2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)
July 6-11, 2014, Beijing, China

# Tracking Control for a Non-Holonomic Car-Like Robot Using Dynamic Feedback Linearization Based on Piecewise Bilinear Models

Tadanari Taniguchi, Luka Eciolaza, and Michio Sugeno

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

## I. Introduction

This paper deals with the tracking control of a car-like robot using dynamic feedback linearization based on piecewise bilinear (PB) 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 PB model as a piecewise approximation model of the car-like 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 PB 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.

T. Taniguchi is with IT Education Center, Tokai University, Hiratsuka, Kanagawa, 2591292 Japan, email: taniguchi@tokai-u.jp. L. Eciolaza and M. Sugeno are with European Centre for Soft Computing, 33600 Mieres, Asturias, Spain, emails: luka.eciolaza@softcomputing.es, michio.sugeno@gmail.com

This work was supported by a URP grant from Ford Motor Company which the authors thankfully acknowledge. In addition, this work was supported by Grant-in-Aid for Young Scientists (B: 23700276) of The Ministry of Education, Culture, Sports, Science and Technology in Japan.

We propose a dynamic feedback linearization for PB control system and apply the tracking control [15] to a car-like 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 PB 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 PB models are not of class of $C^\infty$. In the car-like robot control, we have to calculate the third derivatives of the output. Therefore the derivative PB models lose some dynamics. Thus we propose the derivative PB models of the trigonometric functions.

This paper is organized as follows. Section II introduces the canonical form of PB 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 PB model of the car-like 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 PB models suggested in [10]. We deal with the two-dimensional case without loss of generality. Define vector $d(\sigma,\tau)$ and rectangle $R_{\sigma\tau}$ in two-dimensional space as $d(\sigma,\tau) \equiv (d_1(\sigma), d_2(\tau))^T$,

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

$\sigma$ 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 PB system is expressed as

$$\begin{cases} \dot{x} = \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_1^i(x_1)\omega_2^j(x_2) f_o(i,j), \\ x = \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_1^i(x_1)\omega_2^j(x_2) d(i,j), \end{cases} \tag{1}$$

where $f_o(i,j)$ is the vertex of nonlinear system $\dot{x} = f_o(x)$,

$$\begin{cases} \omega_1^{\sigma}(x_1) = (d_1(\sigma+1) - x_1)/(d_1(\sigma+1) - d_1(\sigma)), \\ \omega_1^{\sigma+1}(x_1) = (x_1 - d_1(\sigma))/(d_1(\sigma+1) - d_1(\sigma)), \\ \omega_2^{\tau}(x_2) = (d_2(\tau+1) - x_2)/(d_2(\tau+1) - d_2(\tau)), \\ \omega_2^{\tau+1}(x_2) = (x_2 - d_2(\tau))/(d_2(\tau+1) - d_2(\tau)), \end{cases} \tag{2}$$

and $\omega_1^i(x_1), \omega_2^j(x_2) \in [0,\ 1]$. 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 $\omega_1^i(x_1)$ and $\omega_2^j(x_2)$, just as in the case of $\dot{x}$. As seen in equation (2), $x$ 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 (1) parametric expression.

Representing $\dot{x}$ with $x$ in Eqs. (1) and (2), we obtain the state space expression of the model found to be bilinear (biaffine) [10], so the derived PB model has simple nonlinearity. In PL approximation, a PL model is built on simplexes partitioned in state space, triangles in the two-dimensional case. Note that any three points in three-dimensional space are spanned with an affine plane: $y = a + bx_1 + cx_2$. A PL model is continuous. It is, however, difficult to handle simplexes in the rectangular coordinate system.

Note that any four points in three-dimensional space are spanned with a biaffine plane: $y = a + bx_1 + cx_2 + dx_1x_2$. In contrast to a PL model, a PB model as such is built on rectangles with the four vertices $d(i,j)$, on hypercubes in general dimensional space, partitioned in state space; it matches the rectangular coordinate system well, so PB models are applicable to control purposes.

### B. Closed-Loop Systems

We consider a two-dimensional nonlinear control system.

$$\begin{cases} \dot{x} = f_o(x) + g_o(x)u(x), \\ y = h_o(x). \end{cases} \tag{3}$$

The PB model (4) is constructed from a nonlinear system (3).

$$\begin{cases} \dot{x} = f(x) + g(x)u(x), \\ y = h(x), \end{cases} \tag{4}$$

where

$$\begin{cases} f(x) = \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_1^i(x_1)\omega_2^j(x_2) f_o(i,j), \\ g(x) = \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_1^i(x_1)\omega_2^j(x_2) g_o(i,j), \\ h(x) = \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_1^i(x_1)\omega_2^j(x_2) h_o(i,j), \\ \quad x = \sum_{i=\sigma}^{\sigma+1} \sum_{j=\tau}^{\tau+1} \omega_1^i(x_1)\omega_2^j(x_2) d(i,j), \end{cases} \tag{5}$$

and $f_o(i,j)$, $g_o(i,j)$, $h_o(i,j)$ and $d(i,j)$ are vertices of the nonlinear system (3). 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.
2) Compute vertices $f_o(i,j)$, $g_o(i,j)$ and $h_o(i,j)$ in equation (5) 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 (3).

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

## III. Dynamic Feedback Linearization of Car-Like Robot

We consider a car-like robot model.

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ \frac{1}{L}\tan\psi \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_2, \tag{6}$$

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 vehicle and the $x$ axis, $\psi$ is the steering angle with respect to the car. The control inputs are represented as

$$\begin{aligned} u_1 &= v_s \cos\psi \\ u_2 &= \dot{\psi}, \end{aligned}$$

where $v_s$ is the driving speed. Fig 1 shows the kinematic model of car-like robot. The steering angle $\psi$ is constrained by

$$\|\psi\| \le M,\ 0 < M < \pi/2.$$

The constraint [8] is represented as

$$\psi = M \tanh w,$$

where $w$ is an auxiliary variable. Thus we get

$$\begin{aligned} \dot{\psi} &= M \mathrm{sech}^2 w \mu_2 = u_2, \\ \dot{w} &= \mu_2 \end{aligned}$$

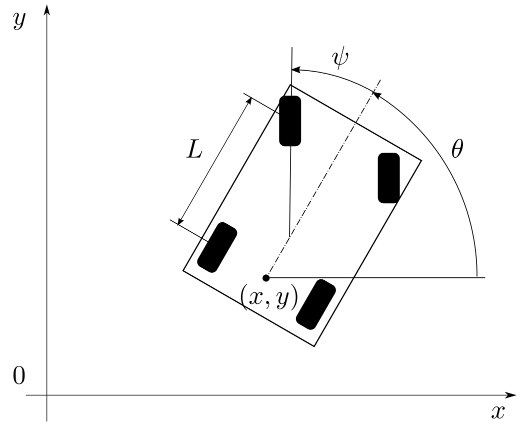


Fig. 1. Kinematic model of car-like robot

We substitute the equations of $\psi$ and $w$ into the car-like robot model. The model is obtained as

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{w} \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ \frac{1}{L}\tan(M\tanh w) \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \mu_2 \tag{7}$$

In this case, we consider $\eta = (x, y)^T$ as the output, the time derivative of $\eta$ is calculated as

$$\dot{\eta} = \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 \\ \sin\theta & 0 \end{pmatrix} \begin{pmatrix} u_1 \\ \mu_2 \end{pmatrix}.$$

The linearized system of (7) 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 car-like robot model (7) can be dynamic feedback linearizable. The extended model is obtained as

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{w} \\ \dot{u}_1 \\ \dot{\nu}_1 \end{pmatrix} = \begin{pmatrix} u_1\cos\theta \\ u_1\sin\theta \\ u_1\frac{1}{L}\tan(M\tanh w) \\ 0 \\ \nu_1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \mu_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \mu_2 \tag{8}$$

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

$$\ddot{\eta} = \begin{pmatrix} L_f^2 h_1 \\ L_f^2 h_2 \end{pmatrix} = \begin{pmatrix} \nu_1\cos\theta - u_1^2\frac{1}{L}\tan(M\tanh w)\sin\theta \\ \nu_1\sin\theta + u_1^2\frac{1}{L}\tan(M\tanh w)\cos\theta \end{pmatrix},$$

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

$$\begin{aligned} \eta^{(3)} =& L_f^3 h + L_g L_f^2 h\mu \\ =& \begin{pmatrix} L_f^3 h_1 \\ L_f^3 h_2 \end{pmatrix} + \begin{pmatrix} L_{g_1}L_f^2 h_1 & L_{g_2}L_f^2 h_1 \\ L_{g_1}L_f^2 h_2 & L_{g_2}L_f^2 h_2 \end{pmatrix} \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \end{aligned} \tag{9}$$

where

$$\begin{aligned} L_f^3 h_1 =& -3\nu_1\frac{1}{L}\{\tan(M\tanh w)\}u_1\sin\theta \\ & - u_1^3\left(\frac{1}{L}\tan(M\tanh w)\right)^2\cos\theta, \\ L_f^3 h_2 =& 3\nu_1\frac{1}{L}\{\tan(M\tanh w)\}u_1\cos\theta \\ & - u_1^3\left(\frac{1}{L}\tan(M\tanh w)\right)^2\sin\theta, \\ L_{g_1}L_f^2 h_1 =& \cos\theta, \ L_{g_1}L_f^2 h_2 = \sin\theta, \\ L_{g_2}L_f^2 h_1 =& -u_1^2\frac{1}{L}\frac{\partial}{\partial w}\{\tan(M\tanh w)\}\sin\theta, \\ L_{g_2}L_f^2 h_2 =& u_1^2\frac{1}{L}\frac{\partial}{\partial w}\{\tan(M\tanh w)\}\cos\theta, \\ g_1 =& (0,0,0,0,0,1)^T, g_2 = (0,0,0,1,0,0)^T. \end{aligned}$$

Equation (9) shows clearly that the system is input-output linearizable because state feedback control

$$\mu = -(L_g L_f^2 h)^{-1} L_f^3 h + (L_g L_f^2 h)^{-1} v$$

reduces the input-output map to $y^{(3)} = v$.

The matrix $L_g L_f^2 h$ 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 $v$ is added to the original input $(u_1, u_2)$. Finally, the stabilizing controller of the car-like robot system (6) is presented as a dynamic feedback controller:

$$\begin{cases} \dot{u}_1 = \nu_1, \ \dot{\nu}_1 = \mu_1, \\ u_2 = M\mathrm{sech}^2 w\mu_2 \end{cases} \tag{10}$$

## IV. PB Modeling and Tracking Controller Design of the Car-Like Robot Model

### A. PB Model of the Car-Like Robot Model

We construct PB model of the car-like robot system (8). The state spaces of $\theta$ and $w$ in the car-like robot model (8) are divided by the 13 vertices $x_3 \in \{-\pi, -5\pi/6, \ldots, \pi\}$ and the 13 vertices $x_4 \in \{-3.0, -2.5, \ldots, 3.0\}$. The state variable is $x = (x_1, x_2, x_3, x_4, x_5, x_6)^T = (x, y, \theta, w, u_1, v_1)^T$.

$$\dot{x} = \begin{pmatrix} \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_1(d_3(i_3)) x_5 \\ \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_2(d_3(i_3)) x_5 \\ \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^{i_4}(x_4) f_3(d_4(i_4)) x_5 \\ 0 \\ x_6 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \mu_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \mu_2. \tag{11}$$

We can construct PB models with respect to $f_1(x)$, $f_2(x)$ and $f_3(x)$. The PB model structures are independent of the vertex positions $x_5$ and $x_6$ since $x_5$ and $x_6$ are the linear terms. This paper constructs the PB models with respect to the nonlinear terms of $x_3$ and $x_4$.

Note that trigonometric functions of the car-like robot (8) are smooth functions and are of class $C^\infty$. The PB models are not of class $C^\infty$. In the car-like robot control, we have to calculate the third derivatives of the output $y$. Therefore the derivative PB models lose some dynamics. In this paper we propose the derivative PB models of the trigonometric functions (see Appendix I).

### B. Tracking Controller Design Using Dynamic Feedback Linearization Based on PB Model

We define the output as $\eta = (x_1, x_2)^T$ in the same manner as the previous section, the time derivative of $\eta$ is calculated as

$$\dot{\eta} = \begin{pmatrix} L_{f_p} h_1 \\ L_{f_p} h_2 \end{pmatrix} = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) \begin{pmatrix} f_1(d_3(i_3))x_5 \\ f_2(d_3(i_3))x_5 \end{pmatrix}$$

where the vertices are $f_1(d_3(i_3)) = \cos d_3(i_3)$ and $f_2(d_3(i_3)) = \sin d_3(i_3)$. The time derivative of $\eta$ doesn't contain the control inputs $(\mu_1, \mu_2)$. We calculate the time derivative of $\dot{\eta}$. We get

$$\begin{aligned}
\ddot{\eta}_1 =& L_{f_p}^2 h_1 = \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_1(d_3(i_3)) x_6 \\
&+ \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_1'(d_3(i_3)) \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^{i_4}(x_4) f_3(d_4(i_4)) x_5^2, \\
\ddot{\eta}_2 =& L_{f_p}^2 h_2 = \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_2(d_3(i_3)) x_6 \\
&+ \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_2'(d_3(i_3)) \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^{i_4}(x_4) f_3(d_4(i_4)) x_5^2,
\end{aligned}$$

where $f_3(d_4(i_4)) = \tan(M \tanh d_4(i_4))/L$. The vertices $f_1'(d_3(i_3))$ and $f_2'(d_3(i_3))$ are given in Appendix I. We continue to calculate the time derivative of $\ddot{\eta}$. We get

$$\begin{aligned}
\eta_1^{(3)} =& L_{f_p}^3 h_1 + L_{g_1} L_{f_p}^2 h_1 \mu_1 + L_{g_2} L_{f_p}^2 h_1 \mu_2 \\
=& x_5^3 \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_1^{''}(d_3(i_3)) \left( \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^{i_4}(x_4) f_3(d_4(i_4)) \right)^2 \\
&+ 3x_5 x_6 \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_1^{'}(d_3(i_3)) \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^{i_4}(x_4) f_3(d_4(i_4)) \\
&+ \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_1(d_3(i_3)) \mu_1 \\
&+ x_5^2 \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_1'(d_3(i_3)) \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^{i_4}(x_4) f_3'(d_4(i_4)) \mu_2,
\end{aligned}$$

$$\begin{aligned}
\eta_2^{(3)} =& L_{f_p}^3 h_2 + L_{g_1} L_{f_p}^2 h_2 \mu_1 + L_{g_2} L_{f_p}^2 h_2 \mu_2 \\
=& x_5^3 \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_2^{''}(d_3(i_3)) \left( \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^{i_4}(x_4) f_3(d_4(i_4)) \right)^2 \\
&+ 3x_5 x_6 \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_2^{'}(d_3(i_3)) \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^{i_4}(x_4) f_3(d_4(i_4)) \\
&+ \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_2(d_3(i_3)) \mu_1 \\
&+ x_5^2 \sum_{i_3=\sigma_3}^{\sigma_3+1} w_3^{i_3}(x_3) f_2'(d_3(i_3)) \sum_{i_4=\sigma_4}^{\sigma_4+1} w_4^{i_4}(x_4) f_3'(d_4(i_4)) \mu_2.
\end{aligned}$$

The vertices $f_1^{''}(d_3(i_3))$, $f_2^{''}(d_3(i_3))$ and $f_3^{'}(d_4(i_4))$ are also given in Appendix I. The controller of (11) is designed as

$$\begin{aligned}
(\mu_1, \mu_2)^T =& -(L_g L_{f_p}^2 h)^{-1} L_{f_p}^3 h + (L_g L_{f_p}^2 h)^{-1} v \\
=& -\begin{pmatrix} L_{g_1} L_{f_p}^2 h_1 & L_{g_2} L_{f_p}^2 h_1 \\ L_{g_1} L_{f_p}^2 h_2 & L_{g_2} L_{f_p}^2 h_2 \end{pmatrix}^{-1} \begin{pmatrix} L_{f_p}^3 h_1 \\ L_{f_p}^3 h_2 \end{pmatrix} \\
&+ \begin{pmatrix} L_{g_1} L_{f_p}^2 h_1 & L_{g_2} L_{f_p}^2 h_1 \\ L_{g_1} L_{f_p}^2 h_2 & L_{g_2} L_{f_p}^2 h_2 \end{pmatrix}^{-1} v
\end{aligned}$$

where $v$ is the linear controller of the linear system (12).

$$\begin{cases} \dot{z} = Az + Bu, \\ y = Cz, \end{cases} \tag{12}$$

where $z = (h_1,\ L_{f_p} h_1,\ L_{f_p}^2 h_1,\ h_2,\ L_{f_p} h_2,\ L_{f_p}^2 h_2)^T \in \Re^6$,

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}, C = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}^T.$$

If $x_5 \neq 0$, there exists a controller $(\mu_1, \mu_2)^T$ of the car-like robot model (11) since $\det(L_g L_{f_p}^2 h) \neq 0$.

In this case, the state space of the car-like robot model is divided into $13 \times 13$ vertices. Therefore the system has $12 \times 12$ local PB models. Note that all the linearized systems of these PB models are the same as the linear system (12).

In the same manner of (10), the dynamic feedback linearizing controller of the PB system is designed as

$$\begin{cases} \ddot{u}_1 = \mu_1, \\ u_2 = M \mathrm{sech}^2 x_4 \mu_2, \\ \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} = L_{f_p}^3 h + L_g L_f^2 h v. \end{cases} \tag{13}$$

The stabilizing linear controller $v = -Fz$ of the linearized system (12) is designed so that the transfer function $C(sI - A)^{-1}B$ is Hurwitz.

Note that the dynamic controller (13) based on PB model is simpler than the conventional one (10). Since the nonlinear terms of controller (13) contain not the original nonlinear terms (e.g., $\sin x_3$, $\cos x_3$, $\tan(M \tanh x_4)$) but the piecewise approximation models.

### C. Tracking Control for PB System

We apply a tracking control [15] to the car-like robot model (6). Consider the following reference signal model

$$\begin{cases} \dot{x}_r = f_r, \\ \eta_r = h_r. \end{cases}$$

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

$$\dot{e} = \dot{\eta} - \dot{\eta}_r = \begin{pmatrix} L_{f_p} h_{p1} \\ L_{f_p} h_{p2} \end{pmatrix} - \begin{pmatrix} L_{f_r} h_{r1} \\ L_{f_r} h_{r2} \end{pmatrix}.$$

Furthermore the time derivative of $\dot{e}$ is calculated as

$$\ddot{e} = \ddot{\eta} - \ddot{\eta}_r = \begin{pmatrix} L_{f_p}^2 h_{p1} \\ L_{f_p}^2 h_{p2} \end{pmatrix} - \begin{pmatrix} L_{f_r}^2 h_{r1} \\ L_{f_r}^2 h_{r2} \end{pmatrix}$$

Since the controller $\mu$ doesn't appear in the equation $\ddot{e}$, we calculate the time derivative of $\ddot{e}$.

$$\begin{aligned} e^{(3)} &= \eta^{(3)} - \eta_r^{(3)} \\ &= \begin{pmatrix} L_{f_p}^3 h_{p1} \\ L_{f_p}^3 h_{p2} \end{pmatrix} + L_g L_{f_p}^2 h \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} - \begin{pmatrix} L_{f_r}^3 h_{r1} \\ L_{f_r}^3 h_{r2} \end{pmatrix} \end{aligned}$$

The tracking controller is designed as

$$\begin{cases} \dot{u}_1 = \mu_1, \\ u_2 = M \operatorname{sech}^2 x_4 \mu_2, \\ \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} = L_{f_p}^3 h - L_{f_r}^3 h_r + L_g L_{f_p}^2 h v. \end{cases} \tag{14}$$

The linearized system (12) and controller $v = -Fz$ are obtained in the same manners as the previous subsection. The coordinate transformation vector is $z = (e_1,\ \dot{e}_1,\ \ddot{e}_1, e_2,\ \dot{e}_2,\ \ddot{e}_2)^T$.

Note that the dynamic controller (14) based on PB model is simpler than the conventional one on the same reason of the previous subsection.

## V. Simulation Results

We consider two tracking trajectories as the reference signal models. Although the controllers are simpler than the conventional I/O feedback linearization controllers, the tracking performance based on PB model is the same as the conventional one [8].

### A. Circle-shaped reference trajectory

Consider a circle-shaped reference trajectory [8] as the reference model.

$$\begin{pmatrix} x_{r_1} \\ x_{r_2} \end{pmatrix} = \begin{pmatrix} R \sin(\omega t) \\ R \cos(\omega t) \end{pmatrix},$$

where $R = 15$ and $\omega = 0.01\pi$. The feedback gain is calculated as

$$F = \begin{pmatrix} 0.0316 & 0.2035 & 0.6387 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0316 & 0.2035 & 0.6387 \end{pmatrix}.$$

The initial positions are set at $(x, y) = (0, 0)$ and $(x_r, y_r) = (R, 0)$. In this simulation, the constraint of the steering angle is $M = \pi/3$ and the wheel base is $L = 1$. Fig 2 shows the trajectories of $x$-$y$ plane. The solid line is the state response $(x, y)$ of the PB control system and the dotted line is the reference signal. Fig 3 shows the errors of the trajectories. Fig 4 shows the control inputs $(u_1, u_2)$ and the steering angle $\psi$. These results confirm the feasibility of the tracking control performance and the constraint of the steering angle.

### B. Eight-shaped reference trajectory

Consider an eight-shaped reference trajectory [7] as the reference model.

$$\begin{pmatrix} x_{r_1} \\ x_{r_2} \end{pmatrix} = \begin{pmatrix} \sin \frac{t}{10} \\ \sin \frac{t}{20} \end{pmatrix}$$

The feedback gain is calculated as

$$F = \begin{pmatrix} 0.3162 & 1.011 & 1.456 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.3162 & 1.011 & 1.456 \end{pmatrix}.$$

The initial positions are set at $(x, y) = (-1, -1)$ and $(x_r, y_r) = (0, 0)$. We set that the constraint of the steering angle is $M = \pi/3$ and the wheel base is $L = 1$. Fig 5 shows the trajectories of $x$-$y$ plane. The solid line is the state response $(x, y)$ of the PB control system and the dotted line is the reference signal. Fig 6 shows the errors of the trajectories. Fig 7 shows the control inputs $(u_1, u_2)$ and the steering angle $\psi$. These results also confirm the feasibility of the tracking control performance and the constraint of the steering angle. Although the controller (14) and the PB model (11) are simpler than the conventional dynamic feedback linearization controller and model [7], the control performance based on PB model is the same as the conventional one [8].

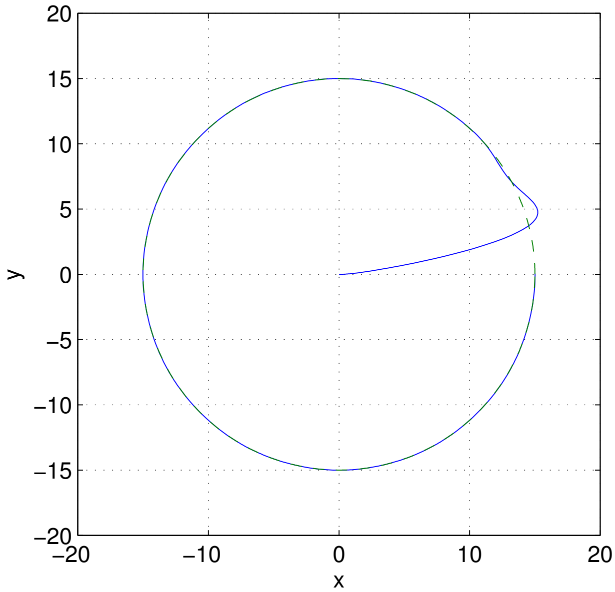


Fig. 2. Circle-shaped trajectories on the $x$-$y$ plane

## VI. Conclusions

We have proposed a dynamic feedback linearization of a car-like robot as a non-holonomic system with PB models. The approximated model is fully parametric. I/O dynamic

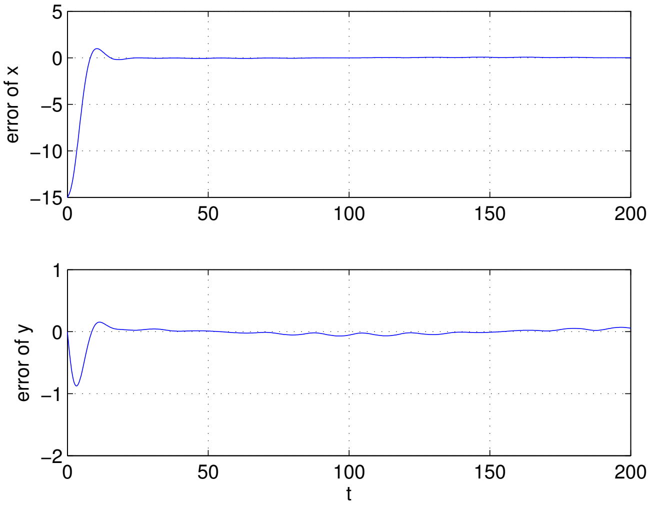


Fig. 3. Errors of circle-shaped trajectories

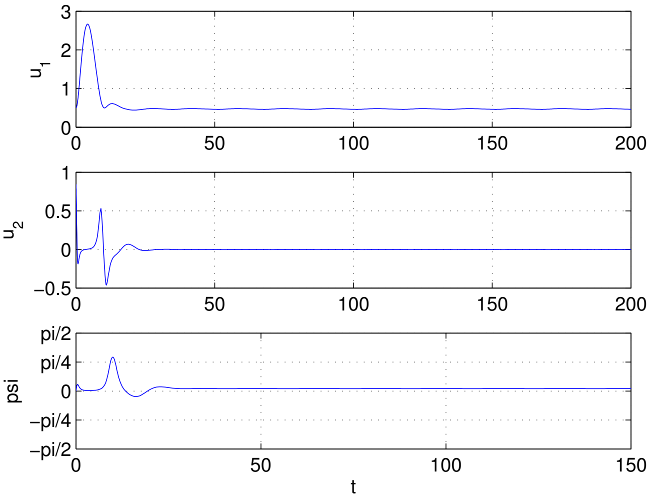


Fig. 4. Control inputs $(u_1, u_2)$ and the steering angle $\psi$

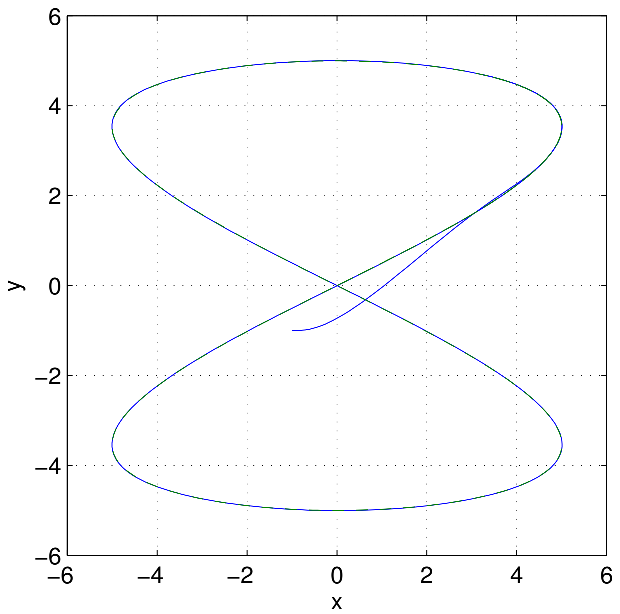


Fig. 5. Eight-shaped trajectories on the $x$-$y$ plane

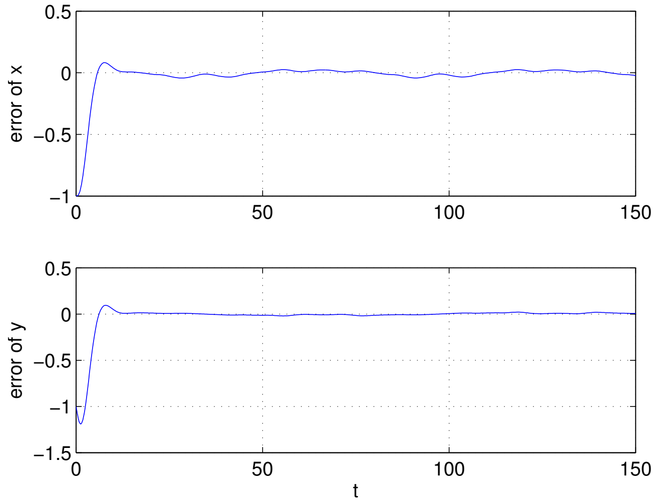


Fig. 6. Errors of eight-shaped trajectories

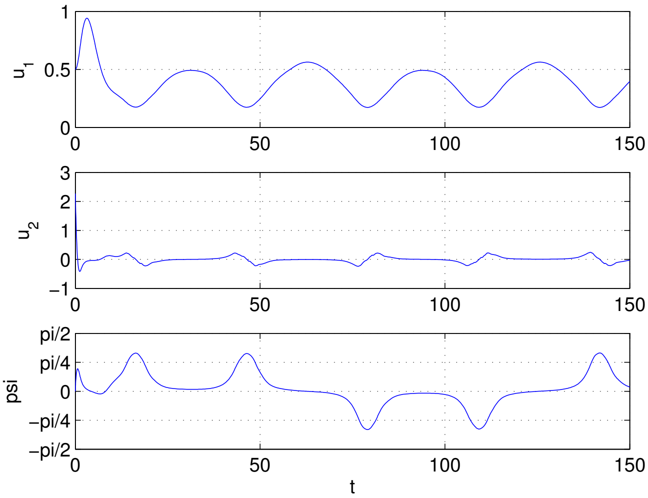


Fig. 7. Control inputs $(u_1,u_2)$ and the steering angle $\psi$

feedback linearization is applied to stabilize PB control system. PB 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 car-like robot. Although the controller is simpler than the conventional I/O feedback linearization controller, the tracking performance based on PB model is the same as the conventional one. Examples have been shown to confirm the feasibility of our proposals by computer simulations.

## APPENDIX I
### DERIVATIVE PB MODELS OF TRIGONOMETRIC FUNCTIONS

Trigonometric functions are smooth functions and are of class $C^\infty$. The PB models are not of class $C^\infty$. In the car-like robot control, we have to calculate the third derivatives of the output $y$. Therefore the derivative PB models lose some dynamics. In this paper we propose the derivative PB models of the trigonometric functions ($\cos x_3$, $\sin x_3$ and $\tan(M \tanh x_4)/L$).

The PB models of these trigonometric functions are constructed as

$$\xi_1 = \sum_{i_3=\sigma_3}^{\sigma_3+1} \omega_3^{i_3}(x_3) f_1(d_3(i_3)),\ \xi_2 = \sum_{i_3=\sigma_3}^{\sigma_3+1} \omega_3^{i_3}(x_3) f_2(d_3(i_3)),$$

$$\xi_3 = \sum_{i_4=\sigma_4}^{\sigma_4+1} \omega_4^{i_4}(x_4) f_3(d_4(i_4)),$$

where $f_1(d_3(i_3)) = \cos(d_3(i_3))$, $f_2(d_3(i_3)) = \sin(d_3(i_3))$, $f_3(d_4(i_4)) = \tan(M \tanh d_4(i_4))/L$, $i_3 = \sigma_3, \sigma_3+1$, $i_4 = \sigma_4, \sigma_4+1$.

The derivative PB models $\xi_1$, $\xi_2$ and $\xi_3$ are defined as

$$\frac{\partial \xi_1}{\partial x_3} := \sum_{i_3=\sigma_3}^{\sigma_3+1} \omega_3^{i_3}(x_3) f_1'(d_3(i_3)),$$

$$\frac{\partial \xi_2}{\partial x_3} := \sum_{i_3=\sigma_3}^{\sigma_3+1} \omega_3^{i_3}(x_3) f_2'(d_3(i_3)),$$

$$\frac{\partial \xi_3}{\partial x_4} := \sum_{i_4=\sigma_4}^{\sigma_4+1} \omega_4^{i_4}(x_4) f_3'(d_4(i_4)),$$

where $f_1'(d_3(i_3)) = -\sin(d_3(i_3))$, $f_2'(d_3(i_3)) = \cos(d_3(i_3))$, $f_3'(d_4(i_4)) = -M(\tan^2(M \tanh d_4(i_4))/L^2 + 1)(\tanh^2 d_4(i_4) - 1)/L$.

The derivative PB models $\partial\xi_1/\partial x_3$ and $\partial\xi_2/\partial x_1$ are also defined as

$$\frac{\partial^2 \xi_1}{\partial x_3^2} := \sum_{i_3=\sigma_3}^{\sigma_3+1} \omega_3^{i_3}(x_3) f_1^{''}(d_3(i_3)),$$

$$\frac{\partial^2 \xi_2}{\partial x_3^2} := \sum_{i_3=\sigma_3}^{\sigma_3+1} \omega_3^{i_3}(x_3) f_2^{''}(d_3(i_3)),$$

where $f_1^{''}(d_3(i_3)) = -\cos(d_3(i_3))$, $f_2^{''}(d_3(i_3)) = -\sin(d_3(i_3))$.

Table I shows a part of PB models with respect to $f_1(d_3(i_3))$ and $f_2(d_3(i_3))$. Table II shows a part of PB model with respect to $f_3(d_4(i_4))$.

TABLE I

A PART OF PB MODELS WITH RESPECT TO $f_1$ AND $f_2$

| $d_3(d_3(i_3))$ | $-\pi/3$ | $-\pi/6$ | 0 | $\pi/6$ | $\pi/3$ |
|---|---|---|---|---|---|
| $f_1(d_3(i_3))$ | 0.5 | 0.866 | 1 | 0.866 | 0.5 |
| $f_1'(d_3(i_3))$ | 0.866 | 0.5 | 0 | -0.5 | -0.866 |
| $f_1^{''}(d_3(i_3))$ | -0.5 | -0.866 | -1 | -0.866 | -0.5 |
| $f_2(d_3(i_3))$ | -0.866 | -0.5 | 0 | 0.5 | 0.866 |
| $f_2'(d_3(i_3))$ | 0.5 | 0.866 | 1 | 0.866 | 0.5 |
| $f_2^{''}(d_3(i_3))$ | 0.866 | 0.5 | 0 | -0.5 | -0.866 |

TABLE II

A PART OF PB MODEL WITH RESPECT TO $f_3$ $(M = \pi/3, L = 1)$

| $d_4(d_4(i_4))$ | -1.0 | -0.5 | 0 | 0.5 | 1.0 |
|---|---|---|---|---|---|
| $f_3(d_4(i_4))$ | -1.025 | -0.5256 | 0 | 0.5256 | 1.025 |
| $f_3'(d_4(i_4))$ | 0.9015 | 1.051 | 1.047 | 1.051 | 0.9015 |


## ACKNOWLEDGMENT

The authors would like to thank Dr. Dimitar Filev and Dr. Yan Wang of Ford Motor Company for his valuable comments and discussions.


## REFERENCES


[1] B. d'Andréa-Novel, G. Bastin, and G. Campion, "Modeling and control of non holonomic wheeled mobile robots," in *the 1991 IEEE International Conference on Robotics and Automation*, 1991, pp. 1130–1135.

[2] R. Fierro and F. L. Lewis, "Control of a nonholonomic mobile robot: backstepping kinematics into dynamics," in *the 34th Conference on Decision and Control*, 1995, pp. 3805–3810.

[3] T.-C. Lee, K.-T. Song, C.-H. Lee, and C.-C. Teng, "Tracking control of unicycle-modeled mobile robots using a saturation feedback controller," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 2, pp. 305–318, 2001.

[4] J. Guldner and V. I. Utkin, "Stabilization of non-holonomic mobile robots using lyapunov functions for navigation and sliding mode control," in *the 33rd Conference on Decision and Control*, 1994, pp. 2967–2972.

[5] J. Yang and J. Kim, "Sliding mode control for trajectory tracking of nonholonomic wheeled mobile robots," *IEEE Transactions on Robotics and Automation*, pp. 578–587, 1999.

[6] B. d'Andréa-Novel, G. Bastin, and G. Campion, "Dynamic feedback linearization of nonholonomic wheeled mobile robot," in *the 1992 IEEE International Conference on Robotics and Automation*, 1992, pp. 2527–2531.

[7] G. Oriolo, A. D. Luca, and M. Vendittelli, "WMR control via dynamic feedback linearization: Design, implementation, and experimental validation," *IEEE Transaction on Control System Technology*, vol. 10, no. 6, pp. 835–852, 2002.

[8] E. Yang, D. Gu, T. Mita, and H. Hu, "Nonlinear tracking control of a car-like mobile robot via dynamic feedback linearization," in *University of Bath, UK*, no. ID-218, 2004.

[9] A. D. Luca, G. Oriolo, and M. Vendittelli, "Stabilization of the unicycle via dynamic feedback linearization," in *the 6th IFAC Symposium on Robot Control*, 2000.

[10] M. Sugeno, "On stability of fuzzy systems expressed by fuzzy rules with singleton consequents," *IEEE Trans. Fuzzy Syst.*, vol. 7, no. 2, pp. 201–224, 1999.

[11] K.-C. Goh, M. G. Safonov, and G. P. Papavassilopoulos, "A global optimization approach for the BMI problem," in *Proc. the 33rd IEEE CDC*, 1994, pp. 2009–2014.

[12] T. Taniguchi and M. Sugeno, "Piecewise bilinear system control based on full-state feedback linearization," in *SCIS & ISIS 2010*, 2010, pp. 1591–1596.

[13] ——, "Stabilization of nonlinear systems with piecewise bilinear models derived from fuzzy if-then rules with singletons," in *FUZZ-IEEE 2010*, 2010, pp. 2926–2931.

[14] ——, "Design of LUT-controllers for nonlinear systems with PB models based on I/O linearization," in *FUZZ-IEEE 2012*, 2012, pp. 997–1022.

[15] T. Taniguchi, L. Eciolaza, and M. Sugeno, "Look-Up-Table controller design for nonlinear servo systems with piecewise bilinear models," in *FUZZ-IEEE 2013*, 2013.