Sensors & Transducers, Vol. 21, Special Issue, May 2013, pp. 42-47

www.sensorsportal.com

Sensors & Transducers

http://www.sensorsportal.com

Path Following Control of a Spherical Robot Rolling on an Inclined Plane

Tao Yu, Hanxu Sun, Qingxuan Jia, Wei Zhao

School of Automation, Beijing University of Posts and Telecommunications, 100876, Beijing, China
Tel.: +861062282571, fax: +861062282571
E-mail: yutaogzm@sohu.com

Received: 31 March 2013 /Accepted: 14 May 2013 /Published: 30 May 2013

Abstract: In this paper, the path following control of a spherical robot rolling without slipping on an inclined plane is discussed. We first study the kinematic constraints of the spherical robot and develop the dynamic model of the robot through the constrained Lagrange method. We then present a state space realization of this constrained system through the null space method and using nonlinear feedback. We investigate the path following control algorithms and develop a variable structure approach to the control of this nonholonomic system. By choosing appropriate output equations for path following, we design the sliding surfaces as the functions of the output tracking errors. Using Lyapunov stability theorem and exponential reaching law, we derive the sliding mode control law. The asymptotic stability of the sliding surfaces is theoretically proved, and the validity of the proposed path following method is further validated through MATLAB simulations. *Copyright © 2013 IFSA.*

Keywords: Spherical robot, Inclined plane, Path following, Sliding mode control, Nonholonomic system.

1. Introduction

Most mobile robots we have today have wheels. That is an obvious choice as there is a considerable amount of knowledge about this type of locomotion. However, more and more possible applications occur where wheeled robots have some flaws. Spherical robots could be a solution to some of these problems. As the robot is encompassed in a ball it is possible to effectively seal everything to enable the robot to withstand exposure to dust, humidity, dangerous substances and other environmental threats. As we can understand, this could be very handy in such applications as planetary exploration, surveillance and others. The above mentioned situations often involve dealing with difficult terrain as well. While wheeled robots can cope with it pretty good, the risk of falling over still persists. A spherical robot, on the other hand, can't fall over at all.

Over the last few decades, there has been considerable interest in the development of powerful methods for motion control of mobile robots. The problems addressed in the literature can be roughly classified into three groups: trajectory tracking, path following and point stabilization [1]. With respect to spherical robots, there have not been established methodologies to resolve these control problems, although many studies have been made during the past. Alves and Dias [2] presented a line tracking method of a spherical robot based on kinematics. Zhan and Liu et al. [3] discussed the trajectory tracking problem of a spherical robot using backstepping approach. Zheng and Zhan et al. [4] investigated the trajectory tracking algorithm for a

Article number P_SI_339

spherical robot based on the RBF-PD controller. Cai and Zhan et al. [5] proposed a real-time fuzzy guidance scheme for trajectory tracking of a spherical robot. Liu and Sun et al. [6] developed a line following controller for a spherical robot based on sliding modes.

Current researches on motion control of spherical robots usually assume that the robot remains strictly on a level plane. As a result, the dynamic model fails to represent the actual motion when the robot rolls up a slope. This paper focuses on practical solutions to trajectory tracking and path following control of a spherical robot rolling on an inclined plane. The main contributions of the paper include two parts. Firstly, the kinematics and dynamics of the robot subject to no-slip and no-spin constraints are derived. Secondly, a sliding mode control scheme for path following of the robot is proposed.

2. Mathematical Model

2.1. System Description

BYQ-VIII is a pendulum-driven spherical robot, the mechanical structure of the robot is illustrated in Fig. 1. The robot is mainly consisted of three parts: the spherical shell 1, the internal gimbals 2 and the pendulum 3. The robot has the internal driving unit mounted inside the spherical shell. The steering motion of the robot is achieved by tilting the pendulum, and the driving motion is performed by swinging the pendulum indirectly through the internal gimbal.

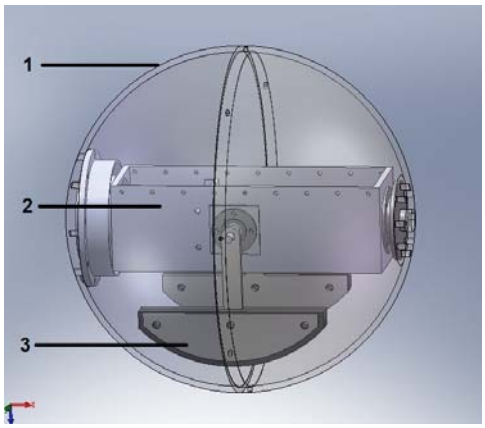

Fig. 1. Three-dimensional model of BYQ-VIII.

2.2. Kinematic Constraints

We first assign two coordinate frames. Let $\sum_O XYZ$ be a fixed inertial frame whose XY plane is anchored to the surface of the incline and Z is the vertical position to the surface. Let $\sum_B X_b Y_b Z_b$ be the body coordinate frame whose origin is located at the center of the sphere B. We denote (ϕ, θ, ψ) to be the ZYX Euler angles from the inertial frame $\sum_O$ to the body coordinate frame $\sum_B$.

Fig. 2 presents the geometrical model of the rolling sphere and definition of necessary variables to deduce the mathematical model. Here R represents the radius of the sphere, C is the contact point between the sphere and slope surface with its coordinates (x_c, y_c) with respect to $\sum_O$, τ_ψ and τ_θ denote the torques exerted on the sphere along the axis X_b and Y_b respectively.

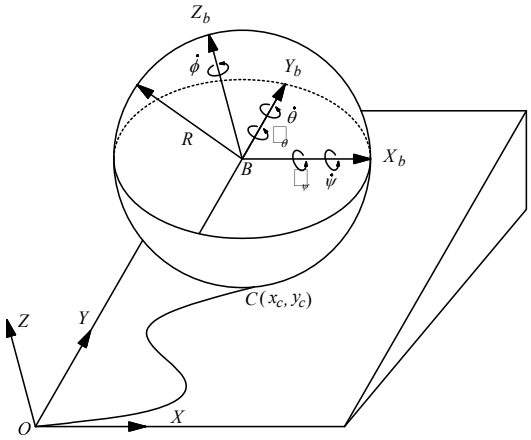

Fig. 2. Definition of system variables for BYQ-VIII.

We now derive the kinematics and dynamics of the spherical robot on the basis of the following assumptions: (i) no-slip constraint: the sphere rolls on the perfectly flat surface of the incline without slipping, (ii) no-spin constraint: rotations of the rolling sphere around the Z axis are not allowed.

Let $\boldsymbol{v}_B$ and $\boldsymbol{\omega}_B$ denote the velocity of the center of mass of the sphere and its angular velocity with respect to the inertia frame $\sum_O$. Then, we have

$$\begin{aligned} \boldsymbol{\omega}_B &= \omega_x \boldsymbol{i} + \omega_y \boldsymbol{j} + \omega_z \boldsymbol{k} \\ \omega_x &= \dot{\psi}\cos\phi\cos\theta - \dot{\theta}\sin\phi \\ \omega_y &= \dot{\theta}\cos\phi + \dot{\psi}\sin\phi\cos\theta \\ \omega_z &= \dot{\phi} - \dot{\psi}\sin\theta \end{aligned}, \tag{1}$$

where $\boldsymbol{i}$, $\boldsymbol{j}$, $\boldsymbol{k}$ are the unit vectors of $\sum_O$.

The no-spin constraint can be formulated as

$$\dot{\phi} - \dot{\psi}\sin\theta = 0 \tag{2}$$

The constraint in (2) represents a nonholonomic constraint. The constraints result from the requirement that the sphere rolls without slipping on the incline, i.e., the velocity of the contact point on the sphere is zero at any instant $\boldsymbol{v}_C = 0$. Now we can express $\boldsymbol{v}_B$ as

$$\boldsymbol{v}_B = \boldsymbol{\omega}_B \times r_{BC} + \boldsymbol{v}_C, \tag{3}$$

where $r_{BC} = -R\boldsymbol{k}$ represents the vector from point C to B. Substituting (1) into (3) gives

$$\boldsymbol{v}_B = \dot{x}_c\boldsymbol{i} + \dot{y}_c\boldsymbol{j} + \dot{Z}\boldsymbol{k} \tag{4}$$

where

$$\dot{x}_c + R\left(\dot{\theta}\cos\phi + \dot{\psi}\sin\phi\cos\theta\right) = 0 \tag{5}$$

$$\dot{y}_c - R\left(\dot{\psi}\cos\phi\cos\theta - \dot{\theta}\sin\phi\right) = 0 \tag{6}$$

$$\dot{Z} = 0 \tag{7}$$

The constraints in (5) and (6) are nonholonomic, whereas the constraint in (7) is holonomic and can be integrated to obtain $Z = R$.

Therefore the configuration of the robotic system can be described by a vector of five generalized coordinates $\boldsymbol{q} = (x_c, y_c, \phi, \theta, \psi)^T$.

Combing (2), (5) and (6), the nonholonomic constraints can be written as

$$\boldsymbol{A}(\boldsymbol{q})\dot{\boldsymbol{q}} = \boldsymbol{0} \tag{8}$$

where

$$\boldsymbol{A}(\boldsymbol{q}) = \begin{bmatrix} 1 & 0 & 0 & R\cos\phi & R\sin\phi\cos\theta \\ 0 & 1 & 0 & R\sin\phi & -R\cos\phi\cos\theta \\ 0 & 0 & 1 & 0 & -\sin\theta \end{bmatrix}$$

2.3. Robot Dynamics

We study the motion equations by calculating the Lagrangian $L = T - P$ of the system, where T and P are the kinetic energy and potential energy of the system respectively. The spherical shell is assumed to have mass m and the moment of inertia I. Then T and P of the robotic system can be calculated as follows

$$\begin{aligned} T &= \frac{1}{2}m\left(\dot{x}_c^2 + \dot{y}_c^2\right) + \frac{1}{2}I\left(\omega_x^2 + \omega_y^2 + \omega_z^2\right) \\ P &= mg\left(y_c\sin\gamma + R\cos\gamma\right) \end{aligned}, \tag{9}$$

where γ is the inclination angle of the slope.

Using the constrained Lagrangian method, the motion equations of the robotic system are described by

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{V}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{E}(\boldsymbol{q})\boldsymbol{t} - \boldsymbol{A}^{\mathrm{T}}(\boldsymbol{q})\boldsymbol{\lambda} \tag{10}$$

where

$$\boldsymbol{M}(\boldsymbol{q}) = \begin{bmatrix} m & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 \\ 0 & 0 & I & 0 & -I\sin\theta \\ 0 & 0 & 0 & I_s & 0 \\ 0 & 0 & -I\sin\theta & 0 & I \end{bmatrix}$$

$$\boldsymbol{V}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \begin{bmatrix} 0 \\ mg\sin\gamma \\ -I\dot{\theta}\dot{\psi}\cos\theta \\ I\dot{\phi}\dot{\psi}\cos\theta \\ -I\dot{\phi}\dot{\theta}\cos\theta \end{bmatrix} \quad \boldsymbol{E}(\boldsymbol{q}) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 & \lambda_2 \end{bmatrix}^T \quad \boldsymbol{t} = \begin{bmatrix} t_\theta & t_\psi \end{bmatrix}^T$$

To eliminate the Lagrange multipliers [7], we first partition $\boldsymbol{A}(\boldsymbol{q})$ as $\boldsymbol{A} = [\boldsymbol{A}_1 \mid \boldsymbol{A}_2]$, where

$$\boldsymbol{A}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \boldsymbol{A}_2 = \begin{bmatrix} R\cos\phi & R\sin\phi\cos\theta \\ R\sin\phi & -R\cos\phi\cos\theta \\ 0 & -\sin\theta \end{bmatrix}$$

Let

$$\boldsymbol{C}(\boldsymbol{q}) = \begin{bmatrix} -\boldsymbol{A}_1^{-1}\boldsymbol{A}_2 \\ \boldsymbol{I}_{2\times 2} \end{bmatrix} \tag{11}$$

where $\theta \neq \frac{\pi}{2} \pm k\pi$, $(k = 1,2,\ldots)$.

It is straightforward to verify that $\boldsymbol{C}(\boldsymbol{q})$ satisfies that $\boldsymbol{A}(\boldsymbol{q})\boldsymbol{C}(\boldsymbol{q}) = \boldsymbol{0}$. If we choose θ and ψ to be the two quasi-coordinates,

$$\boldsymbol{\upsilon}(t) = \begin{bmatrix} \upsilon_1 \\ \upsilon_2 \end{bmatrix} = \begin{bmatrix} \dot{\theta} \\ \dot{\psi} \end{bmatrix}$$

and we can verify that (12) is satisfied.

$$\dot{\boldsymbol{q}} = \boldsymbol{C}(\boldsymbol{q})\boldsymbol{\upsilon}(t) \tag{12}$$

Differentiating (12) yields

$$\ddot{\boldsymbol{q}} = \dot{\boldsymbol{C}}\boldsymbol{\upsilon} + \boldsymbol{C}\dot{\boldsymbol{\upsilon}} \tag{13}$$

Using (10) and (11), we have $\boldsymbol{C}^T\boldsymbol{E} = I$. Substituting (13) into (10), and premultiplying both sides by C^T gives

$$\boldsymbol{C}^{\mathrm{T}}\boldsymbol{M}\boldsymbol{C}\dot{\boldsymbol{\upsilon}}(t) + \boldsymbol{C}^{\mathrm{T}}\boldsymbol{M}\dot{\boldsymbol{C}}\boldsymbol{\upsilon}(t) + \boldsymbol{C}^{\mathrm{T}}\boldsymbol{V} = \boldsymbol{t} \tag{14}$$

Using the state variable $x = (q^T, \upsilon^T)^T$, we have

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \boldsymbol{C}\boldsymbol{\upsilon} \\ \boldsymbol{f}_1 \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} \\ \bar{\boldsymbol{M}}^{-1} \end{bmatrix}\boldsymbol{t}, \tag{15}$$

where $\bar{\boldsymbol{M}} = \boldsymbol{C}^T\boldsymbol{M}\boldsymbol{C}$, $\boldsymbol{f}_1 = -\bar{\boldsymbol{M}}^{-1}\left(\boldsymbol{C}^T\boldsymbol{M}\dot{\boldsymbol{C}}\boldsymbol{\upsilon} + \boldsymbol{C}^T\boldsymbol{V}\right)$.

We apply the following nonlinear feedback [8]

$$\boldsymbol{t} = \bar{\boldsymbol{M}}\left(\boldsymbol{u} - \boldsymbol{f}_1\right), \tag{16}$$

where $\boldsymbol{u} = (u_1, u_2)^T$ represent the new control inputs.

Then the state equation simplifies to the form

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{b}(\boldsymbol{x})\boldsymbol{u} \tag{17}$$

where

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{C}(\boldsymbol{q})\boldsymbol{\upsilon} \\ \boldsymbol{0} \end{bmatrix} \quad \boldsymbol{b}(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{I} \end{bmatrix}$$

3. Path Following

3.1. Controller Design

In the path following task, the controller is given a geometric description of the assigned Cartesian path. For this task, time dependence is not relevant because one is concerned only with the geometric displacement between the robot and path. The path following problem is rephrased as the stabilization to zero of a suitable scalar path error function. Since the robotic system has two control inputs, we may choose two output variables. By appropriately choosing the output variables h_1 and h_2 we can achieve path following control.

Suppose the reference path is

$$\tilde{f}(x_c, y_c) = 0 \tag{18}$$

We define the path error function e_f as

$$e_f = \tilde{f}(x_c, y_c) \tag{19}$$

Then h_1 can be chosen as

$$h_1(\boldsymbol{q}) = e_f = \tilde{f}(x_c, y_c) \tag{20}$$

The other output variable h_2 is chosen to be one of the quasi-velocities of the robotic system.

$$h_2(\boldsymbol{\upsilon}) = \dot{\psi} = \upsilon_2 \tag{21}$$

Differentiating (20) once and twice respectively gives

$$\begin{aligned} \dot{y}_1 &= \frac{h_1(\boldsymbol{q})}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}} = \boldsymbol{J}_{h_1}(\boldsymbol{q})\boldsymbol{C}(\boldsymbol{q})\boldsymbol{\upsilon} \\ \ddot{y}_1 &= \frac{\partial\left(\boldsymbol{J}_{h_1}\boldsymbol{C}\boldsymbol{\upsilon}\right)}{\partial \boldsymbol{q}}\boldsymbol{C}\boldsymbol{\upsilon} + \boldsymbol{J}_{h_1}(\boldsymbol{q})\boldsymbol{C}(\boldsymbol{q})\boldsymbol{u} \end{aligned} \tag{22}$$

where $\boldsymbol{J}_{h_1} = \begin{bmatrix} \dfrac{\partial \tilde{f}}{\partial x_c} & \dfrac{\partial \tilde{f}}{\partial y_c} & 0 & 0 & 0 \end{bmatrix}$.

Differentiating (21) yields

$$\dot{y}_2 = \frac{h_2(\boldsymbol{\upsilon})}{\partial \boldsymbol{\upsilon}}\dot{\boldsymbol{\upsilon}} = \boldsymbol{J}_{h_2}\boldsymbol{u} \tag{23}$$

where $\boldsymbol{J}_{h_2} = \begin{bmatrix} 0 & 1 \end{bmatrix}$.

We define the following sliding surfaces

$$\begin{aligned} s_1 &= \dot{y}_1 + c_1 y_1 \\ s_2 &= y_2 - y_{2d} \end{aligned} \tag{24}$$

where c_1 is a real positive constant, y_{2d} is the desired value for y_2.

Let $\boldsymbol{S}_1$ be the vector of components s_1 and s_2.

$$\boldsymbol{S}_1 = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} \dot{y}_1 + c_1 y_1 \\ y_2 - y_{2d} \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}_{h_1}\boldsymbol{C}\boldsymbol{\upsilon} + c_1 y_1 \\ y_2 - y_{2d} \end{bmatrix} \tag{25}$$

Differentiating (25) yields

$$\dot{\boldsymbol{S}}_1 = \boldsymbol{f}_2 + \boldsymbol{g}\boldsymbol{u} \tag{26}$$

where

$$\boldsymbol{g} = \begin{bmatrix} \boldsymbol{J}_{h_1}\boldsymbol{C} \\ \boldsymbol{J}_{h_2} \end{bmatrix} \quad \boldsymbol{f}_2 = \begin{bmatrix} \dfrac{\partial\left(\boldsymbol{J}_{h_1}\boldsymbol{C}\boldsymbol{\upsilon}\right)}{\partial \boldsymbol{q}}\boldsymbol{C}\boldsymbol{\upsilon} + c_1\boldsymbol{J}_{h_1}\boldsymbol{C}\boldsymbol{\upsilon} \\ 0 \end{bmatrix}$$

Let

$$\boldsymbol{\eta}\,\mathrm{sgn}(\boldsymbol{S}_1) = \begin{bmatrix} \eta_1 \,\mathrm{sgn}(s_1) & \eta_2 \,\mathrm{sgn}(s_2) \end{bmatrix}^T \tag{27}$$

where η_1, η_2 are real positive constants.

We choose the control law $\boldsymbol{u}$ as follows

$$\boldsymbol{u} = \boldsymbol{g}^{-1}\left[-\boldsymbol{\eta}\,\mathrm{sgn}(\boldsymbol{S}_1) - \boldsymbol{K}\boldsymbol{S}_1 - \boldsymbol{f}_2\right] \tag{28}$$

where $\boldsymbol{K} = \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix}$, k_1 and k_2 are real positive constants.

3.2. Stability Analysis

Theorem 1: Suppose that the system in (17) is controlled by the control law given by (28), then the sliding surfaces s_1, s_2 are asymptotically stable.

Proof: Substituting (28) into (26) yields

$$\begin{aligned} \dot{s}_1 &= -\eta_1 \,\mathrm{sgn}(s_1) - k_1 s_1 \\ \dot{s}_2 &= -\eta_2 \,\mathrm{sgn}(s_2) - k_2 s_2 \end{aligned} \tag{29}$$

Consider the Lyapunov function candidates

$$V_i = \frac{1}{2}s_i^2, \quad i = 1,2$$

Differentiating V_i with respect to time gives

$$\dot{V}_i = s_i \dot{s}_i = -\eta_i |s_i| - k_i s_i^2 \le 0 \tag{30}$$

Integrating both sides of (30), we can obtain

$$\begin{aligned} V_i(t) &= \frac{1}{2} s_i^2 \le V_i(0) < \infty \\ \lim_{t\to\infty} \int_0^t \left(\eta_i |s_i| + k_i s_i^2\right) d\sigma &\le V_i(0) < \infty \end{aligned} \tag{31}$$

From (31), we have $s_i \in L_\infty$, $s_i \in L_2$. From (30), we have $\dot{s}_i \in L_\infty$. Consequently, according to Babalat's lemma we have $\lim_{t\to\infty} s_i = 0,\ i = 1, 2$.

4. Simulation Study

We developed a computer simulation in order to verify the validity of the control algorithms discussed in the previous sections. The dimensions and inertial parameters are representative of the spherical robot. According to the notation introduced before:
$m = 0.85$ kg, $R = 0.14$ m, $I = 0.0111$ kg·m^2, $\gamma = 10$ º.

4.1. Basic Paths

Consider a straight line path $y = x$, as shown in Fig. 3. The initial values of the system configuration variables are such that

$$(x_c, y_c, \phi, \theta, \psi) = (0.4, 0.2, 0, 0, 0)$$

and the initial velocity is zero. For the path following algorithm,

$$h_1 = x_c - y_c,\ y_{2d} = 2.5$$

with the controller parameters

$$c_1 = 4.8,\ k_1 = 4.3,\ k_2 = 4.5,\ \eta_1 = 2.6,\ \eta_2 = 2.2$$

The performance of the robot in straight line path following is shown in Fig. 3. The star marker indicates the starting point of the robot.

Consider a circular path

$$(x - 0.8)^2 + (y - 1)^2 = 0.25$$

as shown in Fig. 4. The initial values of the system configuration variables are such that

$$(x_c, y_c, \phi, \theta, \psi) = (1.2, 1.4, 0, 0, 0)$$

and the initial velocity is zero. For the path following algorithm,

$$h_1 = (x_c - 0.8)^2 + (y_c - 1)^2 - 0.25,\ y_{2d} = 2.5$$

with the controller parameters

$$c_1 = 4.9,\ k_1 = 4.5,\ k_2 = 1.3,\ \eta_1 = 2,\ \eta_2 = 1$$

The performance of the robot in circular path following is shown in Fig. 4.

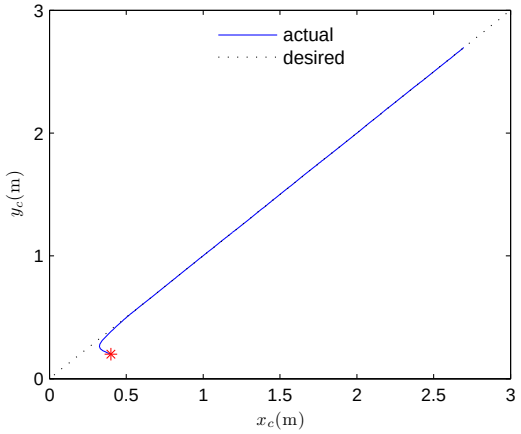

Fig. 3. Control performance in straight line path following.

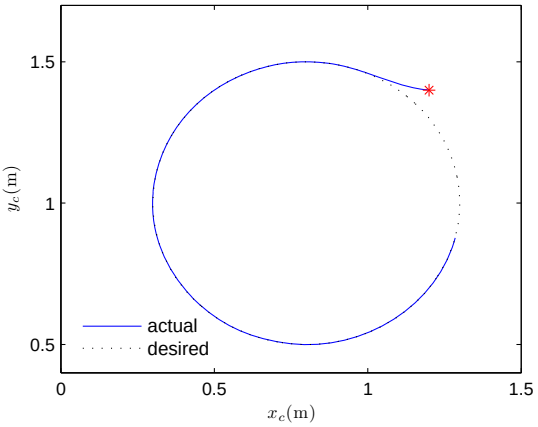

Fig. 4. Performance of the robot in circular path following.

In both cases, the reference point of the robot is able to reach the path and stay on the path. The path following algorithm seems to exhibit a gradual merge, with the path following controller the actual path followed is smooth.

4.2. Piecewise Continuous Paths

An example of a composite path is shown in Fig. 5, which is composed of two circular arcs and a straight line. The performance of the control system is acceptable as seen from the figure. The discontinuities in curvature are negotiated without any difficulty and there is almost no deviation of the actual path from the desired one.

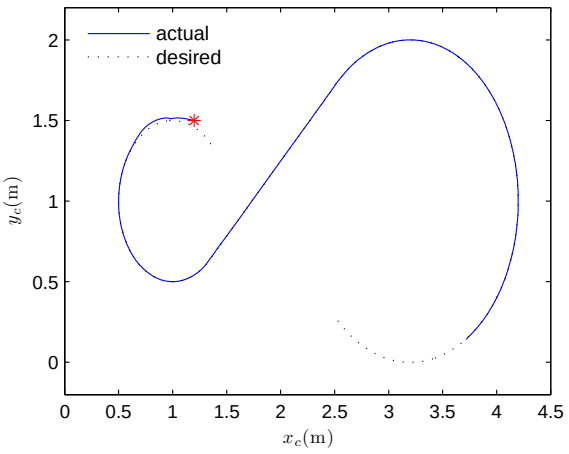

Fig. 5. Control performance in composite path following.

5. Conclusions

We presented a variable structure method for path following control of a spherical robot moving on an inclined plane. We derived the kinematics by imposing the constraint conditions of no-slip and no-spin. We deduced the robot dynamics using the constrained Lagrange formulation. We eliminated the Lagrange multipliers to obtain a state space description of the system. We devised a sliding mode scheme to achieve output tracking. We considered the choice of output variables for path following and derived the sliding mode control laws to satisfy the existence condition of sliding modes. Computer simulation results were presented to illustrate the performance of the proposed control algorithm.

Acknowledgements

The authors wish to acknowledge the financial support provided by the National Natural Science Foundation of China (No. 51175048), the Cultivation Fund of the Key Scientific and Technical Innovation Project, Ministry of Education of China (No. 708011) and Beijing Natural Science Foundation Program and Scientific Research Key Program of Beijing Municipal Commission of Education (No. KZ200810005002).

References

[1]. Soeanto D., Lapierre L., Pascoal A., Adaptive non-singular path-following control of dynamic wheeled robots, in *Proceedings of 42nd IEEE Conference on Decision and Control*, 2, 2003, pp. 1765-1770.

[2]. Alves J., Dias J., Design and control of a spherical mobile robot, in *Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering*, 217, 2003, No. 6, pp. 457-467.

[3]. Liu Z., Zhan Q., Cai Y., Motion control of a spherical mobile robot for environment exploration, *Acta Aeronautica Et Astronautica Sinica*, 29, 2008, No. 6, pp. 1673-1678.

[4]. Zheng M., Zhan Q., Liu J., Cai Y., Trajectory tracking of a spherical robot based on an RBF neural network, in *Proceedings of the IEEE International Conference on Electrical Engineering and Automatic Control*, 7, 2010, pp. 151-154.

[5]. Cai Y., Zhan Q., Xi X., Path tracking control of a spherical mobile robot, *Mechanism and Machine Theory*, 51, 2012, pp. 58-73.

[6]. Liu D., Sun H., Jia Q., Stabilization and path following of a spherical robot, in *Proceedings of the IEEE International Conference on Robotics, Automation and Mechatronics*, 2008, pp. 676-682.

[7]. Bloch A. M., Reyhanoglu M., McClamroch N. H., Control and stabilization of nonholonomic dynamic systems, *IEEE Trans. Aut. Control*, 37, 1992, No. 11, pp. 1746-1757.

[8]. Sarkar N., Yun X., Kumar V., Control of mechanical systems with rolling constraints: application to dynamic control of mobile robots, *International Journal of Robotics Research*, 13, 1994, No. 1, pp. 55-69.