A Novel Non-time based Tracking Controller for Nonholonomic Mobile Robots*

Eric Hu¹, Simon X. Yang¹ and Ning Xi²
¹School of Engineering, University of Guelph, Guelph, ON N1G 2W1, Canada
²Department of Electrical Engineering, Michigan State University
East Lansing, MI 48824-1226, USA

Abstract

Event based controller design was first proposed by Xi (1993), where a suitable non-time motion reference is introduced to represent the desired and measurable system output. It has been successfully applied to many areas such as robot motion control, multi-robot coordination, force and impact control, robotic teleoperation and manufacturing automation. In this paper, a novel non-time based tracking controller for a nonholonomic mobile robot is proposed by combining the conventional event based control technique with a biologically inspired additive model first proposed by Grossberg (1982). The proposed control algorithm can generate the smooth and continuous velocity control commands, which remove the small tracking error constraints in the conventional non-time based controllers. The stability of the control system and convergence of the tracking errors are guaranteed. The effectiveness of the proposed controller is demonstrated by simulation and comparison studies.

1 Introduction

Effective robot control design is a fundamentally important issue in robotics. There are many studies on control of robotic systems. Most of the previous control approaches are time based, where the time plays an important role of action reference in both desired trajectory information and measurable system feedback. A typical conventional time-based control system is shown in Figure 1A, where $y_d(t)$ is the desired robot trajectory, $y(t)$ is the actual robot position, and $e(t)$ is the tracking error of the mobile robot.

Xi [1] proposed a non-time based control method, which produces a better solution to some control problems. The basic idea of non-time based control design is to introduce the concept of an action reference parameter other than time, which is directly relevant to the sensory measurement and the event (thus it is also called event-based control). A typical non-time based control system is shown in Figure 1B, where the variable $s$ is the action reference parameter, and $y_d(s)$ and $e(s)$ are the desired robot path and the tracking error as the functions of $s$. There are many successfully theoretical and practical studies of non-time based controllers [1]-[10], such as robot motion control [2], multi-robot coordination [3], force and impact control [4], robotic teleoperation [5], manufacturing automation [6], and Internet based teleoperation [10]. Recently, the concept of non-time based control design is applied to tracking the control of nonholonomic mobile robots [7, 8, 9], which is capable of tracking an arbitrary twice differentiable robot path. In addition, the construction of the control system has integrated planning capability, thus the planning and control become a closed-loop system. However, this controller can deal with small tracking errors only [8].

![Figure 1: Schematic diagrams of robot control system. A: conventional control; B: non-time based control.](image)

---

*This work was supported by Natural Sciences and Engineering Research Council (NSERC) of Canada.
by incorporating a biologically inspired additive model into a conventional non-time based controller. From the analysis of the generated velocity commands of the conventional non-time based controller, it was found that there is an oscillation at the initial phase in both the linear and angular velocities when there is a large tracking error. This is not practically feasible. To resolve this oscillation problem, the proposed controller replaces the component that results in the oscillation with a new control component from an additive neural model proposed by Grossberg [12] to understand the dynamic behavior of some biological systems. The proposed novel non-time based tracking controller is capable of generating smooth, continuous velocity commands. The stability of the control system and the convergence of the tracking errors to zeros are guaranteed.

2 The Control Algorithm

In this section, the concept of non-time based controller and tracking control problem for nonholonomic mobile robots are briefly introduced. Then, the conventional non-time based tracking controller proposed by [7] is presented. After that, the novel non-time based controller with a biologically inspired additive neural model is proposed. Finally, the analysis of the system stability and convergence of tracking errors to zeros are provided.

2.1 Non-time based Controller

In the non-time based control scheme shown in Figure 1B, the feedback depends on the action reference parameter $s$ measured by single or multiple sensors. The planner not only generates the desired path to the system, but also is part of the feedback to adjust the actual path related to the desired path. Thus, non-time based control becomes a high level control. The dynamic system can be modeled by differential equations in which the free variable is the time variable $t$. A desired trajectory is often modeled as a function of time denoted by $x_d(t)$, where $x$ represents the state of the system. The controller can be designed so that the trajectory of the plan system $x(t)$ asymptotically approaches the desired trajectory $x_d(t)$. This is a typical tracking control problem. We prefer that a feedback controller is independent on time, which can be defined by parametric equations denoted by a non-time action reference parameter $s$.

A dynamic control system can be defined by $\dot{x} = f(x, u)$, where $u$ is the control input and $x$ is the state of the system. Assume that $y = h(x)$ represents the output of the system, and the desired path is given by a parametric equation $y = y_d(s)$, where $s$ is the action motion reference. The non-time based control design can be achieved as follows: Step 1 is to define a corresponding path in time domain. If $s$ is not time, then we assign an strictly increasing function $x = v(t)$; Step 2 is to find a feedback $u(x, t)$ to track the path $y_d(v(t))$. The feedback satisfied $\lim_{u \to \infty} (h(x(t)) - y_d(v(t))) = 0$; Step 3 is to find a suitable transformation $s = \Gamma(x)$ from the state space to the reference $s$. The transformation satisfies $\gamma(x_d(s)) = s$; Step 4 is to construct the feedback. Let $\tilde{u}(x) = u(x, v^{-1}(\gamma(x)))$, where $\tilde{u}(x, t)$ is the feedback found in Step 2, $\gamma(x)$ is a state-to-reference projection. Therefore, we get the feedback input $\tilde{u}(x)$ is independent on time.

2.2 Tracking Control of Mobile Robots

Considering the mobile robot in Figure 2, the rear wheels are aligned with the vehicle, while the front wheels are allowed to spin about the vertical axes. The constraints on the system arise by allowing the wheels to roll and spin, but not slip. Let $(x, y, \theta, \phi)$ denote a configuration of the robot (see Figure 2), which is driven by the front wheels, thus the kinematics of the mobile robot can be presented as follows:

$$\dot{x} = u_1 \cos \theta,$$
$$\dot{y} = u_1 \sin \theta,$$
$$\dot{\theta} = \frac{u_1}{l} \tan \phi,$$
$$\dot{\phi} = u_2,$$

where $u_1$ represents the linear velocity of the rear robot wheels, and $u_2$ represents the angular velocity of the steering wheels, $\theta$ is the angle of the robot body with respect to the $X$-axis in the global frame, and $\phi$ is steering angle with respect to the $X_R$-axis (i.e., the driving direction) in the local frame that is attached on the car. The spatial location of the robot is represented by $(x, y)$ in the global frame, which is the center of the two rear wheels. Parameter $l$ is the length between the front and the rear wheels.

The tracking control design in this paper is to implement a mapping between the known information (e.g., the desired path information and the measurable system output) and the velocity commands designed to achieve the robot’s task. The controller design problem can be described as: given the desired robot path $x_d(s)$, $y_d(s)$ and $\theta_d(s)$, design a control
law for the linear velocity \( u_1(t) \) and angular velocity \( u_2(t) \), which drive the robot to move, such that the actual robot position \( x(s), y(s) \) and \( \theta(s) \) will precisely track the desired robot path \( x_d(s), y_d(s) \) and \( \theta_d(s) \).

2.3 Conventional Non-time based Controller for Mobile Robots

By using the state-to-reference projection, the desired robot path can be represented as a function of the action reference parameter \( s \), \( x = x_d(s) \) and \( y = y_d(s) \). The measurable robot position is also represented by \( s \) as \( x = x(s) \) and \( y = y(s) \). Thus the tracking error can be obtained as \( e_x = x - x_d \) and \( e_y = y - y_d \), which is also a function of \( s \). The velocity commands of a conventional non-time based tracking controller is defined as [7]

\[
\begin{align*}
    u_1 &= \frac{\dot{x}_d - e_x}{\cos \theta}, \\
    u_2 &= \frac{\alpha_1}{\beta_1} + u_1(a_1 e_1 + a_2 e_2 + a_3 e_3),
\end{align*}
\]

where

\[
\begin{align*}
    e_1 &= e_y, \\
    e_2 &= \frac{1}{l} u_1 e_y, \\
    e_3 &= \frac{1}{l} \tan \phi \cos \theta, \\
    \alpha_1 &= \frac{-u_1}{l^2} \sin \theta (\tan \phi)^2, \\
    \beta_1 &= \frac{1}{l} \cos \theta (\cos \phi)^2.
\end{align*}
\]

where \( a_1, a_2 \) and \( a_3 \) are control parameters.

From equations (7)-(11), we have the error dynamics of the control system as

\[
\begin{align*}
    \dot{e}_1 &= u_1 e_2, \\
    \dot{e}_2 &= u_1 e_3, \\
    \dot{e}_3 &= u_1(a_1 e_1 + a_2 e_2 + a_3 e_3). 
\end{align*}
\]

To guarantee the system stability and convergence of tracking error to zero, parameters \( a_1, a_2 \) and \( a_3 \) can be selected, such that all the eigenvalues of the matrix

\[
A = \begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
a_1 & a_2 & a_3
\end{bmatrix}
\]

are on the left half plane [7].

2.4 The Proposed Non-time based Controller with an Biological Additive Equation

From the simulation study and the analysis of the control law for \( u_1 \) and \( u_2 \) and the error input \( e(s) \), it is found that there is a large oscillation at the initial period, which results from the initial tracking error \( e_y \). This oscillation could cause the damage of the vehicle. Actually, from the definition of \( e_1(t), e_2(t), e_3(t) \), \( e(t) \) is actually a kind of offset of the robot from the desired trajectory. The tracking error \( e(t) \) tends to zero when \( (x, y, \theta) \) tends to \((x_d, y_d, \theta_d)\). The error \( e(t) \) reflects the state error and it holds on only when the offset is small enough, which means that the controller based on the nonlinear transformation is only locally asymptotically stable [8].

Inspired by the dynamic characterizations of the additive neural model, which was derived from Hodgkin and Huxley’s [14] biological membrane model for the dynamic behavior of neural potential, a simple additive model was employed in the proposed non-time based controller design to solve the oscillation problem. A typical additive model is given as

\[
\frac{d\xi_i}{dt} = -A\xi_i + I + \sum_{j=1}^{n} w_{ij} f(x_j),
\]

where \( A \) is the passive decay rate of the neural activity. The term \( I + \sum_{j=1}^{n} w_{ij} f(x_j) \) represents the total external input \( I \) and internal input from the lateral neural connections to the \( i \)th neuron. The parameter \( w_{ij} \) is the connection weight from the \( j \)th to the \( i \)th neuron, and \( f(a) \) is the activation function from the neural activity to the neural output. The additive model was first proposed by Grossberg [12]. It has a lot of applications to many areas such as vision, associative pattern learning, pattern recognition and robotic path planning [12, 13, 11].
In the proposed non-time based tracking control design, since the initial oscillation results from the sudden change of tracking error \( \epsilon_y \), a control component from the additive neural model is used to replace the \( \epsilon_1 \) and \( \epsilon_2 \) that are directly related to \( \epsilon_y \). In the additive model in (15), the total input term 
\[ I + \sum_{j=1}^{n} w_{ij} f(x_j) \] 
is replaced by \( a_1 \epsilon_1 + a_2 \epsilon_2 \) in (6). Therefore, the velocity commands of the proposed non-time based tracking controller are given as
\[ \frac{dv}{dt} = -Av + a_1 \epsilon_1 + a_2 \epsilon_2, \]  
(16)
\[ u_1 = \frac{x_d - \epsilon_x}{\cos \theta}, \]  
(17)
\[ u_2 = \frac{\epsilon_1}{\beta_1} + u_1 (kv + a_3 \epsilon_3), \]  
(18)
where \( k \) is a positive parameter, variables \( \epsilon_1, \epsilon_2, \epsilon_3, \alpha_1 \) and \( \beta_1 \) are defined as the same as (7)-(11).

For the start period, we hope that the robot speed increases exponentially and reaches the desired speed smoothly. According to the time response property of the first order exponential time-variant system, the reference velocity can be defined as:
\[ v_d = k_1 (1 - e^{-s/\tau}) \]  
(19)
where \( k_1 \) is the proportional parameter, \( \tau \) is the time constant, and \( s \) is the action reference.

2.5 Stability Analysis

From the additive equation in (15), at the steady-state, \( dv/dt = 0 \), we have
\[ v = \frac{1}{A} (a_1 \epsilon_1 + a_2 \epsilon_2). \]  
(20)
Similar to the conventional control design, the error dynamics of the proposed non-time based tracking controller is given as
\[ \dot{\epsilon}_1 = u_1 \epsilon_2, \]  
(21)
\[ \dot{\epsilon}_2 = u_1 \epsilon_3, \]  
(22)
\[ \dot{\epsilon}_3 = u_1 (kv + a_3 \epsilon_3). \]  
(23)
Substitute (20) into (23), the error dynamics of \( \epsilon_3 \) becomes
\[ \dot{\epsilon}_3 = u_1 \left( \frac{ka_1}{A} \epsilon_1 + \frac{ka_2}{A} \epsilon_2 + a_3 \epsilon_3 \right). \]  
(24)
Similarly, to guarantee the system stability and convergence, all the eigenvalues of the matrix
\[ A_v = \begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
\frac{ka_1}{A} & \frac{ka_2}{A} & a_3
\end{bmatrix} \]
will be chosen on the left half plane [7]. From the equation (17), we can prove that \( \int_0^t u_1 (t) dt \) approaches infinite as \( t \) approaches infinite. Therefore, the solution of (9) is \( e(t) = e_0 \exp(A_v \int_0^t u_1 (t) dt) \), since \( A_v \) is stable and also negative, we have \( \lim_{t \to \infty} e(t) = 0 \). Therefore, the control system is stable, and the tracking error will converge to zero.

3 Simulation Studies

For simplicity without losing the generality, to illustrate the effectiveness of the proposed non-time based tracking controller in (16)-(18), this model is applied to a nonholonomic mobile robot to track a simple straight line. The desired path is described as \( y_d = 5 \) and \( x = 0 \). The robot starts at \( (0, 2.5, 0, 0) \). The length of the vehicle is \( l = 1 \). The model parameters in (18) are chosen as \( a_1 = -5, a_2 = -9, a_3 = -5, k = 1.7 \), and \( A = 10 \). The tracking performance is shown in Figure 3A, where the desired path is represented by a dashed line, while the actual robot trajectory is represented by a solid curve. The tracking error \( y_d - y \) is shown in Figure 3B. The generated control commands of the linear velocity \( u_1 \) and the angular velocity \( u_2 \) are shown in Figures 3C and 3D, respectively. It shows that the proposed controller is capable of generating smooth, continuous velocity control commands with zero initial members under the condition of a large initial tracking error. In addition, it

---

**Figure 3:** A nonholonomic mobile robot to track a straight line. A: the tracking performance; B: the tracking error \( y_d - y \); C: the linear velocity command; D: the angular velocity command.
shows the tracking error quickly converges to zero.

A comparison study to a conventional non-time based tracking controller in (5) and (6) is conducted under the same condition with Figure 3. The model parameters are chosen as the same as that in Figure 3. The tracking performance and the error dynamics are shown Figures 4A and 4B, respectively. The generated velocity control commands are shown in Figures 4C and 4D, respectively. It shows that there are very large oscillations in both linear velocity $u_1$ and the angular velocity $u_2$. Therefore, the conventional non-time based tracking controller suffers from the oscillation problem when the tracking error is large.

4 Discussion

The parameter sensitivity of a model is a very important factor to be considered when we propose a mode. A satisfied model should be not very sensitive to the changes of its parameter values. In the proposed non-time based tracking controller, two additional model parameters $A$ and $k$ are introduced, which represent the passive decay rate of the neural dynamics in the additive equation, and the relative contribution of the control component from the additive equation, respectively. Parameter $A$ solely determines the transient response to an input signal. In addition, the steady-state neural activity is nonlinearly dependent on the value of $A$. To illustrate the role of $A$ in the proposed model, two simulations under the same conditions of Figure 3 except with different $A$ values are carried out. First, a very smaller $A$ value is selected, $A = 2$ instead of $A = 10$ in Figure 3. The tracking performance, the error dynamics, and the generated velocity control commands are shown in Figures 3A, 3B, 3C and 3D, respectively. It shows that the mobile can also track the desired robot path. In comparison to Figure 3, with a much smaller $A_1$ and $A_2$ value, the the tracking error converge faster, but the velocity commands are not as smooth as in Figure 3.

Then, a much larger $A$ value is selected, $A = 50$ instead of $A = 10$ in Figure 3. Figures 3A, 3B, 3C and 3D show the tracking performance, the error dynamics, and the generated velocity control commands, respectively. The mobile can track the desired robot path with a slower converge of tracking error, but there are no oscillations.

The parameter $k$ can also be chosen in a wide range, such as $k = [0.01, 20]$. Therefore, the proposed non-time based tracking controller is not very sensitive to model parameters.

5 Conclusion

Non-time based tracking control is a newly developed method to convert the traditional time-based con-
and Automation, Nagoya, Japan, 1995, pp. 899 - 904.