Active Front Steering Using Stable Model Predictive Control Approach via LMI

Shekoofeh Jafari Fesharaki, Heidar Ali Talebi

Electrical Engineering Department of Amirkabir University of Technology
Tehran, Iran (Tel: +989132106369; E-mail: shekoofeh.jafari@aut.ac.ir; alit@aut.ac.ir).

Abstract: In this paper, a model predictive control (MPC) with guaranteed stability is used for controlling an active front steering system in an autonomous vehicle. At each control interval, it is assumed that by using information obtained from the sensors on the vehicle, a trajectory is known over a finite horizon, and then it is the predictive controller duty to compute the front steering angle as the control input until the vehicle follows the trajectory. In addition, for reducing computational load of implementing the controller we convert the MPC optimization problem into linear matrix inequality (LMI) form so the problem can be solved more efficiently. The effectiveness of the proposed MPC formulation is demonstrated by simulation. A comparison is made between the time consumed for solving LMI form of the problem and another approach of solving.

Keywords: Vehicle dynamics, Quasi-infinite horizon, Model Predictive Control, Linear Matrix Inequality, Stability

1. INTRODUCTION

Using Computers, electronics and controls in automotive industries have increased recently, especially in the active safety mode. Passive safety is involved as an accident or crash happened while active safety helps to avoid crashing and is used to improve controllability and stability of a vehicle especially in dangerous situations (*Kolmanovsky et al.,* 2009; Falcon *et al.,* 2007).

In this paper we assume that a trajectory planner is available. The aim is to use the active front steering in order to control the yaw vehicle dynamics. Considering the physical limitation of vehicle, the controller computes the front steering angle so that the car will follow the predetermined path with the lowest error and the side slip angle have to be zero or near zero.

Model Predictive Control (MPC) is a kind of control algorithm that predicts the future behavior of the plant in an interval which is called the prediction horizon using the explicit system model for its prediction. In each time step MPC tries to compute a sequence of control inputs in a control horizon such that the future behavior of the system will be optimized and the current state is used as initial state in each control interval (Qina and Badgwel, 2003; Wang, 2009).

The main feature that makes MPC a popular controller in both theoretical and practical systems is its ability to cope with hard constraints. Moreover, the works of (Wang, 2009; Haeri *et al*., 2003; Falcon, 2008; Scattolini, 2009) show that the MPC can be used to control a vast majority of systems like MIMO systems, non-minimum phase systems and systems contain delays.

The general form of MPC is an optimization problem with a finite horizon cost function and the system dynamics and limitations on the inputs and states of the system as constraints (Bitmead *et al.,* 1990).

There are two main issues in implementing MPC in practical systems. The first is that the general form of MPC does not guarantee the stability of the obtained closed loop system (Bitmead *et al.,* 1990) and the other is the computational burden of such controllers which limits the applications for systems with slow dynamics or large sampling time (Wang and Boyd, 2010).

If we can convert the MPC problem into some kind for which there is an appropriate solution and solver then this problem can partly be overcome.

In order to overcome the first issue (stability problem) several solutions have been proposed so far. In order to use the cost function as a Lyapunov function for proving the stability of the closed loop system, (Keerthi and Gilbert, 1998) propose to increase the control and prediction horizons to infinity.

The same authors proposed another solution which is using a terminal state constraint to the problem in order to force the states at the end of the finite prediction horizon to be zero hence, for a non-disturbed system the states will stay at the origin due to zero control action at the end of the horizon. But these two methods have a large computational load for practical implementing.

It is shown in (Michalska and Mayne, 1998) that the dual mode control is also a method to design a stable MPC. It uses both MPC and linear state feedback controller in order to force the states to become zero at the end of the prediction horizon.

In (Esfanjani and Nikravesh, 2001), a terminal cost function and a terminal region are used to achieve asymptotic stability for some nonlinear constrained systems with discrete and distributed delays.

In this paper, we will use one of the best methods to design an MPC controller with guaranteed stability which has been introduced by (Chen and Allgower, 1998). This method is called quasi-infinite horizon nonlinear MPC. The objective function which should be minimized on-line is a finite horizon cost plus a terminal cost and constraints are system dynamics, limitations on inputs and states and an additional terminal state inequality constraint.

One of the main features of this method is that it has less computational load and is more general than the other methods of designing stable MPC. More details will be given in future sections.

For reducing computational load of implementing MPC in practical systems, several approaches are proposed so far. We will review some for linear MPC.

One of the most successful methods for implementing fast MPC is to create a lookup table in which the elements are functions of the initial condition of the under controlled system (Bemporad *et al.,* 2002; Tondel *et al.,* 2001). But the drawback is that by increasing parameters of MPC such as the horizons and the dimensions of inputs and states the elements in the table will increase exponentially.

Another solution for implementing linear MPC in a fast manner is using Laguerre networks proposed in (Wang, 2009) and used in (Jafari *et al.,* 2012), which are a set of discrete orthonormal basis functions. By using these functions the number of parameters that the controller has to deal in an interval will reduce significantly.

In order to overcome the implementing load of predictive controller we will translate the optimization problem of the designed MPC into Linear Matrix Inequality (LMI) form.

Since various computationally difficult optimization problems can be effectively approximated by LMI problems this translation may help to solve the optimization problems more efficiently. So we can reduce the time needed for MPC optimization problem in fast processes. This helps to implement this kind of controllers for higher speed of the vehicle.

In this paper considering the physical limits on a vehicle, we will design a model predictive controller with guaranteed stability for an active front steering in an autonomous vehicle and it will be formulated in LMI form. The control input is front steering angle in order to force the vehicle to follow a desired trajectory as close as possible.

In continuation, the paper is structured as follows; Section 2 describes the vehicle dynamics which will be used in the controller design. Section 3 briefs stable MPC formulation using quasi-infinite horizon. Section 4 describes how to convert the MPC problem into LMI form. The simulation results of the applying of the proposed controller of the vehicle are presented in Section 5. Finally, paper is concluded in Section 6.

2. VEHICLE MODELLING

This section describes the vehicle and the tire model used for simulations and control design (Rajamani, 2005).

In this paper the longitudinal speed of the vehicle, V considered to be constant. Vehicle states are yaw angle ψ , the vehicle yaw rate $\dot{\psi}$ and the position of its center of gravity [x, y] as illustrated in Figure 1. We show the side slip angle of the vehicle by β and m is the total mass of the vehicle.

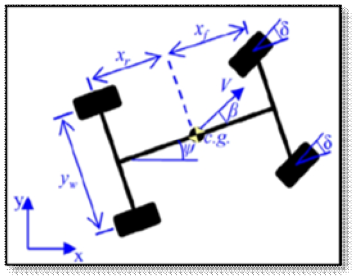

Fig. 1. Vehicle slip model used to MPC controller.

The lateral tire force, Fy is computed as the product of each tire's cornering stiffness (C) and sideslip angle (α) as shown in Figure 2.

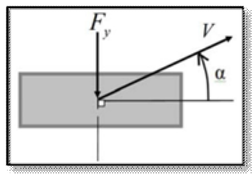

Fig. 2. The linearized tire model used in the equations of motion.

The nonlinear equations of motion for the slip model include:

$$
\begin{aligned}
&\dot{x} = V\cos(\psi+\beta) \\
&\dot{y} = V\sin(\psi+\beta) \\
&\sum F_y = m[\dot{V}\sin(\psi+\beta) + V(\dot{\psi}+\dot{\beta})\cos(\psi+\beta)] \\
&\sum M_z = I_{zz}\ddot{\psi}
\end{aligned}
\tag{1}
$$

By linearizing the above model about a constant speed and assuming small slip angles, the equations become:

$$
\begin{aligned}
&\dot{x} = V \\
&\dot{y} = V(\psi+\beta) \\
&\dot{\beta} = \frac{-(C_r+C_f)}{mV}\beta + (\frac{(C_r x_r - C_f x_f)}{mV^2} - 1)\dot{\psi} + \frac{C_f}{mV}\delta_f \\
&\ddot{\psi} = \frac{(C_r x_r - C_f x_f)}{I_z}\beta - \frac{(C_r x_r + C_f x_f)}{I_z V}\dot{\psi} + \frac{C_f x_f}{I_z}\delta_f
\end{aligned}
\tag{2}
$$

where x_f ,x_r are the distances of front and rear wheels from center of gravity(they are car geometry parameters).

This linear model will be used in designing the MPC controller. As it is mentioned above the state variable is considered as $x = \begin{bmatrix} x & y & \beta & \psi & \dot{\psi} \end{bmatrix}^T$. So the linear state space equation will be described as:

$$\dot{x} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & V & V & 0 \\ 0 & 0 & -(C_r + C_f)/mV & 0 & (C_r X_r - C_f X_f)/mV^2 - 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & (C_r X_r - C_f X_f)/I_{zz} & 0 & -(C_r X^2_r + C_f X^2_f)/I_{zz} V \end{bmatrix}}_{A} x$$

$$+ \underbrace{\begin{bmatrix} 0 \\ 0 \\ C_f / mV \\ 0 \\ C_f X_f / Izz \end{bmatrix}}_{B} u \ , \tag{3}$$

$$y = \underbrace{\begin{bmatrix} 0\,1\,0\,0\,0 \\ 0\,0\,1\,0\,0 \end{bmatrix}}_{C} x$$

The input to this system is steering angle, δ_f which will be controlled such that the position y will be appropriate correspond to the desired trajectory and the side slip angle maintain in zero.

3. PROBLEM FORMULATION

In this section the mathematical expressions and derivations of the MPC algorithm are given.

3.1 Model Predictive Control with guaranteed stability

Now we will use the Quasi-Infinite Horizon MPC scheme to design the controller. It should be pointed out again that the idea of the presented stable MPC is firstly proposed by (Chen and Allgower, 1998).

There are two main parameters in this method which should be determined offline. The first is a region around the origin which is called terminal region and it has been proved that the states at the end of the finite prediction horizon are in this region if the terminal inequality constraint is feasible. The second is a terminal penalty matrix.

A linear state feedback is determined during the design procedure for computing the above parameters, but it has never been applied to the system.

For designing a quasi-infinite horizon MPC for a system, one of the main requirements is that the Jacobian linearization of the nonlinear system or just the linear system which must be controlled have to be stabilizable.

The ordinary differential equations for describing the system are:

$$\dot{x}(t) = f(x(t), u(t)), \quad x(0) = x_0 \tag{4}$$

with state vector $x(t) \in R^n$, input vector $u(t) \in R^m$. The input constraints are:

$$u(t) \in U, \quad \forall t \geq 0 \tag{5}$$

In order to apply this method it is assumed that:

A1- $f : R^n \times R^m \to R^n$ is twice continuously differentiable and $f(0,0) = 0$.

A2- $U \in R^m$ is compact, convex and $0 \in R^m$ is contained in the interior of U.

A3 – There are a unique solution for any initial condition and any piecewise continuous and right-continuous input to the system (4).

Note that the system which will be used in this paper is linear so the conditions A1 and A3 are satisfied. In following we will introduce the constraints on input signal which will satisfy condition A2.

In the following we describe the problem setup. The open-loop optimal control problem at time t with initial state x(t) is formulated as:

$$\min_{\bar{u}(.)} \ J(x(t), u(.)) = \int_t^{t+N_p} \|x(\tau; x(t), t)\|_Q^2 \, d\tau + \int_t^{t+Nu} \|u(\tau)\|_R^2 \, d\tau + \|x(t + T_p; x(t), t)\|_P^2$$

Subj. to

$$\begin{aligned} &\dot{x} = Ax(t) + Bu(t) \\ &y(t) = Cx(t) \\ &u(\tau) \in U, \quad \tau \in [\mathrm{t,t+T}_P] \\ &x(t + T_P; x(t), t) \in \Omega \end{aligned} \tag{6}$$

where $Q \in R^{n \times n}$, $R \in R^{m \times m}$ and $P \in R^{n \times n}$ are positive-definite, symmetric weighting matrices; and P is the terminal matrix penalty; N_p is a finite prediction horizon and N_u is a finite control horizon. The model used for future prediction is initialized by the actual system states x(t) at time t.

The cost function of (6) consists of a finite horizon quadratic cost to specify the desired control performance and a terminal cost to penalize the states at the end of the finite prediction horizon. The terminal inequality constraint will force the states at the end of the prediction horizon to be in the terminal region Ω which is a neighbourhood of the origin.

In order to design the controller for the system used in this paper which is a linear one, a procedure to determine a terminal penalty matrix P and a terminal region Ω, (preferably as large as possible) off-line can be specified as follows:

Step1: Find a locally stabilizing linear state feedback gain K for the linear system

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) \tag{7}$$

The linear optimal control technique (LQR) may be an appropriate method for determining K.

Step2: Choose a constant $\kappa \in [0, \infty)$ satisfying $\kappa < -\lambda_{\max}(A)$ and solve the Lyapunov equation

$$(A_K + \kappa I)^T P + P(A_K + \kappa I) = -Q^*$$
$$Q^* = Q + K^T RK \in R^{n \times n} \tag{8}$$

positive-definite and symmetric matrix *P*.

Step3: Find the largest possible α such that $\forall x \in \Omega_\alpha : Kx \in U$.

Indeed in the design procedure it has been proved that there is an $\alpha \in (0, \infty)$ that specifies a neighbourhood of the origin in the form:

$$\Omega_\alpha := \{x \in R^n \mid x^T Px \le \alpha\} \tag{9}$$

such that:

i. For all $x \in \Omega_\alpha$ we have $Kx \in U$ which means that the linear feedback controller satisfies the input constraints in the region Ω_α .

ii. For the system controlled by linear state feedback $u = Kx$ the terminal region Ω_α is invariant.

iii. For all $x_1 \in \Omega_\alpha$ the infinite horizon cost function subject to linear system (7), with initial state $x(t_1) = x_1$ and controlled by the local linear state feedback $u = Kx$ is bounded from above as $J^\infty(x_1, u) \le x_1^T Px_1$ where $J^\infty(x_1, u) = \int_{t_1}^{\infty} (\|x(t)\|_Q^2 + \|u(t)\|_R^2) dt$.

Note that because the system is linear, the terminal region will only be limited by input constraints.

3.2 Optimization Problem

According to the previous sections we can formulate our problem for an active front steering vehicle. It should be noted that the designed stable Model Predictive Controller will be applied approximately to a discrete system.

Consider a nominal model for the system:

$$x(k+1) = Ax(k) + Bu(k),$$
$$y(k) = Cx(k) \tag{10}$$

where it has been discretized by a sampling time T_s and $x \in R^n$ is the state vector; $u \in R^m$ is the input vector; $y \in R^q$ is the output vector and A, B, C are constant matrixes with appropriate dimensions.

Consider that the constraints in the MPC problem are defined by:

$$J = \sum_{i=1}^{N_p} \|r - y(k+i \mid k)\|_Q^2 + \sum_{i=1}^{N_u - 1} \|u(k+i \mid k)\|_R^2 + \|x(k+N_p \mid k)\|_P^2$$

Subj. to

$$\begin{aligned} & x(k+i+1 \mid k) = \\ & \qquad Ax(k+i \mid k) + Bu(k+i \mid k), \\ & y(k+i \mid k) = Cx(k+i \mid k) \\ & u(k+i \mid k) \in U, \quad \mathrm{i} = 1, \dots, N_u \\ & x(k+T_P \mid k) \in \Omega_\alpha := \{x \in R^n \mid x^T Px \le \alpha\} \end{aligned} \tag{11}$$

where r is the reference signal, $u(.|k), y(.|k)$ are the predicted input and output signals over the control and prediction horizon respectively.

The norms in J are defined as:

$$\|r - y(k+i \mid k)\|_Q^2 = (r - y(k+i \mid k))^T Q (r - y(k+i \mid k)) \tag{12}$$

Similarly for the other norms.

For converting $\|r - y(k+i \mid k)\|_{Q'}^2$ to $\|x(k+i \mid k)\|_Q^2$ simply we can define $Q = C^T Q' C$.

In many problems like active steering control of a vehicle we may have some constraints on the rate of change of input signals $\Delta u(k+i \mid k) \in \Delta U$, $\mathrm{i} = 1, \dots, N_u$ so we have to translate it to the constraints on input signals. Note that in order to be able to apply the method in designing stable MPC, ΔU should be a compact, convex set too.

4. CONVERSION TO A LINEAR MATRIX INEQUALITY PROBLEM

In this section the designed stable MPC will convert to the LMI form.

4.1 Linear Matrix Inequality

A Linear Matrix Inequality (LMI) is an expression in form:

$$F(x) := F_0 + x_1 F_1 + \dots + x_n F_n < 0 \tag{13}$$

where $x = col(x_1, \dots, x_n)$ is a vector of n real numbers called the decision variables and $F_j = F_j^T, j = 0, \dots, n$ are real symmetric matrices. Moreover $F(x) < 0$ in (13) means it is a negative definite matrix, i.e. all eigenvalues of the matrix are negative.

By using LMI we can define convex constraints on x, i.e., $S := \{x \mid F(x) < 0\}$ is convex which means that if $x_1, x_2 \in S, 0 \le \theta \le 1$ then $\theta x_1 + (1-\theta) x_2 \in S$.

Another important feature of using LMI is that solution set of systems of k individual LMIs $F_1(x) < 0, ..., F_k(x) < 0$ are representable as a single LMI as bellow:

$$F(x) = \begin{bmatrix} F_1(x) & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & F_k(x) \end{bmatrix} \tag{14}$$

And the last but most important feature of LMI problem is that nonlinear matrix inequalities can be converted to linear matrix inequalities by Schur complement. Let F be an affine function with:

$$F(x) = \begin{bmatrix} F_{11}(x) & F_{12}(x) \\ F_{21}(x) & F_{22}(x) \end{bmatrix}, F_{11} \text{ is square} \tag{15}$$

then:

$$\begin{aligned} F(x) < 0 \Leftrightarrow & \begin{cases} F_{11}(x) < 0 \\ F_{22}(x) - F_{21}[F_{11}(x)]^{-1} F_{12}(x) < 0 \end{cases} \\ \Leftrightarrow & \begin{cases} F_{22}(x) < 0 \\ F_{11}(x) - F_{12}[F_{22}(x)]^{-1} F_{21}(x) < 0 \end{cases} \end{aligned} \tag{16}$$

We will use the last feature in converting the problem in the form of LMI..

4.2 LMIs forms of the designed MPC problem

Based on the model presented in (10), the predicted states can be calculated by:

$$x(k+i \mid k) = \begin{cases} A^i x(k) + A^{i-1} Bu(k \mid k) + ... \\ +Bu(k+i-1 \mid k), & \text{if } 1 \le i \le N_u \\ \\ A^i x(k) + A^{i-1} Bu(k \mid k) + ... \\ +A^{i-N_u-1} Bu(k+N_u-2) & \text{if } N_u < i \le N_p \\ +(A^{i-N_u} B + ... + B)u(k+N_u-1 \mid k) \end{cases} \tag{17}$$

We define the metrixs:

$$\begin{aligned} \mathcal{U}(k) &= [u^T(k \mid k) \;\; u^T(k+1 \mid k) \;\; ... \\ & \quad u^T(k+N_u-1 \mid k)]^T \\ \mathcal{Y}(k) &= [y^T(k+1 \mid k) \;\; y^T(k+2 \mid k) \;\; ... \\ & \quad y^T(k+T_p \mid k) \;\; \hat{x}^T(k+T_p \mid k)]^T \\ \mathcal{T} &= [r^T \; r^T \; ... \; r^T \; \underbrace{[-1,-1,...,-1]^T}_{n} \;]^T \end{aligned} \tag{18}$$

The last block of $\mathcal{Y}$ and $\mathcal{T}$ is added to consider the terminal penalty in LMI form too.

Now we can rewrite the objective function in (11) in the augmented form:

$$J = (\mathcal{T} - \mathcal{Y}(k))^T \mathcal{Q}(\mathcal{T} - \mathcal{Y}(k)) + \mathcal{U}^T(k)\mathcal{R}\mathcal{U}(k) \tag{19}$$

and the augmented weightings are

$$\begin{aligned} \mathcal{Q} &= diag(Q, Q, ..., Q, P) \\ \mathcal{R} &= diag(R, R, ..., R) \end{aligned} \tag{20}$$

By substituting the predicted states in the form of (17) in into (10) for $i = 1, ..., N_p$ and the augmented vectors and weightings matrixes in (18) and (19), predicted output sequence $\mathcal{Y}(k)$ can be expressed as:

$$\mathcal{Y}(k) = \mathcal{A}x(k) + \mathcal{B}\mathcal{U}(k) \tag{21}$$

where:

$$\mathcal{A} = \begin{bmatrix} CA \\ \vdots \\ CA^{N_u} \\ \vdots \\ CA^{N_p} \\ A^{N_p} \end{bmatrix},$$
$$\mathcal{B} = \begin{bmatrix} CB & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ CA^{N_u-1}B & CA^{N_u-2}B & \cdots & CB \\ \vdots & \vdots & \vdots & \vdots \\ CA^{N_p-1}B & CA^{N_p-2}B & \cdots & C(A^{N_p-N_u}B + ... + B) \\ A^{N_p-1}B & A^{N_p-2}B & \cdots & (A^{N_p-N_u}B + ... + B) \end{bmatrix} \tag{22}$$

where the last row is added because of terminal constraint.

By utilizing an auxiliary scalar t and substituting to (19) the cost function of the MPC problem and the constraints on system dynamics can be replaced by:

$$J_0 = \min_{t, \mathcal{U}(k)} t \tag{23}$$

Subj. to

$$\begin{aligned} & t > J \\ & J = \\ & (\mathcal{T} - (\mathcal{A}x(k) + \mathcal{B}\mathcal{U}(k)))^T \mathcal{Q}(\mathcal{T} - (\mathcal{A}x(k) + \mathcal{B}\mathcal{U}(k))) \\ & + \mathcal{U}^T(k)\mathcal{R}\mathcal{U}(k) \end{aligned}$$

where J_0 is the optimal value of the cost function J and scalar t acting as an upper bound of J. Applying Schur complements to the constraint in the above problem, it is converted into

$$J_0 = \min_{t, \mathcal{U}(k)} \quad t \tag{24}$$

Subj. to

$$t > 0$$

$$\begin{bmatrix} t & (\mathcal{T} - (\mathcal{C}\mathcal{A}x(k) + \mathcal{C}\mathcal{B}\mathcal{U}(k)))^T & \mathcal{U}^T(k) \\ * & \mathcal{Q}^{-1} & 0 \\ \mathcal{U}(k) & 0 & \mathcal{R}^{-1} \end{bmatrix} \geq 0$$

where * indicates symmetric terms in the matrix and x(k) is the state measurement at instant k. This problem is in the form of a semi-definite optimization problem.

Now we will convert the constraints on the input and rate of input changes into LMI form. Since the U is a compact, convex set we can write:

$$\begin{aligned} &u_1^{\min} \leq u_1(k) \leq u_1^{\max} \\ &u_2^{\min} \leq u_2(k) \leq u_2^{\max} \\ &\vdots \\ &u_m^{\min} \leq u_m(k) \leq u_m^{\max} \end{aligned} \tag{25}$$

then it is identical to:

$$\begin{bmatrix} -I \\ I \end{bmatrix} U \leq \begin{bmatrix} -U^{\min} \\ U^{\max} \end{bmatrix} \tag{26}$$

Similarly for the rate of input change

$$\left.\begin{aligned} &\Delta u_1^{\min} \leq \Delta u_1(k) \leq \Delta u_1^{\max} \\ &\Delta u_2^{\min} \leq \Delta u_2(k) \leq \Delta u_2^{\max} \\ &\vdots \\ &\Delta u_m^{\min} \leq \Delta u_m(k) \leq \Delta u_m^{\max} \end{aligned}\right. \Rightarrow \Delta U^{\min} \leq \Delta U \leq \Delta U^{\max} \tag{27}$$

$$\Rightarrow \begin{aligned} -\Delta U &\leq -\Delta U^{\min} \\ \Delta U &\leq \Delta U^{\max} \end{aligned}$$

In order to convert the constraints on rate of input change into constraints on input signals and to state them in the form of LMI, we can write the input signals as:

$$\begin{bmatrix} u(k|k) \\ u(k+1|k) \\ u(k+2|k) \\ \vdots \\ u(k+N_c-1|k) \end{bmatrix} = \begin{bmatrix} I \\ I \\ I \\ \vdots \\ I \end{bmatrix} u(k-1|k) + \begin{bmatrix} I & 0 & 0 & \cdots & 0 \\ I & I & 0 & \cdots & 0 \\ I & I & I & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I & I & \dots & I & I \end{bmatrix} \begin{bmatrix} \Delta u(k|k) \\ \Delta u(k+1|k) \\ \Delta u(k+2|k) \\ \vdots \\ \Delta u(k+N_c-1|k) \end{bmatrix} \tag{28}$$

then:

$$\rightarrow \begin{bmatrix} \Delta u(k|k) \\ \Delta u(k+1|k) \\ \Delta u(k+2|k) \\ \vdots \\ \Delta u(k+N_c-1|k) \end{bmatrix} = \underbrace{\begin{bmatrix} I & 0 & 0 & \cdots & 0 \\ I & I & 0 & \cdots & 0 \\ I & I & I & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ I & I & \dots & I & I \end{bmatrix}}_{C_2}{}^{-1} \times \left(\begin{bmatrix} u(k|k) \\ u(k+1|k) \\ u(k+2|k) \\ \vdots \\ u(k+N_c-1|k) \end{bmatrix} - \underbrace{\begin{bmatrix} I \\ I \\ I \\ \vdots \\ I \end{bmatrix}}_{C_1} u(k-1|k) \right) \tag{29}$$

so

$$\begin{aligned} -C_2^{-1}U + C_2^{-1}C_1u(k-1|k) &\leq -\Delta U_{\min} \\ C_2^{-1}U - C_2^{-1}C_1u(k-1|k) &\leq \Delta U_{\max} \end{aligned} \tag{30}$$

Finally by augmenting the constraints on input signals and rate of input signals we have:

$$\begin{bmatrix} M_1 \\ M_2 \end{bmatrix} U \leq \begin{bmatrix} N_1 \\ N_2 \end{bmatrix} \tag{31}$$

where:

$$\begin{aligned} &M_1 = \begin{bmatrix} -C_2^{-1} \\ C_2^{-1} \end{bmatrix}; N_1 = \begin{bmatrix} -\Delta U^{\min} - C_2^{-1}C_1u(k-1|k) \\ \Delta U^{\max} + C_2^{-1}C_1u(k-1|k) \end{bmatrix}; \\ &M_2 = \begin{bmatrix} -I \\ I \end{bmatrix}; N_2 = \begin{bmatrix} -U^{\min} \\ U^{\max} \end{bmatrix} \end{aligned} \tag{32}$$

then:

$$MU \leq \gamma \tag{33}$$

where M is a representative of constraints with its number of rows equal to the number of constraints and number of columns equal to the dimension of u (Wang, 2009).

5. SIMULATION RESULTS

In this section the parameters in the introduced quasi-infinite model predictive control will be determined for the linear model of the described linear model of the vehicle and then it will be converted in the form of LMI. The simulation results are given too.

We will use the parameters which are represented in table 1.

The model of the car which is used in controller design is stabilizable. So the controller design method can be applied for controlling this system.

Table 1. Car Parameters.

$M\,(Mass)$	$1070\,kg$
$I_{zz}\,(Moment\ of\ Inertia)$	$2100\,kg.m^2$
$C_f\,(cornering\ stifness\ of\ forward\ tire)$	$90624\,N\ /\ rad$
$C_r\,(cornering\ stifness\ of\ rare\ tire$	$90624\,N\ /\ rad$
x_r	$1.3\ m$
x_f	$1.1\ m$
$V\,(lingitudinal\ velocity)$	$50\ km\ /\ h$

The weighting matrix for output and control input signals are (Falcon, 2007):

$$Q=\begin{bmatrix}1&0&0&0\\0&1&0&0\end{bmatrix}^T\begin{bmatrix}75&0\\0&500\end{bmatrix}\begin{bmatrix}1&0&0&0\\0&1&0&0\end{bmatrix}=\begin{bmatrix}75&0&0&0\\0&500&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix},R=150 \quad (34)$$

By choosing $\kappa=7$ the design parameters of the stable MPC computed as follows. The penalty matrix P is:

$$P=1.0\text{e}+004 * \begin{bmatrix}0.0192&0.1135&0.1574&0.0002\\0.1135&0.7559&1.0562&0.0067\\0.1574&1.0562&1.4623&0.0090\\0.0002&0.0067&0.0090&0.0001\end{bmatrix}$$

and the terminal region is:

$$x(k+T_P\,|\,k)\in\Omega_\alpha := \{x\in R^n\,|\,x^T Px \le 0.1216\}$$

The car model is discretized by using a sampling time $T_s=0.2$.

In all simulations we use the LMI Toolbox of MATLAB software and a Core2Duo 2.3GHz CPU Laptop.

For a straight roadway which the goal is reaching the middle line of the road and the prediction and control horizon are chosen as $Np=7, Nu=5$ respectively, the simulation results are as follows(the road boundary lines are y=1 and y=5:

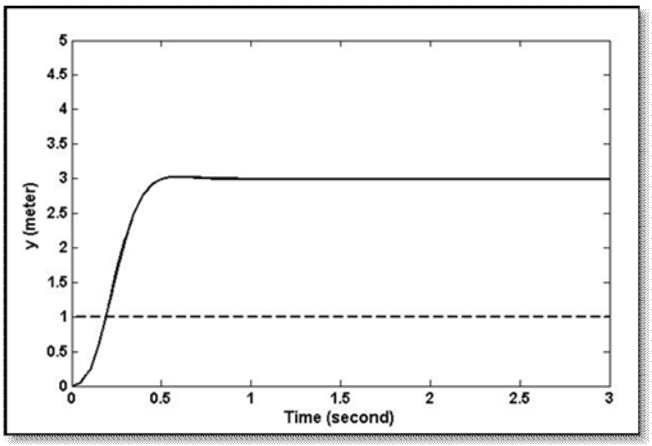

Fig. 3. The trajectory followed by the car for a straight roadway.

where the dashed lines show the boundary of the roadway.

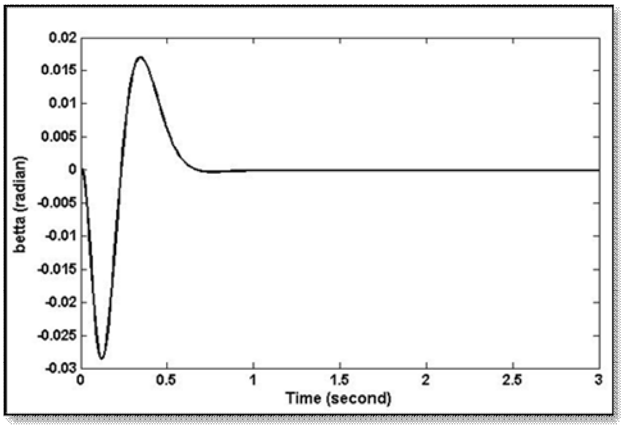

Fig. 4. Side slip angle of the car for a straight roadway.

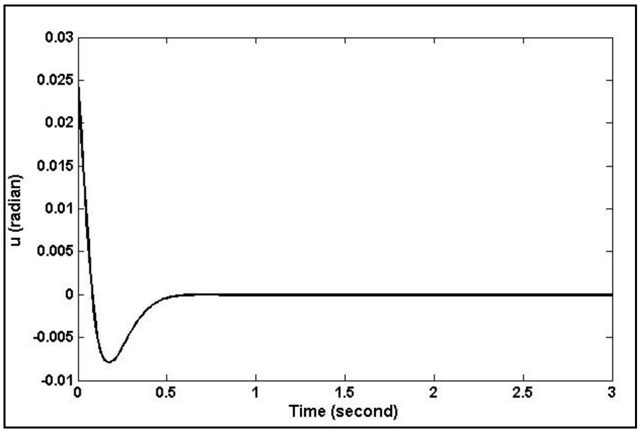

Fig. 5. The steering angle(control signal) for a straight roadway.

As it is clear the optimal output is reached and the constraints on input signals have been met.

Now we apply the controller for a lane-keeping in another kind of roadway like a sinusoidal. Here the prediction and control horizon are chosen as $Np=9, Nu=7$ respectively. So we will have:

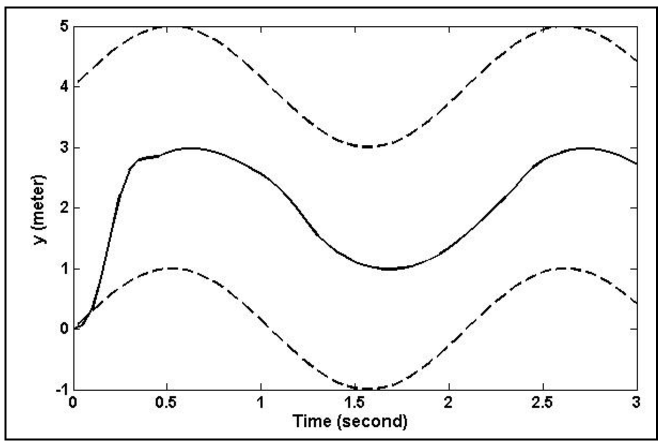

Fig. 6. The trajectory followed by the car for a sinusoidal roadway.

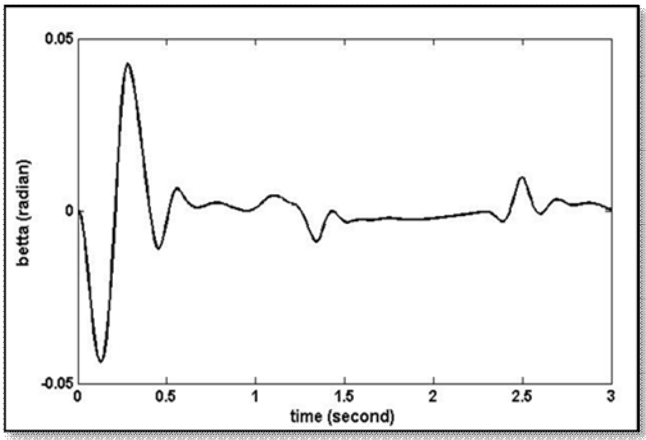

Fig. 7. Side slip angle of the car for a sinusoidal roadway.

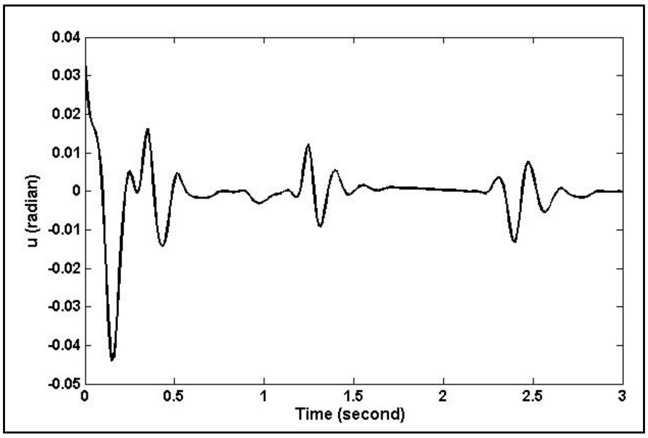

Fig. 8. The steering angle (control signal) for a sinusoidal roadway.

In this case the optimal output is reached too while the constraints on input signals have been met.

The proposed linear MPC in the form of LMI scheme has computational advantages when compared to other existing methods of solving MPC problems. To show this, we compare the proposed controller using the LMI Toolbox of MATLAB to solve and Yalmip Toolbox for solving the problem in the form (11).

For a total simulation time of 3 time-units, the elapsed CPU times for the LMI form using LMI Toolbox is 44 seconds while for the other one this time is about 97 seconds.

It is clearly seen that the proposed controller needs less CPU time than the other form.

REFERENCES

H. Chen and F. Allgower. (1998). A Quasi-Infinite Horizon Nonlinear Model Predictive Control Scheme with Guaranteed Stability. *Automatca,* vol. 34, pp. 1205-1217.

P. Falcone. (2007). Nonlinear Model Predictive Control for Autonomous Vehicles. *PhD, Department of Engineering University of Sannio*, Benevento.

P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat. Predictive Active Steering Control for Autonomous Vehicle Systems. (2007). *IEEE Transaction on Control Systems Technology,* vol. 15, pp. 566 - 580.

P. Falcone, F. Borrelli, H. E. Tseng, J. Asgari, and D. Hrovat. (2008). Hierarchical Model Predictive Control Framework for Autonomous Ground Vehicles. Presented at the *American Control Conferenc*e, Washington, USA.

L. Wang, *Model Predictive Control System Design and Implementation Using MATLAB*. (2009) Springer.

Y. Wang and S. Boyd. (2010). Fast Model Predictive Control Using Online Optimization. *IEEE Transaction on Control Systems Technology,* vol. 18, pp. 267 - 278.

I. V. Kolmanovsky, E. G. Gilbert, and H. E. Tseng. (2009). Constrained Control of Vehicle Steering. *presented at the IEEE International Conference on Control Applications*, Saint Petersburg, Russia.

S. J. Qina and T. A. Badgwell. (2003). A survey of industrial model predictive control technology. *Control Engineering Practice,* vol. 11, pp. 733–764.

M. Haeri and H. Z. Beik. (2003). Extension of Nonlinear DMC for MIMO Systems. *Presented at the International Conference on Control and Automation*, Montreal, Canada.

R. Scattolini. (2009). Architectures for distributed and hierarchical Model Predictive Control – A review. *Journal of Process Control,* vol. 19, pp. 723-731.

R. R. Bitmead, M. Gevers, and V. Wertz. (1990). *Adaptire Optimal Control-- The Thinking Man's GPc*. New York: Prentice-Hall.

S. S. Keerthi and E. G. Gilbert. (1998). Optimal Infinite-horizon Feedback Laws for a General Class of Constrained Discrete-time Systems: Stability and Moving-horizon Approximations. *Optimality and Theory Application,* vol. 57.

H. Michalska and D. Q. Mayne. (1995). Moving Horizon Observers and Observer-based Control. *IEEE Transaction on Automatic Control,* vol. 40, pp. 995 -

R. M. Esfanjani and K. Y. Nikravesh. (2011). Stabilizing model predictive control for constrained nonlinear distributed delay systems. *ISA Transactions,* vol. 50, pp. 201-206.

A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. (2002). The explicit linear quadratic regulator for constrained systems. *Automatica,* vol. 38, pp. 3-20, 2002.

P. Tøndel, T. A. Johansen, and A. Bemporad. (2001). An algorithm for multi-parametric quadratic programming and explicit MPC solutions. Presented at the *IEEE Conference Publications*.

S. J. Fesharaki, F. Towhidkhah, and H. A. Talebi. (2012). Satellite Control Method Using Laguerre Model Predictive Control Approach. Presented at the *Satellites for Sustainable Development* Iran,Tehran.

R. Rajamani. (2005). *Vehicle Dynamics and Control*: Springer.