CONTROL AND STABILIZATION OF A PENDULUM-DRIVEN SPHERICAL MOBILE ROBOT ON AN INCLINED PLANE: I-SAIRAS 2012

TURIN, ITALY
4-6 SEPTEMBER 2012

Tao Yu (1), Hanxu Sun (1), Yanheng Zhang (2), Wei Zhao (1)

(1) Beijing University of Posts and Telecommunications, Room 202, HongTong Building, No.10 Xitucheng Road, Haidian District, 100876, Beijing, P.R. China, Email: yutaolanjie@163.com
(2) Beijing University of Posts and Telecommunications, Mailbox 191, No.10 Xitucheng Road, Haidian District, 100876, Beijing, P.R. China, Email: hxsun@bupt.edu.cn

ABSTRACT

In this paper, the dynamics and control aspects of a pendulum-driven spherical mobile robot rolling without slipping along an inclined plane are investigated. The planar dynamic model of the robot rolling ahead on an inclined plane is derived using Lagrangian dynamics. Based on the equivalent control methodology and Lyapunov stability theorem, a decoupled sliding mode control approach is then presented for stable control of the planar motion. Utilizing the constrained Lagrange method the 3-D dynamics of the robot is developed. Then based on input-output feedback linearization we develop a control algorithm for stabilizing the robot to track a desired trajectory. The validity of the proposed controllers is then validated through simulation study.

1. INTRODUCTION

Most mobile robots we have today have wheels. That is an obvious choice as there is considerable amount of knowledge about this type of locomotion. However, more and more possible applications occur where wheeled robots have some flaws. Spherical mobile robots or robots that resemble a ball could be a solution to some of these problems. As the robot is in fact encompassed in a ball it is possible to effectively seal everything to enable the robot to withstand exposure to dust, dangerous substances, humidity or 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 mobile robot, on the other hand, can't fall over at all. Also, it can be quite a task to deploy a wheeled robot without direct human intervention - the landing spot has to be carefully chosen, the robot has to land with wheels down, etc. However a robotic ball can be simply dropped out above the desired position. BYQ-VIII is a novel spherical mobile robot using a pendulum-based design, and the structure of the robot is shown in Figure 1. The robot has the internal driving unit mounted inside the spherical shell. The robot is steered by tilting the pendulum and driven by swinging the pendulum indirectly through the internal gimbal.

The outline of this paper is as follows. In Section 2, the planar dynamics and control schemes of BYQ-VIII are discussed. In Section 3, the 3-D dynamics and stabilization methods of the robot are investigated. We carry out a simulation study to verify the effectiveness of the proposed controllers in Section 4. Finally, the conclusion is given in Section 5.

Figure 1. The structure of BYQ-VIII
The idealized planar model of the spherical mobile robot is shown in Fig. 2. In addition, the definitions for the model parameters are listed in Table 1.

![Figure 2. Simplified planar model of the robot](image)

**Table 1. Parameter definitions for the planar model**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
<td>mass of the outer shell</td>
</tr>
<tr>
<td>$m$</td>
<td>mass of the pendulum</td>
</tr>
<tr>
<td>$I$</td>
<td>moment of inertia of the shell</td>
</tr>
<tr>
<td>$l$</td>
<td>length of the link</td>
</tr>
<tr>
<td>$R$</td>
<td>radius of the shell</td>
</tr>
<tr>
<td>$g$</td>
<td>gravitational acceleration</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>slope inclination angle</td>
</tr>
<tr>
<td>$\tau$</td>
<td>torque applied to the pendulum</td>
</tr>
</tbody>
</table>

We assume that the mass of the link is negligible relative to the shell and eccentric mass, and we also model the eccentric mass as a point instead of a rigid body. We impose the assumptions to reduce the system to two coordinates: $\theta$ for sphere rotation and $\phi$ for pendulum rotation angle. Using the Lagrange equation, the dynamics of the planar motion of the robot can be described as

$$
\begin{align*}
\begin{bmatrix}
(M + m)r^2 + I & mrI \cos(\phi - \gamma) \\
MrI \cos(\phi - \gamma) & ml^2
\end{bmatrix} \begin{bmatrix}
\dot{\theta} \\
\dot{\phi}
\end{bmatrix} + 
\begin{bmatrix}
(M + m)gr\sin\gamma - mrl\sin(\phi - \gamma) \phi^2
\end{bmatrix} = 
\begin{bmatrix}
\tau \\
\tau
\end{bmatrix}
\end{align*}
$$

(1)

Considering the dynamic equation in (1), two states of equilibrium can be easily derived. First, consider the case in which the robot sits stationary on the slope. In this case, all angular velocities and accelerations reduce to zeros. When we enforce this condition upon (1), they reduce to

$$
(M + m)gr\sin\gamma = mgl\sin\phi_e
$$

(2)

We use the notation that $\phi_e$ denotes the equilibrium value of $\phi$. It stands to reason that there exists a limiting value of the slope $\gamma$ after which the robot will be incapable of holding its position. To determine this operational boundary we solve for $\phi_e$ in the above equation. The resulting solution is

$$
\phi_e = \arcsin\left(\frac{(M + m)r\sin\gamma}{ml}\right)
$$

(3)

Clearly, $\gamma$ must be bounded above and below to ensure an inverse sine operand less than unity:

$$
-arcsin\left(\frac{ml}{(M + m)r}\right) \leq \gamma \leq arcsin\left(\frac{ml}{(M + m)r}\right)
$$

(4)

The bounds associated with (4) correspond to stable node bifurcations at which the equilibrium solutions coalesce and disappear. This phenomenon is associated with the dynamic condition of whirling [1] in which the robot unsuccessfully attempts to either remain stationary or climb the slope.

The second equilibrium condition can be defined by assuming the robot maintains a constant velocity over constant slope terrain. This condition can also be satisfied by setting $\dot{\phi} = 0$. When this condition is enforced upon (1), we can find the equilibrium pendulum angle continues to satisfy (3).

**2.2. Control Design**

Consider the following coupling nonlinear systems which can be divided into two subsystems as

$$
\begin{align*}
A: \begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{bmatrix} = 
\begin{bmatrix}
f_1(x) \\
2b(x)u
\end{bmatrix}
\quad B: \begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{bmatrix} = 
\begin{bmatrix}
f_1(x) + h_1(x)u \\
h_2(x)
\end{bmatrix}
\end{align*}
$$

(5)

where $x = [x_1, x_2, x_3, x_4]^T$ is the state vector, $f_1(x)$, $h_1(x)$, $f_2(x)$, $b(x)$ are nonlinear functions, and $u$ is the control input.

Then we construct the following linear functions as sliding surfaces for the two subsystems [2-3]

$$
s_1 = c_1x_1 + x_2 
\quad s_2 = c_2x_2 + x_4
$$

(6)

where $c_1$ and $c_2$ are positive constants.

We define an intermediate variable $z$, which represents the information from subsystem $A$, and it is incorporated into $s_2$. Therefore, the sliding surface $s_2$ can be modified as

$$
s_2 = c_2(x_2 - z) + x_4
$$

(7)

Here the intermediate variable $z$ is related to $s_1$. For decoupling control, we define $z$ as

$$
z = \tanh\left(s_1/\Phi_z\right)z_U
$$

(8)

where $z_U$ is the upper bound of $\text{abs}(z)$, $0 < z_U < 1$, $\Phi_z$ is a positive constant, and $\tanh(\cdot)$ is the hyperbolic tangent function defined as follows

$$
\tanh(s_1/\Phi_z) = \frac{e^{s_1/\Phi_z} - e^{-s_1/\Phi_z}}{e^{s_1/\Phi_z} + e^{-s_1/\Phi_z}}
$$

(9)

Since $z_U$ is less than one, $z$ presents a decaying signal.

As $s_1$ decreases, $z$ decreases too. When $s_1 \to 0$, we have $z \to 0$, $x_1 \to 0$, and then $s_2 \to 0$, and the control objective will be achieved.

Differentiating (8), $\dot{z}$ can be calculated as
\[ \dot{z} = \alpha(s, \Phi, z) \cdot \dot{s} \]  

(10)

where

\[
\alpha(s, \Phi, z) = \frac{\text{sech}^2(s/\Phi)}{\Phi} \cdot z
\]

\[
\text{sech}(s/\Phi) = \frac{2}{e^{s/\Phi} + e^{-s/\Phi}}
\]

Differentiating (7) and using (10), we can calculate \( \dot{s} \) as

\[ \dot{s} = c_2 x_4 + f_2 - ac_2 (c_1 x_2 + f_1) + (b_2 - ac_2 b_1) u \]  

(11)

The equivalent control \( u_{eq} \) can be obtained from \( \dot{s} = 0 \), i.e.

\[ u_{eq} = -\frac{c_2 x_4 + f_2 - ac_2 (c_1 x_2 + f_1)}{b_2 - ac_2 b_1} \]  

(12)

We further assume the system control input \( u \) to have the following form

\[ u = u_{eq} + u_m \]  

(13)

Here \( u_m \) is the switching control. Define the Lyapunov function candidate \( V = \frac{1}{2} s^2 \) and differentiate \( V \) with respect to time, we have

\[ \dot{V} = s_2 \dot{s}_2 = s_2 (b_2 - ac_2 b_1) u_m \]  

(14)

Then we choose the switching control \( u_m \) as follows

\[ u_m = -\eta \text{sgn}(s_2) - ks_2 \]  

(15)

where \( \eta \) and \( k \) are positive constants.

**Theorem 1:** Suppose that the system (5) is controlled by the control input described in (12), (13) and (15), then the sliding surface \( s_2 \) is asymptotically stable.

**Proof:** Substituting (15) into (14), we can obtain

\[ \dot{V} = s_2 \dot{s}_2 \leq -\eta |s_2| - ks_2^2 \leq 0 \]  

(16)

Integrating both sides of (16), we have

\[ V(t) - V(0) \leq \int_{t_0}^{t} (\eta |s_2| - ks_2^2) d\sigma \]  

(17)

From (17), we can further obtain

\[ V(t) = \frac{1}{2} s_2^2 \leq V(0) < \infty \]  

(18)

\[ \lim_{t \to \infty} \int_{t_0}^{t} (\eta |s_2| + ks_2^2) d\sigma \leq V(0) < \infty \]

From (18), we have \( s_2 \in L_\infty \) and \( s_2 \in L_2 \). But from (16) we have \( \dot{V} = s_2 \dot{s}_2 < 0 \), then we can obtain \( \dot{s}_2 \in L_\infty \).

According to babalat’s lemma, we have \( \lim_{t \to \infty} s_2 = 0 \).

\section{3. 3-D DYNAMICS AND CONTROL}

\subsection{3.1. Kinematic Analysis}

In derivation of the 3-D motion equation for the robot, we first make the following assumptions: The shell is a rigid homogeneous sphere, which rolls on perfectly flat surface of an incline without slipping. The pendulum is composed of a massless link and a point mass at its end. The center of mass of the spherical shell is exactly located at the geometric center of the sphere. The center of mass of the internal gimbal coincides with that of the spherical shell.

Then we assign four coordinate frames. Let \( \Sigma_o \) XYZ be a fixed inertial frame whose \( XY \) plane is anchored to the surface of the inclined plane and \( Z \) is the vertical position to the surface. Let \( \Sigma_b \) be the body coordinate frame whose origin is located at the center of the sphere \( B \). Let \( \Sigma_c \) be the internal gimbal coordinate frame, whose center is located at the center of mass of the internal gimbal \( C \). Note that \( Z_c \) is always parallel to \( Z_b \).

Let \( \Sigma_d \) be the pendulum coordinate frame, whose center is located at point \( D \). Note that \( Y_d \) is always parallel to \( Y_c \). The variable definition of the robot model is listed in Table 2 and the coordinate system configurations for the robot is shown in Figure 3.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3.png}
\caption{Coordinate system configuration for the robot}
\end{figure}

We define \( (i, j, k) \) and \( (l, m, n) \) to be the unit vectors of the coordinate frames \( \Sigma_o \) and \( \Sigma_b \) respectively. We denote \( \sin \chi \) and \( \cos \chi \) by \( S_\chi \) and \( C_\chi \) respectively throughout this paper. The transformation between the two coordinate frames is given by

\[
\begin{bmatrix}
i \\
j \\
k
\end{bmatrix} = \begin{bmatrix}
l \\
m \\
n
\end{bmatrix}\begin{bmatrix}
C_\theta C_\phi - S_\theta S_\phi C_\chi \\
- S_\theta C_\phi - C_\theta S_\phi C_\chi \\
S_\phi S_\chi
\end{bmatrix}
\]

(19)

where \( R^o_b \) is the rotation matrix from \( \Sigma_o \) to \( \Sigma_b \).

We define \( \nu \) and \( \omega \) to denote the velocity and angular velocity of the center of mass of the spherical shell with respect to the inertia frame \( \Sigma_o \). Then we have

\[
\omega = (-\dot{C}_\phi + \dot{\psi} S_\phi) i + (-\dot{S}_\phi + \dot{\phi} C_\phi) j + (\ddot{\psi} + \dot{\phi} \dot{\psi}) k
\]

(20)
The constraints result from the requirement that the sphere rolls without slipping on an inclined plane, i.e., the velocity of the contact point on the sphere \( v_E = 0 \). Then we can express \( v_A \) as
\[
v_A = \omega \times r_{BE} + v_E
\]  
(21)

Here \( r_{BE} = -rk \). Substituting (20) into (21) gives
\[
\dot{x}_r + r(\dot{S}_r - \phi C_v S_\beta) = 0
\]  
(22)
\[
\dot{y}_r + r(\dot{C}_v + \phi S_v S_\beta) = 0
\]  
(23)
\[
\dot{z}_r = 0
\]  
(24)

The constraints in (22) and (23) are nonholonomic, and the constraint in (24) is holonomic and can be integrated to obtain
\[
z_r = r
\]  
(25)

Therefore the robot configuration can be described by seven generalized coordinates \( (x_r, y_r, \psi, \theta, \phi, \alpha, \beta) \).

Table 2. Variable definition of the 3-D model

<table>
<thead>
<tr>
<th>( \psi, \theta, \phi )</th>
<th>Precession angle, lean angle and spin angle of the spherical shell respectively</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \alpha, \beta )</td>
<td>Rotation angle of the internal gimbal and pendulum respectively</td>
</tr>
<tr>
<td>( \gamma )</td>
<td>Inclination angle of the plane</td>
</tr>
<tr>
<td>( x_s, y_s, z_s )</td>
<td>Coordinates of the mass center of the spherical shell with respect to the inertial frame</td>
</tr>
<tr>
<td>( x_i, y_i, z_i )</td>
<td>Coordinates of the mass center of the internal gimbal with respect to the inertial frame</td>
</tr>
<tr>
<td>( x_p, y_p, z_p )</td>
<td>Coordinates of the mass center of the pendulum with respect to the inertial frame</td>
</tr>
<tr>
<td>( m, m_i, m_p )</td>
<td>Mass of the spherical shell, internal gimbal and pendulum respectively</td>
</tr>
<tr>
<td>( r )</td>
<td>Radius of the spherical shell</td>
</tr>
<tr>
<td>( l )</td>
<td>Length of the pendulum link</td>
</tr>
<tr>
<td>( g )</td>
<td>Gravitational acceleration</td>
</tr>
<tr>
<td>( I_{xxs}, I_{yy}, I_{zz} )</td>
<td>Moment of inertia of the spherical shell about ( X, Y ) and ( Z ) direction</td>
</tr>
<tr>
<td>( I_{xxi}, I_{yyi}, I_{zzi} )</td>
<td>Moment of inertia of the internal gimbal about ( X, Y ) and ( Z ) direction</td>
</tr>
</tbody>
</table>

\[ P_s = mg \left( xS_r + rC_\gamma \right) \]  
(28)

Then the Lagrangian of the spherical shell is
\[ L_s = T_s - P_s \]  
(29)

From the above assumptions, we have
\[
\begin{bmatrix}
x_s \\
y_s \\
z_s
\end{bmatrix} = \begin{bmatrix}
x_r \\
y_r \\
z_r
\end{bmatrix}
\]  
(30)

We define \( \omega \) to denote the angular velocity of the internal gimbal with respect to \( \Sigma_c \). Then we have
\[
\omega = R_s^c \omega_b + \begin{bmatrix} 0 \\ 0 \\ \alpha \end{bmatrix}
\]
\[
R_s^c = \begin{bmatrix} C_\theta & S_\theta & 0 \\ -S_\theta & C_\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}
\]  
(31)

where \( R_s^c \) is the transformation from \( \Sigma_c \) to \( \Sigma_b \).

Differentiating (30) and using (31), the kinetic energy of the internal gimbal is given by
\[
T_c = \frac{1}{2} m \left( \dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2 \right) + \frac{1}{2} \left( I_{xx} \dot{\alpha}_x^2 + I_{yy} \dot{\alpha}_y^2 + I_{zz} \dot{\alpha}_z^2 \right)
\]  
(32)

The potential energy of the internal gimbal is
\[
P_c = m g \left( xS_r + rC_\gamma \right)
\]  
(33)

Then the Lagrangian of the internal gimbal is
\[ L_c = T_c - P_c \]  
(34)

The transformation from the center of mass of the sphere to that of the pendulum can be described as
\[
\begin{bmatrix}
x_p \\
y_p \\
z_p
\end{bmatrix} = \begin{bmatrix}
x_r + R_s^c l \\ y_r + R_s^c 0 \\ z_r
\end{bmatrix}
\]
\[
R_s^c = \begin{bmatrix} C_\phi & C_\theta \phi & S_\phi \\ -S_\phi \phi & C_\theta & C_\phi \theta \\ 0 & -S_\theta & C_\theta \end{bmatrix}
\]  
(35)

where \( R_s^c \) is the transformation from \( \Sigma_c \) to \( \Sigma_b \) and \( R_c^b \) is the transformation from \( \Sigma_b \) to \( \Sigma_c \).

Differentiating (35) and we can obtain the kinetic energy for the pendulum as
\[
T_p = \frac{1}{2} m_p \left( \dot{x}_p^2 + \dot{y}_p^2 + \dot{z}_p^2 \right)
\]  
(36)

The potential energy of the pendulum is
\[
P_p = m_p g \left( xS_r + z_s C_\gamma \right)
\]  
(37)

Then the Lagrangian of the pendulum is
\[ L_p = T_p - P_p \]  
(38)

The robot consists of the above three parts, then the Lagrangian of the robot system is
\[ L = L_s + L_c + L_p \]  
(39)

Substituting (29), (34) and (38) into (39), then \( L \) is determined. There are only two control torques, i.e., the tilt torque \( \tau_z \) and drive torque \( \tau_x \), available on the system. Consequently, using the constrained Lagrange method, the dynamic equation of the entire system is given by
\[
M(q) \ddot{q} + N(q, \dot{q}) = \dot{A}^T(q) \lambda + B \tau
\]  
(40)

where \( M(q) \in \mathbb{R}^{7 \times 7} \) and \( N(q, \dot{q}) \in \mathbb{R}^{7 \times 4} \) are the inertia matrix and nonlinear terms respectively.
\[ A(q) = \begin{bmatrix} 1 & 0 & 0 & r_{S_y} & -r_{C_y} S_{\phi} & 0 & 0 \\ 0 & 1 & 0 & r_{S_y} & r_{C_y} & S_{\phi} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \]

\[ B = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \]

\[ q = [x, y, \psi, \theta, \phi, \alpha, \beta]^T \]

\[ \lambda = [\lambda_1, \lambda_2]^T \]

The nonholonomic constraints can be formed as

\[ A(q)q = 0 \] (41)

### 3.3. Trajectory Tracking

We first partition \( A(q) \) as \( A = [A_1 \quad A_2] \), where

\[ A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & r_{S_y} & -r_{C_y} S_{\phi} & 0 & 0 \\ 0 & r_{C_y} & r_{S_y} S_{\phi} & 0 & 0 \end{bmatrix} \]

Let

\[ C(q) = \begin{bmatrix} -A_1 A_2 \\ I_{4 \times 5} \end{bmatrix} \] (42)

We define \( \nu = \dot{q} \) and consider the following relation

\[ \dot{q} = C(q)\nu \] (43)

Differentiating (43) yields

\[ \ddot{q} = C\dot{\nu} + \dot{C} \nu \] (44)

Substituting (43) and (44) into (40), and premultiplying both sides by \( C^T(q) \) gives

\[ C^T(MC\nu + MC\dot{\nu} + N) = C^T B\tau \] (45)

Using the state variable \( x = [q^T \quad \nu^T]^T \), we have

\[ \dot{x} = C\nu + \begin{bmatrix} 0 \\ \left( C^T MC \right)^{-1} C^T B \end{bmatrix} \tau \] (46)

where \( f_2 = -\left( C^T MC \right)^{-1} \left( C^T MC\nu + C^T N \right) \).

We apply the following nonlinear feedback

\[ \tau = \left( C^T MC \right)^{-1} C^T B \left( u - f_2 \right) \] (47)

where \( D^+ \) is the generalized inverse of the matrix \( D \).

The state equation simplifies to the form

\[ \dot{x} = f(x) + g(x)u \] (48)

where

\[ f(x) = \begin{bmatrix} C(q) \nu \\ 0 \end{bmatrix}, \quad g(x) = \begin{bmatrix} 0 \\ I \end{bmatrix} \]

The nonholonomic system in (48) can be input-output linearized if a proper set of output equations are chosen. Consider the following output equation

\[ y = h(q) \] (49)

The necessary and sufficient condition for input-output linearization is that the decoupling matrix has full rank [4].

With (49), the decoupling matrix \( \Phi(q) \) for the system is

\[ \Phi(q) = J_x(q) C(q) \] (50)

where \( J_x(q) = \frac{\partial h}{\partial q} \) is the Jacobian matrix.

To achieve input-output linearization, we introduce a new state variable \( z \) defined as follows

\[ z = T(x) = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} h(q) \\ J_x(q) C(q) \\ \Phi(q)\nu \end{bmatrix} \] (51)

Here \( \tilde{h}(q) \) is a vector function such that \( [J_x^T \quad J_k^T] \) has full rank. It is easy to verify that \( T(x) \) is a diffeomorphism. The system under the new state variable \( z \) is characterized by

\[ \dot{z}_1 = \frac{\partial h}{\partial q} \nu + \tilde{h}(q) \] (52)

\[ \dot{z}_2 = J_x Cv + J_q C(J_x C)^{-1} z_2 \]

Using the following state feedback

\[ u = \Phi^{-1}(q)(\nu - \Phi(q)\nu) \]

We achieve the input-output linearization as

\[ \dot{z}_1 = z_2 \] (54)

\[ y = z_1 \]

The zero dynamics of the system is \( \dot{z}_1 = 0 \) [5], which is Lagrange stable. For trajectory tracking we choose the output equation as

\[ y = h(x) = [x, y]^T \] (55)

It can be verified that the robotic system is input-output linearizable with the output equation in (55).

Applying the nonlinear feedback in (53), we can obtain a linearized and decoupled system in the following form

\[ \dot{x}_i = u_i \]

\[ \ddot{y}_i = u_i \] (56)

We define the tracking errors as

\[ e_1 = x - x^d \]

\[ e_2 = y - y^d \] (57)

Here \( x^d \) and \( y^d \) are the desired values for \( x \) and \( y \), respectively. To stabilize the system in (56) and achieve desired performance, an outer linear feedback loop is designed to place the poles of the system

\[ u_1 = \dot{u}_1 - k_1 e_1 = \dot{x} - k_1 e_1 \]

\[ u_2 = \dot{u}_2 - k_2 e_2 = \dot{y} - k_2 e_2 \] (58)

Here \( u_1^d \) and \( u_2^d \) are the desired values for \( \dot{x} \) and \( \dot{y} \), respectively. The gains \( k_i \) (i = 1, 2, 3, 4) are real positive values and are properly chosen so that the dynamics of the following error system is exponentially stable.
\[ \ddot{x} + k_2 \dot{y} + k_1 e_1 = 0 \]
\[ \dot{y} + k_2 \dot{z} + k_1 e_2 = 0 \]

4. SIMULATION STUDY

4.1. Planar Motion

We apply the decoupled sliding mode control scheme to the spherical robot to demonstrate its effectiveness. Defining the state variable \( x = [\theta \ \dot{\theta} \ \phi \ \dot{\phi}]^T \), the planar dynamic equation of the robot in (1) can be converted into the canonical form described in (5) by adopting coordinate transformation. The robot model parameters are selected as in [6] and the controller design parameters are listed in Table 3. We assume the robot to be executing a rest-to-rest manoeuvre along a slope of 15 degrees. The initial values of the states are chosen as \( x_0 = [0 \ 0 \ \phi_0 \ 0]^T \), and we choose the desired values of the system states as \( x^d = [\pi \ 0 \ \phi_0 \ 0]^T \).

Table 3. Sliding mode controller parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( c_1 )</td>
<td>0.5501</td>
</tr>
<tr>
<td>( c_2 )</td>
<td>2.3272</td>
</tr>
<tr>
<td>( z_d )</td>
<td>0.9626</td>
</tr>
<tr>
<td>( \Phi_x )</td>
<td>7.4822</td>
</tr>
<tr>
<td>( \eta )</td>
<td>2.7688</td>
</tr>
<tr>
<td>( k )</td>
<td>0.6769</td>
</tr>
</tbody>
</table>

The tracking result of the proposed controller is shown in Figure 4. We can find that not only the rotation angle of the sphere but also the rotation angle of the pendulum can reach their desired values in a short time. Before the robot reaches its desired position, the pendulum angle has already converged to its equilibrium value only after a few oscillations.

4.2. Three-dimensional Motion

We develop a numerical simulation to verify the effectiveness of the proposed trajectory tracking control algorithm. The proposed controller is used for the robot to track a straight line \( y = x \) on an incline of \( \gamma = 20^\circ \). The dimensions and inertial parameters of the robot are also selected as in [6]. In addition, the initial conditions and control parameters are chosen as listed in Table 4. Here \( x_0 \), \( y_0 \), \( \dot{x}_0 \), \( \dot{y}_0 \) are the initial values and differentiated initial values of \( x \), \( y \) respectively.

Table 4. Control parameters and conditions

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_0 )</td>
<td>0.6 m</td>
</tr>
<tr>
<td>( y_0 )</td>
<td>0.2 m</td>
</tr>
<tr>
<td>( \dot{x}_0 )</td>
<td>0 m/s</td>
</tr>
<tr>
<td>( \dot{y}_0 )</td>
<td>0 m/s</td>
</tr>
<tr>
<td>( k_1 )</td>
<td>64</td>
</tr>
<tr>
<td>( k_2 )</td>
<td>16</td>
</tr>
<tr>
<td>( k_3 )</td>
<td>64</td>
</tr>
<tr>
<td>( k_4 )</td>
<td>16</td>
</tr>
</tbody>
</table>

Figure 5 depicts the trajectory tracking performance of the proposed control scheme with different desired forward velocities, the controller gains \( k_i \) \( (i = 1, 2, 3, 4) \) remain the same in all these cases. The system response for the desired path is satisfactory as seen from the figure, and the robot is able to reach the desired path and stay on the path.

5. CONCLUSION

In this article, we discuss the control and stabilization problems of a spherical mobile robot rolling along an inclined slope. At first the planar dynamics of the robot is developed through the Euler-Lagrange method and a modified decoupled sliding mode control approach is proposed. Then we derive the 3-D dynamics of the spherical robot using the constrained Lagrange equation. We investigate the trajectory tracking control algorithm for the 3-D motion and derive a nonlinear feedback that guarantees input-output stability for the overall system. Finally, the simulation results verify the validity of the proposed control schemes.

6. ACKNOWLEDGEMENT

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 (708011) and the key project of science and technology development projects of Beijing Municipal Education Commission (KZ200810005002).

Figure 5. Trajectory tracking performance for the robot

7. REFERENCES


