Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Constrained Model Predictive Control for Nonholonomic Vehicle Regulation Problem ⋆

Yongjie Zhu and Ümit Özgüner

Department of Electrical and Computer Engineering, The Ohio State University, 2015 Neil Avenue, Columbus, OH 43210,USA (e-mail: zhu.120@osu.edu, ozguner.1@osu.edu).

Abstract: A model predictive control architecture based on discrete time nonlinear car model is derived to solve regulation("parking") problem. Parameters of the proposed controller are chosen by considering terminal state constraints. This setup combined with terminal state penalty in the cost function could assure control stability. The generated trajectory satisfies minimum curvature requirements and actuator saturations of the vehicle are considered in controller design. Obstacle avoidance is realized by considering distance constraints in the open-loop optimization process. Simulation results are given to illustrate the feasibility of the proposed control architecture.

Keywords: model predictive control; regulation problem; autonomous vehicle.

1. INTRODUCTION

Autonomous vehicle regulation("parking") problem is an open and challenging field not only because the nonholonomic constraint makes it impossible to design a time invariant feedback controller Brockett (1983) but also because input and state saturations add constraints on the reference trajectory and control behavior. For solving such kind of problems, Zhu et al. (2007) proposed a geometry solution. In this paper, model predictive control(MPC) for nonlinear discrete time system is introduced to give a suboptimal solution.

Model predictive control or receding horizon control(RHC) is a kind of control algorithm suitable for the case in which pre-computation of a control law is not feasible. In this control strategy, at each sampling instant, the current control law is obtained by solving a finite horizon open-loop optimal control problem. An optimal control sequence is achieved and only the first control in this sequence is used as control input. With the current state as the initial state, this on-line optimal control problem will be solved repeatedly. It is worth noting that MPC can consider input or state constraints directly in the optimal control computation.

There has been historical interest in the topic of applying MPC in the process industry with sufficient slow dynamic. Currently, these applications are extended to faster dynamic systems such as robots and vehicles. In reference Mayne et al. (2000), a thorough survey is given from the theoretical foundations of MPC, the evolution of MPC, the sufficient condition of MPC stability, to the robustness consideration. Various issues, such as tracking, output feedback and adaptive MPC are discussed. Several papers are related to our research topic. In 1990s, MPC strategy was introduced in trajectory generation for nonlinear systems and researchers gave much attention to the stability of MPC algorithm. Recently, the examples of real time applications appeared where MPC is the strategy for dynamic path planning in structured environment with known obstacles and even unstructured environment with unknown moving obstacles. Real-time MPC optimization for obstacle avoidance is realized not only for ground robots Wan et al. (2004) but also for unmanned aerial vehicles Shim et al. (2006). Though using MPC in linear systems is a matured field, nonlinear model based predictive control(NMPC) is still an open topic including how to achieve good stability and robustness performance.

The stability of MPC algorithm has close relation with prediction horizon. Lyapunov arguments can be used to show asymptotic stability of MPC with infinite prediction horizon while finite prediction horizon does not guarantee stability. However, finite prediction horizon is usually adopted in real time systems. Adding end constraints is a common method to solve this problem. Several kinds of end constraints are mentioned including terminal equality constraints Grimm et al. (2003), terminal inequality constraints Michalska et al. (1993), and a terminal cost function. Most recently, the combination of a terminal cost function and terminal state constraints has been explored Scokaert et al. (1998). This method is adopted in tracking control of wheeled mobile robots in reference Gu et al. (2006). Another notable result is contractive MPC for constrained nonlinear systems Kothare et al. (2000) where the contractive constraint renders the closed-loop system exponentially stable in the state feedback case and uniformly asymptotically stable in the output feedback case.

⋆ A project from OSU Center for Automotive Research (CAR) Industrial Consortium

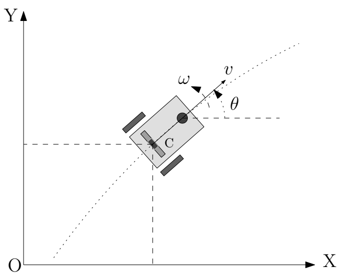

Fig. 1. Coordinate frames for the kinematic model

In this paper, nonlinear model based predictive control architecture is applied to nonholonomic vehicle system to solve constrained regulation problem. Both minimum curvature requirements and actuator saturations are considered in the open-loop optimization process. Based on stability consideration, the parameters in the controller and discrete time step size are properly set up to realize collision-free navigation under the guidance of real-time MPC control module.

The paper is organized as follows. In section 2, the problem is formulated with kinematic error system model. And nonlinear model based predictive control architecture is presented in section 3. In section 4, simulation results of parking maneuver are analyzed, followed by some concluding remarks in section 5.

2. PROBLEM FORMULATION

The kinematic car-like vehicle model (1) is used in this paper. The state is represented by $\chi = [x, y, \theta]' \in \mathcal{C} = \mathbb{R}^2 \times S$, where $\mathcal{C}$ denotes the configuration space including vehicle position and orientation, (x, y, θ) are the Cartesian coordinates of the vehicle and its orientation with respect to an inertial coordinate frame $\{O, X, Y\}$. $\mathrm{u} = [v \; \omega]'$ is the control input, i.e., the linear and the angular velocities, respectively.

$$\begin{cases} \dot{x} = v\cos\theta \\ \dot{y} = v\sin\theta \\ \dot{\theta} = \omega \end{cases} \tag{1}$$

It is assumed that there is a pure rolling contact between the wheels and the ground. Then the vehicle moves without slipping on a plane, that is to say, the vehicle is subject to a nonholonomic constraint (2).

$$\dot{x}\sin\theta - \dot{y}\cos\theta = 0 \tag{2}$$

For this model, the closed-loop control relates to the determination of steering inputs assuring the states of the system asymptotically converge to the origin (parking target). While according to the well known work of Brockett (1983), Cartesian state space representations of car model is among a class of systems which are not controllable by a time invariant feedback control law. In addition, the input v and ω have saturation resulted from physical limitations of the actuators. The minimum turning radius of a vehicle determined the relation between v and ω, that is,

$$|\frac{v}{\omega}| \geq R_{min}. \tag{3}$$

Since autonomous vehicle is usually driven by control signals from computer, it is necessary to discretize the system model. Considering a step size T, using Euler's approximation, the following model (4) can be obtained for the time instant k

$$\begin{cases} x(k+1) = x(k) + v(k)cos\theta(k)T \\ y(k+1) = y(k) + v(k)sin\theta(k)T \\ \theta(k+1) = \theta(k) + \omega(k)T \end{cases} \tag{4}$$

where $\chi_k = [x(k), y(k), \theta(k)]'$. Or, denote it as (5).

$$\chi_{k+1} = f(\chi_k, \mathrm{u}_k) \tag{5}$$

Generally, the MPC control problem to be solved can be described as achieving a control sequence u_k so that the current state χ_k will converge to a desired reference state χ_r when $k \to \infty$. The NMPC algorithm will solve an optimal problem in (6).

$$\begin{aligned} \min_{\mathrm{u}^M} J(k, \mathrm{u}^M) &= g(\chi_f) + \sum_{j=0}^{P-1} \|\chi_{k+j}\|_Q^q \\ &+ \sum_{i=0}^{M-1} (\|\mathrm{u}_{k+i}\|_R^q + \|\Delta \mathrm{u}_{k+i}\|_S^q) \end{aligned} \tag{6}$$

Subject to:

$$\begin{aligned} \chi_{k+j} &= f(\chi_{k+j-1}, \mathrm{u}_{k+j-1}) && for\ j = 0, \cdots, P \\ \underline{\mathrm{u}} \leq \mathrm{u}_{k+i} &\leq \overline{\mathrm{u}} && for\ i = 0, \cdots, M-1 \\ \Delta\underline{\mathrm{u}} \leq \Delta\mathrm{u}_{k+i} &\leq \Delta\overline{\mathrm{u}} && for\ i = 0, \cdots, M-1. \end{aligned} \tag{7}$$

where $g(\chi_f) = \chi_f' Q_f \chi_f$ is the terminal penalty term. Q_f, Q, R, and S are positive definite weight matrices. M is the control length and P is the predictive length. $M \leq P$. $\mathrm{u}_{k+i} = 0$ when $i = M, M+1, \cdots, P$. $\mathrm{u}^M = (\mathrm{u}_k, \cdots, \mathrm{u}_{k+M-1})$ is the control sequence corresponding to each predictive length. The first m inputs $(\mathrm{u}_k, \mathrm{u}_{k+1}, \cdots, \mathrm{u}_{k+m-1})$ will be applied to the system at each time instant.

Without loss of generality, we can consider autonomous parking as regulation problem in which the desired operating point is the origin. Error state system is defined in (8)

$$\begin{cases} x_e(k+1) = x_e(k) - v(k)cos\theta_e(k)T \\ y_e(k+1) = y_e(k) - v(k)sin\theta_e(k)T \\ \theta_e(k+1) = \theta_e(k) - \omega(k)T \end{cases} \tag{8}$$

where $\chi_e = \chi_r - \chi = -\chi$.

Now the regulation("parking") problem in our case can be reformulated as finding a control sequence u_k so that the current state $\chi_e(k)$ will converge to the origin when $k \to \infty$. By choosing $M = P$, the value function we used in this paper is as follows (9).

$$\begin{aligned} J(k, \mathrm{u}^M) &= \chi_e(k+M)'\chi_e(k+M) + \\ &\sum_{i=0}^{M-1} [\chi_e(k+i)'Q\chi_e(k+i) + \mathrm{u}(k+i)'R\mathrm{u}(k+i)] \end{aligned} \tag{9}$$

Subject to:

$$\begin{aligned} \chi_e(k+i) &= f(\chi_e(k+i-1), \mathrm{u}(k+i-1)) \\ & \quad for\ i = 0, \cdots, P \\ \underline{\mathrm{u}} \leq \mathrm{u}(k+i) &\leq \overline{\mathrm{u}} \\ & \quad for\ i = 0, \cdots, M-1 \\ \rho(k) &= \|[x(k+i) \;\; y(k+i)]' - [x_o^h \;\; y_o^h]'\| \geq D \\ & \quad for\ i = 0, \cdots, P \\ & \quad for\ h = 1, \cdots, H \\ \chi_e(k+M) &\in \Omega \end{aligned} \tag{10}$$

where $\rho(k)$ is the distance from vehicle to the obstacle h measured by sensor set up on the vehicle in the real time system. In simulation, it is simulated by the Euclidean distance between current vehicle position and the nearby obstacle. D is the safe distance defined for obstacle avoidance behavior. By adding this constraint, the real time system can realize parking maneuver with obstacle avoidance. Ω is the terminal state region.

The fact that one can solve the closed-loop control problem through a sequence of open-loop optimisations was recognised very clearly in the development of optimal control theory and one can view the NMPC solution as a way of turning an intractable closed-loop computation into a sequence of tractable open-loop calculations Kouvaritakis et al. (2001). In autonomous vehicle field, prediction horizon in MPC strategy is meaningful just as look ahead distance. Usually, in driving process, the driver will consider the road condition several hundred yards ahead(his vision) so that he can predict the potential dangers. This vision is moving along with the vehicle and could not be too small, i.e., the prediction length should be tuned to a fitful value.

3. NONLINEAR MODEL BASED PREDICTIVE CONTROL ARCHITECTURE

3.1 Control Strategy

It is well known that MPC with infinite receding horizon can guarantee stability for nonlinear systems Keerthi et al. (1988) though it is not feasible in practice due to the computation complexity. The proposed MPC in this section is using finite predictive horizon considering the speed requirement of real time systems. To clearly describe the predictive control process, the similar notation as that appeared in Kothare et al. (2000) is adopted here. The evolution of the system will be over time index of the form $t_k^j := t_0 + (j + km)T$, with j varying in the interval $j = 0, \ldots, M-1$, while k is kept constant at $k = 0, 1, 2 \ldots$. Here, we choose $t_0 = 0$ and $P = M$. Then the time index will be as follows.

$$\{\ldots, t_k^0, t_k^1, \ldots, t_k^m = t_{k+1}^0, t_{k+1}^1, \ldots, t_{k+1}^m = t_{k+2}^0, \ldots\}$$
$$\forall k \in \mathbb{Z}_+.$$

As shown in Fig. 2, there are several sets of time duration in which the corresponding optimal problem will be solved. Using the iteration in Fig. 2 as an example, we can get an optimal control sequence $u_{k+1}^0, u_{k+1}^1, \ldots, u_{k+1}^{M-1}$. The first $M - m$ control inputs are called local optimal control denoted by u_{op} and the rest m control inputs are called terminal control denoted by u_T. The corresponding time index $t_{k+1}^0, \ldots, t_{k+1}^{M-m-1}$ and $t_{k+1}^{M-m}, \ldots, t_{k+1}^{M-1}$ are called local optimal control duration and terminal control duration respectively. Only the first m control inputs, $u_{k+1}^0, u_{k+1}^1, \ldots, u_{k+1}^{m-1}$ will be the future control action applied to vehicle steering system.

The control architecture is shown in Fig. 3 where the initial state is current vehicle position and the set point is goal parking position. Future control action is decided by MPC module and this part is motivated by the continuous MPC for tracking control in Gu et al. (2006). Discrete time MPC is used here for considering the real time application and therefore, the main focus will be on how

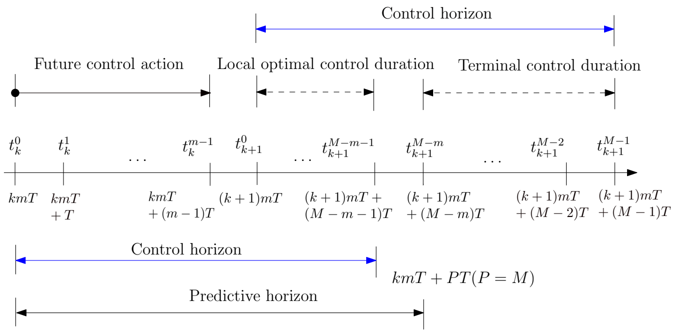

Fig. 2. Time index for optimal problem

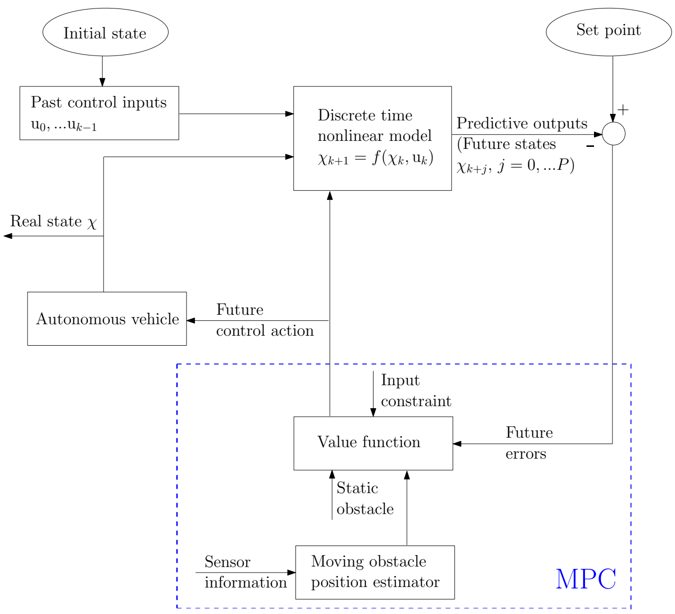

Fig. 3. Control architecture

to choose time step size T and other parameters as well as terminal control inputs so that nonlinear MPC parking controller is stable. Besides, obstacle avoidance is realized by adding constraints based on the real time sensor data and the generated trajectory has the minimum curvature according to the requirement of vehicle. The detail control process is as follows.

Step 1. Get the current error state $\chi_e(t_k^0)$.

Step 2. Solve the following optimal control problem on time index $t_k^0, t_k^1, \ldots, t_k^{M-1}$.

$$\min_{\mathrm{u}^M} J(t_k^j, \mathrm{u}^M) =$$
$$\chi_e(t_k^M)'\chi_e(t_k^M) + \sum_{j=0}^{M-1} [\chi_e(t_k^j)' Q \chi_e(t_k^j) + \mathrm{u}(t_k^j)' R \mathrm{u}(t_k^j)] \tag{11}$$

Subject to:

$$\begin{aligned}
\chi_e(t_k^{j+1}) &= f(\chi_e(t_k^j), \mathrm{u}(t_k^j)) \\
& \quad for\ j = 0, \cdots, M-1 \\
\underline{\mathrm{u}} \leq \mathrm{u}(t_k^j) &\leq \bar{\mathrm{u}} \\
\rho(t_k^{j+1}) &= \|[x(t_k^{j+1})\ \ y(t_k^{j+1})]' - [x_o^h\ \ y_o^h]'\| \geq D \\
& \quad for\ h = 1, \cdots, H \\
\chi_e(t_k^M) &\in \Omega.
\end{aligned} \tag{12}$$

Get the optimal control sequence $\hat{\mathrm{u}} = (\mathrm{u}_k^0, \mathrm{u}_k^1, \ldots, \mathrm{u}_k^{M-1})$ and apply $\mathrm{u}_k^0, \mathrm{u}_k^1, \ldots, \mathrm{u}_k^{m-1}$ to the error state system.

Step 3. Use $\chi_e(t^0_{k+1})$ as initial state and $\mathrm{u}^0_{k+1}, \mathrm{u}^1_{k+1}, \ldots, \mathrm{u}^{M-m-1}_{k+1}, 0, \ldots, 0$ as initial solution of the optimal problem and solve (11) and (12) again, we get a new optimal control sequence among it the first $M-m$ local optimal control inputs and m terminal control inputs together as (13) will be the new initial solution for the optimal control problem.

$$\tilde{u} = (\overbrace{u^0_{k+1}, \ldots, u^{M-m-1}_{k+1}}^{\text{local optimal control}}, \overbrace{u^{M-m}_{k+1}, \ldots, u^{M-1}_{k+1}}^{\text{terminal control}}) \tag{13}$$

where

$$\begin{aligned} \mathrm{u}^{\mathrm{j}}_{\mathrm{k+1}} &= [v^j_{k+1} \quad \omega^j_{k+1}]' \\ &= [\eta\sqrt{(x_e(t^j_{k+1}))^2 + (y_e(t^j_{k+1}))^2} \quad \xi\theta_e(t^j_{k+1})]' \\ &\quad for\ j = M-m, \ldots, M-1. \end{aligned} \tag{14}$$

Step 4. Solve (11) and (12) once more. The first m control inputs among the solution sequence will be applied to error state system.
Step 5. $k+1 \rightarrow k$ and continue this procedure till the parking error is small enough.

As shown in (12), actuator saturations and minimum curvature requirements are considered as constraints in the optimal problem. Besides, the constraint of keeping safe Euclidean distance to each obstacle can assure the generated trajectory would be collision-free. Then, the big concern is how to choose time step T and output feedback gain η and ξ and other parameters so that the NMPC algorithm is stable.

3.2 Stability Proof and Parameter Setup

Define $V(t^j_k)$ and $\hat{V}(t^j_k)$ as the value function for MPC and optimal control respectively. Denote $L = \chi_e' Q \chi_e + \mathrm{u}' R \mathrm{u}$. χ_e is the real state by applying control solutions from MPC to the error system and $\hat{\chi}_e$ is the virtual error state by applying control solutions from the optimal problem in each iteration. $\tilde{\chi}_e$ is the error state by applying the control sequence $\tilde{\mathrm{u}}$. Considering the m control inputs $\mathrm{u}^0_k, \mathrm{u}^1_k, \ldots, \mathrm{u}^{m-1}_k$ are applied to the error state system at the beginning of each iteration(Step 2), first we try to get decreasing series $V(t^0_k), \quad k = 0, 1, 2, \ldots$ at the end of each applied future control action. Then, at each time instant t^0_k, we have

$$\hat{V}(t^0_{k+1}, \tilde{\chi}_e(t^0_{k+1})) \geq \hat{V}(t^0_{k+1}, \hat{\chi}_e(t^0_{k+1})) = V(t^0_{k+1}, \chi_e(t^0_{k+1}))$$

And

$$\begin{aligned} &\hat{V}(t^0_{k+1}, \tilde{\chi}_e(t^0_{k+1})) - V(t^0_k, \chi_e(t^0_k)) = \tilde{\chi}_e(t^M_{k+1})'\tilde{\chi}_e(t^M_{k+1}) \\ &-\hat{\chi}_e(t^M_k)'\hat{\chi}_e(t^M_k) + \sum_{j=0}^{M-1} \tilde{L}(t^0_{k+1} + jT) - \sum_{j=0}^{M-1} \hat{L}(t^0_k + jT) \end{aligned}$$

Since we know $\hat{\chi}_e(t^M_k) = \tilde{\chi}_e(t^M_k)$ from Step 3, the following relation can be achieved.

$$\begin{aligned} &\hat{V}(t^0_{k+1}, \tilde{\chi}_e(t^0_{k+1})) - V(t^0_k, \chi_e(t^0_k)) = \tilde{\chi}_e(t^M_{k+1})'\tilde{\chi}_e(t^M_{k+1}) \\ &-\tilde{\chi}_e(t^M_k)'\tilde{\chi}_e(t^M_k) + \sum_{j=0}^{m-1} \tilde{L}(t^M_k + jT) - \sum_{j=0}^{m-1} \hat{L}(t^0_k + jT) \\ &\leq \tilde{\chi}_e(t^M_{k+1})'\tilde{\chi}_e(t^M_{k+1}) - \tilde{\chi}_e(t^M_k)'\tilde{\chi}_e(t^M_k) + \sum_{j=0}^{m-1} \tilde{L}(t^M_k + jT) \\ &= [\tilde{\chi}_e(t^M_k + mT)'\tilde{\chi}_e(t^M_k + mT) - \tilde{\chi}_e(t^M_k + mT - T)' \\ &\cdot\tilde{\chi}_e(t^M_k + mT - T)] + [\tilde{\chi}_e(t^M_k + mT - T)'\tilde{\chi}_e(t^M_k + mT - T) \\ &\quad -\tilde{\chi}_e(t^M_k + mT - 2T)'\tilde{\chi}_e(t^M_k + mT - 2T)] + \ldots \\ &+[\tilde{\chi}_e(t^M_k + T)'\tilde{\chi}_e(t^M_k + T) - \tilde{\chi}_e(t^M_k)'\tilde{\chi}_e(t^M_k)] + \sum_{j=0}^{m-1} \tilde{L}(t^M_k + jT) \\ &= \sum_{j=0}^{m-1} [\tilde{\chi}_e(t^M_k + (j+1)T)'\tilde{\chi}_e(t^M_k + (j+1)T) - \\ &\quad \tilde{\chi}_e(t^M_k + jT)'\tilde{\chi}_e(t^M_k + jT) + \tilde{L}(t^M_k + jT)] \end{aligned}$$

Substitute (8) in the above equation and omit the time index. One item among the summation items can be analyzed as follows.

$$\begin{aligned} OneItem = &- \overbrace{[2\tilde{x}_e v \cos\tilde{\theta}_e T + 2\tilde{y}_e v \sin\tilde{\theta}_e T - \tilde{v}^2 T^2 - q_{11}\tilde{x}_e^2 - q_{22}\tilde{y}_e^2 - r_{11}\tilde{v}^2]}^{\text{Term1}} \\ &- \overbrace{[2\tilde{\theta}_e\tilde{\omega}T - \tilde{\omega}^2 T^2 - q_{33}\tilde{\theta}_e^2 - r_{22}\tilde{\omega}^2]}^{\text{Term2}} \end{aligned}$$

It worth noting that the related control inputs here are corresponding to the terminal control duration. When $\tilde{v} = \eta\sqrt{\tilde{x}_e^2 + \tilde{y}_e^2}$ and $\tilde{\omega} = \xi\tilde{\theta}_e$, we have

$$\begin{aligned} Term1 = [&2\eta T \frac{\tilde{x}_e}{\sqrt{\tilde{x}_e^2 + \tilde{y}_e^2}} \cos\tilde{\theta}_e + 2\eta T \frac{\tilde{y}_e}{\sqrt{\tilde{x}_e^2 + \tilde{y}_e^2}} \sin\tilde{\theta}_e \\ &-(\eta^2 T^2 + q_{11} + r_{11}\eta^2)](\tilde{x}_e^2 + \tilde{y}_e^2) \end{aligned}$$

and (15) will be one constraint for the terminal control so that $Term1 \geq 0$.

$$\begin{aligned} \frac{\tilde{x}_e}{\sqrt{\tilde{x}_e^2 + \tilde{y}_e^2}} \cos\tilde{\theta}_e + \frac{\tilde{y}_e}{\sqrt{\tilde{x}_e^2 + \tilde{y}_e^2}} \sin\tilde{\theta}_e \\ \geq \frac{(\eta^2 T^2 + q_{11} + r_{11}\eta^2)}{2\eta T} \end{aligned} \tag{15}$$

$$Term2 = [2\xi T - (\xi^2 T^2 + q_{33} + r_{22}\xi^2)]\tilde{\theta}_e^2$$

and (16) will be another constraint for the terminal control so that $Term2 \geq 0$.

$$2\xi T \geq \xi^2 T^2 + q_{33} + r_{22}\xi^2 \tag{16}$$

Therefore, by choosing terminal control inputs, (17) can be achieved.

$$OneItem = Term1 + Term2 \leq 0 \tag{17}$$

Since each item of the summation $\leq$0, we have

$$\hat{V}(t^0_{k+1}, \tilde{\chi}_e(t^0_{k+1})) - V(t^0_k, \chi_e(t^0_k)) \leq 0 \tag{18}$$

Therefore,

$$V(t^0_{k+1}, \chi_e(t^0_{k+1})) - V(t^0_k, \chi_e(t^0_k)) \leq 0 \tag{19}$$

Now consider series $V(t^j_k)$ where $k = 0, 1, 2, \ldots$. while j is fixed and $j \in \{1, 2, \ldots, m-1\}$.

$$\begin{aligned} V(t^j_k, \chi_e(t^j_k)) = \hat{V}(t^j_k, \hat{\chi}_e(t^j_k)) = \hat{\chi}_e(t^{j+M}_k)'\hat{\chi}_e(t^{j+M}_k) \\ + \sum_{i=0}^{M-1} \hat{L}_(t^j_k + iT) \end{aligned}$$

$$\begin{aligned} V(t^0_k, \chi_e(t^0_k)) = \hat{V}(t^0_k, \hat{\chi}_e(t^0_k)) = \hat{\chi}_e(t^M_k)'\hat{\chi}_e(t^M_k) \\ + \sum_{i=0}^{M-1} \hat{L}_(t^0_k + iT) \end{aligned}$$

Parameters	$T(s)$	q_{11}	q_{22}	q_{33}	r_{11}	r_{22}	η	ξ
Value	0.2	0.1	0.1	0.1	0.1	0.1	1	1

Table 1. Parameter values

$$V(t_k^j,\chi_e(t_k^j)) - V(t_k^0,\chi_e(t_k^0)) = \hat{\chi}_e(t_k^{j+M})'\hat{\chi}_e(t_k^{j+M})$$
$$-\hat{\chi}_e(t_k^M)'\hat{\chi}_e(t_k^M) + \sum_{i=0}^{j-1}\hat{L}(t_k^M+iT) - \sum_{i=0}^{j-1}\hat{L}(t_k^0+iT)$$
$$\leq \hat{\chi}_e(t_k^{j+M})'\hat{\chi}_e(t_k^{j+M}) - \hat{\chi}_e(t_k^M)'\hat{\chi}_e(t_k^M) + \sum_{i=0}^{j-1}\hat{L}(t_k^M+iT)$$
$$= [\hat{\chi}_e(t_k^M+jT)'\hat{\chi}_e(t_k^M+jT) - \hat{\chi}_e(t_k^M+(j-1)T)'$$
$$\cdot\hat{\chi}_e(t_k^M+(j-1)T)] + [\hat{\chi}_e(t_k^M+(j-1)T)'\hat{\chi}_e(t_k^M+(j-1)T)$$
$$-\hat{\chi}_e(t_k^M+(j-2)T)'\hat{\chi}_e(t_k^M+(j-2)T)] + \dots$$
$$+[\hat{\chi}_e(t_k^M+T)'\hat{\chi}_e(t_k^M+T) - \hat{\chi}_e(t_k^M)'\hat{\chi}_e(t_k^M)] + \sum_{i=0}^{j-1}\hat{L}(t_k^M+iT)$$
$$= \sum_{i=0}^{j-1}[\hat{\chi}_e(t_k^M+(i+1)T)'\hat{\chi}_e(t_k^M+(i+1)T) -$$
$$\hat{\chi}_e(t_k^M+iT)'\hat{\chi}_e(t_k^M+iT) + \hat{L}(t_k^M+iT)]$$

Again, the related control inputs here are corresponding to the terminal control duration. The similar process as before is followed to achieve terminal control inputs. Obviously,the constraints (15) and (16) can also assure (20) is correct.

$$V(t_k^j,\chi_e(t_k^j)) - V(t_k^0,\chi_e(t_k^0)) \leq 0 \tag{20}$$

From (20), we know for some fixed $j \in \{1,2,\dots,m-1\}$, and $k = 0,1,2\dots$

$$0 \leq V(t_k^j,\chi_e(t_k^j)) \leq V(t_k^0,\chi_e(t_k^0)). \tag{21}$$

From (19), we have

$$\lim_{k\longrightarrow\infty} V(t_k^0,\chi_e(t_k^0)) = 0. \tag{22}$$

Then, according to the squeeze theorem, we have

$$\lim_{k\longrightarrow\infty} V(t_k^j,\chi_e(t_k^j)) = 0. \tag{23}$$

According to Lyapunov theorem for discrete time systems, in our case V is the lyapunov function, the nonlinear discrete error system can be stabilized by the proposed controller design. To satisfy (15) and (16), the parameters are chosen as in Table 1.

4. SIMULATION RESULTS

In this section, simulation results are given to illustrate the performance of NMPC algorithm for parking maneuver. In Fig. 4, corresponding to 8 different initial positions as follows,
$(A)[20,-50,\pi/4]$ $(B)[45,-50,\pi/4]$ $(C)[45,0,-3\pi/2]$
$(D)[45,0,-3\pi/4]$ $(E)[45,0,3\pi/4]$ $(F)[45,0,3\pi/2]$
$(G)[45,50,-\pi/4]$ $(H)[20,50,-\pi/4]$,
the parking trajectories are provided by control architecture proposed in this paper. For all these initial setup, the vehicle could arrive the final parking configuration, (0, 0, π or -π), via a reasonable path. Compared with them, a special case is given in Fig. 5. Since the initial position is pretty close to the target point, the car is adjusted

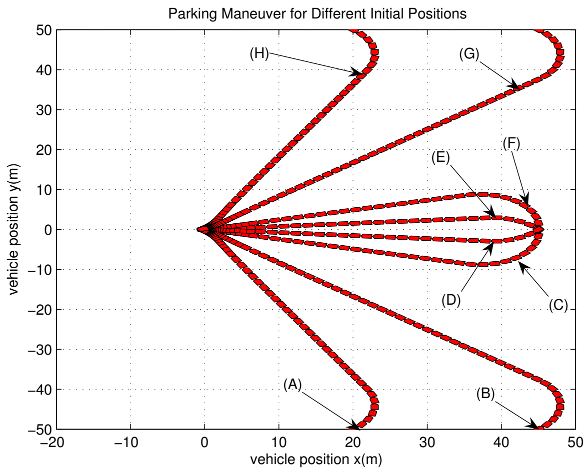

Fig. 4. Parking maneuver from different initial positions

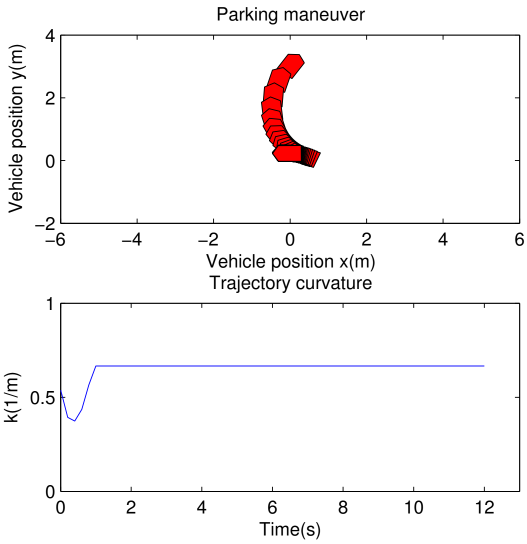

Fig. 5. Parking maneuver and trajectory curvature

to a comfortable configuration by reverse maneuver, and then driven to the set point. The algorithm provides a parking maneuver just like what human driver will do in this case. We set the bound for linear velocity v as between -5 and $5m/s$ and angular velocity ω as between -1.5 and 1.5 rad/s. As shown in Fig. 6, the simulation results for both of them are within the required range. Besides, the curvature of the trajectory is also bounded since a vehicle has minimum turning radius(1.5m in our case). By adding the proposed constraints in our algorithm, the curvature of the trajectory is smooth enough (Fig. 5).

The simulation results in Fig. 7 show the parking maneuver with obstacle avoidance. By setting a safe distance D, as we mentioned before, whenever the sensor detected an obstacle entering this range, the predictive control strategy will modify the path. The velocity v and ω for this case is shown in Fig. 8. They are within the boundary and the result path is smooth enough for our vehicle.

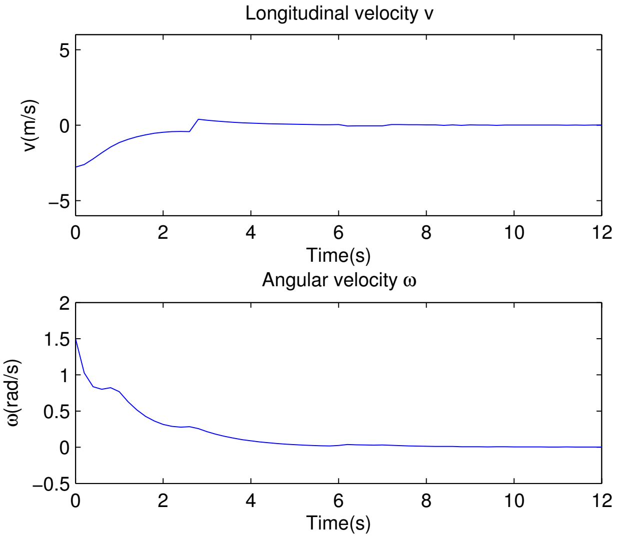

Fig. 6. Control variable v and ω

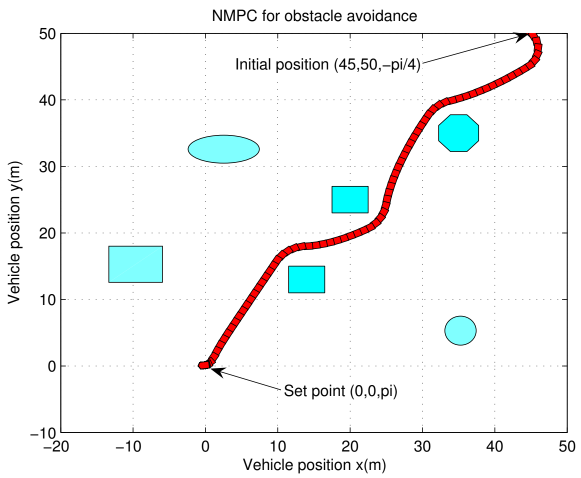

Fig. 7. Parking maneuver with obstacle avoidance

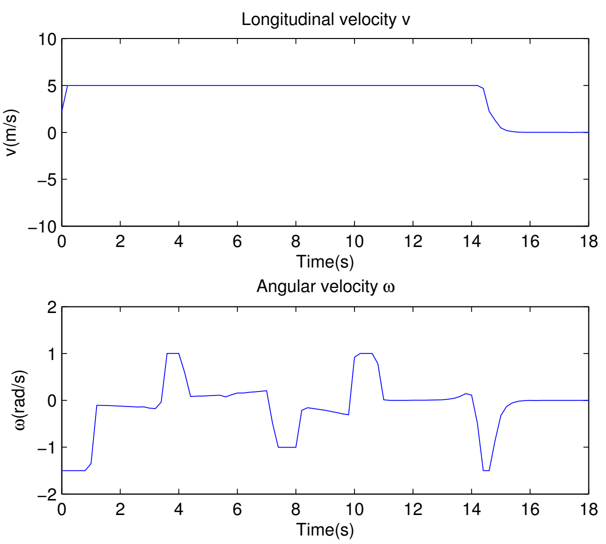

Fig. 8. Control variable v and ω

5. CONCLUSION

In this paper, a model predictive control strategy is applied to nonholonomic vehicle regulation problem. Based on nonlinear discrete time system model, a stable controller is designed in terminal region for each predictive length. Parameters of the controller are chosen by considering the control stability requirement. According to the terminal region constraints, the corresponding requirement on discrete step size is given. The input saturation and minimum turning radius are considered as constraints in the optimal problem so that the given trajectory is feasible for a real car.

REFERENCES

D. Q. Mayne, J. B. Rawlings, C. V. Rao and P. O. M. Scokaert. Constrained model predictive control: stability and optimality. *Automatica*, 36:6, pages 789–814, June 2000.

J. Wan, C. G. Quintero, N. Luo and J. Vehi. Predictive motion control of a mirosot mobile robot. *Proceedings of World Automation Congress*, volume 15, pages 325–330, June 2004.

D. H. Shim, H. Chung and S. S. Sastry. Conflict-free navigation in unknown urban environments. *IEEE Robotics and Automation Magazine*, pages 27–33, September 2006.

G. Grimm, M. J. Messina, S. E. Tuna and A. R. Teel. Nominally robust model predictive control with state constraints. *Proceedings of the 42nd IEEE Conference on Decision and Control*, 2:9–12, pages 1413–1418, December 2003.

H. Michalska and D. Q. Mayne. Robust receding horizon control of constrained nonlinear systems. *IEEE Transactions on Automatic Control*, 38:11, pages 1623–1633, November 1993.

P. O. M. Scokaert and J. B. Rawlings. Constrained linear quadratic regulation. *IEEE Transactions on Automatic Control*, 43:8, pages 1163–1169, 1998.

D. Gu and H. Hu. Receding horizon tracking control of wheeled mobile robots. *IEEE Transactions on Control Systems Technology*, 14:4, pages 743–749, July 2006.

S. L. O. Kothoare and M. Morari. Contractive model predictive control for constrained nonlinear systems. *IEEE Transactions on Automatic Control*, 45:6, pages 1053–1070, June 2000.

R. W. Brockett. Asymptotic stability and feedback stabilisation. *Differential Geometry Control Theory*, pages 181–191, 1983.

B. Kouvaritakis and M. Cannon. Nonlinear predictive control theory and practice. In A.F. Round, editor, *IEE Control Engineering Series*, volume 61. The Institution of Electrical Engineers, United Kingdom, 2001.

S. S. Keerthi and E. G. Gilbert. Optimal infinite-horizon feedback laws for a general class of constrained discrete-time systems: stability and moving-horizon approximations. *Journal of Optimal Theory Application*, volume 57, pages 265–293, 1988.

Y. Zhu, Y. Zheng and Ümit Özgüner. Waypoint selection in constrained domains (for cooperative systems). *Advances in cooperative control and optimization (Lecture Notes in Control and Information Sciences)*, Springer, volume 369, pages 191–202, 2007.