the desired orientation in a single move.

This paper is organized as follows. In the next section, to make the presentation self-contained, we briefly review the kinematics of the studied non-holonomic orienting robot. Then, in Section III, we start with a path planner that generates three-move maneuvers consisting of three consecutive rotations around two orthogonal axes on the plane defined by the non-holonomic constraint. In Section IV, a two-move maneuver is derived consisting of two consecutive rotations around two non-necessarily orthogonal axes of this plane. Finally, in Section V, it is shown how a single rotation around a variable axis that rests on the plane solves the path planning problem. A detailed example is presented in Section VI. We conclude in Section VII with a summary of the main point and some prospects for future research.

II. KINEMATIC MODEL

The path planning problem for the studied non-holonomic parallel robot can be decomposed into the following two steps: (1) first solve the planning problem considering only the sphere and the disk that constrains its motion, and then (2) obtain the required motion for the prismatic joints in the legs using the inverse kinematics of the robot. Separating both problems, instead of considering both at once, leads to an important simplification. However, this does not go without a price as the first step does not take into account the robot singularities. This is not an important drawback if a workspace free from singularities is defined beforehand.

Next, we briefly summarize the mathematics behind both steps (a detailed derivation can be found in [6]).

A. The sphere's motion

Once the disk is put in contact with the sphere, the sphere can still rotate about any axis orthogonal to an axis defined by the contact with the disk. Without loss of generality, we can assume that this constrained axis, say $\hat{\mathbf{r}}$, coincides with the z -axis. Therefore, the sphere can undergo any rotation of the form

$$\mathbf{R}_{\mathbf{n}}(\omega) = \mathbf{R}_z(\theta)\mathbf{R}_x(\omega)\mathbf{R}_z(-\theta), \quad (1)$$

where $\mathbf{n} = [\cos \theta, \sin \theta, 0]^T$. In particular, it can be rotated about the x -axis ($\theta = 0$) or the y -axis ($\theta = \pi/2$).

From the instantaneous kinematics point of view, the sphere can undergo any angular velocity of the form $\omega = (w_x(t), w_y(t), 0)$. Therefore, the sphere's orientation must satisfy the differential equation

$$\dot{\mathbf{R}}(t) = \mathbf{W}(t)\mathbf{R}(t), \quad (2)$$

where

$$\mathbf{W}(t) = \begin{bmatrix} 0 & 0 & w_y(t) \\ 0 & 0 & -w_x(t) \\ -w_y(t) & w_x(t) & 0 \end{bmatrix}. \quad (3)$$

Finding $\mathbf{R}(t)$ means integrating (2). Although this equation describes a non-holonomic dynamic system and hence it is

not integrable, it does not mean that, for particular angular velocities, it can be integrated. For example, for the trivial case in which the angular velocities are constant, \mathbf{W} becomes constant and then, using Rodrigues' formula, we have that

$$\mathbf{R}(t) = e^{\mathbf{W}t} = \mathbf{I} + \frac{\sin(\theta)}{\theta}\mathbf{W} + \frac{1 - \cos(\theta)}{\theta^2}\mathbf{W}^2, \quad (4)$$

where $\theta = t\sqrt{w_x^2 + w_y^2}$.

B. The legs' motion

The lengths of the prismatic actuators for a given orientation of the sphere, \mathbf{R} , will be given by

$$l_i = \|\mathbf{b}_i - \mathbf{a}_i\| = \|\mathbf{R}\mathbf{b}_i^0 - \mathbf{a}_i\|, \quad i = 1, 2, \quad (5)$$

where \mathbf{a}_i stands for the location of the center of the i spherical joint attached to the base, \mathbf{b}_i for the location of the corresponding spherical joint center attached to the moving platform, and \mathbf{b}_i^0 for the location of this latter center at the reference orientation of the moving platform.

The velocities of the prismatic joints, as a function of the angular velocities, ω_x and ω_y , can be expressed as

$$\begin{bmatrix} \dot{l}_1/l_1 \\ \dot{l}_2/l_2 \\ 0 \end{bmatrix} = \mathbf{F} \begin{bmatrix} \omega_x \\ \omega_y \\ 0 \end{bmatrix} \quad (6)$$

where

$$\mathbf{F} = \begin{bmatrix} (\mathbf{a}_1 \times \mathbf{R}\mathbf{b}_1^0)^T \\ (\mathbf{a}_2 \times \mathbf{R}\mathbf{b}_2^0)^T \end{bmatrix} \quad (7)$$

Then, parallel singularities arise when \mathbf{F} is rank-deficient, and serial singularities when \mathbf{a}_i and \mathbf{b}_i are parallel, where $i = 1, 2$ (see [6] and [7] for details).

III. THREE-MOVE MANEUVER

Given the initial and final poses represented by the rotation matrices \mathbf{R}_I and \mathbf{R}_F , respectively, the goal is to find three rotations, $\mathbf{R}_x(\omega_1)$, $\mathbf{R}_y(\omega_2)$, and $\mathbf{R}_x(\omega_3)$, such that

$$\mathbf{R}_F = \mathbf{R}_x(\omega_3)\mathbf{R}_y(\omega_2)\mathbf{R}_x(\omega_1)\mathbf{R}_I, \quad (8)$$

That is,

$$\mathbf{R}_x(\omega_3)\mathbf{R}_y(\omega_2)\mathbf{R}_x(\omega_1) = \mathbf{R}_F\mathbf{R}_I^T = \mathbf{R}_T.$$

Therefore, $\{\omega_1, \omega_2, \omega_3\}$ is a valid set of YX Euler angles representing \mathbf{R}_T . As a consequence, if $\omega_2 \in (0, \pi)$, then

$$\omega_1 = \text{atan2}(r_{21}, -r_{31}), \quad (9)$$

$$\omega_2 = \arccos(r_{11}), \quad (10)$$

$$\omega_3 = \text{atan2}(r_{12}, r_{13}), \quad (11)$$

where r_{ij} stands for the (i, j) entry of \mathbf{R}_F (see, for example, [14] for the derivation of the above formulas).

The robot orientation as a function of time can be expressed as

$$\mathbf{R}(t) = \begin{cases} \mathbf{R}_x\left(\frac{t}{\Delta t}\omega_1\right)\mathbf{R}_I, & 0 \leq t < \Delta t \\ \mathbf{R}_y\left(\frac{t-\Delta t}{\Delta t}\omega_2\right)\mathbf{R}_x(\omega_1)\mathbf{R}_I, & \Delta t \leq t < 2\Delta t \\ \mathbf{R}_x\left(\frac{t-2\Delta t}{\Delta t}\omega_3\right)\mathbf{R}_y(\omega_2)\mathbf{R}_x(\omega_1)\mathbf{R}_I, & 2\Delta t \leq t \leq 3\Delta t \end{cases} \quad (12)$$

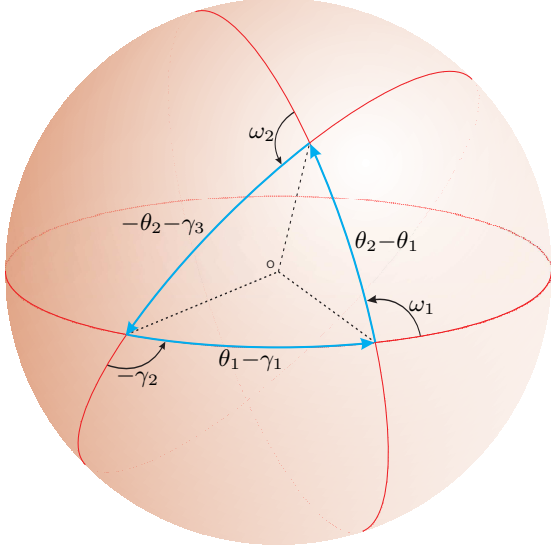


Fig. 2. Spherical triangle whose closure equation is given by (15).

This motion essentially corresponds to that generated by the path planner presented in [6]. The above derivation is performed at a much more elementary level.

IV. TWO-MOVE MANEUVER

Given the initial and final poses, \mathbf{R}_I and \mathbf{R}_F , the goal is to find the angles θ_1 , θ_2 , ω_1 and ω_2 such that

$$\begin{aligned} \mathbf{R}_{n_1}(\omega_1)\mathbf{R}_{n_2}(\omega_2) &= \mathbf{R}_z(\theta_1)\mathbf{R}_x(\omega_1)\mathbf{R}_z(-\theta_1) \\ &\quad \mathbf{R}_z(\theta_2)\mathbf{R}_x(\omega_2)\mathbf{R}_z(-\theta_2) \\ &= \mathbf{R}_F\mathbf{R}_I^T = \mathbf{R}_T. \end{aligned} \quad (13)$$

It is always possible to express \mathbf{R}_T as follows

$$\mathbf{R}_T = \mathbf{R}_z(\gamma_1)\mathbf{R}_x(\gamma_2)\mathbf{R}_z(\gamma_3), \quad (14)$$

where $\{\gamma_1, \gamma_2, \gamma_3\}$ is any set of valid ZZX Euler angles representing \mathbf{R}_T (see [14] for explicit formulas). Then, substituting (14) in (13) and rearranging terms, we obtain

$$\begin{aligned} \mathbf{R}_z(\theta_1 - \gamma_1)\mathbf{R}_x(\omega_1)\mathbf{R}_z(\theta_2 - \theta_1)\mathbf{R}_x(\omega_2) \\ \mathbf{R}_z(-\theta_2 - \gamma_3)\mathbf{R}_x(-\gamma_2) = \mathbf{I}. \end{aligned} \quad (15)$$

This equation can be seen as the closure equation of the spherical triangle shown in Fig. 2 [15]. Then, the analogues of the law of cosines for the angles ω_1 , ω_2 and $-\gamma_2$ of this spherical triangle allow us to write

$$\begin{aligned} \cos(\theta_2 - \theta_1) \cos(\theta_1 - \gamma_1) - \cos(-\theta_2 - \gamma_3) \\ + \sin(\theta_2 - \theta_1) \sin(\theta_1 - \gamma_1) \cos(\pi - \omega_1) = 0, \end{aligned} \quad (16)$$

$$\begin{aligned} \cos(\theta_2 - \theta_1) \cos(-\theta_2 - \gamma_3) - \cos(\theta_1 - \gamma_1) \\ - \sin(\theta_2 - \theta_1) \sin(-\theta_2 - \gamma_3) \cos(\pi - \omega_2) = 0, \end{aligned} \quad (17)$$

$$\begin{aligned} \cos(\theta_1 - \gamma_1) \cos(-\theta_2 - \gamma_3) - \cos(\theta_2 - \theta_1) \\ + \sin(\theta_1 - \gamma_1) \sin(-\theta_2 - \gamma_3) \cos(\pi + \gamma_2) = 0, \end{aligned} \quad (18)$$

Equations (16) and (17) allow us to express ω_1 and ω_2 as a function of θ_1 and θ_2 , respectively, as follows

$$\omega_1 = \arccos \left[\frac{\cos(\theta_2 - \theta_1) \cos(\theta_1 - \gamma_1) - \cos(\theta_2 + \gamma_3)}{\sin(\theta_2 - \theta_1) \sin(\theta_1 - \gamma_1)} \right], \quad (19)$$

$$\omega_2 = \arccos \left[\frac{\cos(\theta_2 - \theta_1) \cos(\theta_2 + \gamma_3) - \cos(\theta_1 - \gamma_1)}{\sin(\theta_1 - \theta_2) \sin(\theta_2 + \gamma_3)} \right], \quad (20)$$

and equation (18), to express θ_1 as a function of θ_2 as

$$\begin{aligned} \theta_1 = \arctan \\ \left[\frac{\cos \gamma_1 \cos \gamma_2 \sin(\theta_2 + \gamma_3) + \sin \gamma_1 \cos(\theta_2 + \gamma_3) - \sin \theta_2}{\sin \gamma_1 \cos \gamma_2 \sin(\theta_2 + \gamma_3) - \cos \gamma_1 \cos(\theta_2 + \gamma_3) + \cos \theta_2} \right]. \end{aligned} \quad (21)$$

Then, the robot orientation as a function of time is given by

$$\mathbf{R}(t) = \begin{cases} \mathbf{R}_{n_1} \left(\frac{t}{\Delta t} \omega_1 \right) \mathbf{R}_I, & 0 \leq t < \Delta t \\ \mathbf{R}_{n_2} \left(\frac{t - \Delta t}{\Delta t} \omega_2 \right) \mathbf{R}_{n_1}(\omega_1) \mathbf{R}_I, & \Delta t \leq t \leq 2\Delta t \end{cases} \quad (22)$$

The motion generated by this two-move maneuver depends on a parameter: θ_2 . This permits to optimize the maneuver using different criteria. An obvious option is to choose the value of θ_2 that minimizes $\omega_1^2 + \omega_2^2$, *i.e.*, the “total motion” of the sphere. This is exemplified in Section VI.

V. SINGLE-MOVE MANEUVER

It was proved in [16] that, if $\mathbf{W}(t)$ in (2) satisfies the differential equation

$$\dot{\mathbf{W}}(t) = \mathbf{N}_1 \mathbf{W}(t) - \mathbf{W}(t) \mathbf{N}_1, \quad (23)$$

then the solution to (2) can be expressed as

$$\mathbf{R}(t) = \exp(\mathbf{N}_1 t) \exp(\mathbf{N}_2 t) \mathbf{R}(0) \quad (24)$$

where $\mathbf{N}_2 = \mathbf{W}(0) - \mathbf{N}_1$.

The substitution of the ansatz

$$\mathbf{N}_1 = \begin{bmatrix} 0 & -\omega & 0 \\ \omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (25)$$

in (23) yields the following system of equations

$$\begin{aligned} \dot{w}_y &= \omega w_x \\ \dot{w}_x &= -\omega w_y \end{aligned} \quad (26)$$

whose integration yields

$$\begin{aligned} w_x &= A \cos(\omega t + \omega_0), \\ w_y &= A \sin(\omega t + \omega_0). \end{aligned} \quad (27)$$

Then,

$$\mathbf{W}(0) = \begin{bmatrix} 0 & 0 & A \sin \omega_0 \\ 0 & 0 & -A \cos \omega_0 \\ -A \sin \omega_0 & A \cos \omega_0 & 0 \end{bmatrix}, \quad (28)$$

and, as a consequence,

$$\mathbf{N}_2 = \begin{bmatrix} 0 & \omega & A \sin \omega_0 \\ -\omega & 0 & -A \cos \omega_0 \\ -A \sin \omega_0 & A \cos \omega_0 & 0 \end{bmatrix}. \quad (29)$$

In order to use the above result to solve the problem of rotating the moving platform from \mathbf{R}_I to \mathbf{R}_F , we can scale the time variable, t , so that the maneuver is completed at $t = 1$. Then, at $t = 1$, we have that

$$\mathbf{R}_F = \exp(\mathbf{N}_1) \exp(\mathbf{N}_2) \mathbf{R}_I = \mathbf{R}_z(\omega) \mathbf{R}_p(\delta) \mathbf{R}_I, \quad (30)$$

where

$$\mathbf{p} = \frac{1}{\sqrt{\omega^2 + A^2}} (A \cos \omega_0, A \sin \omega_0, -\omega)^T, \quad (31)$$

$$\delta = \sqrt{\omega^2 + A^2}. \quad (32)$$

In other words, the goal is to find ω , ω_0 , and A such that

$$\mathbf{R}_z(\omega) \mathbf{R}_p(\delta) = \mathbf{R}_F \mathbf{R}_I^T = \mathbf{R}_T. \quad (33)$$

If the vector of Euler parameters for the rotation defined by \mathbf{R}_T is (a, b, c, d) , it can be checked that the Euler parameters for $\mathbf{R}_z(-\omega) \mathbf{R}_T$ can be expressed, as a function of ω , as

$$a'(\omega) = a \cos \frac{\omega}{2} + d \sin \frac{\omega}{2}, \quad (34)$$

$$b'(\omega) = b \cos \frac{\omega}{2} + c \sin \frac{\omega}{2}, \quad (35)$$

$$c'(\omega) = c \cos \frac{\omega}{2} - b \sin \frac{\omega}{2}, \quad (36)$$

$$d'(\omega) = d \cos \frac{\omega}{2} - a \sin \frac{\omega}{2}. \quad (37)$$

Since, according to (33), $\mathbf{R}_p(\delta) = \mathbf{R}_z(-\omega) \mathbf{R}_T$, we have that

$$\cos \frac{\delta}{2} = a'(\omega), \quad (38)$$

$$\frac{A}{\delta} \cos \omega_0 \sin \frac{\delta}{2} = b'(\omega), \quad (39)$$

$$\frac{A}{\delta} \sin \omega_0 \sin \frac{\delta}{2} = c'(\omega), \quad (40)$$

$$-\frac{\omega}{\delta} \sin \frac{\delta}{2} = d'(\omega). \quad (41)$$

Now, observe that (38) and (41) depend only on ω and δ . From (38), we have that

$$\delta = \pm 2 \arccos(a'(\omega)). \quad (42)$$

Moreover, equation (38) can be rewritten as $\sin(\delta/2) = \pm \sqrt{1 - [a'(\omega)]^2}$. Then, dividing this expression by (41), we conclude that $-\delta/\omega = \pm \sqrt{1 - [a'(\omega)]^2}/d'(\omega)$. In other words,

$$\delta = \mp \omega \frac{\sqrt{1 - [a'(\omega)]^2}}{d'(\omega)}. \quad (43)$$

Therefore, equating (42) and (43) yields the following transcendental equation in ω

$$2d'(\omega) \arccos(a'(\omega)) + \omega \sqrt{1 - [a'(\omega)]^2} = 0. \quad (44)$$

Unfortunately, as it is usually the case for transcendental equations, no explicit solution has been found for (44). Thus, we have to rely at this point on a numerical method.

If we plot δ as a function of ω using (42) and (43), the intersection of both curves will correspond to the sought solutions. Fig. 3 depicts a typical example of the obtained plots.

Observe that if (δ, ω) is a solution of (42) and (43), then $(-\delta, \omega)$ is a solution as well, but they both correspond to the

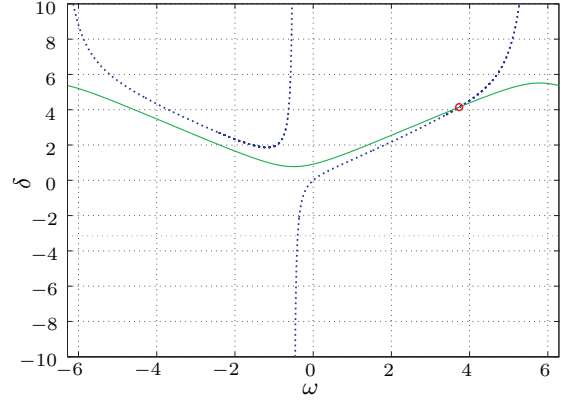


Fig. 3. Typical example of plots representing the positive branches of (43) and (42) shown in dashed blue and solid green lines, respectively.

same physical motion. This simply accounts for the double covering of $SO(3)$ when using Euler parameters.

Finally, with the obtained solutions for ω and δ , it is concluded from (32) that

$$A = \sqrt{\delta^2 - \omega^2}, \quad (45)$$

and, from (40) and (39), that

$$\omega_0 = \arctan\left(\frac{c'(\omega)}{b'(\omega)}\right). \quad (46)$$

In conclusion, the robot orientation as a function of time can simply be expressed as

$$\mathbf{R}(t) = \mathbf{R}_z\left(\frac{t}{\Delta t} \omega\right) \mathbf{R}_p\left(\frac{t}{\Delta t} \delta\right) \mathbf{R}_I, \quad 0 \leq t \leq \Delta t. \quad (47)$$

It can be verified that the equivalent axis of rotation of $\mathbf{R}_z\left(\frac{t}{\Delta t} \omega\right) \mathbf{R}_p\left(\frac{t}{\Delta t} \delta\right)$ lies on the XY plane for any value of t , as required [9, p. 85].

At this point, it is worth remembering that a maneuver is a sequence of moves connected by zero-velocity points. Thus, a one-move maneuver is not, strictly speaking, a maneuver and hence the title of this paper.

VI. EXAMPLE

Let us consider the non-holonomic orienting robot shown in Fig. 4. The center of the sphere is located at the origin, the spherical joints attached to the base are centered at $\mathbf{a}_1 = (1, 0, 0)^T$ and $\mathbf{a}_2 = (0, 1, 0)^T$, and those attached to the moving platform, in the reference orientation, at $\mathbf{b}_1^0 = (0.5, 0, 0)^T$ and $\mathbf{b}_2^0 = (0, 0.5, 0)^T$. Due to the non-holonomic constraint, the sphere cannot rotate about $\hat{\mathbf{r}}$, which is assumed to be aligned with the z -axis, as above.

Let us also assume that the initial and final orientations of the moving platform are given by

$$\mathbf{R}_I = \begin{bmatrix} 0.7513 & -0.5275 & -0.3965 \\ 0.5071 & 0.8460 & -0.1646 \\ 0.4223 & -0.0774 & 0.9031 \end{bmatrix} \quad (48)$$

and

$$\mathbf{R}_F = \begin{bmatrix} 0.9970 & 0.0534 & 0.0553 \\ -0.0192 & 0.8700 & -0.4927 \\ -0.0744 & 0.4902 & 0.8684 \end{bmatrix}, \quad (49)$$

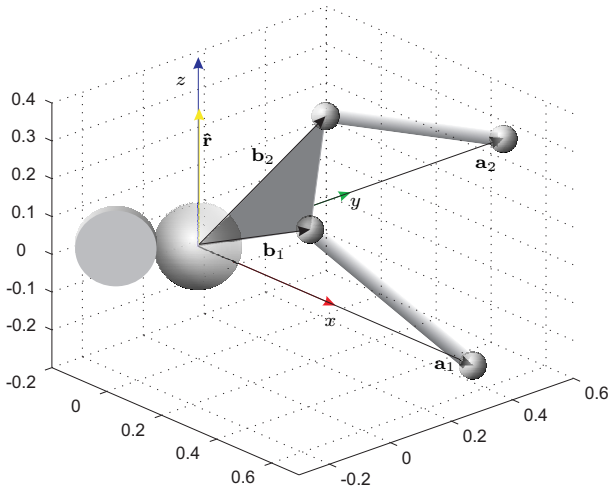
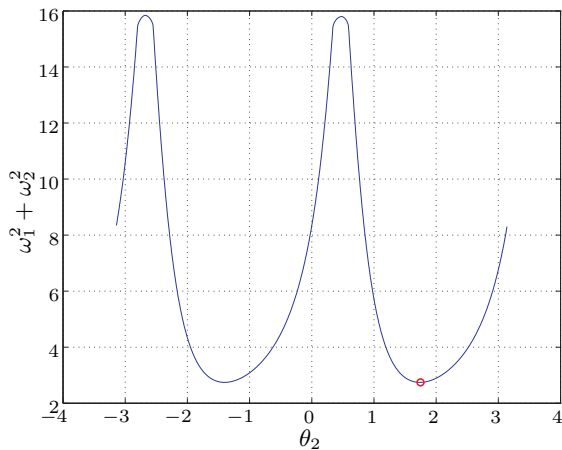


Fig. 4. Location of the joints and the disk for the robot used in the example.

Fig. 5. The minimum of $\omega_1^2 + \omega_2^2$ as a function of θ_2 is attained at 1.7384. This value determines the two-move maneuver used in the example.

respectively. Then,

$$\mathbf{R}_T = \mathbf{R}_F \mathbf{R}_I^{-1} = \begin{bmatrix} 0.6991 & 0.5416 & 0.4668 \\ -0.2780 & 0.8074 & -0.5204 \\ -0.6588 & 0.2340 & 0.7150 \end{bmatrix}. \quad (50)$$

A. Three-move maneuver

The three-step maneuver results from applying equations (9), (10) and (11). This yields

$$\omega_1 = 0.8594, \quad \omega_2 = 0.7967, \quad \omega_3 = -0.3993.$$

A representation of the robot motion following the resulting trajectory can be seen in Fig. 6(top row).

B. Two-move maneuver

To obtain a two-move maneuver, the first step is to compute a set of values for γ_1 , γ_2 , and γ_3 satisfying (14). Then, we can express ω_1 and ω_2 as a function of θ_2 using (19), (20), and (21). Since we can arbitrarily choose θ_2 , we can pick out the value that minimizes $\omega_1^2 + \omega_2^2$. This minimum is attained

at $\theta_2 = 1.7384$ (see Fig. 5). The substitution of this value in (21) yields $\theta_1 = 2.5829$. Then, the axes of rotation are

$$\mathbf{n}_1 = (-0.848, 0.530, 0)^T, \quad \mathbf{n}_2 = (-0.167, 0.986, 0)^T,$$

and the rotated angles about them are obtained substituting the values of θ_1 and θ_2 in (19) and (20), respectively. This yields

$$\omega_1 = 1.1713, \quad \omega_2 = -1.1708.$$

A representation of the robot motion following the resulting trajectory can be seen in Fig. 6(center row).

C. One-move maneuver

Following the procedure detailed in Section V, the first step consists in computing the Euler parameters for the rotation defined by (50). The result is $(0.8974, 0.2102, 0.3136, -0.2283)$. Then, we can plot δ as a function of ω using both (42) and (43). The result is plotted in Fig. 3. The intersection of both curves occurs at

$$\delta = 4.1578, \quad \omega = 3.7496.$$

Then, the substitution these values in (45) and (46) yields

$$A = 1.7965, \quad \omega_0 = -0.8945.$$

A representation of the motion followed by the robot along the resulting path can be seen in Fig. 6(bottom row).

D. Comparing the three path planners

Fig. 6 shows the motion generated by the three path planners organized in three rows. From top to bottom we can see the generated three-move, two-move, and single-move maneuvers. In the left column, we have light gray and dark gray triangles representing the moving platform in its initial and final orientation, respectively. The sequence of small reference frames illustrate the path followed by \mathbf{b}_1 . This is the motion generated as seen from the base reference frame. If we fix the observer to the moving reference frame, the motion followed by the disk on the sphere is better appreciated. This is represented in the right column.

All three trajectories behave well and quite similarly in this example. A greater difference is observed when translating the generated motions into a variation of the two leg lengths using (5). The result is represented in Fig. 7. The single-move maneuver generates a differentiable path. It can lead to faster motions because the generated trajectories do not contain zero-velocity points, thus making the use of the studied parallel robot possible in a larger range of applications. Finally, it is worth noting that, although using the single-move path planner the time variation of the legs' lengths are differentiable, the path followed by the point of contact between the disk and the sphere contains cusp points (see Fig. 8). These points play a fundamental role in non-holonomic path planners [12]. An important result is that they are here automatically generated.

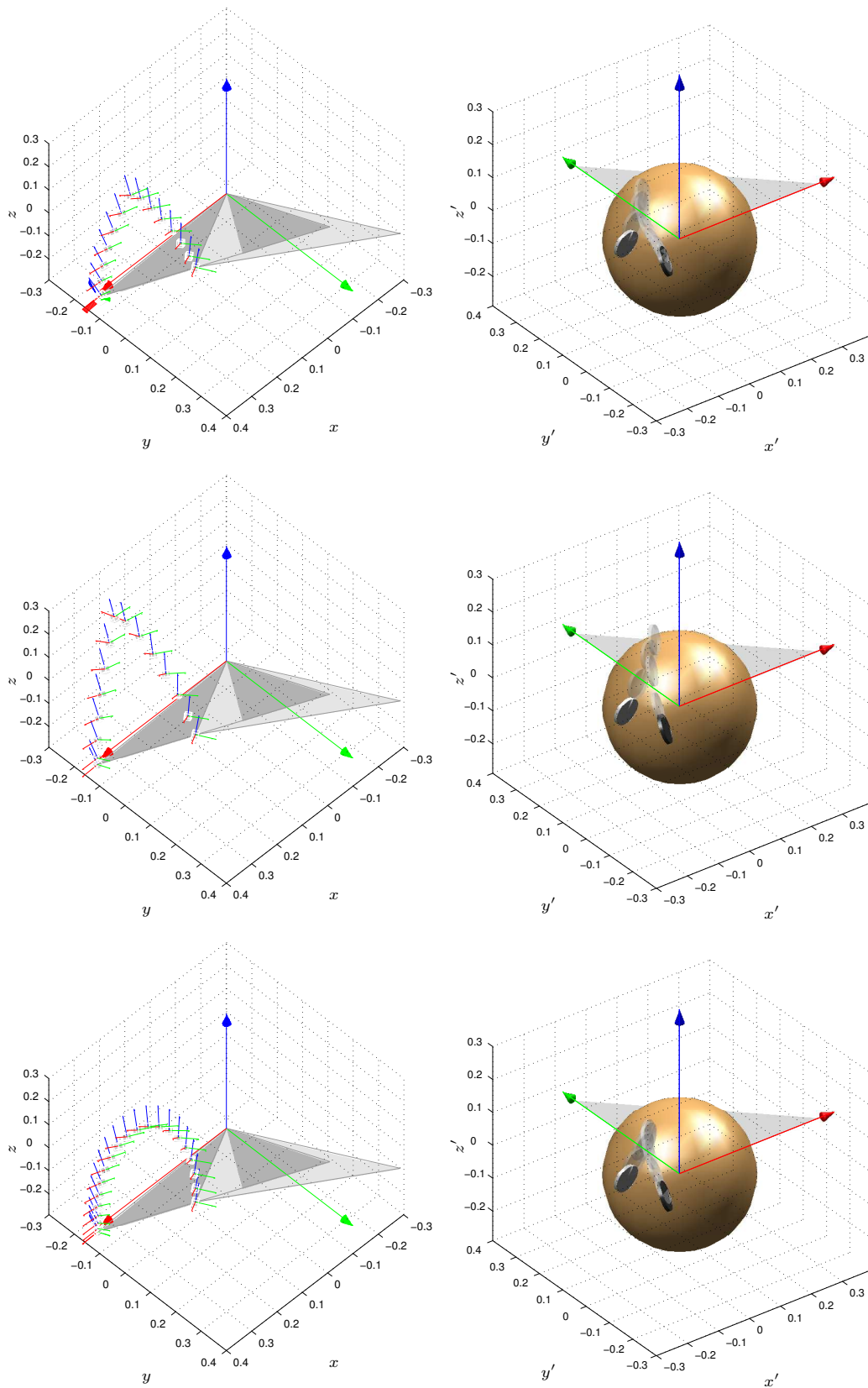


Fig. 6. Example of the motions generated by the three described path planners shown with respect to both the reference frame of the base and that of the moving platform (see text for details).

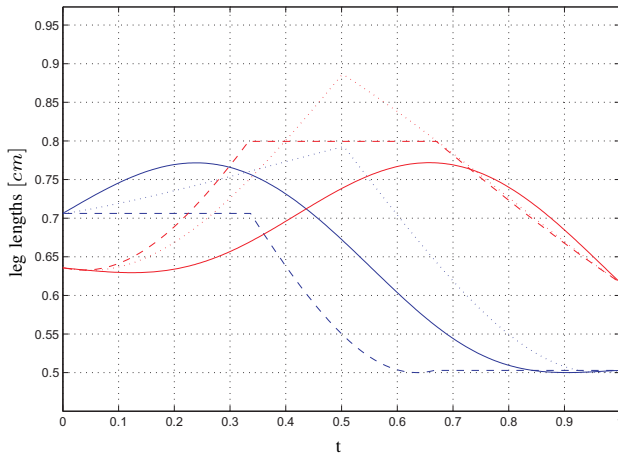


Fig. 7. Temporal variation of the leg lengths (l_1 in blue and l_2 in red) for the three generated trajectories. The leg lengths for the three-move maneuver are represented in dashed lines, for the two-move maneuver, in dotted lines, and for the single-move maneuver, in solid lines.

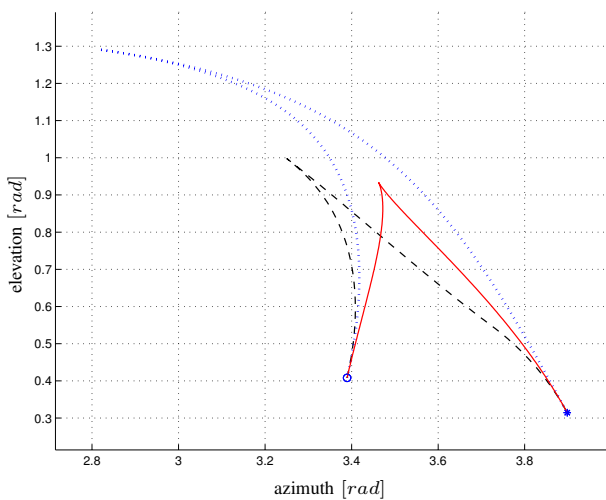


Fig. 8. Paths followed by the point of contact between the disk and the sphere for the three path planners. The initial and final configurations are indicated by 'o' and '*', respectively. The same line code as in Fig. 7 is used to indicate the results generated by the different path planners.

VII. CONCLUSION

It has been shown that it is possible to find smooth paths connecting two arbitrary orientations of a sphere constrained in its motion by a disk that can freely roll in contact with it without slipping. The generated paths result from choosing the ansatz in (25), but there are other alternatives. This means that there are probably infinitely many L^∞ paths connecting two arbitrary orientations in $SO(3)$ that satisfy the non-holonomic constraint. This opens the possibility of optimizing the path according to some criterion, or even the possibility of finding paths with closed-form formulas for their defining parameters, instead of relying on a numerical method.

The presented path planners are open-loop methods. However, it would be desirable to construct the input as a function of the system state to compensate for noises and errors in the system. Path planners that generate maneuvers cannot trans-

lated into control systems in an obvious way. The situation changes with the presented single-move path planner. This is certainly a point that deserves further attention.

REFERENCES

- [1] M. Callegari, M. Battistelli, and R. Di Gregorio, "Design of a non-holonomic spherical wrist," *Journal of Intelligent and Robotic Systems*, to appear, 2015.
- [2] P. Grosch, R. Di Gregorio, and F. Thomas, "Generation of under-actuated manipulators with non-holonomic joints from ordinary manipulators," *ASME Journal Mech. Robotics*, vol. 2, no. 1, 2010.
- [3] R. Di Gregorio, "Kinematic analysis of the (nS)-2SPU underactuated parallel wrist," *ASME Journal Mech. Robotics*, vol. 4, no. 3, 2012.
- [4] R. Di Gregorio, "Position analysis and path planning of the S-(nS)PU-SPU and S-(nS)PU-2SPU underactuated wrists," *ASME Journal Mech. Robotics*, vol. 4, no. 2, 2012.
- [5] R. Di Gregorio, "Type synthesis of underactuated wrists generated from fully-parallel wrists," *ASME Journal Mech. Des.*, vol. 134, no. 12, 2012.
- [6] P. Grosch and F. Thomas, "A bilinear formulation for the motion planning of non-holonomic parallel orienting platforms," *Proc. of the 2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pp. 953-958, 2013.
- [7] J. Jakubiak, W. Magiera, and K. Tchon, "Control and motion planning of a non-holonomic parallel orienting platform," *ASME J. of Mechanisms and Robotics*, vol. 7, no. 4, 2015.
- [8] R. Di Gregorio, "Position analysis, path planning, and kinetostatics of single-loop RU-(nS)PU wrists," *Mech. Machine Theory*, vol. 74, pp. 117133, 2014.
- [9] P. Grosch, *Parallel Robots with Unconventional Joints to Achieve Under-Actuation and Reconfigurability*, PhD dissertation, Technical University of Catalonia, 2015 (to appear).
- [10] J.A. Zimmerman, "Optimal control of the sphere S^n rolling on E^n ," *Math. Control Signals Systems*, vol. 17, no. 1, pp. 14-37, 2005.
- [11] V. Jurdjevic and J.A. Zimmerman, "Rolling problems on spaces of constant curvature. Lagrangian and Hamiltonian methods for nonlinear control," *Lecture Notes in Control and Inform. Sci.*, vol. 366, pp. 221-231, Springer, Berlin, 2007.
- [12] J.P. Laumond, S. Sekhavat, and F. Lamiraux, *Guidelines in Nonholonomic Motion Planning for Mobile Robots*, in *Robot Motion Planning and Control*, J.P. Laumond (ed.), Lecture Notes in Control and Information Science, 129, pp. 1-53, Springer, 1998.
- [13] R.M. Murray and S.S. Sastry, "Nonholonomic motion planning: Steering using sinusoids," *IEEE Trans. on Automatic Control*, vol. 38, no. 5, pp. 700-716, 1993.
- [14] D. Eberly, "Euler angle formulas," *Geometric Tools, LLC*. Available at <http://www.geometrictools.com/>
- [15] C.W. Wampler, "Displacement analysis of spherical mechanisms having three or fewer loops," *ASME J. Mech. Design*, vol. 126, no. 1, pp. 93-100, 2004.
- [16] M.Y. Wu, "Some new results in linear time-varying systems," *IEEE Trans. on Automatic Control*, vol. 20, pp. 159-161, 1975.