Tracking Control of Wheeled Inverted Pendulum  
Based on the Time-State Control Form  

Satoko YAMAKAWA*  

*Toyo University, Kujirai2100, Kawagoe 3508585, JAPAN  
(e-mail: yamakawa@toyo.jp)  

Abstract: In this study, a tracking controller for an inverted pendulum robot with two wheels is proposed. Two control inputs of the system are separately designed. One control input is used for stabilizing the body angle and controlling the translational velocity. The other input is designed for asymptotically converging to a given trajectory in a horizontal plane based on the time-state control form. In numerical simulations and experimental results, it was confirmed that the robot tracks a given desired trajectory.  

Keywords: Nonlinear control system, Stabilization, Tracking, Wheeled inverted pendulum  

1. INTRODUCTION  
A wheeled inverted pendulum robot is a nonlinear system which has an unstable equilibrium. It has attracted researcher's attentions like as an inverted pendulum system(Pathak et al., 2005). Recently, it has become widely known because some vehicles such as Segway have been developed based on the control theory for this robot. Although the cart of classical inverted pendulum system moves along a linear path, the inverted pendulum robot whose control method is discussed in this paper can move on a two dimensional horizontal plane by controlling two wheels. The horizontal motion is nonholonomic due to the constraint condition between wheels and the ground.  

The purpose of this study is to propose a tracking controller for a two wheeled inverted pendulum robot to asymptotically track a given desired trajectory while stabilizing the body angle. As for stabilizing the body angle, practical stabilizing controllers can be easily obtained based on the linear approximation of the motion equation around the vertical equilibrium. However, the kinematic equation describing the horizontal motion of two wheeled vehicle can not be approximated to a linear model. Therefore, we have to consider the nonlinearity of the motion.  

Many nonlinear controllers have been proposed for nonholonomic systems including two wheeled vehicle(Banavar et al., 2006). The method based on the time-state control form is one of them(Sampei et al., 1994,1996). We clarified the condition of this method for stabilization(Fujimoto et al., 2000 and Yamakawa et al.,2003). In such methods of controlling nonholonomic vehicles, the translational and rotational velocities are usually considered as independent control inputs. However, the body angle of pendulum robot is controlled by the force acting on a joint of the pendulum, which is generated by the translational acceleration of the robot. Therefore, in the tracking problem of inverted pendulum robot, the translational velocity is not considered as an independent control input.  

Paying attention to this fact, the controller design process is separated into two phases in this study. First, the controller for stabilizing the body angle is derived based on a linearly approximated model. Second, the input concerned with the rotational velocity is designed as a function of the translational velocity in order to track a given horizontal trajectory. Since the method of controller design based on the time-state control form treats the translational and rotational velocities separately, it is easily applied to this controller design process for the inverted pendulum robot. Furthermore, the gains of proposed nonlinear controller are able to be comparatively easily modified based on the linear control theory. In section 2, a linear controller for tracking a given desired trajectory is described. In the following sections, the model of inverted pendulum robot is described and the tracking controller for the robot is proposed. The proposed controller is applied to a small robot and the controlled trajectories are evaluated in numerical simulations and experiments.  

2. PRELIMINARY  
Consider an asymptotic tracking control problem for generally controllable SISO systems.  

\[
\begin{align*}  
\dot{x} &= Ax + bu  
y &= cx  
\end{align*}  
\]  

where \( x \) is the \( n \times 1 \) state vector. By applying a coordinate transformation \( x_0 = T^{-1}x \), system(1) is transformed into a controllable canonical form.  

\[
\begin{align*}  
\dot{x}_0 &= A_0x_0 + b_0u  
y &= c_0x_0  
\end{align*}  
\]
Lemma 1 Consider a vector
\[
x_r = \begin{pmatrix} x_r(t) & \dot{x}_r(t) & \cdots & \frac{d^{n-1}x_r(t)}{dt^{n-1}} \end{pmatrix}^T
\]
which consists of an \(n\) times differentiable function \(x_r(t)\) and its derivatives. The state vector \(x_0\) of the system(3) asymptotically converges to \(x_r\) by using
\[
u = k(x_0 - x_r) - ax_r + \frac{d^n x_r(t)}{dt^n},
\]
where \(k\) is the gain vector such that \(A_0 + b_0 k\) is a stable matrix. □

Proof The error dynamics of \((x_0 - x_r)\) is derived from (3)(4) and (5).
\[
\frac{d}{dt}(x_0 - x_r) = A_0 x_0 + b_0 (k(x_0 - x_r) - ax_r + \frac{d^n x_r(t)}{dt^n}) - \dot{x}_r,
\]
where \(k\) is the gain vector such that \(A_0 + b_0 k\) is a stable matrix.
\[
= (A_0 + b_0 k)(x_0 - x_r) = (A_0 + b_0 a)x_r + b_0 \frac{d^n x_r(t)}{dt^n} - \dot{x}_r,
\]
\[
= (A_0 + b_0 k)(x_0 - x_r)
\]
Therefore, the error \((x_0 - x_r)\) asymptotically converges to 0 if \(A_0 + b_0 k\) is stable.

When \(x_0\) converges to \(x_r\), the output \(y\) converges to
\[
y = c_0 x_r,
\]
Therefore, a result for output tracking problem is derived from the above lemma.

Theorem 1 Assume that the differential equation
\[
c_{n+1} \frac{d^{n+1} x_r(t)}{dt^{n+1}} + \cdots + c_1 \dot{x}_r(t) + c_0 x_r(t) = y_{ij}(t)
\]
has a solution \(x_r(t)\) for a given \(y_{ij}(t)\) and that the solution is \(n\) times differentiable. Substitute the solution \(x_r(t)\) into (4)(5). Then, the output \(y(t)\) of system(1) with controller(5) asymptotically converges to \(y_{ij}(t)\).

Generally, nonhomogeneous differential equations may not have an analytical solution or a systematic method for finding its solutions, depending on the class of \(y_{ij}(t)\). However, for instance, we can easily find a solution by using the method of undetermined coefficients[Kreyszig, 1983] if \(y_{ij}(t)\) is an elementary function such as polynomial of \(t\), trigonometric function or exponential function.

3. MODEL OF WHEELED INVERTED PENDULUM
The wheeled inverted pendulum moves on a horizontal plane according to its translational and rotational velocities. The velocities are controlled by independently driving two wheels. The angle of the body from the vertical axis \(z\) is \(\theta\) (Fig.1). The angles of the right and left wheel with respect to the body are \(\phi\) and \(\phi\), respectively. The average angle \(\phi\) is defined as
\[
\phi = \frac{\phi + \phi}{2}.
\]
The translational velocity \(v_1\) is represented as
\[
v_1 = r(\dot{\phi} + \dot{\theta}).
\]
The robot rotates around the vertical axis according to the difference between two wheel velocities. Rotational velocity \(v_2\) is described as
\[
v_2 = \frac{r(\dot{\phi} - \dot{\phi})}{d},
\]
where \(d\) is the distance between two wheels.

Assuming that the wheels don't slip with respect to the ground, the kinematic equation of motion in a horizontal plane is obtained as
\[
\dot{x} = v_1 \cos \psi
\]
\[
\dot{y} = v_1 \sin \psi
\]
\[
\psi = v_2,
\]
where \((x, y)\) denotes the position of the midpoint of two wheels and \(\psi\) is the orientation of the robot with respect to the \(x\) axis (Fig.1).

The mass of robot's wheel unit is \(M\) and the mass of body is \(m\). The inertia of body and wheel about the axle of wheels are \(J_p\) and \(J_s\), respectively. The inertia of motor's rotor is \(J_m\). The gear ratio is \(n\). Although the inertia of robot about the \(z\) axis actually changes according to the angle \(\theta\), it is approximated as a constant \(J_M\). The coefficient of friction of wheels is \(c\). The input voltages to right and left wheel are \(V_r\) and \(V_l\), respectively. The driving force and steering torque are
\[ f_1 = KV + V_j \]
\[ f_2 = K \frac{d(V_i - V_j)}{2} \]

where \( K \) is a constant. The nonlinear equation of motion of the system is derived from Lagrange's equations, which is omitted in this paper. The obtained nonlinear equation is approximated to a linear equation, assuming that the angle \( \theta \) and angular velocity \( \dot{\theta} \) are sufficiently small. Eventually, the motion of the robot is described by the following equations.

\[
A\begin{bmatrix} \ddot{\theta} \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} -mg\theta \\ 2c\phi \end{bmatrix} = \begin{bmatrix} 0 \\ f_1 \end{bmatrix} \quad (15a)
\]
\[
J_r \ddot{\psi} + c_d \ddot{\psi} = f_2 \quad (15b)
\]
\[
\ddot{x} = \rho(\dot{\theta} + \phi) \cos \psi \\
\ddot{y} = \rho(\dot{\theta} + \phi) \sin \psi
\]

where,
\[
A = \begin{bmatrix} J_r + J_n(1-n) + ml^2 + mlr^2 + J_n(n + mlr) \\ J_n(n(1-n) + mlr) + (M + m)r^2 + J_n + J_n^2 \end{bmatrix}
\]

4. TRACKING CONTROLLER

As shown in (15a), the motion of angle \( \theta \) is independent of \( x \) and \( \psi \). Therefore, first, we design the control input \( f_1 \) so that the angle of body is stable and the translational velocity \( v_1 \) converges to a desired velocity \( v_d(t) \). Second, a steering control input \( f_2 \) for tracking a desired horizontal trajectory \( v_1(t) \) is designed as a function of velocity \( v_1 \).

4.1 Stabilization of Body Angle and Tracking of Translational Velocity

First of all, an input transformation is applied to simplify the following calculations.

\[ u = f_1 - 2c\phi \quad (16) \]

When the state vector is chosen as \( x = (\theta \ \dot{\theta} \ \dot{\theta} + \phi)^T \), the following state equation is obtained from (15a).

\[
\ddot{x} = A_1 x + b_1 u \\
A_1 = \begin{bmatrix} 0 & 1 & 0 \\ mgl_{a_1} & 0 & 0 \\ mgl_{a_2} & 0 & 0 \end{bmatrix}, \quad b_1 = \begin{bmatrix} 0 \\ a_{12} \\ a_{22} \end{bmatrix}
\]

where
\[
\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \Lambda^{-1}.
\]

In order to discuss the tracking problem of velocity \( v_1 \), the output equation is chosen as

\[
y = v_1 = (0 \ 0 \ r)x \quad (18)
\]

By using a coordinate transformation

\[ x_n = T^{-1} x \]
\[ T = \begin{bmatrix} a_{12} & 0 \\ 0 & a_{11} \end{bmatrix}, \quad \gamma = \frac{mg}{\det \Lambda}, \]

system (17)(18) is transformed into a controllable canonical form as follows.

\[
\dot{x}_n = \begin{bmatrix} 0 & 1 & 0 \\ 0 & mgl_{a_1} & 0 \\ 0 & mg \end{bmatrix} x_n + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u \\
y = (-\gamma r \ 0 \ a_{22}) x_n
\]

From the result of Theorem 1, the control input is designed as

\[ u = k_1(T^{-1} x - x_\nu) - mgl_{a_1}\frac{dx_\nu(t)}{dt} + d^3 x_\nu(t) + 2c\phi \quad (22) \]

where \( k_1 \) is a state feedback gain vector such that the system (20) is stable. Moreover, \( x_\nu(t) \) is a solution of the differential equation

\[ a_{22} \dot{r}_\nu(t) - \gamma r x_\nu(t) = v_d(t) \quad (23) \]

and \( v_d(t) \) is a given desired velocity. The reference vector is

\[ x_\nu = (x_\nu(t) \ \dot{x}_\nu(t) \ \ddot{x}_\nu(t))^T. \quad (24) \]

Eventually, the control input \( f_1 \) for asymptotic tracking of translational velocity \( v_1 \) is obtained by substituting (22) to (16).

\[ f_1 = k_1(T^{-1} x - x_\nu) - mgl_{a_1}\frac{dx_\nu(t)}{dt} + d^3 x_\nu(t) + 2c\phi \quad (25) \]

When controller (25) is applied, the state \( x \) converges to the following steady state.

\[ \lim_{t \to \infty} x(t) = T x^*_\nu \]

Especially, when \( \ddot{v}_d(t) = 0 \), it is sufficient to use \( x_\nu(t) = -v_d(t) / (\gamma r) \), which is a trivial solution of (23). In this case, the angular velocity \( \dot{\theta} \) converges to 0 because \( \dot{x}(t) = 0 \). Furthermore, the angle \( \theta \) of body also converges to 0 if \( v_d(t) = 0 \). On the other hand, if \( \ddot{v}_d(t) \neq 0 \), then \( \dot{x}(t) \neq 0 \). The angle \( \theta \) of body becomes large according to \( \dot{x}(t) \). Thus, the assumptions for linear approximation (15a) might not be satisfied and the system might not be stabilized by controller (25) if the time-derivative of \( v_d(t) \) is too large.
4.2 Tracking a Horizontal Trajectory

The translational and rotational velocities are usually considered as independent inputs in the control problems of nonholonomic vehicles. However, the translational velocity $v_1$ of inverted pendulum robot has already been designed as (25). Namely, $v_1$ can not be used as a control input for controlling the trajectory any longer. Thus, in this study, the states $y$ and $\psi$, which dominate the horizontal trajectory, are controlled by the steering input $f_z$. The tracking control input $f_z$ is derived by using the transformation to the time-state control form. In this control design method, a nonlinear system described as the chained form is transformed into a linear system by considering one of the states as a time axis. Consequently, a controller can be designed based on linear control theory.

However, when the system (15b)-(15d) is transformed into the time-state control form, the obtained nonlinear controller would need model parameters such as inertia. Therefore, referring the control problem of mobile robots (Siciliano et al., 2009), consider the rotational velocity $\psi_2$ in (12) as a control input $u_2$.

$$\psi = u_2$$

(27)

We design the input $u_2(27)$ so that the trajectory $y(x)$ converges to a given desired trajectory $y_d(x)$. The robot is actually controlled by input $f_z$ as

$$f_z = -k(\psi - u_2), \quad k > 0.$$  

(28)

When the gain $k$ is appropriately large, the state $\psi$ sufficiently approaches $u_2$ by using (28).

Moreover, there are singular states when system (15c)-(15d) (27) is transformed into the time-state control form. Therefore, the following controller design is considered under the assumption that $\cos \psi > 0$. If $\cos \psi(0) \leq 0$, it is enough to use a constant input $f_z$ until satisfying $\cos \psi > 0$.

We first consider the controller design when $v_1 = r(\dot{\phi} + \dot{\theta}) > 0$. In this case, from (15c), the state $x$ monotonously increases with time $t$. Thus, $x$ is able to be regarded as a time instead of actual time $t$.

By applying a nonlinear coordinate transformation

$$z = \left(y \tan \psi \right)'$$  

(29)

and an input transformation

$$\mu = \frac{1}{v_1 \cos \psi} u_2$$  

(30)

to (15c)-(15d)-(27), the state-space equation is transformed into the time-state control form.

$$\frac{d}{dx} z = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} z + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \mu$$

(31)

The output equation is chosen so that the output is $y$.

$$y = (1 \ 0) z$$  

(32)

The system is described as a controllability canonical form (31) which time axis is $x$. Therefore, we can obtain a tracking controller from the result of Theorem 1, when the desired trajectory $y_d(x)$ is 2 times differentiable with respect to $x$. The reference vector $z_*$ is obtained without solving any differential equation because equation (8) becomes an algebra equation when $c_0 = (1 \ 0)$.

$$z_* = \begin{pmatrix} y_d(x) \\ \frac{dy_d(x)}{dx} \end{pmatrix}^T$$  

(33)

The output $y(x)$ converges to $y_d(x)$ by using the controller

$$\mu = k_2(z_* - z_*) + \frac{d^2 y_d(x)}{d x^2}.$$  

(34)

where $k_2$ is a state feedback gain such that system (31) is stable.

Reversely, the state $x$ monotonously decreases with time $t$ when $v_1 < 0$. In this case, $x' = -x$ is used as a time axis instead of $x$. Consequently, the system is transformed into

$$\frac{d}{dx} z = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} z + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \mu, \quad \mu = diag(1, -1) z_*.$$  

(35)

As for the reference vector, the derivative $dy_d(x)/dx'$ has the opposite sign of $dy_d(x)/dx$ therefore,

$$\begin{pmatrix} y_d(x) \\ \frac{dy_d(x)}{dx'} \end{pmatrix}^T = diag(1, -1) z_*.$$  

(36)

By noticing the symmetry between (31)-(33) and (35)-(36), the tracking controller for $v_1 < 0$ is designed as

$$\mu = k_2, diag(1, -1)(z_* - z_*) + \frac{d^2 y_d(x)}{d x^2}.$$  

(37)

Eventually, the tracking controller such that $y$ asymptotically converges to $y_d(x)$ is derived as a function of $v_1$.

$$u_2 = \mu v_1 \cos \psi$$  

(38)

$$\mu = k_2, diag(1, \text{sgn}(v_1))(z_* - z_*) + \frac{d^2 y_d(x)}{d x^2},$$

where $\text{sgn}(v_1)$ is a function that extracts the sign of $v_1$ and $\text{sgn}(0)$ equals 0. From the Theorem 1 and the previous result (Yamakawa, S. et al., 2003), it is confirmed that state error $(z - z_*)$ converges to 0 if the average interval for changing the sign of $v_1$ is greater than a constant.

5. NUMERICAL SIMULATIONS

In numerical simulations, the following model parameters (ZMP inc., 2007) were used: $m = 0.6113, J_p = 2.65 \times 10^{-7}, J_r = 4.694 \times 10^4, J_m = 1.30 \times 10^7, n = 30, r = 0.02485, l = 0.1104, c = 0.5 \times 10^7, d = 0.175, J_{\psi} = 3.0 \times 10^{-4}$ and $g = 9.80665$. 

10694
Controller gains $k_1$, $k_2$ were designed based on pole assignment method. $k_1 = (-3375, -794.884, -45)$ assigns three poles of system(20) to $-15$ and $k_2 = (-100, -20)$ assigns two poles of system(31) to $-10$. The gain in (28) was appropriately chosen as $k=1.4$.

5.1 Tracking a Parallel Line, Going Back and Forth

A desired trajectory is given as $y_d(x)=0.3$ which is parallel to the $x$ axis. A desired velocity of $v_1$ is given as

$$v_1(t) = \alpha \sin \omega t$$

so that the robot goes back and forth. A solution of the differential equation(23) is obtained as

$$x_1(t) = \frac{-\alpha}{\lambda_2} \sin \omega t \cdot$$

Thus, the reference vector $x_1$ is

$$x_1 = \left[ \begin{array}{c} -\alpha \cos \omega t \\ \alpha \sin \omega t \\ -\omega \sin \omega t \\ \omega \cos \omega t \end{array} \right].$$

For example, when $\alpha=0.2$ and $\omega=0.5$, the controller is calculated as

$$f_1 = 17.49\theta + 2.545\dot{\theta} + 0.2340\dot{\phi} - 1.869\sin 0.5t - 0.1874 \cos 0.5t$$

$$f_2 = -1.4(\psi - u_2)$$

$$u_2 = v_1 \cos^3 \psi (-100(y - 0.3) - 20 \text{sgn}(v_1) \tan \psi).$$

In numerical simulations, the responses were calculated by using the nonlinear model without applying the linear approximation with respect to $\theta$ and $\dot{\theta}$. The simulation results starting from the conditions that $(x, y, \psi) = (0, 0, 0)$ and $(\theta, \phi, \dot{\theta}, \dot{\phi}) = (0, 0, 0, -0.001)$ are plotted as solid lines in Fig.2-3. Although the controller is designed based on linearly approximated model(15), the translational velocity $v_1$ tracks $y_d(t)$ well as shown in Fig.2. However, in this condition, the angle of body does not converge to 0 because the acceleration of robot changes continuously. From (26)(41), the angle of body $\theta$ oscillates in steady state as

$$\theta = \frac{a_1}{\lambda_2} \frac{-\alpha}{\lambda_2} \omega \cos \omega t$$

$$= 0.0126\cos 0.5t.$$  

As for the horizontal motion, the robot go back and forth and its trajectory converges to $y_d(x)=0.3$ as shown in Fig.3.

The steering angle changed relatively large at an early stage as shown in Fig.2. Thus, the controller gain $k_2$ was modified to reduce the steering angle. The gain $k_2$ was chosen as $k_2 = (-25, -10)$ so that the poles of (31) are assigned to $-5$ which is greater than $-10$. The simulation results are plotted as the dotted lines in Fig.2-3. The rotation angle $\psi$ became small and the trajectory slowly converged to $y_d(x)$. We can modify the controller gains based on the design method of linear controllers as shown in the above example. In addition, the angle $\theta$ and velocity $v_1$ were not changed even if $k_2$ was changed.

5.2 Tracking a Sine Wave Trajectory with a Constant Velocity

Another problem of moving the robot on a desired trajectory $y_d(x) = \beta \sin wx$ with constant velocity $v_d(t) = 0.1$ is considered. The reference vectors are obtained from (24)(33).

$$x_r = \left[ \begin{array}{c} 0.1 \\ 0 \\ 0 \\ 0 \end{array} \right]$$

$$z_r = \beta (\sin wx \ w \cos wx)$$

The control inputs are obtained as

$$f_1 = 17.49\theta + 2.545\dot{\theta} + 0.2340\dot{\phi} - 0.9418$$

$$f_2 = -1.4(\psi - u_2)$$

$$u_2 = \mu v_1 \cos^3 \psi$$

$$\mu = (-100 - 20 \text{sgn}(v_1) (\tan \psi - \beta w \cos wx) - \beta w^2 \sin wx).$$

When $\beta = 0.2$, $w = 7$ and the initial states are the same as the previous simulations, the simulation results are plotted as the dotted lines in Fig.4-6. It was confirmed that the translational velocity $v_d(t)$ and the trajectory $y(x)$ converged to $v_d(t)$ and $y_d(x)$, respectively. Furthermore, the angle $\theta$ converged to 0 because $v_d(t)$ was constant.
6. EXPERIMENTAL RESULTS

The small inverted pendulum robot was used in experiments. The model parameters of the robot were the same as those in section 5. The controller for tracking a sine wave trajectory designed in section 5.2 was applied to the robot. As shown in Fig.4, the body angle fluctuated because of the disturbances such as the backlash of gears and so on. Therefore, the velocity \( v_1 \) also fluctuated in order to stabilize the attitude of body. Furthermore, the velocity \( v_1(t) \) periodically became greater than \( v_d = 0.1 \) because there was the difference between dynamical properties of right and left wheel units. Namely, the resistance force of the robot became small when turning left. Thus, \( x(t) \) did not equal the simulation result as shown in Fig.5. Since the rotational velocity was modified by controller(38) depending on the actual velocity \( v_1(t) \), \( y(t) \) was also different from the simulation result. As a result, the trajectory \( y(x) \) finely converged to the given desired trajectory like the simulation result, as shown in Fig.6. The fact that the controller \( f_2 \) did not directly use the model parameters also contributed to achieving the fine tracking.

7. CONCLUSIONS

In this study, the tracking controller for inverted pendulum robot with two wheels was proposed. After the controller for stabilizing the body angle was designed, the controller for asymptotically converging to a given trajectory was designed based on the time-state control form. The control gains of proposed controller are comparatively easily modified based on the linear control theory. In numerical simulations and experiments, it was confirmed that the robot moved along a given trajectory in the horizontal plane even if the translational velocity fluctuates because of model errors or disturbances.

REFERENCES


Siciliano, B. et al. (2009). Robotics, Springer