Trajectory Tracking Controller Design for A Tricycle Robot Using Piecewise Multi-Linear Models
Tadanari Taniguchi and Michio Sugeno

Abstract—This paper deals a tracking trajectory controller design of a tricycle robot as a non-holonomic system with a piecewise multi-linear (PML) model. The approximated model is fully parametric. Input-output (I/O) dynamic feedback linearization is applied to stabilize PML control system. We also apply a method for a tracking control based on PML models to the tricycle robot. Although the controller is simpler than the conventional I/O feedback linearization controller, the control performance based on PML model is the same as the conventional one. Examples are shown to confirm the feasibility of our proposals by computer simulations.

Index Terms—piecewise model, tracking trajectory control, dynamic feedback linearization.

I. INTRODUCTION

We propose the tracking trajectory control of a tricycle robot using dynamic feedback linearization based on piecewise multi-linear (PML) models. Wheeled mobile robots are completely controllable. However they cannot be stabilized to a desired position using time invariant continuous feedback control [1]. The wheeled mobile robot control systems have a non-holonomic constraint. Non-holonomic systems are much more difficult to control than holonomic ones. Many methods have been studied for the tracking control of wheeled robots. The backstepping control methods are proposed in (e.g. [2], [3]). The sliding mode control methods are proposed in (e.g., [4], [5]), and also the dynamic feedback linearization methods are in (e.g., [6], [7], [8]). For non-holonomic robots, it is never possible to achieve exact linearization via static state feedback [9]. It is shown that the dynamic feedback linearization is an efficient design tool to solve the trajectory tracking and the setpoint regulation problem in [6], [7].

In this paper, we consider PML model as a piecewise approximation model of the tricycle robot dynamics. The model is built on hyper cubes partitioned in state space and is found to be bilinear (bi-affine) [10], so the model has simple nonlinearity. The model has the following features: 1) The PML model is derived from fuzzy if-then rules with singleton consequents. 2) It has a general approximation capability for nonlinear systems. 3) It is a piecewise nonlinear model and second simplest after the piecewise linear (PL) model. 4) It is continuous and fully parametric. The stabilizing conditions are represented by bilinear matrix inequalities (BMIs) [11], therefore, it takes long computing time to obtain a stabilizing controller. To overcome these difficulties, we have derived stabilizing conditions [12], [13], [14] based on feedback linearization, where [12] and [14] apply input-output linearization and [13] applies full-state linearization.

We propose a dynamic feedback linearization for PML control system and apply the tracking control [15] to a tricycle robot system. The control system has the following features: 1) Only partial knowledge of vertices in piecewise regions is necessary, not overall knowledge of an objective plant. 2) These control systems are applicable to a wider class of nonlinear systems than conventional I/O linearization. 3) Although the controller is simpler than the conventional I/O feedback linearization controller, the tracking performance based on PML model is the same as the conventional one. Wheeled robot dynamics has some trigonometric functions. The trigonometric functions are smooth functions and of class $C^\infty$. The PML models are not of class $C^\infty$. In the tricycle robot control, we have to calculate the third derivatives of the output. Therefore the derivative PML models lose some dynamics. Thus we propose the derivative PML models of the trigonometric functions.

This paper is organized as follows. Section II introduces the canonical form of PML models. Section III presents a dynamic feedback linearization of the car-like robot. Section IV proposes a tracking controller design using dynamic feedback linearization based on PML model of the tricycle robot. Section V shows examples demonstrating the feasibility of the proposed methods. Finally, section VI summarizes conclusions.

II. CANONICAL FORMS OF PIECEWISE BILINEAR MODELS

A. Open-Loop Systems

In this section, we introduce PML models suggested in [10]. We deal with the two-dimensional case without loss of generality. Define vector $\delta(\sigma, \tau)$ and rectangle $R_{\sigma \tau}$ in two-dimensional space as $\delta(\sigma, \tau) \equiv (d_1(\sigma), d_2(\tau))^T$.

$$R_{\sigma \tau} \equiv [d_1(\sigma), d_1(\sigma + 1)] \times [d_2(\tau), d_2(\tau + 1)]. \quad (1)$$

$\sigma$ and $\tau$ are integers: $-\infty < \sigma, \tau < \infty$ where $d_1(\sigma) < d_1(\sigma + 1)$, $d_2(\tau) < d_2(\tau + 1)$ and $d(0, 0) \equiv (d_1(0), d_2(0))^T$. Superscript $^T$ denotes a transpose operation.
For \( x \in R_{\sigma \tau} \), the PML system is expressed as
\[
\begin{align*}
\dot{x} &= \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_i^d(x_1)\omega_j^d(x_2)f_o(i,j), \\
x &= \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_i^d(x_1)\omega_j^d(x_2)d(i,j),
\end{align*}
\]
(2)
where \( f_o(i,j) \) is the vertex of nonlinear system \( \dot{x} = f_o(x) \),
\[
\begin{align*}
\omega_i^d(x_1) &= (d_1(\sigma + 1) - x_1)/(d_1(\sigma + 1) - d_1(\sigma)), \\
\omega_j^d(x_2) &= (d_2(\tau + 1) - x_2)/(d_2(\tau + 1) - d_2(\tau)),
\end{align*}
\]
(3)
and \( \omega_i^d(x_1), \omega_j^d(x_2) \in [0, 1] \). In the above, we assume \( f(0,0) = 0 \) and \( d(0,0) = 0 \) to guarantee \( \dot{x} = 0 \) for \( x = 0 \).

A key point in the system is that state variable \( x \) is also expressed by a convex combination of \( d(i,j) \) for \( \omega_i^d(x_1) \) and \( \omega_j^d(x_2) \), just as in the case of \( \dot{x} \). As seen in equation (3), \( x \) is located inside \( R_{\sigma \tau} \) which is a rectangle: a hypercube in general. That is, the expression of \( x \) is polytopic with four vertices \( d(i,j) \). The model of \( \dot{x} = f(x) \) is built on a rectangle including \( x \) in state space, it is also polytopic with four vertices \( f(i,j) \). We call this form of the canonical model (2) parametric expression.

### B. Closed-Loop Systems

We consider a two-dimensional nonlinear control system.
\[
\begin{align*}
\dot{x} &= f_1(x) + g_1(x)u(x), \\
y &= h_1(x).
\end{align*}
\]
(4)
The PML model (5) is constructed from a nonlinear system (4).
\[
\begin{align*}
\dot{x} &= f(x) + g(x)u(x), \\
y &= h(x),
\end{align*}
\]
(5)
where
\[
\begin{align*}
f(x) &= \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_i^d(x_1)\omega_j^d(x_2)f_o(i,j), \\
g(x) &= \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_i^d(x_1)\omega_j^d(x_2)g_o(i,j), \\
h(x) &= \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_i^d(x_1)\omega_j^d(x_2)h_o(i,j), \\
x &= \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_i^d(x_1)\omega_j^d(x_2)d(i,j),
\end{align*}
\]
(6)
and \( f_o(i,j), g_o(i,j), h_o(i,j) \) and \( d(i,j) \) are vertices of the nonlinear system (4). The modeling procedure in region \( R_{\sigma \tau} \) is as follows:

1. Assign vertices \( d(i,j) \) for \( x_1 = d_1(\sigma), d_1(\sigma + 1), x_2 = d_2(\tau), d_2(\tau + 1) \) of state vector \( x \), then partition state space into piecewise regions (see Fig. 1).

2. Compute vertices \( f_o(i,j), g_o(i,j) \) and \( h_o(i,j) \) in equation (6) by substituting values of \( x_1 = d_1(\sigma), d_1(\sigma + 1) \) and \( x_2 = d_2(\tau), d_2(\tau + 1) \) into original nonlinear functions \( f_o(x), g_o(x) \) and \( h_o(x) \) in the system (4). Fig. 1 shows the expression of \( f(x) \) and \( x \in R_{\sigma \tau} \).

The overall PML model is obtained automatically when all vertices are assigned. Note that \( f(x), g(x) \) and \( h(x) \) in the PML model coincide with those in the original system at vertices of all regions.

### III. DYNAMIC FEEDBACK LINEARIZATION OF TRICYCLE ROBOT

We consider a tricycle robot model.
\[
\begin{align*}
\dot{x} &= \left( \begin{array}{c} \cos \theta \\ \sin \theta \\ \frac{1}{2} \tan(M\tan w) \end{array} \right) u_1 + \left( \begin{array}{c} 0 \\ 0 \\ 0 \end{array} \right) u_2,
\end{align*}
\]
(7)
where \( x \) and \( y \) are the position coordinates of the center of the rear wheel axis, \( \theta \) is the angle between the center line of the car and the \( x \) axis, \( \psi \) is the steering angle with respect to the car. The control inputs are represented as
\[
\begin{align*}
u_1 &= u_\psi \cos \psi \\
u_2 &= \psi,
\end{align*}
\]
where \( u_\psi \) is the driving speed. Fig 2 shows the kinematic model of tricycle robot. The steering angle \( \psi \) is constrained by
\[
||\psi|| \leq M, \ 0 < M < \pi/2.
\]
The constraint (8) is represented as
\[
\psi = M \text{sech}^2 w \mu_2 = u_2,
\]
\[
\frac{w}{\mu_2} = \psi
\]
where \( w \) is an auxiliary variable. Thus we get
\[
\begin{align*}
\dot{\psi} &= M \text{sech}^2 w \mu_2 = u_2, \\
\dot{w} &= \mu_2
\end{align*}
\]
We substitute the equations of \( \psi \) and \( w \) into the tricycle robot model. The model is obtained as
\[
\begin{align*}
\dot{x} &= \left( \begin{array}{c} \cos \theta \\ \sin \theta \\ \frac{1}{2} \tan(M\tan w) \end{array} \right) u_1 + \left( \begin{array}{c} 0 \\ 0 \\ 0 \end{array} \right) u_2,
\end{align*}
\]
(8)
In this case, we consider \( \eta = (x, y)^T \) as the output, the time derivative of \( \eta \) is calculated as

\[
\dot{\eta} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{w} \\ \dot{u}_1 \end{bmatrix} = \begin{bmatrix} u_1 \cos \theta \\ u_1 \sin \theta \\ \frac{1}{2} \tan(M \tan w) \\ \nu_1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \mu_1 \\ \mu_2 \end{bmatrix}.
\]

The linearized system of (8) at any points \((x, y, \theta, w)\) is clearly not controllable and the only \(u_1\) affects \(\dot{\eta}\). To proceed, we need to add some integrators of the input \(u_1\). Using dynamic compensators as

\[
\dot{u}_1 = \nu_1, \quad \dot{\nu}_1 = \mu_1,
\]

the tricycle robot model (8) can be dynamic feedback linearizable. The extended model is obtained as

\[
\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{w} \\ \dot{u}_1 \end{bmatrix} = \begin{bmatrix} u_1 \cos \theta \\ u_1 \sin \theta \\ \frac{1}{2} \tan(M \tan w) \\ \nu_1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \mu_1 \\ \mu_2 \end{bmatrix}.
\]

The time derivative of \(\dot{\eta}\) is calculated as

\[
\ddot{\eta} = \begin{bmatrix} L_2^2 h_1 \\ L_2^2 h_2 \end{bmatrix} = \begin{bmatrix} \nu_1 \cos \theta - \nu_2 \frac{1}{2} \tan(M \tan w) \sin \theta \\ \nu_1 \sin \theta + \nu_2 \frac{1}{2} \tan(M \tan w) \cos \theta \end{bmatrix},
\]

where \((h_1, h_2) = (x, y)\). Since the controller \((\mu_1, \mu_2)\) doesn’t appear in the equation \(\ddot{\eta}\), we continue to calculate the time derivative of \(\dot{\eta}\). Then we get

\[
\eta^{(3)} = L_3^2 h + L_0 L_2^2 h \mu
\]

\[
= \begin{bmatrix} L_3^2 h_1 \\ L_3^2 h_2 \end{bmatrix} + \begin{bmatrix} L_{g_1} L_2^2 h_1 \\ L_{g_2} L_2^2 h_2 \\ L_{g_3} L_2^2 h_3 \end{bmatrix} (\mu_1, \mu_2).
\]

Equation (10) shows clearly that the system is input-output linearizable because state feedback control

\[
\mu = -(L_0 L_2^2 h)^{-1} L_3^2 h + (L_0 L_2^2 h)^{-1} \nu
\]

reduces the input-output map to \(\eta^{(3)} = \nu\).

The matrix \(L_0 L_2^2 h\) multiplying the modified input \((\mu_1, \mu_2)\) is non-singular if \(u_1 \neq 0\). Since the modified input is obtained as \((\mu_1, \mu_2)\), the integrator with respect to the input \(\nu\) is added to the original input \((u_1, u_2)\). Finally, the stabilizing controller of the tricycle robot system (7) is presented as a dynamic feedback controller:

\[
\begin{cases}
\dot{u}_1 = \nu_1, & \dot{\nu}_1 = \mu_1, \\
u_2 = M \text{sech}^2 w \mu_2
\end{cases}
\]
where  \( f_3(d_i) = \tan(M \tanh d_i) / L \). We continue to calculate the time derivative of \( \dot{\eta} \). We get

\[
\eta_1^{(3)} = L_3^{(3)} h_1 + L_{g2} L_{f_3} h_1 \mu_1 + L_{g2} L_{f_3} h_1 \mu_2 \\
= x_3^2 \left( \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^i(x_3) f_1'(d_i) \right) \left( \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^i(x_4) f_3(d_i) \right) \\
+ 3 x_3 x_6 \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^i(x_3) f_1'(d_i) \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^i(x_4) f_3(d_i) \\
+ \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^i(x_3) f_2'(d_i) \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^i(x_4) f_3(d_i) \\
+ x_3^2 \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^i(x_3) f_2'(d_i) \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^i(x_4) f_3'(d_i) \mu_2.
\]

where \( \eta_1^{(3)} \) is the same as the linear system (13).

**C. Tracking Control for PML System**

We apply a tracking control [15] to the tricycle robot model (7). Consider the following reference signal model

\[
\begin{align*}
\dot{x}_r &= f_x, \\
\eta_r &= h_r.
\end{align*}
\]

The controller is designed to make the error signal \( e = (e_1, e_2)^T = \eta - \eta_r \to 0 \) as \( t \to \infty \). The time derivative of \( e \) is obtained as

\[
\dot{e} = \dot{\eta} - \dot{\eta}_r = \left( L_{f_x} h_{p_1} - L_{f_x} h_{p_2} \right) - \left( L_{f_x} h_{r_1} - L_{f_x} h_{r_2} \right).
\]

Furthermore, the time derivative of \( \dot{e} \) is calculated as

\[
\ddot{e} = \ddot{\eta} - \ddot{\eta}_r = \left( L_{f_x}^2 h_{p_1} - L_{f_x}^2 h_{p_2} \right) - \left( L_{f_x}^2 h_{r_1} - L_{f_x}^2 h_{r_2} \right).
\]

Since the controller \( \mu \) doesn’t appear in the equation \( \ddot{e} \), we calculate the time derivative of \( \dot{e} \).

\[
e^{(3)} = \eta^{(3)} - \eta^{(3)} = \left( L_{f_x}^3 h_{p_1} \right) + L_{f_x} L_{f_x} h_{p_2} - \left( L_{f_x}^3 h_{r_1} L_{f_x} h_{r_2} \right)
\]

The tracking controller is designed as

\[
\begin{align*}
\dot{u}_1 &= \mu_1, \\
u_2 &= M \sech^2 x_4 \mu_2, \\
\mu_1 &= \frac{1}{L_{f_x}^2 h} - L_{f_x} h_{r_1} + L_{f_x} L_{f_x} h_{r_2}.
\end{align*}
\]

The linearized system (13) and controller \( v = -F_z \) are obtained in the same manners as the previous subsection. The coordinate transformation vector is \( z = (e_1, e_2, \dot{e}_1, \dot{e}_2, \ddot{e}_1, \ddot{e}_2)^T \).

V. SIMULATION RESULTS

We apply the previous tracking control method to the tricycle robot model (7). Although the controller is simpler than the conventional I/O feedback linearization controller, the tracking performance based on PML model is the same as the conventional one. In addition, the controller is capable to use a nonlinear system with chaotic behavior as the reference model. In the following simulations, the tricycle length \( L \) is 1.0 [m] and the angle constrain \( M \) is \( \pi / 3 \) [rad.].
A. Ellipse-shaped reference trajectory

Consider an ellipse model as the reference trajectory.

\[
\begin{pmatrix} x_{r1} \\ x_{r2} \end{pmatrix} = \begin{pmatrix} R_1 \cos \theta + x_{r1}(0) \\ R_2 \sin \theta + x_{r2}(0) \end{pmatrix},
\]

where \(R_1\) and \(R_2\) are the semiminor axes and \((x_{r1}(0), x_{r2}(0))\) is the center of the ellipse. Fig. 3 shows the simulation result. The dotted line is the reference signal and the solid line is the tricycle tracking trajectory. The semiminor parameters \(R_1\) and \(R_2\) are 10 and 25. The initial positions are set at \((x(0), y(0)) = (5,0)\) and \((x_r(0), y_r(0)) = (10,0)\). Fig. 4 shows the control inputs \(u_1\) and \(v_1\) of the tricycle. Fig. 5 shows the error signals of the tricycle position \((x, y)\).

B. Trajectory tracking control using ellipse-shaped reference models

Arbitrary tracking trajectory control can be realized using the ellipse-shaped tracking trajectory method. The controller design procedure is as follows:

1) Assign passing points \((p_x(i), p_y(i)), i = 1, \ldots, n\).

We consider the passing points: \((0,0), (10,20), (26,30), (18,50)\) and \((2,70)\)

2) Construct some ellipses trajectories to connect the passing points smoothly.

From \((0,0)\) to \((10,20)\), the trajectory 1:

\[
\begin{pmatrix} x_{r1} \\ x_{r2} \end{pmatrix} = \begin{pmatrix} 10 \cos \theta + 10 \\ 20 \sin \theta \end{pmatrix},
\]

where \(\pi/2 \leq \theta \leq \pi\).

From \((10,20)\) to \((26,30)\), the trajectory 2:

\[
\begin{pmatrix} x_{r1} \\ x_{r2} \end{pmatrix} = \begin{pmatrix} 16 \cos \theta + 10 \\ 10 \sin \theta + 30 \end{pmatrix},
\]

where \(-\pi/2 \leq \theta \leq 0\).

From \((26,30)\) to \((18,50)\), the trajectory 3:

\[
\begin{pmatrix} x_{r1} \\ x_{r2} \end{pmatrix} = \begin{pmatrix} 8 \cos \theta + 18 \\ 20 \sin \theta + 30 \end{pmatrix},
\]

where \(0 \leq \theta \leq \pi/2\).

From \((18,50)\) to \((2,70)\), the trajectory 4:

\[
\begin{pmatrix} x_{r1} \\ x_{r2} \end{pmatrix} = \begin{pmatrix} 16 \cos \theta + 18 \\ 10 \sin \theta + 60 \end{pmatrix},
\]

where \(-\pi/2 \leq \theta \leq -\pi/2\).

3) Design the controllers (15) for the ellipse tracking trajectories (16)-(19).

We show a tracking trajectory control example for the tricycle robot system. Fig. 6 shows the reference signals (16)-(19) and the tricycle tracking trajectory. The dotted line is the reference signal and the solid line is the tricycle tracking trajectory. Fig. 7 shows the control inputs \(u_1\) and \(v_1\) of the tricycle. Fig. 8 shows the error signals with respect to the tricycle position \((x, y)\).
VI. Conclusions

We have proposed a trajectory tracking controller design of a tricycle robot as a non-holonomic system with PML models. The approximated model is fully parametric. I/O dynamic feedback linearization is applied to stabilize PML control system. PML modeling with feedback linearization is a very powerful tool for analyzing and synthesizing nonlinear control systems. We also have applied a method for tracking controller to the tricycle robot. Although the controller is simpler than the conventional I/O feedback linearization controller, the tracking performance based on PML model is the same as the conventional one. Examples have been shown to confirm the feasibility of our proposals by computer simulations.

REFERENCES


