IEEE/RSJ International Workshop on Intelligent Robots and Systems IROS '91, Nov. 3-5, 1991, Osaka, Japan. IEEE Cat. No. 91TH0375-6

A Stable Tracking Control Method for a Non-Holonomic Mobile Robot

Yutaka Kanayama

Dept of Computer Science
Naval Postgraduate School
Monterey, CA 93943

Yoshihiko Kimura

Center for Robotic Systems
University of California
Santa Barbara, CA 93106

Mitsubishi Material Corp.
Tokyo, Japan

Fumio Miyazaki

Osaka University
Dept of Mechanical Engineering
Toyonaka, Japan

Tetsuo Noguchi

Center for robotic Systems
University of California
Santa Barbara, CA 93106

ABSTRACT

The major objective of this paper is to propose a stable control rule to find a reasonable target linear and rotational velocities $(v, \omega)^t$. The stability of the rule is proved through the use of a Liapunov function. The rule contains three parameters, K_x, K_y and K_θ. Although any set of positive parameters makes the system stable, a condition on the parameters for the system being critically damped for a small disturbance is obtained through linearizing the system's differential equation. This method was successfully implemented on the autonomous mobile robot Yamabico-11. Experimental results obtained turn out to be close to the results with the velocity/acceleration limiter.

1. Introduction

The purpose of this paper is to propose a stable tracking control method for a non-holonomic vehicle using a Liapunov function. Tsumura proposed a method in which the reference point sequence is stored in memory. In each cycle of the locomotion control the reference point and the future position of the robot are compared to determine the next steering orders [1]. Kanayama proposed a method using straight line reference for the robot's locomotion instead of a sequence of points [2]. Its velocity and steering control method has some similarities to the one proposed in this paper. Crowley developed a locomotion control system whose organization has a three layered structure [3]. He defines the concept of "virtual vehicle" which is useful for constructing a system which is robot independent. In its command system independent control of linear and rotational motion is possible, thus enabling smooth clothoid curves [4]. Singh used an inverse kinematic and a quintic polynomial method for compensating errors in vehicle tracking [5]. In a second method he connects the current point and a future reference point by a smooth curve. Kanayama proposed the use of a reference and current configurations for vehicle control, the use of a local error coordinate system, and a PI control algorithm for linear/rotational velocity rules in an earlier locomotion control method on the Yamabico-11 robot [6]. Nelson proposed a locomotion control method for a cart with a front steering wheel, in which they also used the error coordinate system [7]. They adopted a linear function in control rules for steering and linear velocity. These two papers are regarded as the pioneers of this paper.

In this paper a new control rule for determining vehicle's linear and rotational velocities is given, which is different from both of [6] and [7]. One of the novelties in this paper is that the stability of the control rule given in this paper is proven using a Liapunov function [8][9][10]. The use of the trace function $(1 - \cos\theta)$ of the rotation matrix of θ led to a success in finding an appropriate Liapunov function [10]. One of the difficulties of this problem lies in the fact that ordinary vehicles possess only two degrees of freedom (linear velocity v and rotational velocity ω) for locomotion control, although vehicles have three degrees of freedom, x, y and θ in its positioning. (in other words, just because the vehicle is non-holonomic). Another difficulty is in the non-linearity of the kinematic relation between $(v, \omega)^t$ and $(\dot{x}, \dot{y}, \dot{\theta})^t$. The use of a Liapunov function resolves these difficulties.

This method was implemented and verified on the autonomous mobile robot Yamabico-11 which has been developed at the University of Tsukuba, at the University of California at Santa Barbara, and at Naval Postgraduate School [11][12][13].

2. Problem Statements

Before stating the problem, we give a few preliminary definitions.

2.1. Path Representation and Vehicle Kinematics

Suppose there is a mobile robot which is located on a 2D plane in which a global Cartesian coordinate system is defined. We also assume a local coordinate system fixed on the robot, where its X-axis coincides with its front orientation. The *configuration*,

$$\mathbf{p} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \tag{1}$$

of the robot represents the position and orientation of the local coordinate system in the global frame, where the orientation θ is taken counterclockwise from the global X-axis. This configuration stands for the three degrees of freedom which the robot possesses in its world [15]. Let $\mathbf{0}$ denote a *identity configuration* $(0, 0, 0)^t$.

The vehicle's motion is controlled by its *linear velocity* v

and *rotational velocity* ω, which are also functions of time. The vehicle's kinematics is defined by a *Jacobian matrix J*:

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = \dot{\mathbf{p}} = J\mathbf{q} \equiv \begin{bmatrix}\cos\theta & 0\\ \sin\theta & 0\\ 0 & 1\end{bmatrix}\mathbf{q} \tag{2}$$

where $\mathbf{q} = (v, \omega)^t$. This kinematics is common to all kinds of vehicles which are not omni-directional. (For instance, an automobile, a bicycle, a vehicle with two parallel independent power wheels (a power wheeled steering system), and a tricycle) The linear and rotational velocities of this kind of vehicle are controlled by its accelerator and steering wheel (or handle) respectively.

2.2. Error Configuration

In this control system, two configurations are used; the *reference configuration* $\mathbf{p}_r = (x_r, y_r, \theta_r)^t$ and the *current configuration* $\mathbf{p}_c = (x_c, y_c, \theta_c)^t$. The reference configuration is its goal configuration of the vehicle. The current configuration is its "real" configuration at this moment. We will define the *error configuration* $\mathbf{p}_e$ of the two, which is a transformation of the reference configuration $\mathbf{p}_r$ in a local coordinate system with an origin of (x_c, y_c) and the X-axis in the orientation of θ_c [6][7]. This is the "difference" between $\mathbf{p}_r$ and $\mathbf{p}_c$:

$$\mathbf{p}_e = \begin{bmatrix}x_e\\ y_e\\ \theta_e\end{bmatrix} = \begin{bmatrix}\cos\theta_c & \sin\theta_c & 0\\ -\sin\theta_c & \cos\theta_c & 0\\ 0 & 0 & 1\end{bmatrix}(\mathbf{p}_r - \mathbf{p}_c) \equiv T_e(\mathbf{p}_r - \mathbf{p}_c) \tag{3}$$

If $\mathbf{p}_r = \mathbf{p}_c$, the error configuration $\mathbf{p}_e = \mathbf{0}$. If $\mathbf{p}_r$ is ahead of $\mathbf{p}_c$ (the vehicle is behind of the goal), $x_e > 0$.

2.3. Problem

Now we are able to state the architecture of a tracking control system for the vehicle (Fig. 1). The global input of the system is the reference configuration $\mathbf{p}_r$ and *reference velocities* $\mathbf{q}_r = (v_r, \omega_r)^t$, which are both variables of time. The global output of the system is the current configuration $\mathbf{p}_c$. The purpose of the tracking controller is to converge the error configuration to $\mathbf{0}$. Let us describe each component in Figure 1 from left to right. The first component calculates an error configuration from $\mathbf{p}_r$ and $\mathbf{p}_c$ using Equation 3. The second box is a *control rule* for the vehicle, which calculates a *target velocities* $\mathbf{q} = (v, \omega)^t$ using the error configuration $\mathbf{p}_e$ and the reference velocities $\mathbf{q}_r = (v_r, \omega_r)$:

$$\mathbf{q} = \begin{bmatrix}v\\ \omega\end{bmatrix} = \begin{bmatrix}v(\mathbf{p}_e, \mathbf{q}_r)\\ \omega(\mathbf{p}_e, \mathbf{q}_r)\end{bmatrix} \tag{4}$$

The third box T stands for the vehicle hardware capability of transforming target velocities to vehicle's real current velocities. In Sections 3 and 4, specifically, we assume the identity transformation:

$$\mathbf{q}_c = \begin{bmatrix}v_c\\ \omega_c\end{bmatrix} = \begin{bmatrix}v\\ \omega\end{bmatrix} = \mathbf{q} \tag{5}$$

This *perfect velocity tracking assumption* simplifies the forthcoming analysis.

The fourth box is the kinematics matrix M in Equation 2 to produce the derivative of a current configuration $\mathbf{p}_c$. The last box is for integration. Thus, only unknown component in this system is the control rule, Equation 4. Since the system's input $\mathbf{p}_r$ is time-variable, it is called "non-autonomous" by the definition in the control theory [8].

3. A Control Scheme and Its Stability

In this section, we will find a stable control rule using a Liapunov function [8]. The following lemma follows the system depicted in Figure 1.

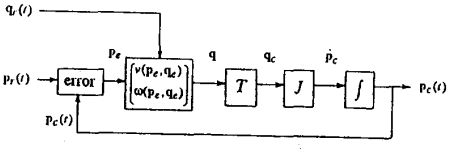

Fig. 1 Architecture of Tracking Controller

Lemma 1.

$$\dot{\mathbf{p}}_e = \begin{bmatrix}\dot{x}_e\\ \dot{y}_e\\ \dot{\theta}_e\end{bmatrix} = \begin{bmatrix}y_e\omega_c - v_c + v_r\cos\theta_e\\ -x_e\omega_c + v_r\sin\theta_e\\ \omega_r - \omega_c\end{bmatrix} \tag{6}$$

Proof. Using Equation 3 and an equality $\dot{x}_r\sin\theta_r = \dot{y}_r\cos\theta_r$ from Equation 2,

$$\begin{aligned}\dot{x}_e &= (\dot{x}_r - \dot{x}_c)\cos\theta_c + (\dot{y}_r - \dot{y}_c)\sin\theta_c\\ &\quad - (x_r - x_c)\dot{\theta}_c\sin\theta_c + (y_r - y_c)\dot{\theta}_c\cos\theta_c\\ &= y_e\omega_c - v_c + \dot{x}_r\cos\theta_c + \dot{y}_r\sin\theta_c\\ &= y_e\omega_c - v_c + \dot{x}_r\cos(\theta_r - \theta_e) + \dot{y}_r\sin(\theta_r - \theta_e)\\ &= y_e\omega_c - v_c + \dot{x}_r(\cos\theta_r\cos\theta_e + \sin\theta_r\sin\theta_e)\\ &\quad + \dot{y}_r(\sin\theta_r\cos\theta_e - \cos\theta_r\sin\theta_e)\\ &= y_e\omega_c - v_c + (\dot{x}_r\cos\theta_r + \dot{y}_r\sin\theta_r)\cos\theta_e\\ &\quad + (\dot{x}_r\sin\theta_r - \dot{y}_r\cos\theta_r)\sin\theta_e\\ &= y_e\omega_c - v_c + v_r\cos\theta_e\end{aligned}$$

$$\begin{aligned}\dot{y}_e &= -(\dot{x}_r - \dot{x}_c)\sin\theta_c + (\dot{y}_r - \dot{y}_c)\cos\theta_c\\ &\quad - (x_r - x_c)\dot{\theta}_c\cos\theta_c - (y_r - y_c)\dot{\theta}_c\sin\theta_c\\ &= -x_e\omega_c + \dot{x}_c\sin\theta_c - \dot{y}_c\cos\theta_c - \dot{x}_r\sin\theta_c + \dot{y}_r\cos\theta_c\\ &= -x_e\omega_c - \dot{x}_r\sin(\theta_r - \theta_e) + \dot{y}_r\cos(\theta_r - \theta_e)\\ &= -x_e\omega_c - \dot{x}_r(\sin\theta_r\cos\theta_e - \cos\theta_r\sin\theta_e)\\ &\quad + \dot{y}_r(\cos\theta_r\cos\theta_e + \sin\theta_r\sin\theta_e)\end{aligned}$$

$$= -x_e\omega_c + (\dot{x}_r\cos\theta_r + \dot{y}_r\sin\theta_r)\sin\theta_e$$
$$+ (\dot{y}_r\cos\theta_r - \dot{x}_r\sin\theta_r)\cos\theta_e$$
$$= -x_e\omega_c + v_r\sin\theta_e$$

$$\dot{\theta}_e = \dot{\theta}_r - \dot{\theta}_c = \omega_r - \omega_c$$

□

Lemma 2. Assuming the perfect velocity tracking assumption,

$$\dot{\mathbf{p}}_e = \begin{bmatrix} \omega\, y_e - v + v_r\cos\theta_e \\ -\omega\, x_e + v_r\sin\theta_e \\ \omega_r - \omega \end{bmatrix} \tag{7}$$

Proof. Use Lemma 1, Equation 5, and Equation 4. □

Equation 7 demonstrates that $\dot{\mathbf{p}}_e$ is a function of $\mathbf{p}_e$, $\mathbf{p}_r$ and $\mathbf{q}(\mathbf{p}_e, \mathbf{q}_r)$. Since $\mathbf{p}_r$ and $\mathbf{q}_r$ are functions of time t, $\dot{\mathbf{p}}_e$ can be considered as a function $\mathbf{f}$ of t and $\mathbf{p}_e$. We propose a specific instance of the control rule (Equation 4) for the target velocities as follows:

$$\mathbf{q} = \begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} v(\mathbf{p}_e, \mathbf{q}_r) \\ \omega(\mathbf{p}_e, \mathbf{q}_r) \end{bmatrix} = \begin{bmatrix} v_r\cos\theta_e + K_x x_e \\ \omega_r + v_r(K_y y_e + K_\theta \sin\theta_e) \end{bmatrix} \tag{8}$$

where K_x, K_y and K_θ are positive constant *parameters*. The first term in each velocity is a feedforward part.

Lemma 3. If the control rule of Equation 8 is adopted,

$$\dot{\mathbf{p}}_e = \mathbf{f}(t,\ \mathbf{p}_e) = \begin{bmatrix} (\omega_r + v_r(K_y y_e + K_\theta \sin\theta_e))\, y_e - K_x x_e \\ -(\omega_r + v_r(K_y y_e + K_\theta \sin\theta_e))\, x_e + v_r\sin\theta_e \\ -v_r(K_y y_e + K_\theta \sin\theta_e) \end{bmatrix} \tag{9}$$

Proof. Substitute Equation 8 into Equation 7. □

Proposition 1. If we use Equation 8 as a control rule, $\mathbf{p}_e = \mathbf{0}$ is a stable equilibrium point if the reference velocity $v_r > 0$.

Proof. We propose a scalar function V as a Liapunov function candidate [8]:

$$V = \frac{1}{2}(x_e^2 + y_e^2) + (1 - \cos\theta_e)/K_y \tag{10}$$

Clearly, $V \geq 0$. If $\mathbf{p}_e = \mathbf{0}$, $V = 0$. If $\mathbf{p}_e \neq \mathbf{0}$, $V > 0$. Furthermore, by Lemma 3:

$$\dot{V} = \dot{x}_e x_e + \dot{y}_e y_e + \dot{\theta}_e \sin\theta_e / K_y$$
$$= [(\omega_r + v_r(K_y y_e + K_\theta \sin\theta_e))y_e - K_x x_e]\, x_e$$
$$+ [-(\omega_r + v_r(K_y y_e + K_\theta \sin\theta_e))x_e + v_r\sin\theta_e]\, y_e$$
$$+ [-v_r(K_y y_e + K_\theta \sin\theta_e)]\, \sin\theta_e / K_y$$
$$= -K_x x_e{}^2 - v_r K_\theta \sin^2\theta_e / K_y \leq 0$$

Then, V becomes a Liapunov function. □

The following proposition demonstrates the uniformly asymptotic stability around $\mathbf{p}_e = \mathbf{0}$ under some conditions:

Proposition 2. Assume that (a) v_r and ω_r are continuously differentiable and bounded, (b) there exits a positive constant δ such that $v_r \geq \delta$ for all $t \geq 0$, (c) K_x, K_y, and K_θ are positive constants, and (d) $\dot{v}_r$ and $\dot{\omega}_r$ are sufficiently small. Under these conditions, $\mathbf{p}_e = \mathbf{0}$ is a uniformly asymptotically stable point of Equation 9 over $[0, \infty)$.

Proof. By linearizing the differential Equation 9 around $\mathbf{p}_e = \mathbf{0}$:

$$\dot{\mathbf{p}}_e = A(t)\, \mathbf{p}_e \tag{11}$$

where

$$A(t) = \left[\frac{\partial \mathbf{f}(t,\ \mathbf{p}_e)}{\partial \mathbf{p}_e} \right]_{\mathbf{p}_e = \mathbf{0}} = \begin{bmatrix} -K_x & \omega_r & 0 \\ -\omega_r & 0 & v_r \\ 0 & -v_r K_y & -v_r K_\theta \end{bmatrix} \tag{12}$$

Then, $A(t)$ is continuously differentiable and is bounded. The characteristic equation for $A(t)$ is:

$$a_3 s^3 + a_2 s^2 + a_1 s + a_0 = 0 \tag{13}$$

where

$$\begin{cases} a_3 = 1 \\ a_2 = K_\theta v_r + K_x \\ a_1 = K_y v_r^2 + K_x K_\theta v_r + \omega_r^2 \\ a_0 = K_x K_y v_r^2 + \omega_r^2 K_\theta v_r \end{cases}$$

Substituting $s = x - \varepsilon$ into Equation 13, where ε is a positive real number, we have

$$b_3 z^3 + b_2 z^2 + b_1 z + b_0 = 0, \tag{14}$$

If we choose a positive number ε such that

$$\varepsilon < min\left(\frac{K_x}{3},\ \frac{\delta K_\theta}{3},\ \frac{\delta K_y}{K_\theta}\right),$$

α, β and γ are positive. Therefore $b_i > 0$ for each i and

$$b_1 b_2 - b_0 b_3 =$$
$$\omega_r^2(\alpha + \varepsilon) + K_y v_r^2(\beta + \varepsilon) + (K_\theta v_r + \alpha)(\alpha\beta + \varepsilon\alpha + \varepsilon\beta)$$
$$+ (\alpha + 2\varepsilon)(\beta + 2\varepsilon)\varepsilon > 0$$

From the previous inequalities and the Routh-Hurwitz Criterion, the real parts of all roots of Equation 14 are non-positive for all $t \geq 0$. Therefore, all the roots of Equation 13 have real parts less than or equal to $-\varepsilon$. By the theory of slowly varying systems (Corollary 41 on page 223 in [8]), $\mathbf{p}_e = \mathbf{0}$ is a uniformly asymptotically stable point of Equation 11. Using Liapunov's indirect method, the Proposition is proved. □

4. Effects of Control Parameters

In the previous section, we demonstrated that the system is stable for any combination of parameter values of K_x, K_y, and K_θ. However, since we want a non-oscillatory, but not too slow response of the robot, we have to find an optimal parameter set. In order to simplify the analysis we consider only situa-

tions in which the reference configuration is moving on the x axis in the direction at a constant velocity V_r:

$$\mathbf{p}_r(t) = \begin{bmatrix} x_r(t) \\ y_r(t) \\ \theta_r(t) \end{bmatrix} = \begin{bmatrix} V_r t \\ 0 \\ 0 \end{bmatrix} \text{ and } \mathbf{q}_r(t) = \begin{bmatrix} v_r(t) \\ \omega_r(t) \end{bmatrix} = \begin{bmatrix} V_r \\ 0 \end{bmatrix} \quad (15)$$

This condition is called the *linear reference motion.* In addition, we assume that:

$$|\theta_e| \ll 1 \text{ and } |\dot{\theta}_e| \ll 1 \quad (16)$$

Proposition 3. Under Conditions 15 and 16,

$$\dot{\mathbf{p}}_c = \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\theta}_c \end{bmatrix} = \begin{bmatrix} -K_x & 0 & 0 \\ 0 & 0 & V_r \\ 0 & -V_r K_y & -V_r K_\theta \end{bmatrix} \begin{bmatrix} x_c - V_t t \\ y_c \\ \theta_c \end{bmatrix} + \begin{bmatrix} V_r \\ 0 \\ 0 \end{bmatrix} \quad (17)$$

Proof. By substituting Equation 11 by Equation 3,

$$\dot{\mathbf{p}}_c = T_e^{-1} (A\, T_e - \dot{T}_e)(\mathbf{p}_c - \mathbf{p}_r) + \dot{\mathbf{p}}_r \quad (18)$$

By Equation 3, Conditions 15 and 16, $|\theta_c| = |\theta_e| \ll 1$, $|\dot{\theta}_c| = |\dot{\theta}_e| \ll 1$. Therefore T_e and $\dot{T}_e$ in Equation 18 can be considered as the identity matrix and null matrix respectively. Therefore,

$$\dot{\mathbf{p}}_c = A\,(\mathbf{p}_c - \mathbf{p}_r) + \dot{\mathbf{p}}_r$$

By substituting the previous equation by Equations 12 and 15, we obtain Equation 17. □

Equation 17 shows that the behavior x_c is independent of y_c and θ_c in this small perturbation case. $1/K_x$ corresponds to the time constant of the exponential decay.

Corollary 1 When $x_c = \Delta x$ at $t = 0$,

$$x_c = V_r t + \Delta x\, e^{-K_x t} \quad (19)$$

By cancelling θ_e in Equation 17,

$$\ddot{y}_c + 2\zeta\xi\dot{y}_c + \xi^2 = 0 \quad (20)$$

where,

$$\zeta = \frac{K_\theta}{2\sqrt{K_y}} \text{ and } \xi = V_r \sqrt{K_y}$$

Corollary 2 The condition for critical damping is

$$K_\theta{}^2 = 4 K_y \quad (21)$$

Corollary 3 In critical damping and if $y_c = \Delta y$ and $\theta_c = 0$ at $t = 0$,

$$\mathbf{p}_c = \begin{bmatrix} V_r t \\ \Delta y\,(1 + \xi t) e^{-\xi t} \\ -\Delta y\, \xi^2\, t\, e^{-\xi t} \end{bmatrix} \quad (22)$$

In this motion, the error, y_c, becomes 9.2% of its initial value Δy at $x_c = 4/\sqrt{K_y}$.

Simulation results on three distinct convergence characteristics are shown in Figure 2. Here, the robot's $\mathbf{p}_r$ and $\mathbf{p}_c$ were moving on the x axis to the positive direction, when y_r suddenly jumps up by $\Delta y = 5cm$ while continuing a parallel horizontal reference motion. The common parameters are $v_r = 30cm/sec$, $K_y = 6.4 \times 10^{-3}/cm^2$, $\xi = 2.4/sec$, and $K_x = 10/sec$. An oscillatory case ($\zeta = 0.75$ and $K_\theta = 0.12/cm$), a critical damping case ($\zeta = 1$ and $K_\theta = 0.16/cm$), and an over damping case ($\zeta = 1.25$ and $K_\theta = 0.20/cm$) are used. With the small Δy perturbation of 5cm, the result of simulation and analysis matches. In the following experiments and in our real implementation we adopt the critical damping condition, $\zeta = 1$, since the convergence is fastest under non-oscillatory condition.

5. Velocity/Acceleration Limiting

From a path planner's viewpoint, it is convenient if non-smooth paths are allowed to be used. However, in that case, (i) either or both of the target velocities (v, ω) by Equation 8 might become too large to be attained by a real vehicle, and (ii) the linear/rotational acceleration might become too large caus-

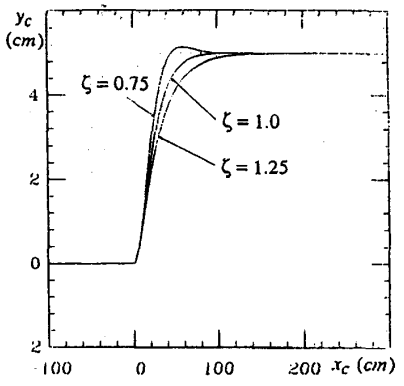

Fig. 2 Responses of Lateral Step Input at Various Damping Parameters (Theoretical, without Limiter)

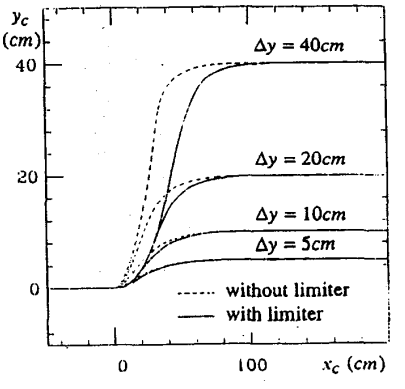

Fig. 3 Responses of Lateral Step Inputs (Theoretical, with/without Limiter)

ing the robot's slippage. (Any slippage is a cause of a severe error in dead-reckoning.) Therefore, in order to handle these non-smooth reference paths, we need some limiter for velocities and accelerations. This modification is added in the box T in Figure 1.

Figure 3 shows simulation results for various values of Δy's with and without the velocity/acceleration limiter. Figure 4 shows simulation results for $\Delta\theta$ discontinuous jumps without limitation ($\Delta\theta = \pi/4$, $\pi/2$ and $3\pi/4$). Figure 5 shows simulation results for $\Delta\theta$ discontinuous jumps with velocity/acceleration limitation.

The last pair of simulation results is on a circular reference path, i.e. a non-zero ω_r case. In Figure 6, the reference path is a circle with a radius of 50(cm) and with a center of (0, 70). The initial reference configuration and reference velocities are (x_r, y_r, θ_r) = (0, 20, 0) and (v_r, ω_r) = (30cm/sec, 0.6rad/sec) respectively. The initial current configuration is $(x_c, y_c, \theta_c) = (0, 0, 0)$. This result shows how the initial lateral difference in the current and reference configurations converges even in the presence of ω_r. Figure 7 shows a result with velocity/acceleration limitation. Both results demonstrates the effectiveness of the feedforwarding of ω_r in Equation 8.

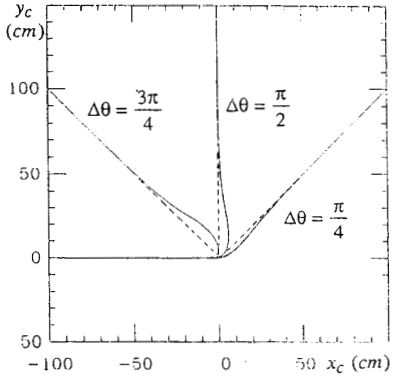

Fig. 4 Responses of Orientational Step Inputs (Theoretical, without Limiter)

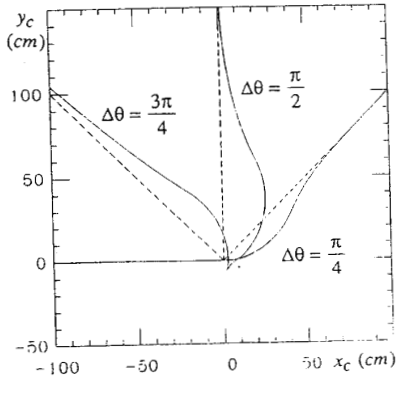

Fig. 5 Responses of Orientational Step Inputs (Theoretical, with Limiter)

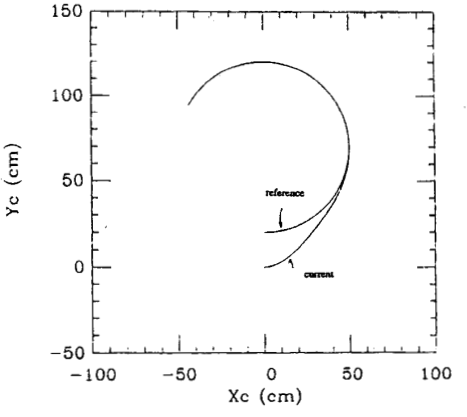

Fig. 6 Responses of Lateral Step Input at Circular Reference (Theoretical, without Limiter)

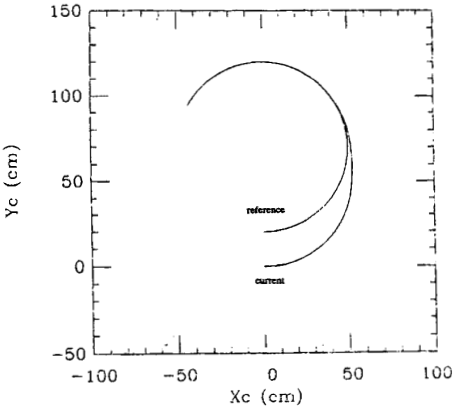

Fig. 7 Responses of Lateral Step Input at Circular Reference (Theoretical, with Limiter)

6. Experimental Results

The results presented in Sections 3, 4 and 5 were hardware independent. This theory was implemented on the robot Yamabico-11. After deep investigation and experimental observations, the parameters have been decided as follows: $K_x = 10/sec$, $K_y = 6.4\times10^{-3}/cm^2$ and $K_\theta = 0.16/cm$. With these K_x, K_y and K_θ, no oscillations were seen. The errors x_e

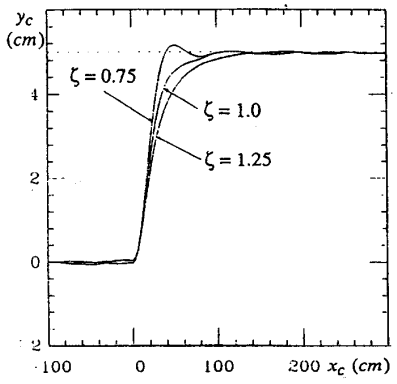

Fig. 8 Responses of Lateral Step Input at Various Damping Parameters (Experimental)

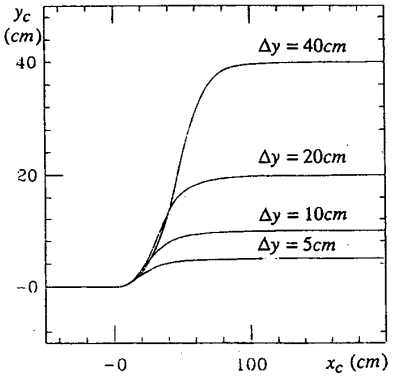

Fig. 9 Responses of Lateral Step Inputs (Experimental)

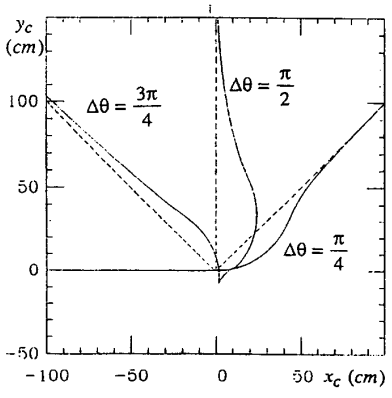

Fig. 10 Responses of Orientational Step Inputs (Experimental)

and y_e at constant reference velocity of $30cm/sec$ are about $2mm$ and less than $1mm$ respectively.

We conducted a few experiments to make sure that these values of K_y and K_θ are reasonable. Figure 8 shows experimental results with three distinct values of ζ, which corresponds to Figure 2. Figure 9 shows results on Δy, which corresponds to Figure 3. Figure 10 shows results on $\Delta\theta$, which corresponds to Figure 6. (As shown here, the results on the real vehicle are close to that of simulation with a velocity/acceleration limiter.) In Figures 8-10, the trajectories are plotted using the current configuration $\mathbf{p}_c$ which is obtained by the vehicle's dead reckoning.

References

[1] T. Tsumura, N. Fujiwara, T. Shirakawa and M. Hashimoto, "An Experimental System for Automatic Guidance of Robot Vehicle, following the route stored in Memory," Proceeding 11th International Symposium on Industrial Robots, October 1981, pp. 187-193.

[2] Y. Kanayama and S. Yuta, "Vehicle Path Specification by a Sequence of Straight Lines", IEEE Journal of Robotics and Automation, vol. 4, no. 3, pp. 265-276, 1988.

[3] J. Crowley, "Asynchronous Control of Direction and Displacement in a Robotic Vehicle", Proc. IEEE Conference on Robotics and Automation, pp. 1277-1282, 1989.

[4] Y. Kanayama and N. Miyake, "Trajectory Generation for Mobile Robots", Robotics Research, vol. 3, pp. 333-340, The MIT Press, 1986.

[5] S. Singh and D. H. Shin, "Position Based Path Tracking for Wheeled Mobile Robots", Proc. IEEE International Workshop on Intelligent Robots and Systems, in Tsukuba, Japan, pp. 386-391, September 1989.

[6] Y. Kanayama, A. Nilipour and C. Lelm, "A Locomotion Control Method for Autonomous Vehicles", Proc. IEEE Conference on Robotics and Automation, pp. 1315-1317, 1988.

[7] W. Nelson and I. Cox, "Local Path Control for an Autonomous Vehicles", Proc. IEEE Conference on Robotics and Automation, pp. 1504-1510, 1988.

[8] M. Vidyasagar, "Nonlinear Systems Analysis", Prentice-Hall Inc, Englewood Cliffs, N.J., 1978.

[9] F. Miyazaki, Y Masutani, C. Lelm and Y. Kanayama, "Precise Trajectory Following Control for Autonomous Vehicles", Proc. Annual Conf. of Institute of Systems, Control and Information Engineers, in Kyoto, Japan, May 1989.

[10] D. Koditschek, "Application of a New Lyapunov Function: Global Adaptive Inverse Dynamics for a Single Rigid Body", Repot of Center for Systems
Science - 8806, Yale University, 1988.

[11] Y. Kanayama and T. Noguchi, "Locomotion Functions for a Mobile Robot Language", Proc. International Workshop on Advanced Robots and Intelligent Systems, pp. 542-549, September 1989.

[12] Y. Kanayama and M. Onishi, "Locomotion Functions in the Mobile Robot Language, MML", Proc. IEEE International Conference on Robotics and Automation, pp. 1110-1115, Sacramento, CA, April 7-12, 1991.

[13] Y. Kanayama and B. Hartman, "Smooth Local Path Planning for Autonomous Vehicles", Proc. IEEE International Conference on Robotics and Automation, pp. 1265-1270, 1989. Also to appear in International Journal of Robotics Research, 1991.