Proceedings of the 18th World Congress
The International Federation of Automatic Control
Milano (Italy) August 28 - September 2, 2011

Model Predictive Control based on LMIs Applied to an Omni-Directional Mobile Robot

Humberto X. Araújo * **André G. S. Conceição** *
Gustavo H. C. Oliveira ** **Jônatas Pitanga** *

* *Electrical Engineering Department (DEE), Universidade Federal da Bahia (UFBA), e-mail: humberto.araujo@ufba.br, andre.gustavo@ufba.br, jonatas_pitanga@hotmail.com*
** *Electrical Engineering Department (DELT), Universidade Federal do Paraná (UFPR), e-mail: gustavo@eletrica.ufpr.br*

Abstract: The paper presents a methodology for state feedback MPC synthesis applied to the trajectory tracking control problem of a three wheeled omnidirectional mobile robot. The MPC design used here is based on a cost function developed over finite horizon and LMI framework. It is shown that MPC concepts well established for robot applications, for instance, the use of open loop predictions, receding horizon control, constraints manipulation, are preserved in this new formulation. The stability of the closed loop system is guaranteed by LMI conditions related with the cost function monotonicity. Simulation results of navigation are provided to demonstrate the performance of the proposed control strategy.

Keywords: Omni-Directional Robot, Model Predictive Control, Trajectory Control, Closed-loop Stability, Input constraints, Linear Matrix Inequalities.

1. INTRODUCTION

In the last years, there has been a great interest in model based predictive control techniques for trajectory tracking of mobile robots, as it can be seen in Oliveira and Carvalho [2003] and Raffo et al. [2009].

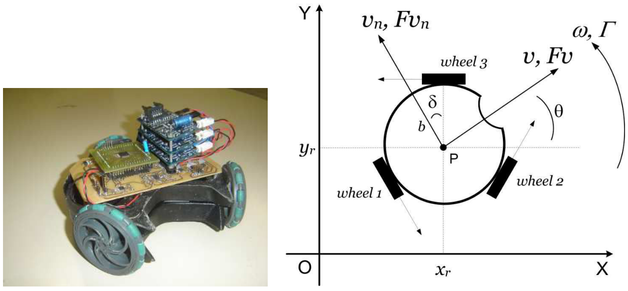
Many methods in this context have been developed. In Jiang et al. [2005], a tracking method for a mobile robot is presented, where the predictive control is used to predict the position and the orientation of the robot, while the fuzzy control is used to deal with the nonlinear characteristics of the system. A path tracking scheme for mobile robot based on neural predictive control is presented in Gu and Hu [2002], where a multi-layer back-propagation neural network is employed to model non-linear kinematics of the robot. In this work, the approach to perform the minimization of the controller's cost function is based on convex optimization, involving LMIs. Many reasons justify the use of LMIs (Boyd et al. [1994]).

Omnidirectional mobile robots have the ability to move simultaneously and independently in translation and rotation. However, nonlinearities, like motor dynamic constraints or friction related to robot's velocity, can greatly affect the behavior of the robot. So, the predictive control technique must be able to take into account the nonlinearities. A quite simple model has be used in Conceição et al. [2009], where methods for parameter estimation were proposed and validated through experimental results. A similar model is used in this work.

On the other hand, in the last decades, several LMI-based approaches have been proposed for design control laws, mainly in $\mathcal{H}_2/\mathcal{H}_\infty$ control. In the context of MPC, the use of LMIs was firstly proposed in the nineties, with the paper of Kothare et al. [1996]. In this work, a state space structure with parameter uncertainties is assumed and LMI-based constraints obtained from Lyapunov functions guarantee closed loop stability. These constraints should be satisfied for all model realizations. The state feedback control law minimizes an infinite horizon cost function subject to input and output constraints. Other references related with this approach are Mao [2003], Casavola et al. [2004] and Lee and Park [2006].

The present paper contains a methodology for state feedback MPC synthesis based on the LMI framework applied to omnidirectional mobile robot. An important property of the proposed algorithm is the use of finite horizon together with a sufficient condition for closed-loop stability. Constraints in the input and output signals are also tackled by the control law design. The stability condition is obtained by guaranteeing that an upper bound for the finite horizon cost function is monotonically non-increasing. These approach for achieving closed loop stability is different for the one described in Kothare et al. [1996] and related works, for infinite horizon, where stability is achieved by the state feedback control law u = Fx due to the well known change of variables $F = YQ^{-1}$.

The controller architecture considers both the kinematic and the dynamic control in a cascade structure, where a MPC is used to make the robot follow a reference velocity trajectory, generated by an inverse kinematics block which translates position objectives into velocity references.



(a) Mobile robot. (b) Geometric parameters and reference frames.

Fig. 1. Omnidirectional robot

The structure of this paper is presented in the following. In Section 2, the omnidirectional mobile robot is modeled into discrete space-state equations. In Section 3, the optimization problem is formulated. Section 4 contains the new synthesis of contrained stable MPC based on LMI framework. The trajectory generation method is defined in 5. Simulation results are presented in Section 6 and, the conclusions are addressed in Section 7.

2. THE OMNI-DIRECTIONAL MOBILE ROBOT MODEL

The robot pose is fully described by the vector: $\xi = [x_r \;\; y_r \;\; \theta]^T$, where x_r and y_r is the localization of the point P in the world frame, and θ the angular difference between the world and robot frames. $\dot{\xi_r} = [v \;\; v_n \;\; w]^T$ is the velocity vector at the robot frame and describes the linear velocity of the robot at the point P, represented by the orthogonal components v and v_n, and the angular velocity of the robot's body, represented by w. By geometric parameters of the robot and the robot's body frame, in Fig. 1(b), the motion equations are as follows:

$$\dot{\xi} = R^T(\theta)\dot{\xi_r} \tag{1}$$

where an orthogonal rotation matrix $R(\theta)$ is defined to map the world frame into the robot frame, and vice versa:

$$R(\theta) = \begin{bmatrix} cos(\theta) & sin(\theta) & 0 \\ -sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

By Newton's law of motion and the robot's body frame, in Fig. 1(b), the dynamic model is described by

$$F_v(t) - B_v v(t) - C_v sgn(v(t)) = M\frac{dv(t)}{dt}, \tag{3}$$

$$F_{v_n}(t) - B_{v_n} v_n(t) - C_{v_n} sgn(v_n(t)) = M\frac{dv_n(t)}{dt}, \tag{4}$$

$$\Gamma(t) - B_w w(t) - C_w sgn(w(t)) = I_n\frac{dw(t)}{dt}, \tag{5}$$

where $F = [F_v \;\; F_{v_n} \;\; \Gamma]^T$ represents the vector force (F_v and F_{v_n}) in the robot frame and the moment (Γ) around the center of gravity for the mobile robot (point P). The robot mass is M and the robot inertia moment is I_n. The viscous frictions ($B_v v(t)$, $B_{v_n} v_n(t)$ and $B_w w(t)$) represent a retarding force, which is a linear relationship between the applied force and the velocity. The coulomb frictions ($C_v sgn(v(t))$, $C_{v_n} sgn(v_n(t))$ and $C_w sgn(w(t))$) represent a retarding force that has constant amplitude with respect to the change of velocity, but the sign changes with the reversal of the direction of velocity. B_v, B_{v_n} and B_w are the viscous friction coefficients related to v, v_n and w respectively, and C_v, C_{v_n} and C_w the coulomb friction coefficients. The relationships between the robot's traction forces and the wheel's traction forces are,

$$F_v(t) = f_2(t)cos(\delta) - f_3(t)cos(\delta), \tag{6}$$

$$F_{v_n}(t) = -f_1(t) + f_2(t)sin(\delta) + f_3(t)sin(\delta), \tag{7}$$

$$\Gamma(t) = (f_1(t) + f_2(t) + f_3(t))b. \tag{8}$$

where δ is $\pi/6$. The wheel's traction force on each wheel i (for $i = 1, ...3$) is

$$f_i(t) = \frac{T_i(t)}{r_i}. \tag{9}$$

with T_i the rotation torque of the wheels. The dynamics of each DC motor i (for $i = 1, ..., 3$) can be described using the following equations,

$$u_i(t) = R_{a_i} i_{a_i}(t) + K_{v_i} w_{m_i}(t), \tag{10}$$

$$T_i(t) = l_i K_{t_i} i_{a_i}(t), \quad 0 \leq i_{a_i}(t) \leq i_{max} \tag{11}$$

where R_{a_i} are the motor's armature resistance, l_i are the motor's gear ratio reduction, and w_{m_i} the angular velocity of the wheels.

3. LMI FORMULATION OF FINITE HORIZON CONSTRAINED MPC FOR A CLASS OF MOBILE ROBOTS

The state-space equations in discrete-time obtained for the mobile robot are given by:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) + Ksign(x(k)) \\ y(k) = Cx(k), \end{cases} \tag{12}$$

where $u(k) = [u_1(k) \;\; u_2(k) \;\; u_3(k)]^T \in \mathcal{R}^{n_u}$, $n_u = 3$, is the control input corresponding to the motor's armature voltages, the output and the state-variables of the system are the robot velocities, $y(k) = x(k) = [v(k) \;\; v_n(k) \;\; w(k)]^T \in \mathcal{R}^{n_y}$, $n_y = 3$. The state-variable-form matrices defining this system are

$$A = -GQK_tLR_M^{-1}G^T - K_1 \tag{13}$$

$$B = GQ \tag{14}$$

$$K = \begin{bmatrix} -\frac{C_v}{M} & 0 & 0 \\ 0 & -\frac{C_{v_n}}{M} & 0 \\ 0 & 0 & -\frac{C_w}{I_n} \end{bmatrix}, \quad K_1 = \begin{bmatrix} \frac{B_v}{M} & 0 & 0 \\ 0 & \frac{B_{v_n}}{M} & 0 \\ 0 & 0 & \frac{B_w}{I_n} \end{bmatrix}, \tag{15}$$

$$Q = \begin{bmatrix} \frac{l_1K_{t_1}}{MR_{a1}r_1} & 0 & 0 \\ 0 & \frac{l_2K_{t_2}}{MR_{a2}r_2} & 0 \\ 0 & 0 & \frac{l_3K_{t_3}}{MR_{a3}r_3} \end{bmatrix}, \tag{16}$$

$$G = \begin{bmatrix} 0 & cos(\delta) & -cos(\delta) \\ -1 & sin(\delta) & sin(\delta) \\ b & b & b \end{bmatrix}, \quad K_t = \begin{bmatrix} K_{t_1} & 0 & 0 \\ 0 & K_{t_2} & 0 \\ 0 & 0 & K_{t_3} \end{bmatrix}, \tag{17}$$

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad R_M = \begin{bmatrix} r_1 & 0 & 0 \\ 0 & r_2 & 0 \\ 0 & 0 & r_3 \end{bmatrix}, \quad L = \begin{bmatrix} l_1 & 0 & 0 \\ 0 & l_2 & 0 \\ 0 & 0 & l_3 \end{bmatrix}. \tag{18}$$

Motor's parameters, geometric parameters, and estimated dynamic parameters used in this work can be found in Conceicao et al. [2010].

Defining $\Delta u(k) = u(k) - u(k-1)$ and $\Delta x(k) = x(k) - x(k-1)$, system (12) can be rewritten by:

$$\begin{cases} \Delta x(k+1) = A\Delta x(k) + B\Delta u(k) + \\ \qquad\qquad +K[sign(x(k)) - sign(x(k-1))] \\ y(k+1) = C\Delta x(k+1) + y(k). \end{cases} \tag{19}$$

In this work, a quadratic objective function over finite horizon, for the first N_y steps, is given by:

$$\begin{aligned} J(k) = \sum_{j=1}^{N_y} &\|\hat{y}(k+j|k) - r(k+j)\|^2_{Q_j} \\ &+ \|\Delta u(k+j-1|k)\|^2_{R_{j-1}}, \end{aligned} \tag{20}$$

where N_y defines the prediction horizon, $Q_j \geq 0$ and $R_j > 0$ are suitable weighting matrices, $\hat{y}(k+j|k)$ and $\Delta u(k+j-1|k)$ are the predicted output at $k+j$ and the optimal value for the incremental control at $k+j-1$, respectively, computed at time k, by using the model composed by equations (19), and $r(k+j)$ is the set-point at time $k+j$.

The control law is obtained by minimizing cost function (20) in relation to the control moves, that is:

$$\min_{\Delta u(k|k)\cdots\Delta u(k+N_u-1|k)} J(k), \tag{21}$$

where N_u is the control horizon. It is assumed that the control input is constant after sample time $k+N_u-1$, *i.e.*, $\Delta u(k+j|k) = 0$ for $j \geq N_u$. At each sampling time k, the vector $\mathbf{\Delta u}$ containing the sequence of control moves is calculated. Only the first control move is applied to the system. Defining the vectors

$$\hat{\mathbf{y}} = \begin{bmatrix} \hat{y}(k+1|k) \\ \hat{y}(k+2|k) \\ \vdots \\ \hat{y}(k+N_y|k) \end{bmatrix} \quad \mathbf{\Delta u} = \begin{bmatrix} \Delta u(k|k) \\ \Delta u(k+1|k) \\ \vdots \\ \Delta u(k+N_u-1|k) \end{bmatrix}, \tag{22}$$

the predicted outputs can be calculated over the finite horizon by

$$\hat{\mathbf{y}} = H\mathbf{\Delta u} + F\Delta x(k) + \Pi y(k), \tag{23}$$

where

$$H = \begin{bmatrix} CS_0B & 0 & 0 & \cdots \\ CS_1B & CS_0B & 0 & \cdots \\ \vdots & \vdots & \vdots & \\ CS_{N_y-1}B & CS_{N_y-2}B & CS_{N_y-3} & \cdots \end{bmatrix},$$

$$F = \begin{bmatrix} CS_0A \\ CS_1A \\ \vdots \\ CS_{N_y-1}A \end{bmatrix}, \qquad \Pi = [\, I \ I \ I \ \cdots \ I\,]^T,$$

and $S_j = \sum_{i=0}^{j} A^i$, neglecting the non linear part of the model for the MPC. Hence, the objective function in optimization problem (21) is equivalent to

$$J(k) = (\hat{\mathbf{y}} - \mathbf{r})^T Q_y (\hat{\mathbf{y}} - \mathbf{r}) + \mathbf{\Delta u}^T R \mathbf{\Delta u}, \tag{24}$$

where

$$Q_y = \mathrm{diag}(Q_1, Q_2, \cdots, Q_{N_y}),$$
$$R = \mathrm{diag}(R_0, R_1, \cdots, R_{N_u-1}),$$

and $\mathbf{r}$ is the reference input given by

$$\mathbf{r} = \begin{bmatrix} r(k+1) \\ r(k+2) \\ \vdots \\ r(k+N_y) \end{bmatrix},$$

with $\hat{\mathbf{y}}$ given by (23). The system state is assumed available at each sampling time k, by measurement or estimation.

In this work, constraints on actuators are taken in account by component-wise peak bounds on the input $u(k+j|k)$ and its variation:

$$\begin{aligned} u_{min_{ji}} \leq u_i(k+j|k) \leq u_{max_{ji}}, \\ i = 1, 2, \cdots, n_u;\ k,\ j \geq 0, \end{aligned} \tag{25}$$

$$\begin{aligned} v_{min_{ji}} \leq \Delta u_i(k+j|k) \leq v_{max_{ji}}, \\ i = 1, 2, \cdots, n_u;\ k,\ j \geq 0, \end{aligned} \tag{26}$$

where $u_i(k)$ and $\Delta u_i(k+j|k)$ represent the i-th input in the vectors $u(k)$ and $\Delta u(k+j|k)$, respectively. Similarly, for the output, representing performance requirements, component-wise peak bounds on $y(k+j)$ can also be considered:

$$\begin{aligned} y_{min_{ji}} \leq y_i(k+j) \leq y_{max_{ji}}, \\ i = 1, 2, \cdots, n_y;\ k \geq 0,\ j \geq 1. \end{aligned} \tag{27}$$

In this work, an LMI-based synthesis procedure is provided for finite horizon MPC optimization problem (21), with input and output constraints. Next, the representation of the input and output constraints in terms of LMI is presented.

3.1 Input constraints

Consider component-wise peak bounds on input (25) at each present and future sampling time k:

$$\begin{aligned} u_{min_{ji}} \leq u_i(k+j|k) \leq u_{max_{ji}}, \\ i = 1, 2, \cdots, n_u;\ k \geq 0,\ j = 0, 1, \cdots, N_u - 1. \end{aligned} \tag{28}$$

Since

$$\begin{cases} u(k|k) \quad\ = \Delta u(k|k) + u(k-1) \\ u(k+1|k) = \Delta u(k+1|k) + \Delta u(k|k) + u(k-1) \\ \qquad \vdots \end{cases} \tag{29}$$

inequalities (28) can be rearranged, using Schur complement, as the set of $n_u N_u$ inequalities on variable $\mathbf{\Delta u}$:

$$\begin{aligned} u_{min_{0i}} - u_i(k-1) &\leq ([\, I \ 0 \ 0 \ \cdots \ 0\,])_i\, \mathbf{\Delta u} \\ &\leq u_{max_{0i}} - u_i(k-1), \\ u_{min_{1i}} - u_i(k-1) &\leq ([\, I \ I \ 0 \ \cdots \ 0\,])_i\, \mathbf{\Delta u} \\ &\leq u_{max_{1i}} - u_i(k-1), \\ &\ \ \vdots \\ u_{min_{(N_u-1)i}} - u_i(k-1) &\leq ([\, I \ I \ I \ \cdots \ I\,])_i\, \mathbf{\Delta u} \\ &\leq u_{max_{(N_u-1)i}} - u_i(k-1), \end{aligned} \tag{30}$$

with $i = 1, 2, \cdots, n_u$. The entry $u(k-1)$ corresponds to the control move implemented at the last iteration. The notation $(M)_i$ means that only the row i of matrix M is taken. Each side of inequalities (30) are LMI in $\mathbf{\Delta u}$. Hence, there is a set of $2n_uN_u$ LMIs.

To guarantee constraints (26), the following inequalities can be determined from (29)

$$\begin{aligned} v_{min_{0i}} &\leq ([\, I\ 0\ 0\ \cdots\ 0\,])_i\, \mathbf{\Delta u} \leq v_{max_{0i}}, \\ v_{min_{1i}} &\leq ([\, 0\ I\ 0\ \cdots\ 0\,])_i\, \mathbf{\Delta u} \leq v_{max_{1i}}, \\ &\vdots \\ v_{min_{(N_u-1)i}} &\leq ([\, 0\ 0\ 0\ \cdots\ I\,])_i\, \mathbf{\Delta u} \leq v_{max_{(N_u-1)i}}, \end{aligned} \tag{31}$$

with $i = 1, 2, \cdots, n_u$. Inequalities (31) are a set of $2n_uN_u$ LMIs in Δu. LMIs (30) and (31) represent necessary and sufficient conditions to impose constraints on the input.

3.2 Output constraints

Consider component-wise peak bounds on output (27) at each future sampling time k:

$$y_{min_{ji}} \leq \hat{y}_i(k+j|k) \leq y_{max_{ji}},$$
$$i = 1, 2, \cdots, n_y;\ k \geq 0,\ j \geq 1.$$

Since, at sampling time k,

$$\begin{aligned} &y_{min_{ji}} \leq \left(H^{(j)}\right)_i \mathbf{\Delta u} + \left(F^{(j)}\right)_i \Delta x(k) + y_i(k) \leq \\ &y_{max_{ji}},\quad i = 1, 2, \cdots, n_y;\ k \geq 0,\ j = 1, 2 \cdots, N_y, \end{aligned} \tag{32}$$

then inequalities (32) are equivalent to:

$$\begin{aligned} &y_{min_{ji}} - \left(F^{(j)}\right)_i \Delta x(k) - y_i(k) \leq \left(H^{(j)}\right)_i \mathbf{\Delta u} \leq \\ &\qquad y_{max_{ji}} - \left(F^{(j)}\right)_i \Delta x(k) - y_i(k), \\ &\quad i = 1, 2, \cdots, n_y;\ k \geq 0,\ j = 1, 2, \cdots, N_y. \end{aligned} \tag{33}$$

The notation $H^{(j)}$ (or $F^{(j)}$) means that one must take the n_y rows of H (or F) corresponding to each output prediction step $\hat{y}(k+j|k)$ in $\hat{\mathbf{y}}$ (22). In inequalities (33), both sides are LMIs in $\mathbf{\Delta u}$. So, there are $2n_yN_y$ LMIs in (33).

4. THE CONSTRAINED STABLE MPC

In this section, the stability of the closed loop system is analyzed. This result is attained by imposing a non-increasing monotonicity future behavior for the upper bound of the cost function. A sufficient condition is obtained in LMI form that guarantees a stable control law, when added to MPC problem (21). Hence, the MPC problem, subject to input and output constraints, is reduced to a convex optimization involving LMI. The idea used in this formulation, that is, to relate closed loop stability to monotonicity of the cost function, has been exploited in Scokaert and Clarke [1994] , Oliveira et al. [2000], but using others methodologies.

Firstly, an upper bound $\gamma > 0$ is considered on performance objective (21):

$$J(k) \leq \gamma.$$

From (23), $J(k) \leq \gamma$ can be rewritten as

$$\begin{aligned} &[H\mathbf{\Delta u} + F\Delta x(k) + \Pi y(k) - \mathbf{r}]^T Q_y \\ &[H\mathbf{\Delta u} + F\Delta x(k) + \Pi y(k) - \mathbf{r}] + \ \mathbf{\Delta u}^T R \mathbf{\Delta u} \leq \gamma. \end{aligned} \tag{34}$$

By using Schur complement, this upper bound satisfies inequality

$$\begin{bmatrix} \gamma & [H\mathbf{\Delta u} + F\Delta x(k) + \Pi y(k) - \mathbf{r}]^T Q_y^{\frac{1}{2}} & \mathbf{\Delta u}^T R^{\frac{1}{2}} \\ * & I & 0 \\ * & * & I \end{bmatrix} \geq 0, \tag{35}$$

The symbol $*$ is used throughout this paper to induce a symmetric structure. As $\Delta x(k)$, $\mathbf{r}$ and $y(k)$ are available at each sampling time k, inequality (35) is LMI in $\mathbf{\Delta u}$ and γ. Define the matrices $\bar{F} \in R^{(n_yN_y)\times n}$ and $\bar{\mathbf{r}}$ as

$$\bar{F} = \begin{bmatrix} CS_1A \\ CS_2A \\ \vdots \\ CS_{N_y}A \end{bmatrix} \quad \text{and} \quad \bar{\mathbf{r}} = \begin{bmatrix} r(k+2) \\ r(k+3) \\ \vdots \\ r(k+N_y+1) \end{bmatrix},$$

and matrix $\bar{H} \in R^{(n_yN_y)\times[n_u(N_u+1)]}$ and T

$$\bar{H} = \begin{bmatrix} CS_1B & CS_0B & 0 & \cdots \\ CS_2B & CS_1B & CS_0B & \cdots \\ \vdots & \vdots & \vdots & \\ CS_{N_y}B & CS_{N_y-1}B & CS_{N_y-2} & \cdots \end{bmatrix}, \tag{36}$$

$$T = [\, I\ 0\ 0\ \cdots\ 0\,] \in R^{n_u\times(n_uN_u)},$$

where the identity matrix $I \in R^{n_u\times n_u}$.

Matrices $\bar{H}$ (36) can be partitioned as

$$\bar{H} = [\, \bar{h}\ H\,], \tag{37}$$

where $\bar{h} \in R^{(n_yN_y)\times n_u}$ and $H \in R^{(n_yN_y)\times(n_uN_u)}$.

Before discussing the main result of this section, additional constraints on inputs and outputs at the sample time $(k+1)$ are considered. Thus defining the control vector at sample time $(k+1)$ as $\bar{\mathbf{\Delta u}}$, component-wise peak bounds on input (25) at $(k+1)$ are given by

$$\begin{aligned} u_{min_{0i}} - (T)_i\mathbf{\Delta u} - u_i(k-1) &\leq ([\, I\ 0\ 0\ \cdots\ 0\,])_i\, \bar{\mathbf{\Delta u}} \\ &\leq u_{max_{0i}} - (T)_i\mathbf{\Delta u} - u_i(k-1), \\ u_{min_{1i}} - (T)_i\mathbf{\Delta u} - u_i(k-1) &\leq ([\, I\ I\ 0\ \cdots\ 0\,])_i\, \bar{\mathbf{\Delta u}} \\ &\leq u_{max_{1i}} - (T)_i\mathbf{\Delta u} - u_i(k-1), \\ &\vdots \\ u_{min_{(N_u-1)i}} - (T)_i\mathbf{\Delta u} - u_i(k-1) &\leq ([\, I\ I\ I\ \cdots\ I\,])_i\, \bar{\mathbf{\Delta u}} \\ &\leq u_{max_{(N_u-1)i}} - (T)_i\mathbf{\Delta u} - u_i\,(k-1), \end{aligned} \tag{38}$$

with $i = 1, 2, \cdots, n_u$.

In a similar way, constraints are defined for the outputs at sample time $(k+1)$, for component-wise peak bounds (27)

$$\begin{aligned} &y_{min_{ji}} - \left(\bar{F}^{(j)}\right)_i \Delta x(k) - y_i(k) \leq \left(\bar{h}^{(j)}\right)_i T\mathbf{\Delta u} + \\ &\left(H^{(j)}\right)_i \bar{\mathbf{\Delta u}} \leq y_{max_{ji}} - \left(\bar{F}^{(j)}\right)_i \Delta x(k) - y_i(k), \\ &\quad i = 1, 2, \cdots, n_y;\ j = 1, 2, \cdots, N_y. \end{aligned} \tag{39}$$

Theorem 1: Consider discrete-time system (19) and let $y(k)$ and $\Delta x(k)$ be respectively the measured output and the calculated state variation, at sample k. A MPC problem with guaranteed stability, with input and output constraints, can be solved by the following convex optimization problem:

$$\min_{\mathbf{\Delta u},\bar{\mathbf{\Delta u}},\gamma>0,\gamma_f>0} \quad \gamma_f \tag{40}$$

subject to inequality (35) and

$$\begin{bmatrix} \gamma_f & N(k) & \bar{\mathbf{\Delta u}}^T R^{\frac{1}{2}} \\ * & I & 0 \\ * & * & I \end{bmatrix} \geq 0, \tag{41}$$

with $N(k) = [\bar{h}T\mathbf{\Delta u} + H\bar{\mathbf{\Delta u}} + \bar{F}\Delta x(k) + \Pi y(k) - \bar{\mathbf{r}}]^T Q_y^{\frac{1}{2}}$,

$$\gamma_p \geq \gamma, \tag{42}$$

$$\gamma \geq \gamma_f, \tag{43}$$

$$\gamma \geq \gamma_\epsilon, \tag{44}$$

and constraints (30), (31), (38), (33) and (39), depending on the input and output constrains to be imposed, respectively. γ_p is the optimal value of γ computed at sample time $(k-1)$. γ_ϵ is a tuning parameter. At each sampling time k, two sequences of future control moves are calculated, $\mathbf{\Delta u}$ starting at time k and $\bar{\mathbf{\Delta u}}$, at time $(k+1)$. Only the first control move of $\mathbf{\Delta u}$ is applied to the system at time k. This receding horizon control law guarantees that the upper bound of the cost function is monotonically non-increasing, leading to the BIBO stability of closed loop system (19).

Proof: LMIs (35), presented previously, and (41) are related to the cost upper-bound. LMI (41) is similar to (35) for $\bar{\mathbf{\Delta}}_{\mathbf{u}}$. LMIs (41), (42) and (43) guarantee the stability of the closed loop system, as shown in the following.

The non-increasing monotonicity for the upper bound is obtained if

$$\gamma_p \geq \gamma^{(*)} \geq {\gamma_f}^{(*)}, \ \forall k \geq 0, \tag{45}$$

where $(*)$ means the optimal value calculated by problem (40) at each sample time. LMI (42) guarantees that the left part of inequality (45) is true, and $\gamma^{(*)} \geq J^{(*)}(k)$, from LMI (35). The right-hand-side of inequality (45) is achieved by means of LMI (43), and $\gamma_f^{(*)} \geq J^{(*)}(k+1)$, from LMI (41). The condition obtained from LMI (41) is approached as follows.

Using Schur complement, LMI (41) can be obtained from the fact of vector $\hat{\mathbf{y}}$ calculated at sample time $(k+1)$ is given by

$$\hat{\mathbf{y}} = \bar{h}T\mathbf{\Delta u} + \bar{H}\bar{\mathbf{\Delta u}} + \bar{F}\Delta x(k) + \Pi y(k). \tag{46}$$

So, LMI (41) guarantees that $\gamma_f^{(*)} \geq J^{(*)}(k+1)$. As the upper-bound is monotonically non-increasing, to avoid premature stop of the algorithm when great changes in reference input happen, a slack variable γ_ϵ is included.

LMIs (30), (31), (38), (33) and (39) assure that the inputs and outputs constraints at time k and $(k+1)$ be satisfied, and the proof is complete.

5. TRAJECTORY GENERATION

The trajectory of reference is defined as a set of points in the world frame (OXY): $Traj(k+j) = [\bar{x_r}(k+j) \ \bar{y_r}(k+j) \ \bar{\theta}(k+j)]^T$, $j = 0,1,...,N_y - 1$. So, given the current position and heading of the robot, it is necessary to calculate the desired velocities for the next N_y periods of time. The vector of velocity references $\bar{x}(k+j|k)$=$[\bar{v}(k+j|k) \ \bar{v_n}(k+j|k) \ \bar{w}(k+j|k)]^T$, $j = 0,1,...,Hp-1$, where j is an step prediction of the robot velocity made at instant k, is given by

$$\begin{bmatrix} \bar{v}(k+j|k) \\ \bar{v_n}(k+j|k) \\ \bar{w}(k+j|k) \end{bmatrix} = \begin{bmatrix} cos(\theta(k)) & sin(\theta(k)) & 0 \\ -sin(\theta(k)) & cos(\theta(k)) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} e_{vx} \\ e_{vy} \\ e_w \end{bmatrix}, \tag{47}$$

with

$$\begin{bmatrix} e_{vx} \\ e_{vy} \\ e_w \end{bmatrix} = \begin{bmatrix} v_{nav}cos(\varphi) \\ v_{nav}sin(\varphi) \\ \bar{\theta}(k+j|k) - \theta(k) \end{bmatrix}, \tag{48}$$

and

$$\varphi = atan2\left(\bar{y_r}(k+j|k) - y_r(k), \bar{x_r}(k+j|k) - x_r(k)\right) \tag{49}$$

$\xi(k) = [x_r(k) \ \ y_r(k) \ \ \theta(k)]^T$ is the robot pose at time step k, and v_{nav} is the linear velocity of navigation of the robot, which is a design parameter.

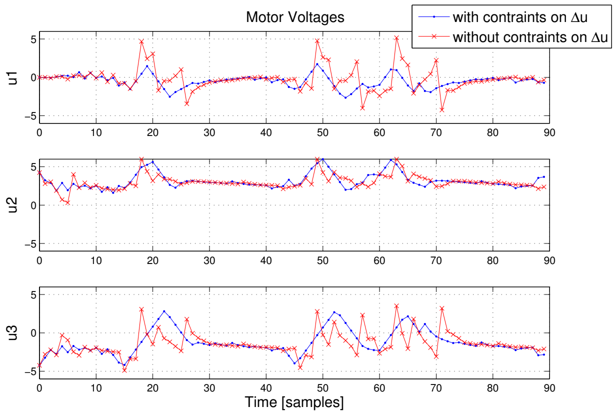


Fig. 2. Control signals.

6. SIMULATION RESULTS

In this section, the MPC with output and input constraints is used to control the omni-directional robot with 3 wheels. The controller parameters are presented in the following. The cost function weight matrices are defined as $Q_y = 100I$ and $R = 0.01I$, and the slack variable is $\gamma_\epsilon = 800$. The prediction and control horizons are selected as $N_y = 5$ and $N_u = 3$. The input and output signals constraints are $|u_i| \leq 6$, $i = 1,2,3$, $|\Delta u_i(k+j|k)| \leq 1$, $i = 1,2,3$, $j \geq 1$, and $|y_i| \leq 2.02$, $i = 1,2$, $|y_3| \leq 20.02$. The linear velocity of navigation was $v_{nav} = 0.6(m/s)$.

The trajectory of test has sudden change of direction and orientation with movements of rotation and translation in the same time, in order to test the controller in hard condition. On the first part of the trajectory, the robot orientation is fixed in 0 deg. On the second part 90 deg, on the third part 180 deg, and on the last part the robot is commanded to rotate 270 deg. The tracking response is illustrated in Figure 4, with good performance, the peak tracking error is under 3cm bound. The orientation angle tracking error is less than 8 deg. In this test, the motor

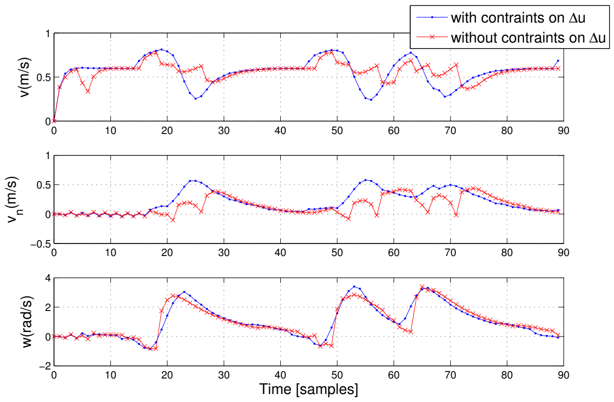


Fig. 3. Output signals.

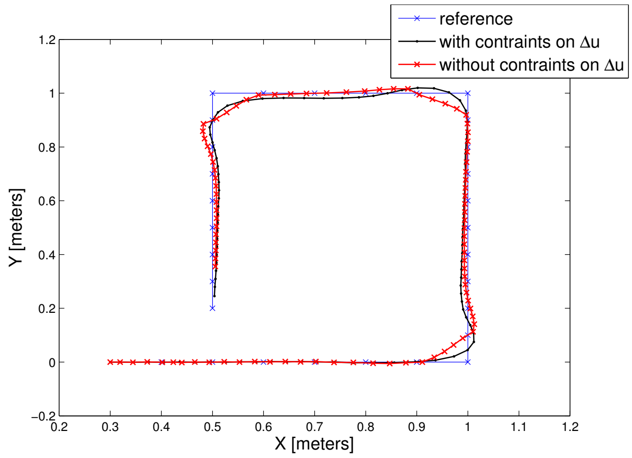


Fig. 4. Tracking performance.

voltages does not exceed the predefined limits, as can be seen in Figure 2. The use of $\Delta u_i \leq 1$ volt represents a smooth control inputs. In the contrary case, voltage variations bigger than 6 volts can be seen, for example on the sample 70 of the Figure 2 (u_3 with variations from -3.1 to 3.2 volts and u_1 with variations from 2 to -4.2 volts), forcing the actuators to operate in peak regions. Another interesting characteristic of smooth inputs was the time to goal. The robot covered a bigger distance with constraints on Δu_i due to a smooth navigation, see Figure 4. The output signals can be observed in Figure 3. The closed loop is stable and the transitory behavior is adequate. The output signals belong to the feasible set, since they were taken into account by the control law design.

7. CONCLUSIONS

In this paper, it has been presented simulation results obtained from application of a new model predictive controller design, with cost function developed over finite horizon and based on LMI framework, for omni-directional mobile robots. This LMI formulation is achieved by defining an upper bound to a quadratic objective function, considering input and output constraints. It has been shown that the control method is adequate for controlling robotic systems. Closed loop system stability is guaranteed by deriving LMI constraints for the monotonicity of the upper bound of the cost function. The constraints on the variation of control inputs has influence in the path-following errors, and also in the time to goal. As seen in Figures, it may be concluded that constraints preserves the actuators with a reasonably performance of control.

8. ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support of FAPESB (Fundação de Amparo à Pesquisa do Estado da Bahia).

REFERENCES

S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM Studies in Applied Mathematics, 1994.

A. Casavola, D. Famularo, and G. Franze. Robust constrained predictive control of uncertain norm-bounded linear systems. *Automatica*, 40:1865–1876, 2004.

A.G.S. Conceicao, C.E.T. Dorea, and J.C.L.B. Sb. Predictive control of an omnidirectional mobile robot with friction compensation. In *Robotics Symposium and Intelligent Robotic Meeting (LARS), 2010 Latin American*, pages 30 –35, 2010. doi: 10.1109/LARS.2010.22.

A.S. Conceição, A.P. Moreira, and P.J. Costa. Practical approach of modeling and parameters estimation for omnidirectional mobile robots. *Mechatronics, IEEE/ASME Transactions on*, 14 (3):377–381, June 2009. ISSN 1083-4435. doi: 10.1109/TMECH.2009.2013615.

D. Gu and H. Hu. Neural predictive control for a car-like mobile robot. *Robotics and Autonomous Systems*, 39(2): 73–86, 2002.

X. Jiang, Y. Motai, and X. Zhu. Predictive fuzzy control for a mobile robot with nonholonomic constraints. In *12th International Conference on Advanced Robotics, ICAR '05*, pages 58–63, 2005.

M. V. Kothare, V. Balakrishnan, and M. Morari. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361–1379, 1996.

S. M. Lee and J. H. Park. Robust model predictive control for norm-bounded uncertain systems using new parameter dependent terminal weighting matrix. *Chaos Solitons and Fractals*, in press, 2006.

Wei-Jie Mao. Robust stabilization of uncertain time-varying discrete systems and comments on "an improved approach for constrained robust model predictive control". *Automatica*, 39(6):1109–1112, 2003.

G. H. C. Oliveira and J. R. H. Carvalho. A non-linear predictive control scheme for nonholonomic mobile robots. In *Proc. of the IFAC Symposium on Robot Control - SYROCO*, Wroclaw, Poland, 2003.

G. H. C. Oliveira, W. C. Amaral, G. Favier, and G. Dumont. Constrained robust predictive controller for uncertain processes modeled by orthonormal series functions. *Automatica*, 36(4):563–572, 2000.

G.V. Raffo, G.K. Gomes, J.E. Normey-Rico, C.R. Kelber, and L.B. Becker. A predictive controller for autonomous vehicle path tracking. *Intelligent Transportation Systems, IEEE Transactions on*, 10(1):92–102, March 2009. ISSN 1524-9050. doi: 10.1109/TITS.2008.2011697.

P. O. M. Scokaert and D. W. Clarke. Stabilising properties of constrained predictive control. *IEE Proceeding: Control Theory*, 141(5):295–304, 1994.