INTERNATIONAL JOURNAL OF ROBUST AND NONLINEAR CONTROL
Int. J. Robust Nonlinear Control 2015; **25**:1168–1182
Published online 2 January 2014 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/rnc.3133

Nonlinear model predictive control for path following problems

Shuyou Yu[1,2,*,†], Xiang Li[2], Hong Chen[1,3] and Frank Allgöwer[2]

[1]*Department of Control Science and Engineering, Jilin University Changchun, China*
[2]*Institute for Systems Theory and Automatic Control, University of Stuttgart Stuttgart, Germany*
[3]*State Key Laboratory of Automotive Simulation and Control, Jilin University Changchun, China*

SUMMARY

This paper presents a general nonlinear model predictive control scheme for path following problems. Path following problem of nonlinear systems is transformed into a parameter-dependent regulation problem. Sufficient conditions for recursive feasibility and asymptotic convergence of the given scheme are presented. Furthermore, a polytopic linear differential inclusion-based method of choosing a suitable terminal penalty and the corresponding terminal constraint are proposed. To illustrate the implementation of the nonlinear model predictive control scheme, the path following problem of a car-like mobile robot is discussed, and the control performance is confirmed by simulation results. Copyright © 2014 John Wiley & Sons, Ltd.

Received 03 April 2012; Revised 06 May 2013; Accepted 22 November 2013

KEY WORDS: model predictive control; nonlinear systems; path following; feasibility and convergence

1. INTRODUCTION

Nonlinear model predictive control (NMPC), also referred to as receding horizon control or moving horizon optimal control, has been viewed as one of standard control techniques for nonlinear systems with input and state constraints [1–3]. A control sequence is obtained by solving online, at each sampling instant, a finite horizon open-loop optimal control problem, which uses the current state of the system as the initial state; the first control action in this sequence is applied to the system. Because it is difficult to obtain an analytical solution to constrained nonlinear optimal control problem by solving the related Hamilton-Jacobi-Bellman equation, NMPC has aroused many interests in both the academic community and the industrial society. Over the last decade, academic researches of NMPC have made significant progresses in issues on both stability and robustness [2, 4], and its applications have spanned a wide range from process control [3] to aerospace [5] and to control of transportation networks [6].

Normally, NMPC is used to deal with the so-called regulation problem, that is, to regulate the state of a system to a fixed target state [7, 8]. However, when the target state changes, the feasibility of the controller may be lost, and the controller fails to track the target states.

Besides the regulation problem, tracking control and path following are the other two fundamental control problems. Tracking control of systems aims at tracking a given time-varying reference trajectory. NMPC for tracking control has been discussed in [9–11] and references therein. Instead of arbitrary but smooth trajectories, only piecewise constant references are considered. A receding horizon control scheme for tracking control of a nonholonomic mobile robot is developed in [12], where a control Lyapunov-based scheme is chosen to determine the terminal penalty and the terminal constraint of the NMPC optimization problem for the considered systems. Thus, the proposed

*Correspondence to: Shuyou Yu, Department of Control Science and Engineering, Jilin University (Nanling Campus), 130025, Changchun, China.
†E-mail: shuyou@jlu.edu.cn

scheme is only typically fit for the given systems. The task of the path following problem is to steer a system to follow a reference path. In contrast to the tracking control problem, the reference path is not parameterized in time but in its geometrical coordinates. The papers [13, 14] propose an NMPC framework for solving the path following problem and give sufficient stability conditions. Although some methods of choosing the terminal ingredients of the MPC optimization problem are proposed, they are either conservative or rely on the system property of differential flatness. The introduced NMPC framework optimizes the time evolution of the path parameter, but the initial value of the path parameter is not considered in the online optimization problem.

This paper presents a more general NMPC scheme for the path following problem, where the time evolution of the path parameter as well as its initial value are determined in the online optimization problem. Firstly, the path following problem of nonlinear systems is transformed into a parameter-dependent regulation problem. Following a discussion of recursive feasibility and asymptotic convergence, a polytopic linear differential inclusion (PLDI)-based method is adopted to choose the terminal penalty and the terminal constraint. To illustrate the implementation of the proposed NMPC scheme, the path following problem of a car-like mobile robot is discussed, and the control performance is confirmed by simulation results.

The remainder of this paper is organized as follows. Section 2 sets up the path following problem. In Section 3, an NMPC scheme to the path following problem is introduced with a proof of asymptotic convergence and recursive feasibility. A method for choosing a suitable terminal penalty and the related terminal constraint is presented in detail. Section 4 shows the implementation of the proposed NMPC scheme for the path following problem of a car-like mobile robot. A short summary is given in Section 5.

2. PATH FOLLOWING PROBLEM

For a system, an intuitive understanding of the path following problem is to approach a reference path as close as possible. Thus, it is necessary to clarify the definition of the system and the reference path before formulating the path following problem.

A continuous time nonlinear system

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \tag{1}$$

is considered, which has state and input constraints

$$\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^n, \quad \mathbf{u} \in \mathcal{U} \subseteq \mathbb{R}^m, \tag{2}$$

where $f(\mathbf{x}, \mathbf{u}) : \mathcal{X} \times \mathcal{U} \longrightarrow \mathbb{R}^n$ is continuously differentiable in $\mathbf{x}$ and $\mathbf{u}$, $\mathcal{U} \subseteq \mathbb{R}^m$ is compact, and $\mathcal{X} \subseteq \mathbb{R}^n$ is connected.

The reference path is a twice continuously differentiable geometric curve, which can be defined as a set of points $\mathbf{r}$ parameterized by a scalar s,

$$\mathcal{P} = \{\mathbf{r} \in \mathbb{R}^n \,|\, \mathbf{r} = \mathbf{p}(s)\}, \tag{3}$$

where the function $\mathbf{p} : \mathbb{R}^1 \to \mathbb{R}^n$ is a twice continuously differentiable function. The scalar s is constrained by $s \in \mathcal{S} \subseteq \mathbb{R}^1$, where $\mathcal{S}$ is a compact set. The time evolution of $s(t)$ is not necessary to be known *a priori* but influenced by a virtual input $v(t)$ that is a DOF to choose,

$$\dot{s}(t) = v(t), \quad v \in \mathcal{V} \subseteq \mathbb{R}^1. \tag{4}$$

Remark 2.1
If the given path is not smooth enough, a continuously differentiable geometric curve can be used to approximate it and can be used as the reference path.

Remark 2.2
If the reference path is in a subspace of $\mathbb{R}^n$, that is, $\mathcal{P}_0 = \{\mathbf{r}_0 \in \mathbb{R}^{n_0} \mid \mathbf{r}_0 = \mathbf{p}(s)\}$ with $n_0 < n$, then $\mathcal{P}$ can be chosen as

$$\mathcal{P} = \left\{\mathbf{r} \in \mathbb{R}^n \mid \mathbf{r} = \begin{bmatrix} \mathbf{r}_0 \\ 0 \end{bmatrix}\right\}$$

with $0 \in \mathbb{R}^{n-n_0}$.

The path following problem is the following:
Given a geometric path $\mathcal{P}$ defined by (3), find admissible control values $\mathbf{u}(t)$ and $v(t)$ such that

$$\lim_{t\to\infty} \mathbf{x}_e(t) = 0, \tag{5}$$

where $\mathbf{x}_e$ is defined by

$$\mathbf{x}_e(t) := \mathbf{x}(t) - \mathbf{p}(s(t)). \tag{6}$$

Two technical assumptions are made:

Assumption 1
The reference path $\mathcal{P}$ is contained in the state constraint set of the system (1), that is, $\mathcal{P} \subseteq \mathcal{X}$.

Assumption 2
There exist admissible inputs $\mathbf{u} \in \mathcal{U}$ and $v \in \mathcal{V}$, such that the dynamics of the state $\mathbf{x(t)} \in \mathcal{X}$ and the parameter $s(t) \in \mathcal{S}$ satisfy

$$\dot{\mathbf{x}}_e(t) = 0, \tag{7}$$

if $\mathbf{x}_e(t) = 0$.

Remark 2.3
Assumption 1 ensures the existence of at least one $\mathbf{x} \in \mathcal{X}$ matching each point on the reference path $\mathcal{P}$. Together, Assumption 1 and Assumption 2 guarantee that the system (1) can indeed follow the given path (3).

The dynamics of the error system (6) is

$$\dot{\mathbf{x}}_e = \dot{\mathbf{x}} - [\mathbf{p}(s)]' = f(\mathbf{x}, \mathbf{u}) - \frac{\partial \mathbf{p}}{\partial s} v. \tag{8}$$

It shows that the error dynamics are continuously differentiable in $\mathbf{x}$, $\mathbf{u}$, s, and v because $f(\cdot,\cdot)$ is continuously differentiable and $\mathbf{p}(\cdot)$ is twice continuously differentiable. The dynamics of the error system (6) is a function of $\mathbf{x}_e$, $\mathbf{u}$, s, and v, because $\mathbf{x} = \mathbf{x}_e + \mathbf{p}(s)$ and

$$\dot{\mathbf{x}}_e = f(\mathbf{x}_e + \mathbf{p}(s), \mathbf{u}) - \frac{\partial \mathbf{p}}{\partial s} v. \tag{9}$$

Assumption 3
There exist a continuously differentiable function $g(\cdot,\cdot)$ and a control input $\mathbf{u}_e$ such that

$$\dot{\mathbf{x}}_e := g(\mathbf{x}_e, \mathbf{u}_e), \tag{10}$$

where the function g is parameter-dependent in s and v, the control input $\mathbf{u}_e \in \mathbb{R}^m$ is an implicit function of $\mathbf{u}$, s, and v.

Because the control input of the error system $\mathbf{u}_e$ is an implicit function of $\mathbf{u}$, s, $\mathbf{x}_e$ and v, abuse of notation, denote $h(\mathbf{x}_e, \mathbf{u}, s, v))$ as the expression of the implicit function. Then, Equation (10) can be rewritten as $\mathbf{x}_e(t) = g(\mathbf{x}_e, h(\mathbf{x}_e, \mathbf{u}, s, v))$, which confirms that the function g is parameter-dependent in s and v.

DOI: 10.1002/rnc

In terms of Assumption 1 and Assumption 2, $(\mathbf{0}, \mathbf{0})$ is the equilibrium of the error dynamics (10), that is, $g(\mathbf{0}, \mathbf{0}) = \mathbf{0}$. The term $\mathbf{u}_e$ has to satisfy the following assumption:

Assumption 4
There exist a compact set $\mathcal{U}_e$ such that $\mathbf{u}_e \in \mathcal{U}_e$ and $0 \in \mathcal{U}_e$.

Remark 2.4
Assumption 4 is only used to find a terminal set that will be introduced in the next section.

Remark 2.5 (A way to find the function $g(\mathbf{x}_e, \mathbf{u}_e)$)
Suppose that there exists continuously differentiable functions $h_1(\mathbf{x}_e, s, v)$ and $h_2(\mathbf{x}_e, s, \mathbf{u}, v)$ such that

$$f(\mathbf{x}_e + \mathbf{p}(s), \mathbf{u}) - \frac{\partial \mathbf{p}}{\partial s} v = h_1(\mathbf{x}_e, s, v) + h_2(\mathbf{x}_e, s, \mathbf{u}, v),$$

and there exist $s \in \mathcal{S}$, $v \in \mathcal{V}$, and $\mathbf{u} \in \mathcal{U}$ such that $h_1(\mathbf{x}_e, s, v) = 0$ and $h_2(\mathbf{x}_e, s, \mathbf{u}, v) = 0$, while $\mathbf{x}_e = 0$, we can choose $\mathbf{u}_e := h_2(\mathbf{x}_e, s, \mathbf{u}, v)$, which results in $g(\mathbf{x}_e, \mathbf{u}_e) = h_1(\mathbf{x}_e, s, v) + \mathbf{u}_e$.

Because of Assumption 2, we know that while $\mathbf{x}_e(t) = 0$, there exist admissible inputs $\mathbf{u} \in \mathcal{U}$ and $v \in \mathcal{V}$, such that the dynamics of the state $\mathbf{x(t)} \in \mathcal{X}$ and the parameter $s(t) \in \mathcal{S}$ satisfy $\dot{\mathbf{x}}_e(t) = 0$. That is,

$$\begin{aligned} 0 =& g(0, \mathbf{u}_e) \\ =& f(\mathbf{p}(s), \mathbf{u}) - \frac{\partial \mathbf{p}}{\partial s} v. \end{aligned}$$

Thus, one option is to choose $h_1(\mathbf{x}_e, s, v) \equiv 0$ and $h_2(\mathbf{x}_e, s, \mathbf{u}, v) = f(\mathbf{x}_e + \mathbf{p}(s), \mathbf{u}) - \frac{\partial \mathbf{p}}{\partial s} v$. Therefore, the existence of the functions h_1 and h_2 is guaranteed.

Remark 2.6
By choosing $g(\cdot, \cdot)$ and $\mathbf{u}_e$ such that Assumption 3 is satisfied, we transform the path following problem into a parameter-dependent regulation problem, where $(\mathbf{0}, \mathbf{0})$ is the target state, s and v are the time-varying parameters.

Remark 2.7
Fundamental control problems can be roughly classified into three groups, which are point stabilization, tracking, and path following. Point stabilization and trajectory tracking problems can be seen as two special cases of the path following problem. While $s(t) \equiv c$ for all t, where c is a constant, the path following problem reduces to a point stabilization (regulation) problem; while $s(t)$ is exactly predefined, the path following problem is equal to a trajectory tracking problem. Comparing with the trajectory tracking problem, the path following problem has one additional DOF, which is to regulate the dynamics of s.

3. NONLINEAR MODEL PREDICTIVE CONTROL FOR PATH FOLLOWING PROBLEMS

In this section, we will discuss an NMPC scheme for the path following problem. After formulating the online optimization problem, a proof of recursive feasibility and asymptotic convergence of the introduced NMPC scheme is presented. Furthermore, a PLDI-based algorithm is proposed to choose a suitable terminal penalty and the related terminal constraint.

3.1. Optimization problem and algorithm

In order to formulate the path following problem within the NMPC framework, we consider the following *online* optimization problem at time instant t:

DOI: 10.1002/rnc

Problem 1

$$\underset{\mathbf{u}(\cdot,\mathbf{x}(t)),v(\cdot,\mathbf{x}(t)),s(t,\mathbf{x}(t))}{\text{minimize}} \quad J(\mathbf{x}(t)) \tag{11a}$$

subject to

$$\dot{\mathbf{x}}(\tau, x(t)) = f(\mathbf{x}(\tau, x(t)), \mathbf{u}(\tau, x(t))), \tag{11b}$$

$$\dot{s}(\tau, x(t)) = v(\tau, x(t)), \quad \mathbf{x}(t, \mathbf{x}(t)) = \mathbf{x}(t), \tag{11c}$$

$$\mathbf{x}_e(\tau, x(t)) = \mathbf{x}(\tau, x(t)) - \mathbf{p}(s(\tau, x(t))), \tag{11d}$$

$$\mathbf{u}(\tau, x(t)) \in \mathcal{U}, \ \mathbf{x}(\tau, x(t)) \in \mathcal{X}, \tag{11e}$$

$$v(\tau, x(t)) \in \mathcal{V}, \ s(\tau, x(t)) \in \mathcal{S}, \tag{11f}$$

$$\mathbf{x}_e(t + T_p, \mathbf{x}(t)) \in \Omega, \tag{11g}$$

with

$$J\,(\mathbf{x}(t)) = E(\mathbf{x}_e(t + T_p, \mathbf{x}(t))) + \int_t^{t+T_p} F(\mathbf{x}_e(\tau, \mathbf{x}(t)), \mathbf{u}_e(\tau, \mathbf{x}(t)))d\tau, \tag{12}$$

where $J(\mathbf{x}(t))$ is the cost functional, and T_p is the prediction horizon. The term $\mathbf{u}_e(\cdot, \mathbf{x}(t))$ denotes the predicted input function of the error system related to $\mathbf{x}(t)$, and $\mathbf{x}_e(\cdot, \mathbf{x}(t))$ represents the predicted state trajectory of the error system under the control $\mathbf{u}_e(\cdot, \mathbf{x}(t))$. The terms $E(\mathbf{x}_e(t+T_p, \mathbf{x}(t)))$ and $\mathbf{x}_e(t+T_p, \mathbf{x}(t)) \in \Omega$ are the terminal penalty and the terminal constraint, respectively, which are used to guarantee recursive feasibility and achieve asymptotic convergence to the given path. The term $F(\cdot, \cdot)$ is the stage cost function, which specifies the desired control performance and satisfies the following condition.

Assumption 5
$F(\cdot, \cdot) : \mathcal{X} \times \mathcal{U} \to \mathbb{R}^1$ is continuous, and $F(\mathbf{0}, \mathbf{0}) = \mathbf{0}$ and $F(\mathbf{x}, \mathbf{u}) > 0$ for all $(\mathbf{x}, \mathbf{u}) \in \mathcal{X} \times \mathcal{U} \setminus \{\mathbf{0}, \mathbf{0}\}$.

For clarity, $\mathbf{u}(\tau, \mathbf{x}(t))$ denotes the predicted input function related to the measured state $\mathbf{x}(t)$ at time instant t, and $\mathbf{x}(\cdot, \mathbf{x}(t))$ represents the predicted state trajectory starting from $\mathbf{x}(t)$ under the control $\mathbf{u}(\tau, \mathbf{x}(t))$, for all $\tau \in [t, t + T_p]$. The notations $v(\tau, \mathbf{x}(t))$, $s(\tau, \mathbf{x}(t))$, and $\mathbf{x}_e(\tau, \mathbf{x}(t))$ refer to the predicted values of v, s, and $\mathbf{x}_e$ at time τ related to $\mathbf{x}(t)$, respectively.

Remark 3.1
The cost functional J and the terminal constraint $\mathbf{x}_e(t + T_p, x(t)) \in \Omega$ do not depend explicitly on the parameter s or v, which is consistent with the fact that s and v only describe a virtual reference motion.

Suppose the sampling time is δ, the proposed NMPC control law is formally described by the following algorithm.

Algorithm 1

Step 1: Measure system state $\mathbf{x}(t)$ at time t,
Step 2: Solve Problem 1 and obtain a feasible (suboptimal) solution $s^0(t, \mathbf{x}(t))$, $\mathbf{u}^0(\cdot, \mathbf{x}(t))$ and $v^0(\cdot, \mathbf{x}(t))$ for $\tau \in [t, t + T_p]$,
Step 3: Take the input value $\mathbf{u}^0(\tau, \mathbf{x}(t))$, $\tau \in [t, t + \delta]$, as the current input for the system,
Step 4: Take the input value $v^0(\tau, \mathbf{x}(t))$ and the initial state $s^0(t, \mathbf{x}(t))$ to update the path parameter $s(\tau, \mathbf{x}(t))$ for $\tau \in [t, t + \delta]$,
Step 5: Set $t := t + \delta$, go to Step 1.

Remark 3.2
The initial state of $\dot{s}(t) = v(t)$ is chosen as a determined variable in Problem 1, that is, the reference motion $s(t)$ is renewed entirely at each time instant. Thus, it provides an extra DOF of optimization problem.

Remark 3.3
Because Problem 1 is a nonlinear and nonconvex optimization problem, in general, it is impossible to obtain the exact globally optimal solution even if there exists a globally optimal solution.

3.2. Feasibility and stability

As important issues of ensuring feasibility and convergence of the NMPC scheme, the terminal penalty $E(\mathbf{x}_e)$, the terminal set Ω, and the corresponding fictitious terminal control law $\pi(\mathbf{x}_e)$ are required to satisfy the following conditions:

B0. $\Omega \subseteq \mathcal{X}$,
B1. $\pi(0) = 0$, and $\pi(\mathbf{x}_e) \in \mathcal{U}_e$ for all $\mathbf{x}_e \in \Omega$,
B2. $E(0) = 0$, and $E(\mathbf{x}_e)$ satisfies

$$\frac{\partial E(\mathbf{x}_e)}{\partial \mathbf{x}_e} g(\mathbf{x}_e, \pi(\mathbf{x}_e)) + F(\mathbf{x}_e, \pi(\mathbf{x}_e)) \leqslant 0, \tag{13}$$

for all $\mathbf{x}_e \in \Omega$.

As a neighborhood of the error state $\mathbf{x}_e = 0$,

$$\Omega := \{\mathbf{x}_e \in \mathbb{R}^n \mid E(\mathbf{x}_e) \leqslant \alpha\}, \tag{14}$$

with $\alpha > 0$.
Clearly, the terminal set Ω has the following additional properties:

1. The point $\mathbf{0} \in \mathbb{R}^n$ is contained in the interior of Ω because of the positive definiteness of $E(\mathbf{x}_e)$.
2. Ω is closed and connected because of the continuity of $E(\mathbf{x}_e)$.
3. Ω is *robustly* invariant for the nonlinear system (10) controlled by $\mathbf{u}_e = \pi(\mathbf{x}_e)$, for all $s(\cdot) \in \mathcal{S}$ and $v(\cdot) \in \mathcal{V}$ because of (13).

Assumption 6
For the error system (10), there exist a locally asymptotically stabilizing controller $\pi(\mathbf{x}_e)$, a terminal set $\Omega \subseteq \mathcal{X}$, and a continuously differentiable, positive semi-definite function $E(\mathbf{x}_e)$ such that conditions B0–B2 are satisfied for all $\mathbf{x}_e \in \Omega$.

Assumption 7
There exist $s \in \mathcal{S}$ and $v \in \mathcal{V}$ such that $\mathbf{x} \in \mathcal{X}$ and $\mathbf{u} \in \mathcal{U}$, while $\mathbf{x}_e \in \Omega$ and $\mathbf{u}_e \in \mathcal{U}_e$.

Now, we are ready to show the recursive feasibility of the considered optimization problem and the asymptotic convergence of the path following problem.

Theorem 1
Suppose that

(a) Assumptions 1–7 are satisfied,
(b) at the initial time instant, Problem 1 has a feasible solution,

then,

1. Problem 1 is feasible for all time instants,
2. the system state $\mathbf{x}(t)$ follows the predefined geometric path $\mathcal{P}$ asymptotically, that is, $\lim_{t\to\infty} \mathbf{x}_e(t) = 0$.

Proof
Assume that Problem 1 has a feasible solution at time instant t, which is $\big(\mathbf{u}^0(\tau, \mathbf{x}(t)), v^0(\tau, \mathbf{x}(t)), s^0(t, \mathbf{x}(t))\big)$ for $\tau \in [t, t + T_p]$. The corresponding input and the state of the error system (10) are $\mathbf{u}_e^0(\tau, \mathbf{x}(t))$ and $\mathbf{x}_e^0(\tau, \mathbf{x}(t))$, respectively.

1. The input $\mathbf{u}^0(\tau, \mathbf{x}(t))$ is implemented, and the related dynamic of the system (1) is $\mathbf{x}^0(\tau, \mathbf{x}(t))$, for all $\tau \in [t, t+\delta]$. The solution $s^0(\tau, \mathbf{x}(t))$ and $v^0(\tau, \mathbf{x}(t))$, $\tau \in [t, t+T_p]$ are used to obtain the evolution of the system (4). Because neither model-plant mismatches nor external disturbances are present, $\mathbf{x}(t+\delta) = \mathbf{x}^0(t+\delta, \mathbf{x}(t))$. Thus, the remaining piece of the inputs $\mathbf{u}^0(\tau, \mathbf{x}(t))$ and $v^0(\tau, \mathbf{x}(t))$, $\tau \in [t+\delta, t+T_p]$ satisfy the constraints of Problem 1. Denote $\mathbf{x}^0(t+T_p) := \mathbf{x}^0(t+T_p, \mathbf{x}(t))$. Because $\mathbf{x}_e^0(t+T_p, \mathbf{x}(t)) \in \Omega$, it follows from Assumptions 6 and 7 that $\pi(\cdot)$ renders Ω invariant, and there exist $s(\tau, \mathbf{x}^0(t+T_p)) \in \mathcal{S}$ and $v(\tau, \mathbf{x}^0(t+T_p)) \in \mathcal{V}$ such that $\mathbf{x}(\tau, \mathbf{x}^0(t+T_p)) \in \mathcal{X}$ and $\mathbf{u}(\tau, \mathbf{x}^0(t+T_p)) \in \mathcal{U}$, for all $\tau \in [t+T_p, t+T_p+\delta]$. The dynamics of the error system (10) under the terminal control law $\pi(\cdot)$ is $\mathbf{x}_e(\tau, \mathbf{x}^0(t+T_p))$ for all $\tau \geq t+T_p$. Therefore, a feasible solution to Problem 1 at time instant $t+\delta$ is $(\mathbf{u}(\tau, \mathbf{x}(t+\delta)), v(\tau, \mathbf{x}(t+\delta)), s(t+\delta, \mathbf{x}(t+\delta)))$ where $s(t+\delta, \mathbf{x}(t+\delta)) := s^0(t+\delta, \mathbf{x}(t))$, and
$$\mathbf{u}(\tau, \mathbf{x}(t+\delta)) := \begin{cases} \mathbf{u}^0(\tau, \mathbf{x}(t)) & \tau \in [t+\delta, t+T_p), \\ \mathbf{u}(\tau, \mathbf{x}^0(t+T_p)) & \tau \in [t+T_p, t+T_p+\delta], \end{cases}$$
$$v(\tau, \mathbf{x}(t+\delta)) := \begin{cases} v^0(\tau, \mathbf{x}(t)) & \tau \in [t+\delta, t+T_p), \\ v(\tau, \mathbf{x}^0(t+T_p)) & \tau \in [t+T_p, t+T_p+\delta]. \end{cases}$$

2. Let us define a Lyapunov-like function candidate as
$$V(\mathbf{x}(t)) := J(\mathbf{x}(t)), \tag{15}$$
for fixed $\mathbf{u}^0(\tau, \mathbf{x}(t))$, $v^0(\tau, \mathbf{x}(t))$, and $s^0(t, \mathbf{x}(t))$, with $\tau \in [t, t+T_p]$. Note that $0 \leqslant V(\mathbf{x}(t)) < +\infty$, which follows directly from the definition of $V(\cdot)$ and $V(\mathbf{x}(t)) = 0$, while $\mathbf{x}(t) = \mathbf{p}(s(t))$.
At time instant t, the cost functional is
$$V(\mathbf{x}(t)) = E\left(\mathbf{x}_e^0(t+T_p, \mathbf{x}(t))\right) + \int_t^{t+T_p} F\left(\mathbf{x}_e^0(\tau, \mathbf{x}(t)), \mathbf{u}_e^0(\tau, \mathbf{x}(t))\right) d\tau. \tag{16}$$
Note that $V(x)$ is not unique because only a feasible solution is considered. Considering the feasible solution at time instant $t+\delta$ for Problem 1, and recalling $\pi(\cdot)$, which renders Ω invariant, we have
$$\begin{aligned} J(\mathbf{x}(t+\delta)) = {} & E\left(\mathbf{x}_e(t+\delta+T_p, \mathbf{x}^0(t+T_p))\right) \\ & + \int_{t+\delta}^{t+T_p} F\left(\mathbf{x}_e^0(\tau, \mathbf{x}(t)), \mathbf{u}_e^0(\tau, \mathbf{x}(t))\right) d\tau \\ & + \int_{t+T_p}^{t+T_p+\delta} F\left(\mathbf{x}_e(\tau, \mathbf{x}^0(t+T_p)), \pi(\mathbf{x}_e(\tau, \mathbf{x}^0(t+T_p)))\right) d\tau. \end{aligned} \tag{17}$$
Because the ‘possible" solution is better than the feasible solution, otherwise, we can use directly the feasible solution, we have $V(\mathbf{x}(t+\delta)) \leqslant J(\mathbf{x}(t+\delta))$. Thus,
$$\begin{aligned} V(\mathbf{x}(t+\delta)) - V(\mathbf{x}(t)) & \leqslant J(\mathbf{x}(t+\delta)) - V(\mathbf{x}(t)) \\ & = -\int_t^{t+\delta} F\left(\mathbf{x}_e^0(\tau, \mathbf{x}(t)), \mathbf{u}_e^0(\tau, \mathbf{x}(t))\right) d\tau \\ & \quad + \int_{t+T_p}^{t+T_p+\delta} F\left(\mathbf{x}_e(\tau, \mathbf{x}^0(t+T_p)), \pi(\mathbf{x}_e(\tau, \mathbf{x}^0(t+T_p)))\right) d\tau \\ & \quad + E\left(\mathbf{x}_e(t+\delta+T_p, \mathbf{x}^0(t+T_p))\right) - E\left(\mathbf{x}_e^0(t+T_p, \mathbf{x}(t))\right). \end{aligned}$$
From the integration of inequality (13), the aforementioned inequality
$$\begin{aligned} & E\left(\mathbf{x}_e(t+\delta+T_p, \mathbf{x}^0(t+T_p))\right) - E\left(\mathbf{x}_e^0(t+T_p, \mathbf{x}(t))\right) \\ & \leqslant -\int_{t+T_p}^{t+T_p+\delta} F\left(\mathbf{x}_e(\tau, \mathbf{x}^0(t+T_p)), \pi(\mathbf{x}_e(\tau, \mathbf{x}^0(t+T_p)))\right) d\tau \end{aligned}$$

DOI: 10.1002/rnc

results in

$$V(\mathbf{x}(t+\delta)) - V(\mathbf{x}(t)) \leqslant -\int_t^{t+\delta} F(\mathbf{x}_e^0(\tau, \mathbf{x}(t)), \mathbf{u}_e^0(\tau, \mathbf{x}(t)))d\tau.$$

Clearly, $V(\mathbf{x}(t))$ is a monotonically decreasing function and has zero as its low bound. The state of the error system (10) will converge to zero as time increases [7]. Accordingly, the state of the system (1) will finally follow the reference path (3), that is, $\lim_{t\to\infty} \mathbf{x}_e(t) = 0$.

□

Remark 3.4
The aforementioned proof shows that applying feasible solution to optimization problem at each time instant is sufficient to guarantee both recursive feasibility and asymptotic convergence. This is one of the main advantages of the proposed NMPC scheme.

Remark 3.5
Because $\lim_{t\to\infty} x_e(t) = 0$ rather than $x_e \equiv 0$, while $t \to \infty$; see Theorem 1, the reference path is not necessary an admissible trajectory of the systems.

3.3. Terminal set with a static terminal control law

To choose a suitable pair of terminal penalty and terminal constraint that satisfies all the assumptions and conditions earlier, we will propose a PLDI-based method. The calculated terminal control law is robust with respect to the parameters v and s, and the terminal set is a corresponding robust invariant set.

Firstly, we discuss how to guarantee the satisfaction of inequality (13), while a quadratic stage cost $F(\mathbf{x}_e, \mathbf{u}_e) := \mathbf{x}_e^T Q\mathbf{x}_e + \mathbf{u}_e^T R\mathbf{u}_e$ is considered.

Because $\mathbf{x}_e = 0$ is an equilibrium of the error system (10), there exists a set Σ_0 such that, for all $\mathbf{x}_e \in \mathcal{X}_e$ and $\mathbf{u}_e \in \mathcal{U}_e$,

$$g(\mathbf{x}_e, \mathbf{u}_e) \in \Sigma_0 \begin{bmatrix} \mathbf{x}_e \\ \mathbf{u}_e \end{bmatrix},$$

where $\mathcal{X}_e \subseteq \mathcal{X}$ is a compact set, $\Sigma_0 \subseteq \mathbb{R}^{n\times(n+m)}$ is a parameterized differential inclusion of the nonlinear system (10) with respect to the parameters s and v.

Because $s \in \mathcal{S}$ and $v \in \mathcal{V}$, and $\mathcal{S}$ and $\mathcal{V}$ are compact sets, Σ_0 can be bounded by a PLDI Σ

$$\Sigma := \mathrm{Co}\left\{\begin{bmatrix} A_{1x} & B_{1u} \end{bmatrix}, \ldots, \begin{bmatrix} A_{Nx} & B_{Nu} \end{bmatrix}\right\}, \tag{18}$$

that is, $\Sigma_0 \subseteq \Sigma$. The term N is the number of vertex matrices, $\begin{bmatrix} A_{ix} & B_{iu} \end{bmatrix}$ is the vertex matrix of the set Σ, where $i \in [1, N]$, $A_{ix} \in \mathbb{R}^{n\times n}$, and $B_{iu} \in \mathbb{R}^{n\times m}$.

Note that, since the error system (10) is continuously differentiable, the set Σ can be obtained by choosing

$$\begin{bmatrix} \frac{\partial g(\mathbf{x}_e,\mathbf{u}_e)}{\partial \mathbf{x}_e} & \frac{\partial g(\mathbf{x}_e,\mathbf{u}_e)}{\partial \mathbf{u}_e} \end{bmatrix} \in \Sigma,$$

for all $\mathbf{x}_e \in \mathcal{X}_e$, $\mathbf{u}_e \in \mathcal{U}_e$, $s \in \mathcal{S}$, and $v \in \mathcal{V}$.

Based on the PLDI in (18), a sufficient condition that guarantees the satisfaction of Equation (13) is proposed, which is given in terms of linear matrix inequalities (LMIs).

Theorem 2
Let $Q \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{m\times m}$ be given. Suppose that the PLDI of the system (10) is described by (18) and suppose that there exist a matrix $X > 0$ with $X \in \mathbb{R}^{n\times n}$ and a matrix $Y \in \mathbb{R}^{m\times n}$ such that

$$\begin{bmatrix} A_{ix}X + B_{iu}Y + (A_{ix}X + B_{iu}Y)^T & X & Y^T \\ X & -Q^{-1} & 0 \\ Y & 0 & -R^{-1} \end{bmatrix} \leqslant 0, \tag{19}$$

for all $i \in [1, N]$. Then, for the system (10), inequality (13) is satisfied, where $E(\mathbf{x}_e) := \mathbf{x}_e^T P\mathbf{x}_e$, $P = X^{-1}$, and $\pi(\mathbf{x}_e) = YX^{-1}\mathbf{x}_e$.

DOI: 10.1002/rnc

Proof
For simplicity, denote $K := YX^{-1}$. By substituting $P = X^{-1}$ and $Y = KX$ in (19) and by performing a congruence transformation with the matrix $\{X^{-1}, I, I\}$, we obtain

$$\begin{bmatrix} A_{i,cl}^T P + PA_{i,cl} & I & K^T \\ I & -Q^{-1} & 0 \\ K & 0 & -R^{-1} \end{bmatrix} \leqslant 0,$$

where $A_{i,cl} := A_{ix} + B_{iu}K$, I is the compatible identity matrix. It follows from the Schur complement that the matrix inequality (19) is sufficient to guarantee

$$[g(\mathbf{x}_e, K\mathbf{x}_e)]^T P + Pg(\mathbf{x}_e, K\mathbf{x}_e) + Q + K^T RK \leqslant 0. \tag{20}$$

We choose $E(\mathbf{x_e}) = \mathbf{x_e}^T P\mathbf{x_e}$ as a Lyapunov function candidate, and $\mathbf{u}_e = K\mathbf{x}_e$, the time derivative of $E(\mathbf{x}_e)$ along the trajectory of (10) is given as follows:

$$\begin{aligned} \frac{dE(\mathbf{x}_e(t))}{dt} &= \dot{\mathbf{x}}_e(t)^T P\mathbf{x}_e(t) + \mathbf{x}_e(t)^T P\dot{\mathbf{x}}_e(t) \\ &= \mathbf{x}_e(t)^T \left\{[g(\mathbf{x}_e, K\mathbf{x}_e)]^T P + Pg(\mathbf{x}_e, K\mathbf{x}_e)\right\} \mathbf{x}_e(t). \end{aligned}$$

Because of (20), we have

$$\frac{dE(\mathbf{x}_e(t))}{dt} \leqslant -\mathbf{x}_e(t)^T Q\mathbf{x}_e(t) - \mathbf{x}_e(t)^T K^T RK\mathbf{x}_e(t).$$

Thus, the inequality (13) holds, and $\pi(\mathbf{x}_e)$ is the associated terminal control law. □

Theorem 2 shows that $\pi(\mathbf{x}_e)$ and $E(\mathbf{x}_e)$ as given earlier can serve as a terminal control law and a terminal penalty, respectively, for the proposed NMPC scheme.

From the aforementioned discussion, an algorithm is proposed to determine a terminal penalty matrix P and a terminal set Ω *offline* such that inequality (13) holds true, and the input constraints $\mathbf{u}_e \in \mathcal{U}_e$ are satisfied.

Algorithm 2

Initialization: The matrices $Q \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{m\times m}$ be given.

Step 1: Solve LMI (19) to obtain a locally stabilizing linear state feedback law $\pi(\mathbf{x}_e)$ and a positive definite matrix P,

Step 2: Find the largest positive α such that $\Omega \in \mathcal{X}_e$ and $\pi(\mathbf{x}_e) \in \mathcal{U}_e$ for all $\mathbf{x}_e \in \Omega$.

Remark 3.6
An algorithm for simultaneous satisfaction of (19) and the input and state constraints are discussed in [15]. As an alternative, similar algorithms that satisfy (19) and the constraints separately are proposed in [7, 16].

Remark 3.7
Because the system (1) can follow the reference path if $\mathbf{x}_e = 0$ because of Assumptions 1 and 2, the reference path can be chosen as the terminal set, that is, $\Omega := \{0\}$, while LMI (19) has no feasible solution. To satisfy the Assumption 6, the corresponding terminal penalty and the terminal control law can be $E(\cdot) := 0$ and $\pi(\cdot) := 0$, respectively.

4. PATH FOLLOWING CONTROL OF A CAR-LIKE MOBILE ROBOT

To illustrate the implementation of the proposed NMPC scheme, the path following problem of a car-like mobile robot is considered in this section.

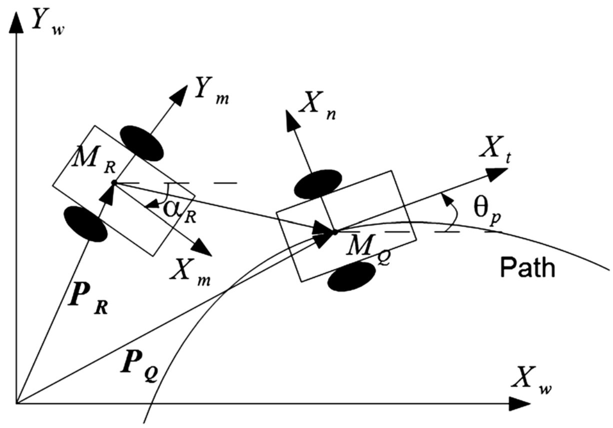

Figure 1. Path following problem of a car-like mobile robot.

4.1. Problem formulation

A car-like mobile robot is a kind of nonholonomic robot, which is not able to move in the direction parallel to the wheels' axes. With definition of a world coordinate system $\{W\}$ composed of axes X_w and Y_w shown in Figure 1, the kinematics model of the car-like mobile robot is described by

$$\begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\alpha}_R \end{bmatrix} = \begin{bmatrix} v_R \cos \alpha_R \\ v_R \sin \alpha_R \\ \omega_v \end{bmatrix}, \tag{21}$$

where x_R and y_R denote the position of the robot center of mass with respect to $\{W\}$, v_R is the magnitude of the robot translational velocity, α_R denotes the robot moving direction, and ω_v is the angular velocity of α_R.

To depict the path following problem of the car-like mobile robot, Figure 1 shows also a mobile robot coordinate system $\{M_R\}$ with axes X_m and Y_m, which is located at the mass of center $\mathbf{M}_R$ of the robot, and a path coordinate system $\{M_Q\}$ composed of axes X_t and Y_n, which plays a role of the body-axis of a *Virtual Vehicle* to be followed. The *Virtual Vehicle* moves along the reference path $\mathcal{P}$ and its position $\mathbf{M}_Q$ is determined by the curvilinear abscissa $s(t)$. Let vectors $\mathbf{P}_R$ and $\mathbf{P}_Q$ describe the positions of $\mathbf{M}_R$ and $\mathbf{M}_Q$ in $\{W\}$, ${}^wR_m$ and ${}^wR_p$ present the transformation matrices from $\{M_R\}$ to $\{W\}$ and from $\{M_Q\}$ to $\{W\}$, respectively, the position relationship can be deduced:

$$\mathbf{P}_R = \mathbf{P}_Q + {}^wR_p \begin{pmatrix} x_e \\ y_e \end{pmatrix}, \tag{22}$$

where x_e and y_e denote the robot position $\mathbf{M}_R$ with respect to $\{M_Q\}$.

Computing the time derivatives of (22) and some calculation leads to the error kinematics model of the path following problem

$$\dot{\mathbf{x}}_e = \begin{bmatrix} (y_e c(s) - 1)v + v_R \cos \alpha_e \\ -x_e c(s)v + v_R \sin \alpha_e \\ \omega_v - c(s)v \end{bmatrix} \tag{23}$$

where the error vector with respect to $\{M_Q\}$ is $\mathbf{x}_e := [x_e, y_e, \alpha_e]^T$, $\alpha_e = \alpha_R - \theta_p$ presents the angular error between the robot moving direction α_R and the path tangent direction θ_p. The value of s is the arc length measured along the path from its origin to the point $\mathbf{p}(s)$, and $c(s)$ denotes the path curvature at point $\mathbf{M}_Q$.

Based on the error kinematics model (23), the path following problem of a car-like mobile robot can be formulated as follows:

Given a geometric path $\mathcal{P}$ defined by (3), find suitable control laws of v and ω_v to drive the errors x_e, y_e, and α_e to zero, while v_R is assigned with a nonzero magnitude value of the desired velocity.

It is noticed in model (23) that the errors can stay on the equilibrium ($\mathbf{x}_e = 0$) when v approaches v_R.

DOI: 10.1002/rnc

4.2. Simulation setup

Considering the geometrical symmetry and sharp changes in curvature, an eight-shaped curve is adopted as the reference path,

$$\begin{aligned} x_P &= 1.8sin(\psi) \\ y_P &= 1.2sin(2\psi), \end{aligned} \tag{24}$$

where ψ is a path parameter and determines the path point $[x_P, y_P]$ with respect to the world coordinate system. It has bounded curvature value of $-3.28 \leqslant c(s) \leqslant 3.28$. The changing velocity of s is bounded by $0 \leqslant v \leqslant 1.2$ m/s.

The car-like robot is required to move with a constant velocity $v_R = 0.7$ m/s. The angular velocity is bounded by $-2.5 \leqslant \omega_v \leqslant 2.5$ rad/s.

By choosing different Lyapunov functions of the positional and angular errors, many nonlinear control approaches have been presented [17–21]. However, most of them rarely take the robot constraints into account, which are crucial for robot performance, and may even destroy the convergence in some cases [22, 23]. In contrast to them, the proposed NMPC scheme takes the constraints into account and achieves the expected performance.

4.3. Nonlinear model predictive control law design

To implement the proposed NMPC scheme for the path following problem of a car-like mobile robot, the error kinematics model (23) needs to be transformed into the form satisfying $g(0,0) = 0$. Denoting states $\mathbf{x}_e = [x_{e1}, x_{e2}, x_{e3}]^T = [x_e, y_e, \alpha_e]^T$ and defining inputs $\mathbf{u}_e := [u_{e1}, u_{e2}]^T$ with

$$\begin{bmatrix} u_{e1} \\ u_{e2} \end{bmatrix} = \begin{bmatrix} -v + v_R \cos x_{e3} \\ \omega_v - c(s)v \end{bmatrix}, \tag{25}$$

the model (23) is rewritten as a candidate of the required error model, where

$$\dot{\mathbf{x}}_e = \begin{bmatrix} x_{e2}c(\mathbf{s})v + u_{e1} \\ -x_{e1}c(\mathbf{s})v + v_R \sin x_{e3} \\ u_{e2} \end{bmatrix}. \tag{26}$$

A quadratic function is selected as the stage cost function,

$$F(\mathbf{x}_e, \mathbf{u}_e) = \mathbf{x}_e^T Q \mathbf{x}_e + \mathbf{u}_e^T R \mathbf{u}_e. \tag{27}$$

The weight matrices are chosen with $Q = 0.5\mathbf{I}_3$ and $R = 0.5\mathbf{I}_2$, where $\mathbf{I}_j$ denotes the unit diagonal matrix of dimension j. The prediction horizon is 10, and the sampling time of δ is 0.02 sec.

To choose the terminal penalty and the terminal constraint, we use the scheme presented in Section 3.3. The terminal penalty is

$$E(\mathbf{x}_e(t + T_p)) = \mathbf{x}_e(t + T_p)^T P \mathbf{x}_e(t + T_p). \tag{28}$$

The terminal constraint is chosen as a sublevel set of the terminal penalty, that is, $E(\mathbf{x}_e(t+T_p)) \leqslant \alpha$.

The value of P and α comes from Algorithm 2, where the PLDI of the error kinematics model (23) is required. According to the following partial derivative,

$$\begin{bmatrix} \frac{\partial g}{\partial x_{e1}} & \frac{\partial g}{\partial x_{e2}} & \frac{\partial g}{\partial x_{e3}} & \frac{\partial g}{\partial u_{e1}} & \frac{\partial g}{\partial u_{e2}} \end{bmatrix} = \begin{bmatrix} 0 & c(\mathbf{s})v & 0 & 1 & 0 \\ -c(\mathbf{s})v & 0 & v_R \cos x_{e3} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{29}$$

the vertex matrix of the PLDI can be obtained based on the boundary values of v and $c(s)$, while they are all bounded variables defined by the reference path.

Here, by defining $0.05 \leqslant v_R \cos \alpha_e \leqslant 0.7$ and by taking the limits of the car-like robot and the eight-shaped reference path into account, we obtain the following vertex matrices of the PLDI based on (29),

DOI: 10.1002/rnc

$$\begin{bmatrix} A_1 & B_1 \end{bmatrix} = \begin{bmatrix} 0 & 3.28 & 0 & 1 & 0 \\ -3.28 & 0 & 0.7 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\begin{bmatrix} A_2 & B_2 \end{bmatrix} = \begin{bmatrix} 0 & -3.28 & 0 & 1 & 0 \\ 3.28 & 0 & 0.7 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\begin{bmatrix} A_3 & B_3 \end{bmatrix} = \begin{bmatrix} 0 & 3.28 & 0 & 1 & 0 \\ -3.28 & 0 & 0.05 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\begin{bmatrix} A_4 & B_4 \end{bmatrix} = \begin{bmatrix} 0 & -3.28 & 0 & 1 & 0 \\ 3.28 & 0 & 0.05 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

To satisfy Assumption 4, the range of $\mathbf{u}_e$ is chosen as $|u_{e1}| \leqslant 0.5$ and $|u_{e2}| \leqslant 1.44$. The terminal penalty matrix is

$$P = \begin{bmatrix} 28.36 & 0 & 0 \\ 0 & 30.02 & 8.89 \\ 0 & 8.89 & 47.04 \end{bmatrix},$$

and the value of α is 25, which are solved by using Algorithm 2.

4.4. *Comparison controller*

In order to evaluate the proposed NMPC scheme, a control Lyapunov function based scheme according to [18] is exploited. It involves the following equations:

$$\begin{aligned} \sigma(y_e) &= -sign(v_R)\sin^{-1}\frac{k_2 y_e}{\|y_e\| + \epsilon_0} \\ \delta(\alpha_e, \sigma) &= \begin{cases} 1 & if\ \alpha_e = \sigma \\ \frac{\sin\alpha_e - \sin\sigma}{\alpha_e - \sigma} & otherwise \end{cases} \\ v &= v_R\cos\alpha_e + k_3 x_e \\ \omega_v &= c(s)v + \dot{\sigma} - k_1(\alpha_e - \sigma) - y_e v_R \delta. \end{aligned} \tag{30}$$

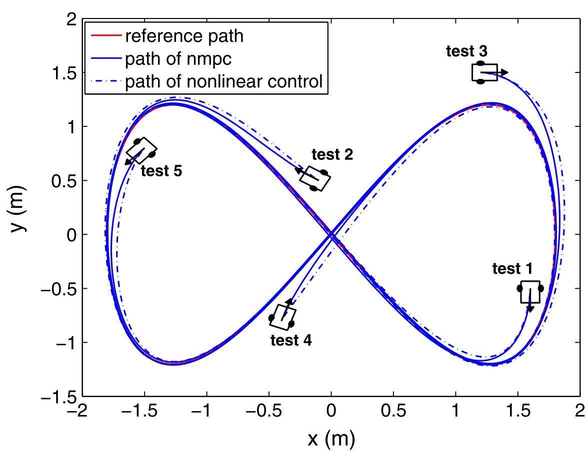

Figure 2. The reference path and real paths based on the nonlinear model predictive control and the nonlinear controller in (30).

DOI: 10.1002/rnc

For some $k_1,\ k_3 > 0$, $0 < k_2 \leqslant 1$, and $\epsilon_0 > 0$, this controller guarantees global stability, which is proven by choosing the Lyapunov function $V = \frac{1}{2}x_e^2 + \frac{1}{2}y_e^2 + \frac{1}{2}(\alpha_e - \sigma)^2$. Here, the control parameters are chosen as $k_1 = 15$, $k_2 = 0.8$, $k_3 = 10$, and $\epsilon_0 = 1$.

4.5. Simulation results

In the simulation, the robot was started from five different initial positions with different heading directions. As Figure 2 shows, both controllers are capable of driving the robot to follow the reference path, but the path based on the NMPC scheme converges faster. Figure 3 shows the values

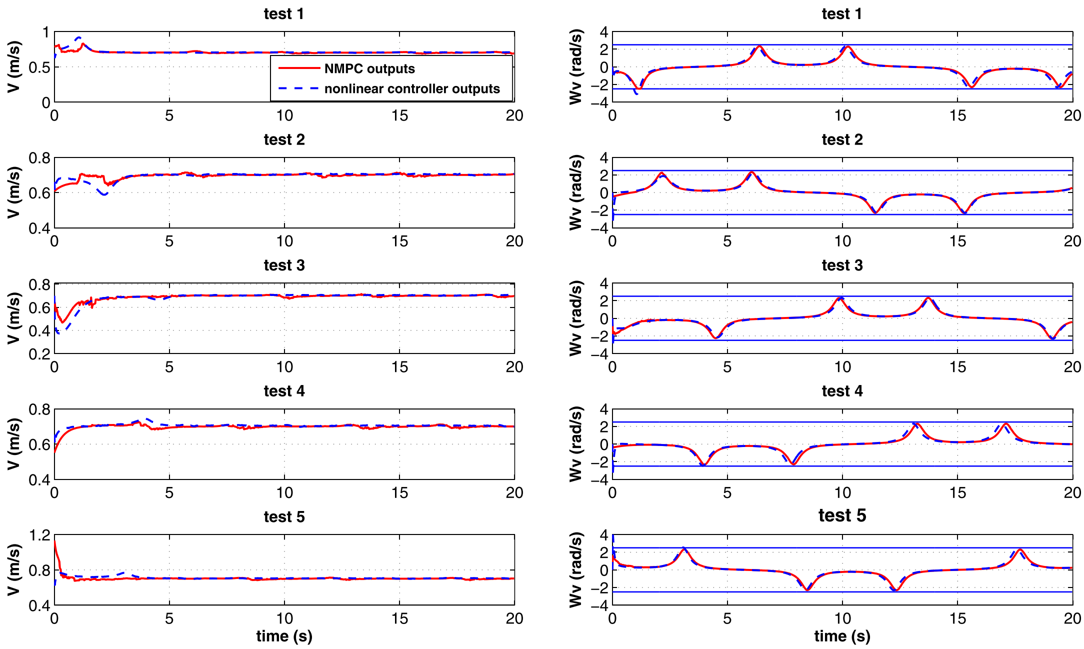

Figure 3. The velocities v and the angular velocities ω_v from the nonlinear model predictive control and the nonlinear controller in (30), which are shown in a solid line and a dashed line, respectively, $t \in [0,\ 20]$.

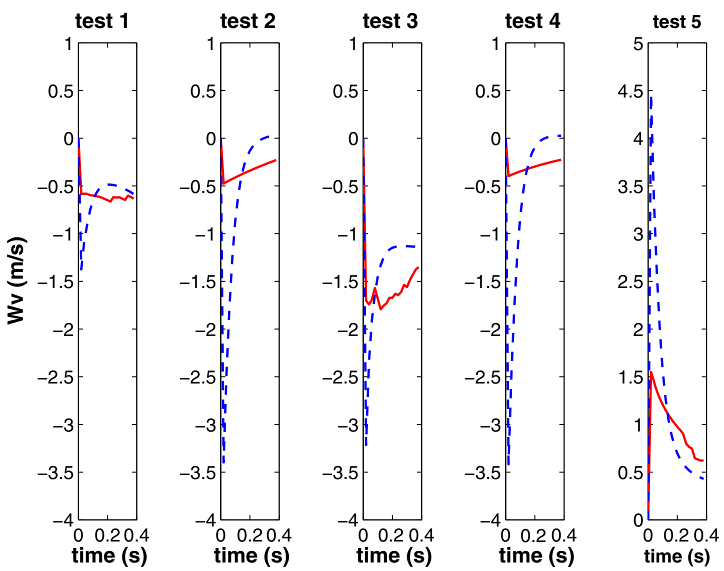

Figure 4. The angular velocities ω_v from the nonlinear model predictive control and the nonlinear controller in (30), which are shown in a solid line and a dashed line, respectively, $t \in [0,\ 0.4]$.

of ω_v and v from the two controllers. It is clear that the difference only appear at the initial period of the simulation, and it follows the reference path once the robot steps on the reference path. The two controllers have similar control performance when the error $\dot{\mathbf{x}}_e$ is small, because the nonlinear controller (30) has similar form of the designed u_e; see (25). Although the values of v generated by the nonlinear controller located inside the boundary $[0, 1.2]$ during the initial period, Figure 4 shows that the initial errors result in big values of the angular velocity ω_v generated by the nonlinear controller. However, the NMPC takes the constraints into account, whose outputs are all inside the boundary values.

5. CONCLUSIONS

This paper presented a general NMPC scheme for the path following problem, where the time evolution of the path parameter and its initial value are all determined online. Not only the asymptotic convergence and the recursive feasibility of the proposed NMPC scheme, but also a PLDI-based method to choose the terminal penalty and the terminal constraint were shown. To illustrate the implementation of the proposed NMPC scheme, the path following problem of a car-like mobile robot was discussed in detail. Compared with a well-known nonlinear control algorithm, the advantage of the proposed NMPC scheme is shown in the simulation results.

ACKNOWLEDGEMENTS

Shuyou Yu and Hong Chen gratefully acknowledge the support by the 973 Program under grant no. 2012CB821202 and by the Program for Changjiang Scholars and Innovative Research Team in University under grant no. IRT1017. Shuyou Yu, Xiang Li and Frank Allgöwer would also like to thank the German Research Foundation (DFG) for their financial support of the project $AL\ 316/6-1$.

REFERENCES

1. Findeisen R, Imsland L, Allgöwer F. State and output feedback nonlinear model predictive control: an overview. *European Journal of Control* 2003; **9**(2-3):179–195.
2. Mayne DQ, Rawlings JB, Rao CV, Scokaert POM. Constrained model predictive control: stability and optimality. *Automatica* 2000; **36**(6):789–814.
3. Qin SJ, Badgwell TA. A survey of industrial model predictive control technology. *Control Engineering Practice* 2003; **11**(7):733–764.
4. Rawlings JB, Mayne DQ. *Model Predictive Control: Theory and Design*. Nob Hill Publishing: Madison, Wisconsin, 2009.
5. Jadbabaie A, Hauser J. Control of a thrust-vectored flying wing: a receding horizon–LPV approach. *International Journal of Robust Nonlinear Control* 2002; **12**:869–896.
6. Negenborn RR, Schutter BD, Hellendoorn H. Multi-agent model predictive control for transportation networks: serial versus parallel schemes. *The 12th IFAC Symposium on Information Control Problems*: IFAC, Saint-Etienne, France, 2006; 339–344.
7. Chen H, Allgöwer F. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica* 1998; **34**(10):1205–1217.
8. Magni L, Nicolao GD, Scattolini R. A stabilizing model-based predictive control algorithm for nonlinear systems. *Automatica* 2001; **37**(9):1351–1362.
9. Chisci L, Zappa G. Dual mode predictive tracking of piecewise constant references for constrianed linear systems. *International Journal of Control* 2003; **76**(1):61–72.
10. Ferramosca A, Limon D, Alvarado I, Alamo T, Camacho EF. MPC for tracking of constrained nonlinear systems. *Proceedings of the 48th IEEE Conference Decision Control and 28th Chinese Control Conference*, Shanghai, P.R. China, 2009; 7978–7983.
11. Limon D, Alvarado I, Camacho EF. MPC for tracking piecewise constant reference for constrained linear systems. *Automatica* 2008; **44**:2382–2387.
12. Gu D, Hu HS. Receding horizon tracking control of wheel mobile robots. *IEEE Transactions on Control Systems Technology* 2006; **14**(4):743–749.
13. Faulwasser T, Findeisen R. Constrained output path-following for nonlinear systems using predictive control. *Proceedings of 8th IFAC Symposium on Nonlinear Control Systems (NOLCOS)*, Bologna, Italy, 2010; 753–758.
14. Faulwasser T, Kern B, Findeisen R. Model predictive path-following for constrained nonlinear systems. *Proceedings of the 48th IEEE Conference on Decision and Control Held Jointly with the 2009 28th Chinese Control Conference CDC/CCC 2009*, Shanghai, China, 2009; 8642–8647.

DOI: 10.1002/rnc

15. Chen WH, O'Reilly J, Ballance DJ. On the terminal region of model predictive control for nonlinear systems with input/state constraints. *International Journal of Adaptive Control and Signal Processing* 2003; **17**(3):195–207.
16. Michalska H, Mayne DQ. Robust receding horizon control of constrained nonlinear systems. *IEEE Transactions on Automatic Control* 1993; **38**(11):1623–1633.
17. Egerstedt M, Hu X, Stotsky A. Control of mobile platforms using a virtual vehicle approach. *IEEE Transactions on Automatic Control* 2001; **46**(11):1777–1782.
18. Ghabcheloo R, Pascoal A, Silvestre C, Kaminer I. Coordinated path following control of multiple wheeled robots with directed communication links. *Proceedings of the 44th IEEE Conference on Decision and Control*, Seville, Spain, 2005; 7084–7089.
19. Li X, Zell A. Motion control of an omnidirectional mobile robot. *Proceedings of the 4th International Conference on Informatics in Control, Automation and Robotics*, Angers, France, 2007; 181–193.
20. Maček K, Petrović I, Siegwart R. A control method for stable and smooth path following of mobile robots. *Proceedings of the 2nd European Conference on Mobile Robots (ECMR)*, Ancona, Italy, 2005; 128–133.
21. Soetanto D, Lapierre L, Pascoal A. Adaptive, non-singular path-following control of dynamic wheeled robots. *Proceedings of International Conference on Advanced Robotics*, Coimbra, Portugal, 2003; 1387–1392.
22. Indiveri G, Paulus J, Plöger PG. Motion control of swedish wheeled mobile robots in the presence of actuator saturation. *Proceedings of the 10th Annual Robocup International Symposium*, Breme, Germany, 2006; 35–46.
23. Conceição AS, Moreira AP, Costa Pj. Trajectory tracking for omni-directional mobile robots based on restrictions of the motor's velocities. *Proceedings of the 8th International IFAC Symposium on Robot Control*, Santa Cristina Convent, Italy, 2006; 121–125.

DOI: 10.1002/rnc