Motion Planning for a Spherical Mobile Robot: Revisiting the Classical Ball-Plate Problem

Ranjan Mukherjee
Department of Mechanical Engineering,
Michigan State University,
East Lansing, MI 48824-1226
e-mail: mukherji@egr.msu.edu

Mark A. Minor
Department of Mechanical Engineering,
University of Utah,
Salt Lake City, UT 84112

Jay T. Pukrushpan
Department of Mechanical Engineering,
University of Michigan,
Ann Arbor, MI 48109

In comparison to wheeled robots, spherical mobile robots offer greater mobility, stability, and scope for operation in hazardous environments. Inspite of these advantages, spherical designs have failed to gain popularity due to complexity of their motion planning and control problems. In this paper, we address the motion planning problem for the rolling sphere, often referred in the literature as the "ball-plate problem," and propose two different algorithms for reconfiguration. The first algorithm, based on simple geometry, uses a standard kinematic model and invokes alternating inputs to obtain a solution comprised of circular arcs and straight line segments. The second algorithm is based on the Gauss-Bonet theorem of parallel transport and achieves reconfiguration through spherical triangle maneuvers. While the second algorithm is inherently simple and provides a solution comprised of straight line segments only, the first algorithm provides the basis for development of a stabilizing controller. Our stabilizing controller, which will be presented in our next paper, will be the first solution to a problem that has eluded many researchers since the kinematic model of the sphere cannot be converted to chained form. Both our algorithms require numerical computation of a small number of parameters and provide the scope for easy implementation. [DOI: 10.1115/1.1513177]

1 Introduction

Mobile robots are typically designed with wheels, likely due to our kinship with automobiles. Relying on traditional use of the wheel as a quasi-static device, mobility and stability of the robots are enhanced using multiple wheels, large wheels, multi-wheel drives, broad wheel bases, traction enhancing devices, articulated body configurations, etc. The single-wheel robot recently proposed by Brown and Xu [1] represents a paradigm shift in mobile robot design. This robot, known as the Gyrover, exploits gyroscopic forces for steering and stability and has certain advantages over traditional designs. Similar to the Gyrover, which differs from traditional quasi-static models, a few designs have been proposed for spherical wheels with internal mechanisms for propulsion [2–4]. The robot designed by Halme et al. [3] incorporates a single-wheeled device constrained inside the spherical wheel; the device generates motion by creating unbalance and changes heading by turning the wheel axis. The design by Bicchi et al. [4] is similar but employs a four-wheeled car to generate the unbalance. The omnidirectional robot by Koshiyama and Yamafuji [2] has a limited range of lateral roll due to its arch-shaped body. Naturally, it fails to completely exploit the maneuverability associated with spherical exo-skeletons.

We are independently engaged in research and development of a spherical mobile robot, and in this paper, we address the nonholonomic motion planning problem. This problem, also important in its own right, refers to the task of converging a sphere, rolling without slipping on a flat surface, from any initial configuration to an arbitrary final configuration. The problem is particularly interesting since the kinematic model of the sphere cannot be reduced to chained-form [5]; this renders regimented nonholonomic motion planning and control algorithms inapplicable [6]. In reality, the rolling sphere is a completely controllable system, and trajectories for reconfiguration exist. Although differential-geometric tools can ascertain the existence of such trajectories [7],

they do not provide solution trajectories. To obtain a feasible solution, a three-step algorithm was proposed by Li and Canny [7]. The position coordinates of the sphere are converged to their desired values in the first step of the algorithm. In the second step, two of the three orientation coordinates are converged using Lie Bracket-like motion. Such motion generates an equatorial spherical triangle on the surface of the sphere. The third step uses a polhode to converge the last orientation coordinate.

The rolling sphere problem, also referred to as the ball-plate problem, has been revisited in recent years. A control input transformation was proposed by Bicchi et al. [6] to obtain a kinematic model of the sphere with a triangular structure. This structure promises to simplify integration of the state equations for alternating inputs and arriving at a system of nonlinear equations. The equations can be solved taking additional criteria into account, such as avoiding workspace limits, or minimizing the length of the path, but their iterative solution demands excessive computational time. In the words of the authors [6], the solution process may fail because of "abnormal extremals encountered along the path." An optimal solution to the ball-plate problem, which minimizes the integral of the kinetic energy of the ball along the path, was proposed by Jurdjevic [8]. The results are elegant and indicate that optimal trajectories are characterized by elliptic functions. It may be noted that elliptic functions have been used earlier to construct the controllability of nonholonomic systems [9], and design optimal trajectories for mobile robots [10]. Despite the elegance and appeal of the closed-form structure, the optimal trajectories require computation of elliptic functions and integrals at every step and pose challenges in implementation.

In this paper, we present two computationally efficient motion-planning algorithms for the rolling sphere. A kinematic model of the sphere is first presented in Section 2. This model provides the basis for the first algorithm, which is presented in Section 3. The first algorithm uses plane trigonometry to obtain polhode solution trajectories. Although these polhodes bear resemblance to the trajectories generated by the third step of Li and Canny's algorithm [7], they are more general and reconfigure the sphere in fewer steps. We present our second algorithm in Section 4. This algorithm, based on spherical trigonometry, requires at most two spherical triangle maneuvers for complete reconfiguration. It is
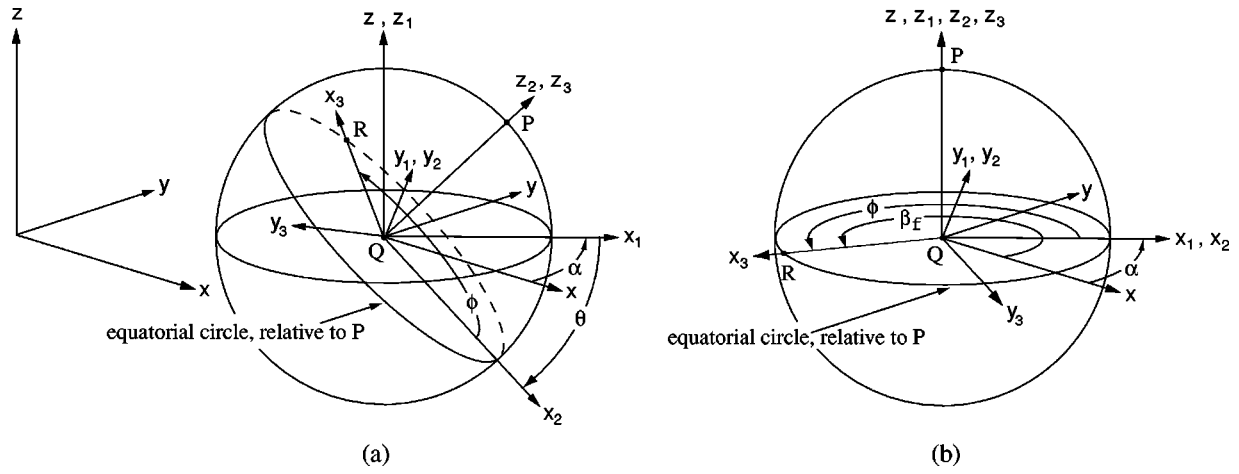
Fig. 1 Initial and final configurations of sphere

similar to the second step of the algorithm by Li and Canny [7], but eliminates the third step by using general spherical triangles, as opposed to equatorial triangles. The Gauss-Bonet theorem [11] provides a basis for our second algorithm but the basis can be independently established using spherical trigonometry [12]. Simulation results for both our algorithms are presented in Section 5. Section 6 provides concluding remarks and future research directions.

2 Problem Statement and Kinematic Model

The configuration of a sphere rolling on a flat surface can be described by the two Cartesian coordinates of the center of the sphere, and three coordinates describing the sphere orientation. In Fig. 1(a) the center of the sphere is defined by point Q; the orientation is described by points P and R, where P is an arbitrary point on the surface of the sphere, and R is an arbitrary point on the equatorial circle, defined relative to P. The complete reconfiguration of the sphere refers to the task of converging Q to the origin of the Cartesian coordinate frame, P to the vertically upright position, and R to some desired orientation, β_f, relative to the x axis. The final desired configuration of the sphere is shown in Fig. 1(b).

To obtain a kinematic model of the sphere, we denote Cartesian coordinates of the sphere center by $Q \equiv (x,y)$. We adopt the z-y-z Euler angle sequence (α, θ, ϕ) to represent the orientation of the sphere. We first translate the xyz frame to the center of the sphere and rotate it about the positive z axis by angle α, $-\pi \leq \alpha \leq \pi$, to obtain frame $x_1 y_1 z_1$. We rotate frame $x_1 y_1 z_1$ about the y_1 axis by angle θ, $0 \leq \theta \leq \pi$, to obtain frame $x_2 y_2 z_2$. The point P is located at the intersection point of the z_2 axis with the sphere surface. The $x_2 y_2 z_2$ frame is rotated about the z_2 axis by angle ϕ to obtain frame $x_3 y_3 z_3$. The point R is located at the intersection point of the x_3 axis and the sphere surface. The frames xyz, $x_1 y_1 z_1$, $x_2 y_2 z_2$, $x_3 y_3 z_3$, and z-y-z Euler angles (α, θ, ϕ) are all shown in Fig. 1(a). The reorientation of the sphere refers to the task of bringing P to the vertically upright position, and R, which then lies on the diametrical circle in the xy plane, to the orientation β_f with respect to the x axis. This can be achieved with $\theta = 0$, and $\alpha + \phi = \beta_f$, irrespective of the individual values of the Euler angles α, ϕ, as shown in Fig. 1(b). Therefore, the sphere can be completely reconfigured by satisfying

$$x=0, \quad y=0, \quad \theta=0, \quad \alpha+\phi=\beta_f \qquad (1)$$

The above equation may create the false impression that our objective is to converge the sphere to a configuration manifold. However, it can be verified from Fig. 1(b) that Eq. (1) represents a unique configuration of the sphere.

In the sequel, we assume the sphere to have unity radius without any loss of generality. Also, we denote the angular velocities of the sphere about the x_1, y_1, z_1 axes as $\omega_x^1, \omega_y^1, \omega_z^1$, respectively. Assuming $\omega_z^1 = 0$, the state equations can be written as

$$\dot{x} = \omega_y^1 \cos \alpha + \omega_x^1 \sin \alpha \qquad (2a)$$

$$\dot{y} = \omega_y^1 \sin \alpha - \omega_x^1 \cos \alpha \qquad (2b)$$

$$\dot{\theta} = \omega_y^1 \qquad (2c)$$

$$\dot{\alpha} = -\omega_x^1 \cot \theta \qquad (2d)$$

$$\dot{\phi} = \omega_x^1 \operatorname{cosec} \theta \qquad (2e)$$

In the model above, the first three equations can be derived simply. The expression for $\dot{\alpha}$ can be obtained from the relative velocity of P with respect to Q, when the sphere rotates with angular velocity ω_x^1. The angular velocity $\dot{\phi}$ is simply the vector sum of the angular velocities $\dot{\alpha}$ and ω_x^1. Alternatively, Eqs. (2c), (2d), and (2e), can be derived from the relation between the z-y-z Euler angle rates $\dot{\alpha}, \dot{\theta}, \dot{\phi}$, and the angular velocities $\omega_x^1, \omega_y^1, \omega_z^1$, subject to the constraint $\omega_z^1 = 0$.

3 Simple Geometric Motion Planner

3.1 Effect of Individual Control Actions. Consider the motion of the sphere described by the kinematic model in Eq. (2), for the individual control actions

$$\text{(A)} \quad \omega_y^1 \neq 0, \quad \omega_x^1 = 0$$

$$\text{(B)} \quad \omega_x^1 \neq 0, \quad \omega_y^1 = 0, \quad \theta \neq 0$$

The motion of the sphere for these actions are explained with the help of Fig. 2. For action (A), the sphere moves along straight line CF as θ changes. Let F be the point on this straight line where the sphere would have $\theta = 0$. Since the sphere rolls without slipping, this point remains invariant under control action (A). For control action (B), the instantaneous radius of the path traced by the sphere on the xy plane can be computed using Eq. (2) as follows

$$\rho = \left| \frac{(\dot{x}^2 + \dot{y}^2)^{3/2}}{\dot{x}\ddot{y} - \dot{y}\ddot{x}} \right| = \tan \theta \qquad (3)$$
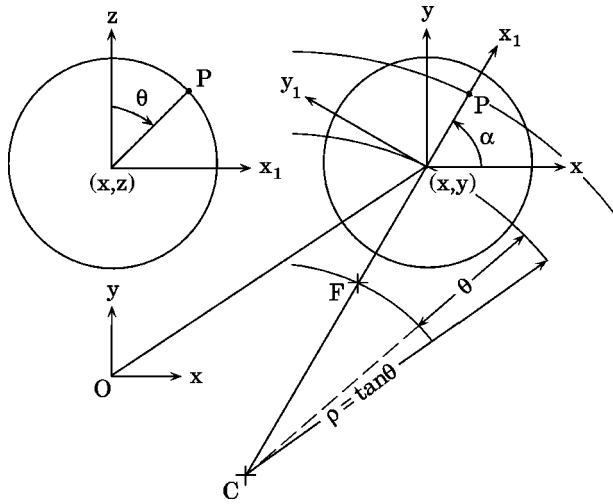
Fig. 2 Motion of the sphere under control actions (A) and (B)

Since $\omega_y^1 = 0$, θ is maintained constant. This implies that the contact point of the sphere moves along a circular path; the center of this circle is located at C in Fig. 2. Along with the contact point, points P and F also move along circular paths; the center of these paths lie on the vertical axis that passes through C. The point C remains fixed under control action (B), but under control action (A) moves away from F, as θ increases, and converges to F, as θ converges to zero.

3.2 Prelude to Complete Reconfiguration. Consider the initial configuration of the sphere in Fig. 3, where

$$x=x_i, \quad y=y_i, \quad \theta=0, \quad \alpha_i + \phi_i = \beta_i \qquad (4)$$

At the initial configuration, where $\theta = 0$, P is at the vertically upright position and R lies on the diametrical circle in the xy plane, at orientation β_i relative to the x axis. Due to singularity of the z-y-z Euler angle representation, α_i is arbitrary. This implies that we can arbitrarily choose the direction of x_1 axis relative to x axis. Once α_i has been chosen, ϕ_i is computed from Eq. (4) according to the relation $\phi_i = \beta_i - \alpha_i$. Now consider the (A)-(B)-(A) sequence of control actions, where

1. Control action (A) changes θ from $\theta=0$, to $\theta=\theta'$,
2. Control action (B) changes the direction of the x_1 axis by angle $\Delta\alpha$, and
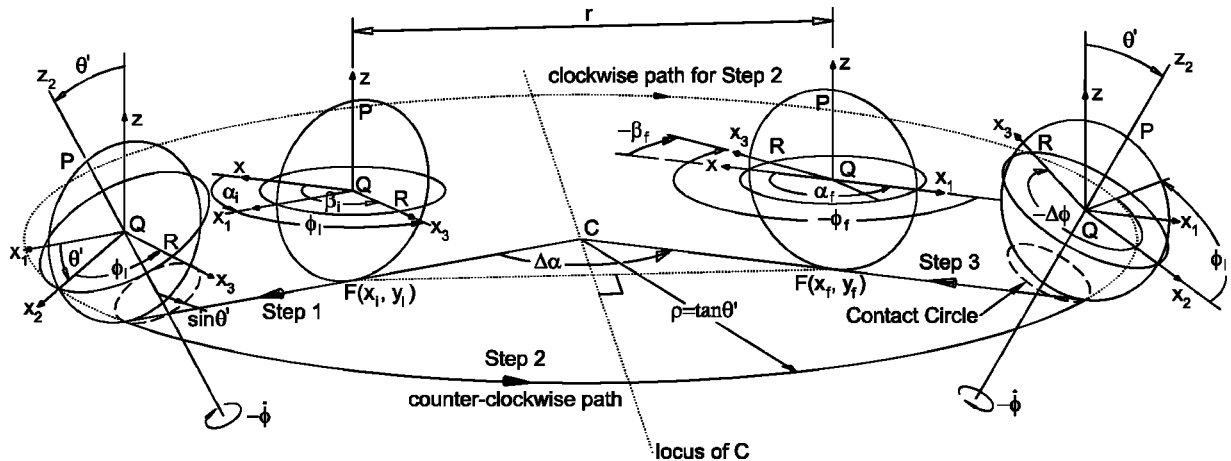3. Control action (A) changes θ from $\theta=\theta'$, back to $\theta=0$.

At the end of the first step, shown in Fig. 3, P moves away from its vertically upright position until QP subtends angle θ' with the vertical. The x_2 axis, initially coincident with the x_1 axis, now subtends angle θ' with the x_1 axis in the vertical plane. During step 2, θ remains constant at $\theta=\theta'$, the sphere traces out a circular arc on the ground, and the x_1 axis changes direction to remain pointed normal to the arc. The x_2 axis moves along with the x_1 axis, maintaining angle θ' with respect to it in the vertical plane. During step 2, the x_3 axis spins about the z_2 axis, and at the end of the maneuver undergoes a net rotation of $\Delta\phi$ about it. Since the x_1 axis undergoes a change of orientation by angle $\Delta\alpha$ during the same time, we can use Eqs. (2d) and (2e) to write

$$\Delta\phi = -\sec\theta' \; \Delta\alpha \qquad (5)$$

During step 3, control action (A) returns the plane containing the axes x_2, y_2, x_3, and y_3, to the plane containing the axes x, y, x_1, and y_1. At the end of the sequence of maneuvers, the sphere is described by the configuration in Fig. 3, where

$$x=x_f, \quad y=y_f, \quad \theta=0, \quad \alpha_f + \phi_f = \beta_f \qquad (6)$$

and where x_f, y_f, denote the final Cartesian coordinates of the sphere, and Euler angles α_f, ϕ_f, are given by the relations

$$\alpha_f = \alpha_i + \Delta\alpha \qquad (7)$$

$$\phi_f = \phi_i + \Delta\phi$$

The negative angles $-\Delta\phi$, $-\phi_f$, $-\beta_f$, and the negative angular velocity $-\dot\phi$, in Fig. 3 imply that these vectors are opposite in sense to their defined positive directions. Using Eqs. (5)–(7), it can be readily shown that

$$\Delta\beta \triangleq (\beta_f - \beta_i) = \Delta\alpha(1 - \sec\theta') \qquad (8)$$

In Eq. (8), $\Delta\beta$ depends on $\Delta\alpha$ and θ'. However, $\Delta\alpha$ and θ' are not independent quantities; it can be verified from Figs. 2 and 3 that they are related by the expression

Fig. 3 Motion of the sphere under (A)-(B)-(A) sequence of control actions

$$|\Delta\alpha| = \begin{cases} 2\arcsin\left[\, r/2\,(\tan\theta' - \theta')\,\right] & \text{for counter-clockwise path in step 2} \\ 2\{\pi - \arcsin\left[\, r/2\,(\tan\theta' - \theta')\,\right]\} & \text{for clockwise path in step 2} \end{cases} \qquad (9a)$$

where $r \triangleq [(x_f - x_i)^2 + (y_f - y_i)^2]^{1/2}$ is the distance between initial and final Cartesian coordinates of the sphere. Considering the fact that $\Delta\alpha$ is positive for counter-clockwise paths and negative otherwise, Eq. (9a) can be written more precisely as

$$\Delta\alpha = \begin{cases} 2\arcsin\left[\, r/2\,(\tan\theta' - \theta')\,\right] & \text{for counter-clockwise path in step 2} \\ 2\{\arcsin\left[\, r/2\,(\tan\theta' - \theta')\,\right] - \pi\} & \text{for clockwise path in step 2} \end{cases} \qquad (9b)$$

From Eqs. (8) and (9b), we can infer that $\Delta\beta$, for the (A)-(B)-(A) sequence of control actions, is implicitly a function of $\Delta\alpha$ alone, for a given value of r. To keep r unchanged, consider variations in $\Delta\alpha$ that do not change the initial and final positions of the sphere in Fig. 3. The feasible variation in $\Delta\alpha$ can be determined from the locus of C, also shown in Fig. 3. Since C must be equidistant from the initial location, $F = (x_i, y_i)$, and the final location, $F = (x_f, y_f)$, the locus of C is a straight line that orthogonally bisects the line joining the initial and final locations. When C is located on the opposite side of this line from the circular arc, $\Delta\alpha$ is an acute angle; it is zero when C lies at infinity and it approaches π as C approaches the line. When C is on the same side of the line as the circular arc, $\Delta\alpha$ is obtuse; it equals π when C is on the line, and it approaches 2π when C approaches infinity. Clearly, $0 \le \Delta\alpha \le 2\pi$ for the counter-clockwise path in step 2. The range of $\Delta\alpha$ for the clockwise path can be similarly shown to be $-2\pi \le \Delta\alpha \le 0$. Using Eq. (9b) to obtain θ' for different values of $\Delta\alpha$, we compute $\Delta\beta$ for both clockwise and counter-clockwise paths using Eq. (8), for different values of r. These plots of $\Delta\beta$ versus $\Delta\alpha$ are shown in Fig. 4. For the specific values of r that we considered, it is clear $\Delta\beta$ continuously assumes values over a range greater than 2π. For example, for $r=2$ and a counter-clockwise path, $\Delta\beta$ assumes values over a range of 2π with $\Delta\alpha$ in the range $a \le \Delta\alpha \le b$. The same holds true for $r=10$ along a clockwise path, when $\Delta\alpha$ in the range $-c \le \Delta\alpha \le -d$. Based on the plots above, we assert that an arbitrary value of $\Delta\beta$ can be achieved over a range of 2π for any fixed value of r. In the sequel, it seems reasonable to summarize that an (A)-(B)-(A) sequence of control actions, apart from pure translation of the sphere, can arbitrarily rotate the sphere about the vertical axis.
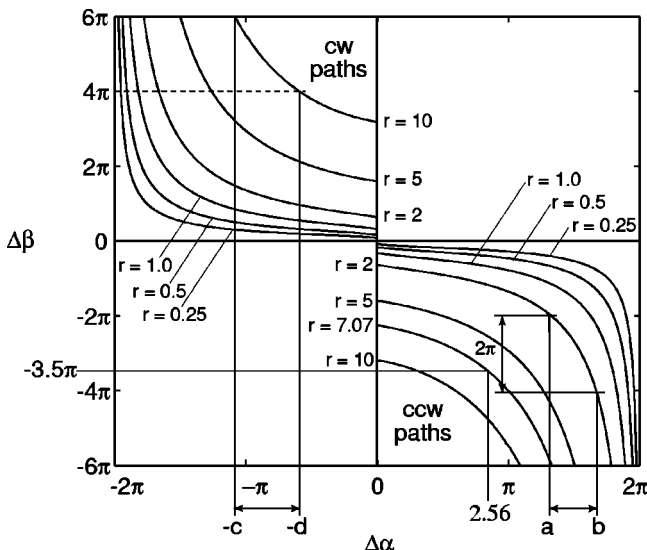

Fig. 4 Plot of $\Delta\beta$ versus $\Delta\alpha$ for different values of r, shown for both counter-clockwise and clockwise paths

3.3 Geometric Algorithm for Complete Reconfiguration.
In this section, we extend the (A)-(B)-(A) sequence of control actions to converge the sphere from its arbitrary initial configuration, described by the Cartesian coordinates and z-y-z Euler angles

$$(x, y, \alpha, \theta, \phi) = (x_0, y_0, \alpha_0, \theta_0, \phi_0) \qquad (10a)$$

to its final configuration satisfying Eq. (1), namely

$$x = 0, \quad y = 0, \quad \theta = 0, \quad \alpha + \phi = \beta_f \qquad (10b)$$

The constraint on the final configuration, namely $\alpha + \phi = \beta_f$, may create the wrong impression that our objective is to reconfigure the sphere onto a manifold in the configuration space, rather than a single point. This impression can be quickly dispelled with the reasoning that all configurations of the sphere, described by Eq. (10b), are physically one and the same. This unique configuration, which lacks a unique description in terms of Euler angle parameters, is also referred to as the singular configuration.

We now present our algorithm with the help of Fig. 5 In this figure, the initial configuration of the sphere, described by Eq. (10a), is denoted by (1). As the first step of our algorithm, we execute control action (A) whereby θ is changed from $\theta = \theta_0$ to $\theta = 0$. This brings the Cartesian coordinates of the sphere center from (x_0, y_0) to (x_i, y_i), given by the relations

$$x_i = (x_0 - \theta_0 \cos\alpha_0), \quad y_i = (y_0 - \theta_0 \sin\alpha_0) \qquad (11)$$

Since $\theta = 0$ at this new configuration, denoted by (2) in Fig. 5, point P is vertically upright and R is located on the equatorial circle, orientated β_i with respect to the x axis. Since $\theta = 0$, the value of β_i can be obtained from the initial configuration parameters as

$$\beta_i = \alpha_0 + \phi_0 \qquad (12)$$

Under control action (A), the center of rotation for control action (B), discussed in Section 3.1, moves from the coordinates denoted by C_0, to (x_i, y_i).

Now, our goal, in terms of reconfiguration, is to translate the sphere from (x_i, y_i) to $(0,0)$, and reorient the sphere through a pure rotation about the vertical, by angle $\Delta\beta$. The desired distance of translation, r, and angle of rotation, $\Delta\beta$, can be computed as

$$r = (x_i^2 + y_i^2)^{1/2}, \quad \Delta\beta = (\beta_f - \beta_i) \qquad (13)$$

To achieve the desired translation, and rotation about the vertical, we execute the (A)-(B)-(A) sequence of control actions, discussed in the previous section. The motion of the sphere under this sequence of control actions is shown in Fig. 5 by path segments (2)-(3), (3)-(4), and (4)-(5), respectively. As the sphere moves from (2)-(3), θ changes from $\theta = 0$ to $\theta = \theta'$, and the center of rotation for control action (B) moves from (x_i, y_i) to $C \equiv (a, b)$. Subsequently, during control action (B), the sphere moves along circular arc (3)-(4) with center of rotation fixed at C. During this motion θ is maintained constant at θ'. Though counter-clockwise direction of motion was chosen with control action (B), clockwise direction of motion can also be chosen. Finally, control action (A) brings the sphere from (4)-(5). The configuration of the sphere at (5) is in accordance with Eq. (10b).

To solve the problem mathematically, we first use the values of r and $\Delta\beta$ obtained from Eq. (13), to numerically compute $\Delta\alpha$ and θ' using Eqs. (8) and (9b). Once $\Delta\alpha$ has been computed, the coordinates of $C\equiv(a,b)$ can be obtained from the following equations

$$(a^2+b^2)=(x_i-a)^2+(y_i-b)^2, \quad \sin(\Delta\alpha/2)=\frac{r}{2(a^2+b^2)} \quad (14)$$

since C is equidistant from both coordinates (x_i,y_i) and $(0,0)$. Having solved for a and b one can verify the value of θ' obtained earlier with that obtained from the relation $(\tan\theta'-\theta')=a^2+b^2$, derived from Fig. 2. From the coordinates of C, we compute α_i from the relation

$$\tan\alpha_i=(y_i-b)/(x_i-a) \quad (15)$$

The value of α_i is useful for computing the direction of rolling for the path segment (2)-(3). The distance of rolling depends on the value of θ', which is known. The remaining maneuvers are comprised of motion of the sphere along: *i*) circular arc (3)-(4) under control action (B) that subtends angle $\Delta\alpha$ at the center of rotation, C, and *ii*) straight line (4)-(5) inclined at angle $\alpha_f=(\alpha_i+\Delta\alpha)$ with respect to the x axis, under control action (A) for a net rotation of θ'.

We conclude this section with a few remarks. Simulation results presented in Section 5 will provide a better understanding of our algorithm, and the steps involved.

1. The results in Fig. 4 indicate that the $\Delta\beta$-$\Delta\alpha$ curves are sharper for smaller values of r. This implies that small errors in the numerical computation of $\Delta\alpha$ (from a given value of $\Delta\beta$) will result in large errors in $\Delta\beta$ at the end of the maneuver, for small values of r. This problem can be trivially eliminated by rolling the sphere farther away from the origin of the Cartesian coordinate frame, at the initial point of time.

2. The graphs in Fig. 4 indicate that $\Delta\beta$ does not assume values in the range $[0,2\pi]$ for $\Delta\alpha$ in the same range, for all values of r. This does not pose problems since $\Delta\beta$ continuously assumes values over a range greater than 2π for different values of r. For $r=5$ and a counter-clockwise path, for example, we can use a value of $\Delta\alpha$ (approximately $\pi/3$) that will result in $\Delta\beta=-2\pi$, equivalent to $\Delta\beta=0$ (a value than cannot be directly achieved with any value of $\Delta\alpha$).

3. The algorithm by Li and Canny [7] requires one trivial and two non-trivial maneuvers for reconfiguration of the sphere. The first non-trivial maneuver is an equatorial spherical triangle maneuver that converges two of the three orientation coordinates of the sphere. The second non-trivial maneuver is based on a polhode trajectory that converges the third orientation coordinate. In comparison, our algorithm requires one trivial maneuver based on control action (A) and one non-trivial maneuver based on the (A)-(B)-(A) sequence of control actions. Clearly, the sphere is reconfigured in fewer number of steps using our algorithm.

4. It is clear from our algorithm that the sphere can be completely reconfigured if x, y, θ can be converged to zero, and β can be converged to an arbitrary value in $[0,2\pi]$. Since the change in β, $\Delta\beta$, is implicitly a function of the change in α, $\Delta\alpha$, for all values of r, and hypothetically there exists $\Delta\alpha$ values for all values of $\Delta\beta$ effectively in $[0,2\pi]$, the controllability of the sphere can be studied from the reduced set of kinematic relations in Eq. (2), namely

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\alpha} \end{pmatrix} = \begin{pmatrix} \sin\alpha \\ -\cos\alpha \\ 0 \\ \cot\theta \end{pmatrix}\omega_x^1 + \begin{pmatrix} \cos\alpha \\ \sin\alpha \\ 1 \\ 0 \end{pmatrix}\omega_y^1 \quad (16)$$

Specifically, by constructing the distribution comprised of the control vectors above, and their iterated Lie brackets, as follows
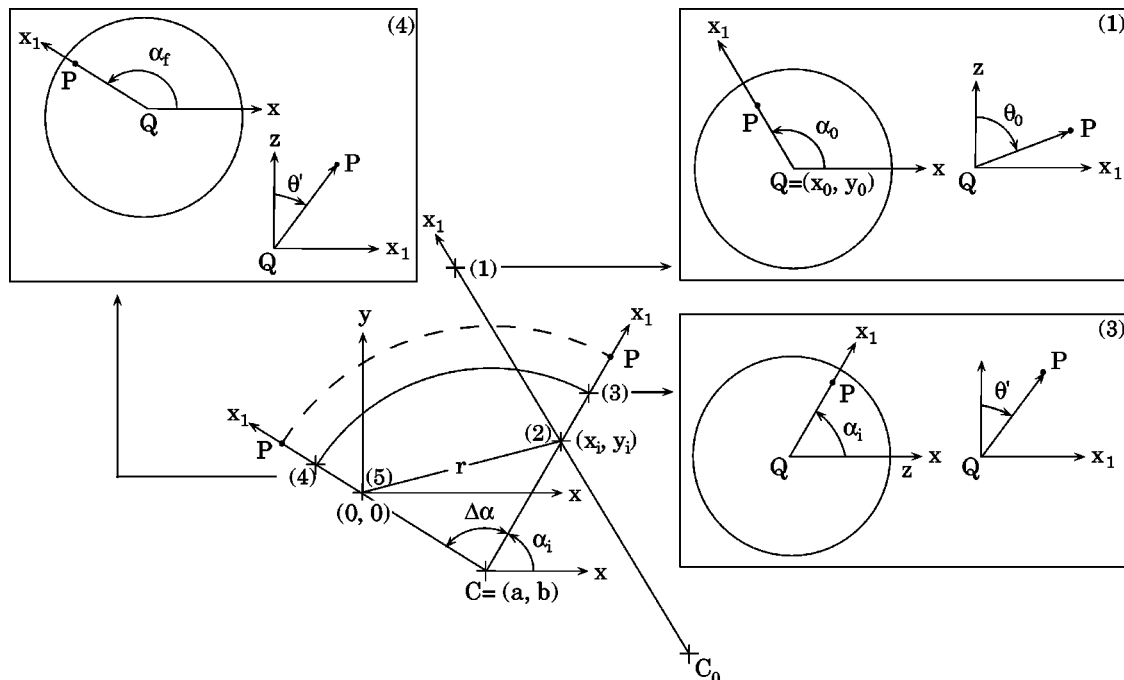

Fig. 5 Basis for complete reconfiguration of the sphere

$$\nabla = \begin{pmatrix} \vdots & \vdots & \vdots & \vdots \\ \mathbf{f} & \mathbf{g} & [\mathbf{f},\mathbf{g}] & [\mathbf{f},[\mathbf{f},\mathbf{g}]] \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix} = \begin{pmatrix} \sin\alpha & \cos\alpha & -\sin\alpha\cot\theta & -\cos\alpha\,\operatorname{cosec}^2\theta(1+\cos^2\theta) \\ -\cos\alpha & \sin\alpha & \cos\alpha\cot\theta & -\sin\alpha\,\operatorname{cosec}^2\theta(1+\cos^2\theta) \\ 0 & 1 & 0 & 0 \\ \cot\theta & 0 & \operatorname{cosec}^2\theta & 0 \end{pmatrix}$$

and showing that it has full rank, one can claim complete controllability. This eliminates the need for using the five-state kinematic model and defining two separate charts of the atlas covering the sphere for the proof of controlability [7].

5. The algorithm by Jurdjevic provides optimal trajectories but require numerical computation of elliptic functions and integrals. During implementation, the elliptic functions and integrals will have to be computed at every step to determine the instantaneous direction of rolling of the sphere. In comparison, our algorithm provides reconfiguration using a circular arc and straight line segments, whose parameters need to be computed only once. Clearly, our algorithm requires significantly less computation than the algorithm by Jurdjevic [8]. Also, due to the complex nature of the optimal trajectories, the error in the final configuration of the sphere is likely to be smaller for our algorithm.

6. Although the motion planning problem provides reconfiguration trajectories that can be tracked using a tracking controller, the more desirable approach to reconfiguration is through the design of a stabilizing controller. The approach by Li and Canny will not lend itself well to the design of a stabilizing controller since it relies on equatorial spherical triangle maneuvers. The same can be said about the optimal trajectories of Jurdjevic since optimal control formulations for nonlinear systems generally do not result in feedback control strategies. In comparison, our motion planning algorithm lends itself very well to the design of a stabilizing controller. Using our algorithm as a basis, we have successfully developed a feedback control strategy and have thereby resolved an important problem that has eluded many researchers till date. The feedback control strategy will be presented in our next paper.

4 Motion Planning Using Spherical Triangles

4.1 Effect of Rolling Along a Spherical Triangle. In this section, we present our second algorithm for complete reconfiguration of the sphere. This algorithm requires the contact point between the sphere and the ground to trace a triangle on the surface of the sphere. Since a spherical triangle is a closed path, the sphere undergoes a net rotation about the vertical axis at the end of the maneuver. The sphere also translates since the image of the spherical triangle on the ground is not a closed path. In this section, we obtain expr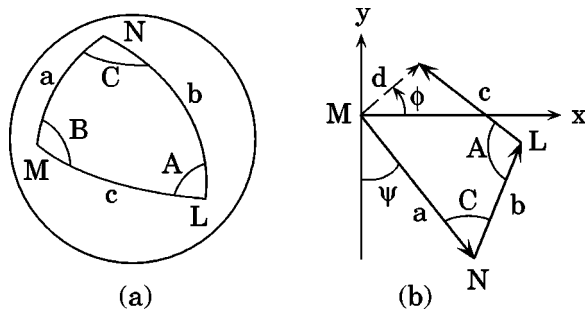essions for the net translation and rotation in terms of dimensions of the spherical triangle. These expressions will be used to solve the reconfiguration problem.

Consider the spherical triangle LMN, inscribed on the surface of the unit sphere, as shown in Fig. 6(a). The length of the sides of the triangle, which are arcs of great circles, are a, b, and c. The angles opposite to these sides are A, B, and C, respectively. Let the sphere initially be at the origin of the xy coordinate frame, shown in Fig. 6(b). At this location, let the sphere be oriented such that its point of contact with the ground is M and the image of MN is the line oriented at angle ψ counter-clockwise with the negative y axis. Since the spherical triangle can be arbitrarily inscribed on the surface of the sphere, the initial direction of roll, ψ, is arbitrary. Starting at point M, the sphere is rolled along this direction by distance a whereby the contact point changes to N. The sphere then rolls distance b after a left turn of angle C with the previous direction of travel. This brings the contact point on the sphere to L. Finally, the sphere rolls distance c after a left turn of angle A. This returns the contact point on the sphere to M. The final position of the sphere in the xy plane is shown in Fig. 6(b). Here, d and ϕ represent the distance and direction of translation, respectively. The direction is measured counter-clockwise with respect to the positive x axis. Using planar trigonometry, we can express d and ϕ by the relations

$$d^2 = a^2 + b^2 + c^2 + 2ac\,\cos(A+C) - 2ab\,\cos C - 2bc\,\cos A \tag{17}$$

$$\phi = \psi + \arctan\left[\frac{a - b\,\cos C + c\,\cos(A+C)}{c\,\sin(A+C) - b\,\sin C}\right] \tag{18}$$

Now, let i, j, k denote unit vectors along the inertially fixed coordinate axes x, y, z, respectively, and let i', j', k' denote orthogonal unit vectors that are fixed to the sphere. Assume that the unit vector triads are coincident before the spherical triangle maneuver. Since rolling along a spherical triangle is equivalent to a pure rotation about the z axis, the unit vector triads at the end of the maneuver will be related by the expression

$$\begin{pmatrix} i' \\ j' \\ k' \end{pmatrix} = \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} i \\ j \\ k \end{pmatrix} \tag{19}$$

where α is the amount of rotation about the positive z axis. To obtain an expression for α, we recall the general expression for rotation of a vector r about axis l by angle γ, to form r'

$$r' = (1 - \cos\,\gamma)(l\,.\,r)\,l + \cos\,\gamma\,\,r + \sin\,\gamma\,(l \times r) \tag{20}$$

Knowing that the spherical triangle maneuver is comprised of three consecutive rotations, with: i) $l = i$, $\gamma = a$; ii) $l = [-\cos Ci + \sin Cj]$, $\gamma = b$, and iii) $l = [\cos(A+C)i - \sin(A+C)j]$, $\gamma = c$, the composite rotation matrix can be computed. After tedious simplification using spherical trigonometric identities [13], the entries of the rotation matrix can be shown [12] to be identical to the entries of the matrix in Eq. (19), with

$$\alpha = \pi - (A + B + C) \leq 0 \tag{21a}$$

The path taken by the sphere in Fig. 6(b) can be considered as left-handed since left turns were taken at N and L. For a right-handed path, α takes the form [12]

Fig. 6 (*a*) A spherical triangle, (*b*) Image of the spherical triangle on the *x*-*y* plane
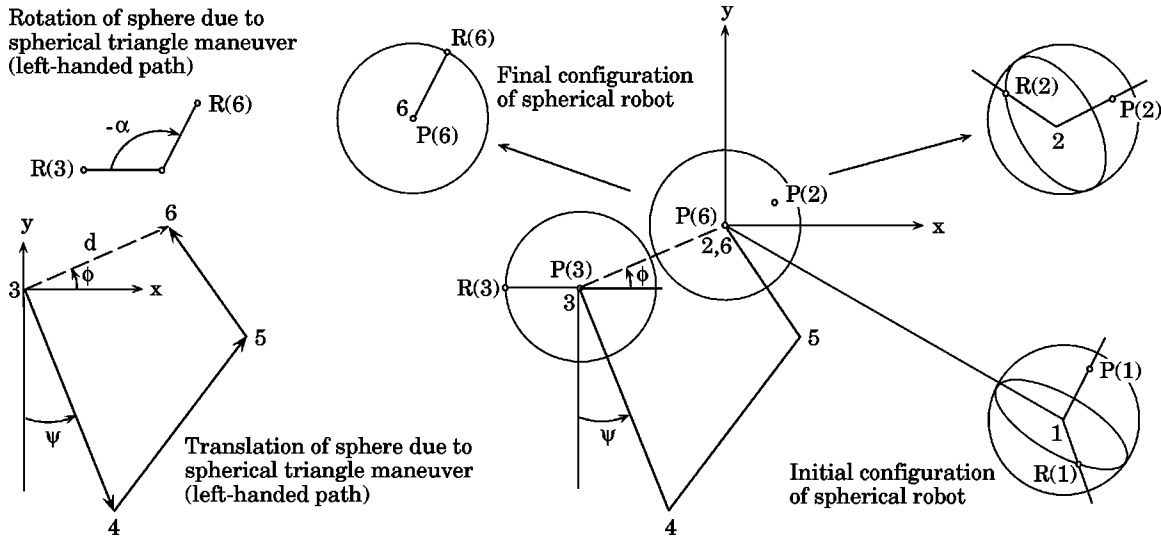
Fig. 7 Open loop reconfiguration using spherical triangles

$$\alpha = (A + B + C) - \pi \geq 0 \qquad (21b)$$

Using the spherical trigonometric relation [13] for the area of a spherical triangle, we can establish from Eq. (21) that the net rotation of a sphere tracing a triangle on its surface is equal to the solid angle subtended by the triangle. The same result is stated by the Gauss-Bonet theorem of parallel transport [1]. In summary, when a sphere rolls along a spherical triangle, the final location of the sphere can be computed using Eqs. (17) and (18); the net rotation of the sphere is about the vertical axis and can be computed using Eq. (21a) or (21b), depending upon whether the path is left-handed or right-handed, respectively.

4.2 Complete Reconfiguration Strategy. In this section we formulate an open loop strategy for completely reconfiguring the sphere. In reference to Fig. 7, this constitutes positioning the sphere at the origin of the xy frame, locating P (an arbitrary point on the sphere) at the top of the sphere, and orienting R (a fixed point on the equatorial circle, defined relative to P) in some desired direction on the horizontal plane. At the initial configuration, marked 1 in Fig. 7, point P is arbitrarily located at $P(1)$, and R is located on the equatorial circle relative to P, at $R(1)$. Our strategy consists of three steps. First, the sphere rolls from the initial configuration to the origin of the xy frame, denoted by the straight line path $1 \rightarrow 2$ in Fig. 7. At this time, P is located at an arbitrary point on the sphere, $P(2)$, and R is located at $R(2)$. Second, the sphere rolls away from the origin to bring P directly to the top, $P(3)$. This is depicted by the straight line trajectory $2 \rightarrow 3$. At this configuration, R, depicted by $R(3)$, is located on the horizontal diametrical plane at some arbitrary orientation. Finally, the sphere executes a spherical triangle maneuver $3 \rightarrow 4 \rightarrow 5 \rightarrow 6$ such that it translates back to the origin of the xy frame and its net rotation about the vertical axis points R in the desired direction. In conformity with Fig. 6, the initial direction of rolling for the spherical triangle maneuver is chosen along the line oriented ψ counterclockwise with respect to the negative y axis.

The first two steps of our algorithm are trivial to implement. The last step requires that we solve for a spherical triangle whose sides and angles satisfy Eqs. (17), (18), and (21a) or (21b), for given values of d, ϕ, and α. The value of ψ is assigned suitably to satisfy Eq. (18), which may not be satisfied otherwise.

4.3 Existense of a Solution. In this section, we show that it is possible to arbitrarily reconfigure the sphere using one or at-most two spherical triangle maneuvers. Since an arbitrary value of ϕ can be achieved through a proper choice of ψ, we only need to show that any value of d and α, $0 \leq d \leq \pi$, $0 \leq \alpha \leq \pi$, can be

achieved with one or two spherical triangle maneuvers. The domain of d is $[0, \pi]$ since the unit sphere rolls no more than π in the second step of our algorithm. The domain of α is defined over $[0, \pi]$ and not $[-\pi, \pi]$ since negative values of α can be achieved using left-handed paths.

To prove existence of a solution, we look at the possible set of d and α values that can be achieved with feasible triangle parameters, namely $0 \leq a, b, C \leq \pi$. For every set of a, b, and C, we determine A, B, and c using the spherical trigonometric identities [13]

$$A - B = 2 \arctan\left(\frac{\sin[(a-b)/2]}{\sin[(a+b)/2] \tan C/2}\right)$$

$$A + B = 2 \arctan\left(\frac{\cos[(a-b)/2]}{\cos[(a+b)/2] \tan C/2}\right) \qquad (22)$$

$$c = 2 \arctan\left(\frac{\tan[(a+b)/2] \cos[(A+B)/2]}{\cos[(A-B)/2]}\right)$$

All values are then substituted in Eqs. (17), (18), and (21b) to numerically compute d and α. These set values are shown by the gray area in the first quadrant in Fig. 8. Clearly, not all values of d and α, $0 \leq d, \alpha \leq \pi$, can be achieved with a single spherical
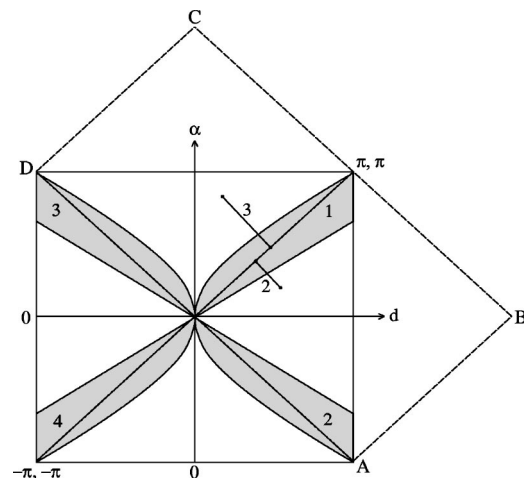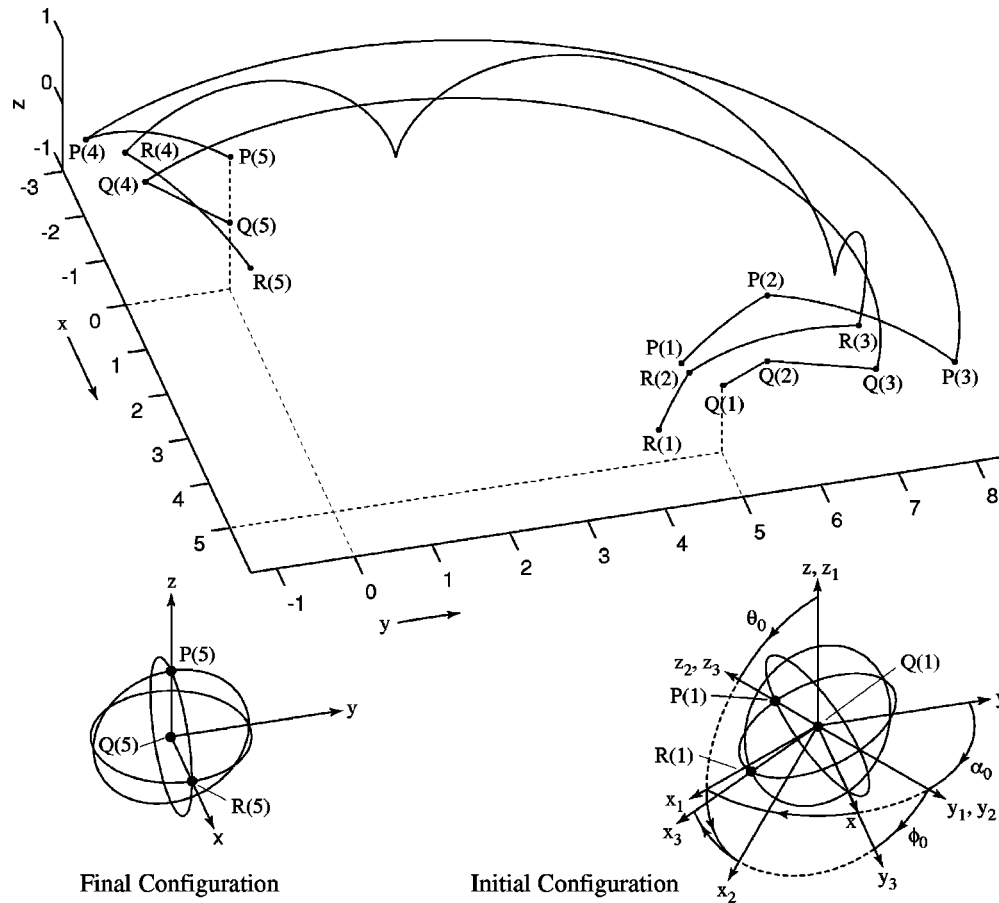

Fig. 8 Reachable space of d-α

Fig. 9 Reconfiguration of the sphere using the geometric motion planner

triangle maneuver. However, we assert that an arbitrary d and α can be achieved with at most two triangle maneuvers.

To prove our assertion, we first realize that ϕ can be arbitrary since ψ is arbitrary, and in particular, can vary by 180° for the same value of d. This implies that for every achievable d in $[0,\pi]$, we can effectively achieve $-d$ in $[-\pi,0]$. This, in addition to the fact that right- and left-handed paths result in opposing signs of α, suggests that we expand the set of achievable d and α, $-\pi \leq d, \alpha \leq \pi$, as shown in Fig. 8. Now consider the straight lines 1, 2, 3, and 4 in Fig. 8, which are conservative estimates of the reachable space of d-α. We translate lines 2 and 3 along line 1 to generate the rectangle $ABCD$. It can be argued that any point inside $ABCD$ can be reached by first moving along line 1 and then along a line parallel to line 2 or 3, by appropriate distances. Since the rectangular area contains the entire region $0 \leq d, \alpha \leq \pi$, it can be argued that the sphere can be arbitrarily reconfigured using at most two spherical triangle maneuvers in the third step of our algorithm.

We conclude this section by summarizing the merits of our algorithm. Our algorithm, based on spherical triangle maneuvers, is similar to the second step of the algorithm proposed by Li and Canny [7]. It is however more general since it is not restricted to the use of equatorial spherical triangles. As a direct consequence, it eliminates the need for the third step in Li and Canny's algorithm [7]. In comparison to the algorithm by Jurdjevic, where the trajectories are defined in terms of elliptic functions and integrals, trajectories designed by our algorithm are comprised only of straight line segments. This provides the scope for much easier implementation.

5 Simulation Results

5.1 Geometric Motion Planner. We simulate a reconfiguration maneuver of an unit sphere radius based on the geometric motion planner presented in Section 3. In accordance with Eq. (10a), the initial configuration of the sphere is assumed to be

$$(x_0, y_0, \alpha_0, \theta_0, \phi_0) \equiv (5, 5, -60°, 45°, -30°) \qquad (23a)$$

where the units are in meters and degrees, repectively. The final configuration, defined by Eq. (10b), is chosen to satisfy

$$x = 0, \quad y = 0, \quad \theta = 0°, \quad \alpha + \phi = 0° \qquad (23b)$$

The final configuration may give the false impression that we are addressing a problem of reconfiguration to an equilibrium manifold. From our discussion in Section 3.3, however, we know that Eq. (23b) represents a unique configuration of the sphere.

The simulation results are shown in Fig. 9. The position and orientation of the sphere is depicted by points P, Q, and R. Here Q denotes the sphere center, and points P and R lie at the intersection of the sphere with the positive z_2 and x_3 axes, respectively, in accordance with our discussion in Section 2. At the initial configuration, inset in Fig. 9, the three points are located at $P(1)$, $Q(1)$, and $R(1)$, respectively. In the first step of our algorithm, they move to locations $P(2)$, $Q(2)$, and $R(2)$, under action (A). At this configuration, $P(2)$ is vertically above $Q(2)$ whose x and y coordinates can be obtained using Eq. (11)

$$x_i = (x_0 - \theta_0 \cos \alpha_0) = 4.61, \quad y_i = (y_0 - \theta_0 \sin \alpha_0) = 5.68 \qquad (24)$$
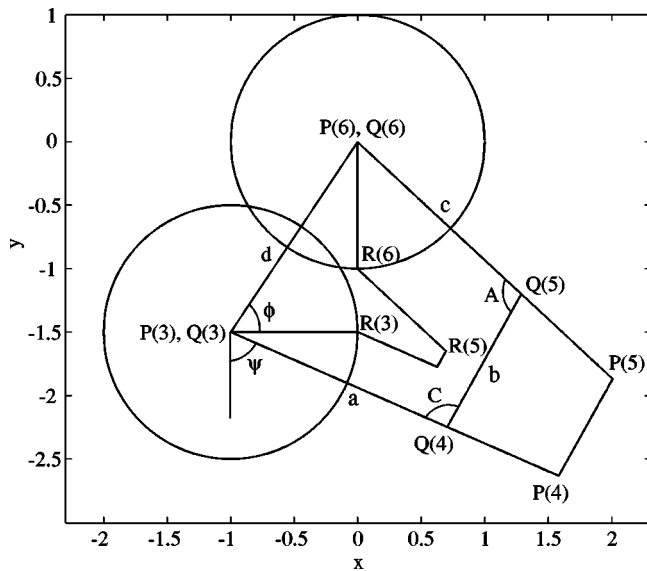
Fig. 10 A spherical triangle maneuver for reconfiguration

Also, $R(2)$ is located on the equatorial circle at an angle of $-90°$ with respect to the x axis. The orientation of $R(2)$ can be verified from Eq. (12)

$$\beta_i = (\alpha_0 + \phi_0) = -60° - 30° = -90° \qquad (25)$$

The reconfiguration problem now requires execution of the (A)-(B)-(A) sequence of control actions that will translate the sphere by distance r, and rotate it about the vertical by angle $\Delta\beta$. The magnitude of r and $\Delta\beta$ can be computed from Eq. (13) as

$$r = (x_i^2 + y_i^2)^{1/2} = 7.07, \quad \Delta\beta = (\beta_f - \beta_i) = 90° \qquad (26)$$

We choose a counter-clockwise path for the sphere under the (A)-(B)-(A) sequence of control actions. However, since there exists no solution for $\Delta\alpha$ (refer to Fig. 4) when $\Delta\beta = \pi/2$ rad and $r = 7.07$, we change $\Delta\beta$ to -3.5π rad by subtracting 4π rad from its actual value. This value of $\Delta\beta$ yields from Eqs. (8) and (9b)

$$\Delta\alpha = 2.56 \text{ rad} = 146.7°, \quad \theta' = 1.38 \text{ rad} = 79.1° \qquad (27)$$

The computed value of $\Delta\alpha$ in Eq. (27), which can be verified from Fig. 4, along with x_i and y_i values in Eq. (24) yields the coordinates of the center of rotation from Eq. (14)

$$a = 3.15, \quad b = 2.15 \qquad (28)$$

The values of x_i, y_i, a and b in Eqs. (24) and (28) yields $\alpha_i = 67.5°$ from Eq. (15).

The motion of the sphere under the (A)-(B)-(A) sequence of control actions is now defined completely. First, the sphere rolls in the direction $\alpha_i = 67.5°$ by $\theta' = 79.1°$. This brings points P, Q, and R to the locations $P(3)$, $Q(3)$, and $R(3)$, respectively. Next, with $(a, b) \equiv (3.15, 2.15)$ as the center of rotation, the sphere moves along a circular path for a net angular displacement of $\Delta\alpha = 146.7°$. At the end of this motion, the three points are located at $P(4)$, $Q(4)$, and $R(4)$, respectively. Finally, the sphere rolls to the origin of the x-y coordinate frame. This completes the reconfiguration which is evident from the coordinates $P(5)$, $Q(5)$, and $R(5)$ in Fig. 9.

5.2 Spherical Trigonometric Motion Planner. The spherical trigonometric motion planner, which was presented in Section 4, is comprised of three steps. However, since the first two steps are trivial, we simulate only the third step.

The initial and final configurations for the third step are chosen such that

$$d = 1.8, \quad \phi = 56.31°, \quad \alpha = -90° \qquad (29)$$

This implies that if P is the top most point on the sphere and R is on the equatorial circle in the direction of the x axis at the initial point of time, P should be again at the top in the final configuration, and R should be on the equatorial circle in the direction of the negative y axis. If we denote the center of the sphere by Q, the coordinates of Q at the initial time can be computed from the values of d and ϕ as $Q = (-1.0, -1.5)$. We choose a left-handed path since α is negative and numerically compute the triangle parameters using Eqs. (17), (18), (21a), and (22). Using a value of $66.31°$ for ψ (the value of ψ can be chosen arbitrarily; we chose it based on the value of ϕ) in Eq. (18), the spherical triangle parameters were obtained numerically

$$a = 1.862, \quad b = 1.198, \quad c = 1.760, \quad A = 103.81°,$$
$$B = 70.78°, \quad C = 95.41° \qquad (30)$$

where the units of the angles are degrees, and that of the sides are meters or radians since the sphere has a unit radius. The triangle parameters and the angle ψ are used to depict the motion of the sphere in Fig. 10. In conformity with our discussion in Section 4.2, the configuration of the sphere at the start of the spherical triangle maneuver is depicted by points $P(3)$, $Q(3)$, and $R(3)$. At the end of the maneuver, the configuration is depicted by points $P(6)$, $Q(6)$, and $R(6)$. Indeed, in the final configuration the sphere is at the origin, point P is at the top, and point R is on the equatorial circle in the direction of the negative y axis.

6 Conclusion

We presented two open-loop control strategies for reconfiguration of the rolling sphere. The reconfiguration problem, which refers to the task of converging the sphere to an arbitrary position with an arbitrary orientation, is challenging since the kinematic model of the sphere cannot be reduced to chained form; this renders all regimented nonholonomic motion planning algorithms inapplicable. Using Euler angle coordinates for description of sphere orientation and choosing the final orientation to lie at the singularity of the description, the first control strategy provides a simple and effective solution. It also simplifies the reachability problem, which can be ascertained from a reduced-order system. The algorithm invokes alternating inputs for reconfiguration and the resulting solution trajectories are comprised of simple circular arcs and straight line segments. Our second control strategy is conceptually simpler and results in trajectories comprised of straight line segments only. Based on the Gauss-Bonet theorem of parallel transport, our second algorithm achieves reconfiguration using spherical triangle maneuvers, and by not restricting the choice to equatorial triangles, provides a generalization of the approach by Li and Canny [7]. As compared to the algorithm by Jurdjevic [8], which requires elliptic functions and integrals to be computed at every step, both our algorithms are simpler, computationally efficient, and easier to implement. Our first algorithm additionally provides the scope for design of feedback control strategies. The algorithms by Li and Canny [7] and Jurdjevic [8] do not lend themselves well to this important problem.

Acknowledgment

References

[1] Brown, H. B., and Xu, Y., 1997, "A Single-Wheel Gyroscopically Stabilized Robot," IEEE Rob. Autom. Mag., **4**(3), pp. 39–44.
[2] Koshiyama, A., and Yamafuji, K., 1993, "Design and Control of All-Direction Steering Type Mobile Robot," Int. J. Robot. Res., **12**(5), pp. 411–419.
[3] Halme, A., Schonberg, T., and Wang, Y., 1996, "Motion Control of a Spherical Mobile Robot," *Proc. of 4th Int. Workshop on Advanced Motion Control* (AMC), Mie Univ., Japan.
[4] Bicchi, A., Balluchi, A., Prattichizzo, D., and Gorelli, A., 1997, "Introducing the Sphericle: An Experimental Testbed for Research and Teaching in Non-

holonomy,'' *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 2620–2625.

[5] Murray, R. M., and Sastry, S. S., 1993, ''Nonholonomic Motion Planning: Steering Using Sinusoids,'' IEEE Trans. Autom. Control, **38**(5), pp. 700–713.

[6] Bicchi, A., Prattichizzo, D., and Sastry, S. S., 1995, ''Planning Motions of Rolling Surfaces,'' *Proc. of 34th IEEE Int. Conf. on Decision and Control*, pp. 2812–2817.

[7] Li, Z., and Canny, J., 1990, ''Motion of Two Rigid Bodies with Rolling Constraint,'' IEEE Trans. Rob. Autom., **6**(1), pp. 62–72.

[8] Jurdjevic, V., 1993, ''The Geometry of the Plate-Ball Problem,'' Arch. Ration. Mech. Anal., **124**, pp. 305–328.

[9] Brockett, R. W., and Dai, L., 1991, ''Nonholonomic Kinematics and the Role of Elliptic Functions in Constructive Controllability,'' *Workshop on Nonholonomic Motion Planning, IEEE Int. Conf. on Robotics and Automation.*

[10] Mukherjee, R., Emond, B. R., and Junkins, J. L., 1997, ''Optimal Trajectory Planning for Mobile Robots using Jacobian Elliptic Functions,'' Int. J. Robot. Res., **16**(6), pp. 826–839.

[11] Levi, M., 1993, ''Geometric Phases in the Motion of Rigid Bodies,'' Arch. Ration. Mech. Anal., **122**(3), pp. 213–229.

[12] Mukherjee, R., and Pukrushpan, J. T., 2000, ''Class of Rotations Induced by Spherical Polygons,'' AIAA J. Guid. Control, **23**(4), pp. 746–749.

[13] Kells, L. M., Kern, W. F., and Bland, J. R., 1940, *Plane and Spherical Trigonometry*, McGraw Hill, New York, NY.