Point Stabilization of Mobile Robots with Nonlinear Model Predictive Control

Felipe Kühne, Walter Fetter Lages and João Manoel Gomes da Silva Jr.
Electrical Engineering Department
Federal University of Rio Grande do Sul
Av. Oswaldo Aranha, 103
Porto Alegre, RS 90035-190 Brazil
{kuhne,fetter,jmgomes}@eletro.ufrgs.br

Abstract— This paper presents an optimal control scheme based on model-based predictive control (MPC) for a wheeled mobile robot (WMR) with nonholonomic constraints. It is shown that, by using MPC, some advantages can be obtained, such as the ability to handle constraints due to state or input limitations and performance improvement. To solve some problems with other MPCs for WMRs, this paper proposes to formulate a cost function in polar coordinates. Considerations regarding the computational effort of the MPC are developed with the purpose of analysing the viability of the proposed technique in real-time.

Index Terms— Nonholonomic systems, wheeled mobile robots, model-based predictive control.

I. INTRODUCTION

The field of mobile robot control has been the focus of active research in the past decades. Despite the apparent simplicity of the kinematic model of a wheeled mobile robot (WMR), the design of stabilizing control laws for those systems can be considered a challenge due to the existence of nonholonomic constraints. Due to Brockett's conditions [1], a smooth static state feedback control law cannot be used to stabilize a nonholonomic system at a given configuration. To overcome these limitations traditional techniques use non-smooth or time-varying control laws [2]–[6].

However, in realistic implementations, the traditional techniques for the control of nonholonomic WMRs often do not present good results, due to constraints on inputs or states that naturally arise. Also, in general, the resulting closed-loop trajectory presents unnecessary oscillatory motions. Furthermore, tuning parameters are difficult to choose in order to achieve good performance since the control laws are not intuitively obtained.

In this paper we show that, by using model-based predictive control (MPC), these disadvantages can be overcome: the tuning parameters are easy to deal with; a cost function is minimized, which makes the control law optimal according to the optimization criterion; constraints on state and control inputs can be considered in a straightforward way. Note that for a WMR the latter is an important feature, since the position of the robot can be restricted to belong to a safe region. Control actions that respect actuators limits can be generated. Furthermore, a control law that respects Brockett's conditions can be implicitly obtained.

On the other hand, the main drawback of MPC schemes is related to its computational burden which, in the past years, had limited its applications only to sufficient slow dynamic systems. However, with the development of increasingly faster processors and efficient numerical algorithms, the use of MPC in faster applications which is the case of WMRs becomes possible.

Although MPC is not a new control method, works dealing with MPC of WMRs are sparse. In [7], [8], GPC (Generalized Predictive Control) is used to solve the path following problem. A linear model is used to compute the distance between the robot and a reference path. The control acts only in the angular velocity, while the linear velocity is constant. In [9] a nonlinear model of the WMR is used for trajectory tracking. The problem is solved considering unknown obstacles in the configuration space. A neural network is used in the optimization problem. In [10] the path following problem is solved. Neural networks are used in the kinematic model which predicts the future behavior of a car-like WMR. The modeling errors are corrected on-line with the neural network model. Using a nonlinear model of the robot, in [11] a nonlinear MPC (NMPC) algorithm in state-space representation is developed, which is applied to both problems of point stabilization and trajectory tracking. A steady-state error is identified and a modified cost function to be minimized is proposed. The disadvantages of the above cited works is that, with a linear model, the problem of point stabilization can not be solved. Furthermore, this model is only valid when the robot is close enough to the reference path. Techniques using neural networks depends on training. Then, the feasibility of the control law can not be guaranteed for all possible situations. By using the controller proposed by [11], it can be noted that the steady-state error can not be totally eliminated.

In this paper a NMPC strategy is developed to solve the problem of point stabilization for a nonholonomic WMR. By using a standard quadratic cost function in the optimization problem, a large steady-state error in the final configuration of the WMR is identified, caused by a coupling between the position states of the kinematic model. By using the idea developed in [11] which uses a modified cost function, this error is reduced but not eliminated, since the coupling of the position

states remains. Therefore, in order to eliminate this steady-state error and also to improve the performance of the closed-loop trajectories, a transformation to polar coordinates which promotes the decoupling of the position states is introduced in the cost function to be minimized. Also, computational effort analysis are carried out to study the viability of the application of the proposed algorithms in real-time.

II. KINEMATIC MODEL OF THE WMR

A mobile robot made up of a rigid body and non deforming wheels is considered (Fig. 1). It is assumed that the vehicle moves without slipping on a plane, i.e., there is a pure rolling contact between the wheels and the ground. The kinematic model of the WMR is then given by [12]:

$$\begin{cases} \dot{x} = v\cos\theta \\ \dot{y} = v\sin\theta \\ \dot{\theta} = w \end{cases} \quad (1)$$

where $\mathbf{x} = [x\ y\ \theta]^T$ describes the configuration (position and orientation) of the center of the axis of the wheels, C, with respect to a global inertial frame $\{O, X, Y\}$. $\mathbf{u} = [v\ w]^T$ is the control input, where v and w are the linear and the angular velocities, respectively.

Since the MPC scheme used here is computed in discrete-time, it is necessary to discretize the kinematic model. Considering a sampling period T, a sampling instant k and applying the Euler's approximation to (1), we obtain the following discrete-time model for the robot dynamics:

$$\begin{cases} x(k+1) = x(k) + v(k)\cos\theta(k)T \\ y(k+1) = y(k) + v(k)\sin\theta(k)T \\ \theta(k+1) = \theta(k) + w(k)T \end{cases} \quad (2)$$

or, in the compact representation,

$$\mathbf{x}(k+1) = f_d(\mathbf{x}(k), \mathbf{u}(k)), \quad (3)$$

III. THE MPC ALGORITHM

MPC is an optimal control strategy that uses the model of the system to obtain an optimal control sequence by minimizing an objective function. At each sampling instant, the model is used to predict the behavior of the system over a prediction horizon. Based on these predictions, the objective

Fig. 1. Coordinate system of the WMR.

function is minimized with respect to the future sequence of inputs, thus requiring the solution of a constrained optimization problem for each sampling instant. Although prediction and optimization are performed over a future horizon, only the values of the inputs for the current sampling interval are used and the same procedure is repeated at the next sampling instant using the updated process measurements and a shifted horizon. This mechanism is known as *moving* or *receding horizon* strategy, in reference to the way in which the time window shifts forward from one sampling instant to the next one.

Considering a robot described by (3), the following prediction model can be formulated:

$$\mathbf{x}(k+j+1|k) = f_d(\mathbf{x}(k+j|k), \mathbf{u}(k+j|k)), \quad j \in [0, N-1],$$

where $j \in [0, N-1]$ and the notation $a(m|n)$ indicates the value of a at the instant m predicted at instant n. Furthermore, we consider the existence of bounds on the amplitude of state and control variables:

$$\underline{\mathbf{x}} \le \mathbf{x}(k+j|k) \le \overline{\mathbf{x}}, \quad j \in [0, N], \quad (4)$$
$$\underline{\mathbf{u}} \le \mathbf{u}(k+j|k) \le \overline{\mathbf{u}}, \quad j \in [0, N-1], \quad (5)$$

where $\underline{\mathbf{x}}$ and $\underline{\mathbf{u}}$ stand for lower bounds and $\overline{\mathbf{x}}$ and $\overline{\mathbf{u}}$ stands for the upper bounds[1].

The cost function to be minimized can be stated as a quadratic function of the states and control inputs:

$$\Phi(k) = \sum_{j=1}^{N} \mathbf{x}^T(k+j|k)\mathbf{Q}\mathbf{x}(k+j|k) + \\ + \mathbf{u}^T(k+j-1|k)\mathbf{R}\mathbf{u}(k+j-1|k), \quad (6)$$

which will be called hereafter the original cost function in cartesian coordinates, where N is the prediction horizon and $\mathbf{Q} \ge 0$, $\mathbf{R} > 0$ are weighting matrices used to penalize the state error and the control effort, respectively.

The optimization problem can therefore be stated as to find a sequence of states and controls such that:

$$\mathbf{x}^\star, \mathbf{u}^\star = \arg\min_{\mathbf{x}, \mathbf{u}} \{\Phi(k)\} \quad (7)$$

s. a.

$$\mathbf{x}(k|k) = \mathbf{x}_0, \quad (8)$$
$$\mathbf{x}(k+j+1|k) = f_d(\mathbf{x}(k+j|k), \mathbf{u}(k+j|k)), \ j \in [0, N-1] \quad (9)$$
$$\mathbf{C}\mathbf{x}(k+j|k) \le \mathbf{c}, \quad j \in [0, N] \quad (10)$$
$$\mathbf{D}\mathbf{u}(k+j|k) \le \mathbf{d}, \quad j \in [0, N-1] \quad (11)$$

where $\mathbf{x}_0$ in (8) is the initial condition which corresponds to the value of the states measured at the current instant.

Constraint (9) represents the prediction model and (10)-(11) are a general form to represent linear bounds in the state and control variables, and they may be present or not in the
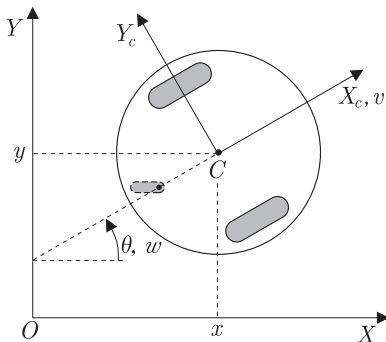
---

[1]The symbol $\le$ stands for componentwise inequalities in this case.

optimization problem. Note that we can rewrite (10)-(11) in the form of (4)-(5) with:

$$\mathbf{C} = \mathbf{D} = \begin{bmatrix} \mathbf{I} \\ -\mathbf{I} \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} \overline{\mathbf{x}} \\ -\underline{\mathbf{x}} \end{bmatrix}, \quad \mathbf{d} = \begin{bmatrix} \overline{\mathbf{u}} \\ -\underline{\mathbf{u}} \end{bmatrix}$$

The optimization problem (7)–(11) is then solved at each time step k, yielding a sequence of optimal states $\{\mathbf{x}^\star(k|k+1), \cdots, \mathbf{x}^\star(k+N|k)\}$, optimal control inputs $\{\mathbf{u}^\star(k|k), \cdots, \mathbf{u}^\star(k+N-1|k)\}$ and the optimal cost $\Phi^\star(k)$. The MPC control law is implicitly given by the first control action of the sequence of optimal control, $\mathbf{u}^\star(k|k)$, and the remaining portion of this sequence is discarded.

IV. MPC WITH COST FUNCTION IN CARTESIAN COORDINATES

In this section we focus on the formulation of MPC for the WMR considering a cost function in cartesian coordinates. In order to evaluate the performance of the approach, simulation results are shown[2]. As a case study, let us consider the robot Twil [13], which have the following limits in the amplitude of the control variables [14]:

$$\underline{\mathbf{u}} = \begin{bmatrix} -0.47 \ m/s \\ -3,77 \ rad/s \end{bmatrix} \qquad \overline{\mathbf{u}} = \begin{bmatrix} 0.47 \ m/s \\ 3.77 \ rad/s \end{bmatrix} \quad (12)$$

Initially, the cost function to be minimized is that showed in (6). Then, a second case with some modifications to the cost function, proposed in [11], is shown. In both cases, constraints only in the amplitude of the control inputs based on the robot Twil, Eq. (12) will be taken into account.

A. Original Cost Function in Cartesian Coordinates.

The weighting matrices used here are $\mathbf{Q} = \text{diag}(1, 1, 0.5)$ (equal penalty in the position states and a half of the penalty in the orientation) and $\mathbf{R} = \text{diag}(0.1, 0.1)$ (less penalty in the control effort). The prediction horizon is $N = 5$. The initial configuration is $\mathbf{x}_0 = [0 \ \ 6 \ \ 0]^T$, and the goal is the origin. The obtained simulation results are shown in Figures 2-3.
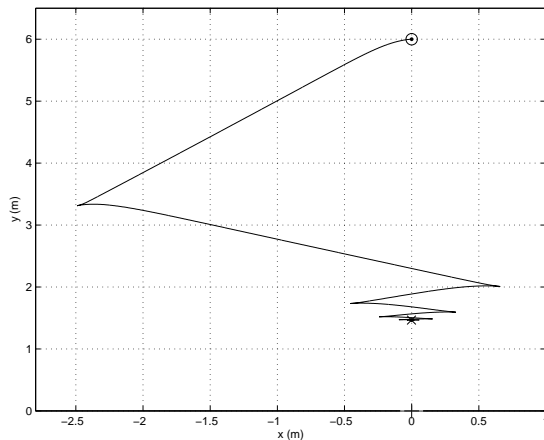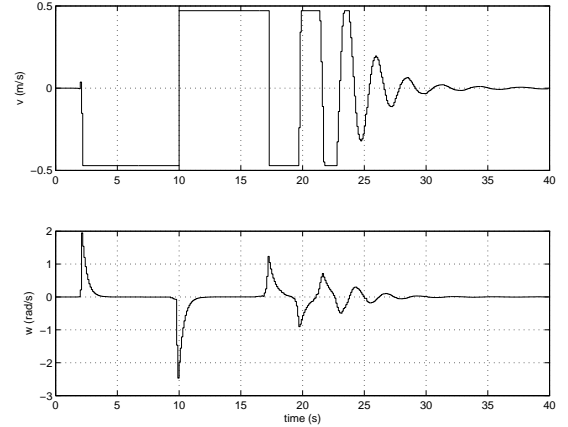
Fig. 2. Trajectory in the XY-plane.

Fig. 3. Control inputs.

Fig. 2 shows the trajectory of the robot in the XY-plane and Fig. 3 shows that the generated control signals respect the imposed constraints. It is easy to note that there is a large steady-state error in one of the position variables (y-state, in this case). Notice also that the robot has already stopped since both control inputs converge to zero. The final configuration is $\mathbf{x}_f = [0 \ \ 1.47 \ \ 0]^T$. This steady-state error can be explained by the fact that both states, x and y, depends on the same control variable, the linear velocity v, as seen in (1). Thus, when the optimization algorithm minimizes with respect to v and x, it can no more minimize with respect to y, since the cost function obeys a monotonic decreasing behavior. In [14] is shown that, by increasing considerably the prediction horizon, this problem can be reduced. However, long prediction horizons are in general undesirable since the computational effort is directly related to it.

B. Modified Cost Function in Cartesian Coordinates.

The steady-state error presented in Section IV-A was also identified in [11]. In the attempt of eliminating it, a modified cost functions have been proposed in order to increase the state penalty over the horizon, thus forcing the states to converge to an acceptable solution. Hence, the idea of exponentially increasing state weighting has been introduced. Also, a terminal state cost has been added to the cost function to be minimized. From these modifications the cost function assumes therefore the following form:

$$\Phi(k) = \sum_{j=1}^{N-1} \mathbf{x}^T(k+j|k)\mathbf{Q}(j)\mathbf{x}(k+j|k)+$$

$$+ \sum_{j=0}^{N-1} \mathbf{u}^T(k+j|k)\mathbf{R}\mathbf{u}(k+j|k) + \Omega(\mathbf{x}(k+N|k)), \quad (13)$$

where $\Omega(\mathbf{x}(k+N|k)) = \mathbf{x}^T(k+N|k)\mathbf{P}\mathbf{x}(k+N|k)$ and $\mathbf{Q}(j) = 2^{j-1}\mathbf{Q}$. Using the same tuning parameters and the same initial condition as in Section IV-A, and with a terminal state weighting of $\mathbf{P} = 50\mathbf{Q}(N)$, the simulation results are shown in Figures 4-5.
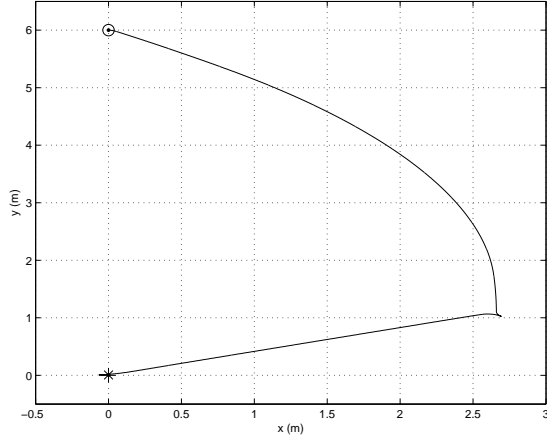
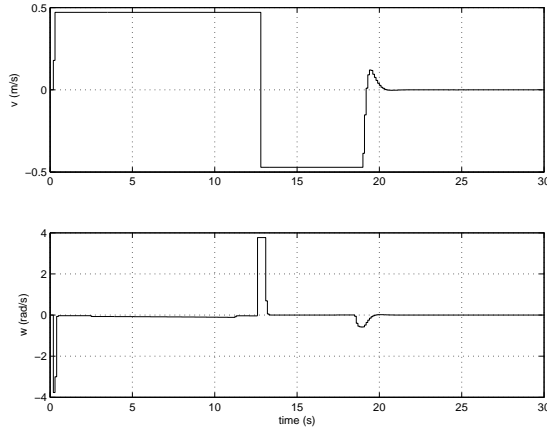Fig. 4. Trajectory in the XY plane.

Fig. 5. Control inputs.

It can be seen in Fig. 4 that the large steady-state error in y-state has disappeared. Also, notice in Fig. 5 that the convergence rate is higher. In the example of Section IV-A the stabilization time is about 40 s, while here it is 21 s. Again, the control signals respect the imposed constraints.

However, even if it can be considered small, the final y-state still presents a persistent steady-state error since the coupling between the states x and y remains. The final configuration is $\mathbf{x}_f = [0 \ \ 0.006 \ \ 0]^T$.

V. MPC WITH COST FUNCTION IN POLAR COORDINATES

Hence, a possible way to overcome the state steady-state error would be the use of some coordinate transformation able to decouple the position states x and y from the same control input, v. This idea is developed here, by using the following state transformations to polar coordinates, as proposed in [15]:

$$e = \sqrt{x^2 + y^2}, \qquad \phi = \mathrm{atan2}(y, x), \qquad \alpha = \theta - \phi,$$

In [15], it can be seen that, if one of the state variables converges to the origin, the other ones will also converge to the origin. Then the objective here is to incorporate this behavior in the cost function to be minimized. By defining

$\mathbf{x}_p = [e \ \ \phi \ \ \alpha]^T$, the following cost function can be formulated:

$$\Phi_p(k) = \sum_{j=1}^N \mathbf{x}_p^T(k+j|k)\mathbf{Q}\mathbf{x}_p(k+j|k)+$$

$$+ \mathbf{u}^T(k+j-1|k)\mathbf{R}\mathbf{u}(k+j-1|k) \quad (14)$$

A. NMPC with Constraints only in Control Variables.

Changing the cost function in (7) by (14) and considering the same tuning parameters and initial condition used in Section IV, the following results shown in Figures 6-7 are obtained. Once more, constraints in the amplitude of the control inputs described by (12) are taken into account.
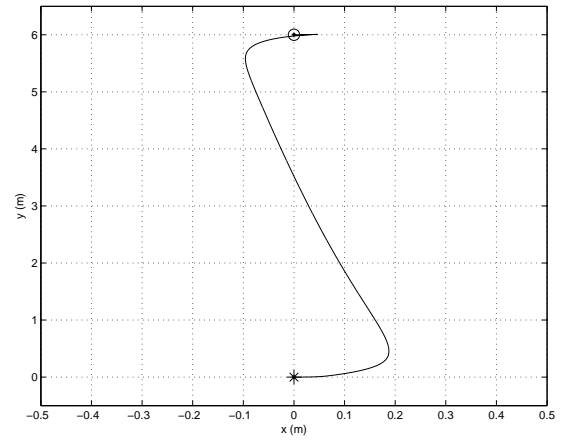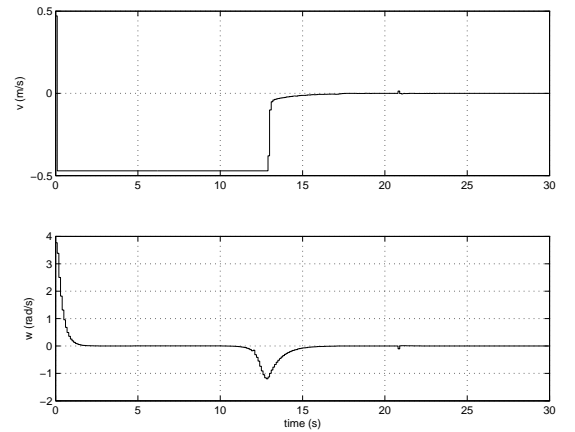

Fig. 6. Trajectory in the XY plane.

Fig. 7. Control inputs.

Comparing the results above with the results obtained in Section IV, significative performance improvements with respect to both state and control trajectories can be noted. The robot needs about 16 s to achieve its objective, against 40 s and 21 s of the previous two cases, respectively. It can be seen in Fig. 6 that the x-state presents a maximum amplitude of about $0.3 \ m$, while for the example in Section IV-B the x-state needs almost $3 \ m$ to maneuver towards the origin (Fig. 4).

Furthermore, in Fig. 7 the control inputs are smoother than the ones presented in Fig. 5.

Also note that, with the transformation to polar coordinates, the decoupling between the position states x and y in the cost function has been possible. Now the final configuration is $\mathbf{x}_f = [0 \ 0 \ 0]^T$. Furthermore, it has been possible to improve the overall performance without the inclusion of another terms in the cost function or a longer prediction horizon.

B. NMPC with Constraints in Control and State Variables.

In this section, we show an illustrative example of the WMR following a corridor. For such a task, the inclusion of constraints in the amplitude of the states such as (4) will be now considered in the optimization problem.

Then, by using the cost function in polar coordinates (14) and the same control constraint and tuning parameters used in the previous examples and an initial condition of $\mathbf{x}_0 = [-4 \ 4 \ \pi]^T$, we obtain the result shown in Fig. 8.
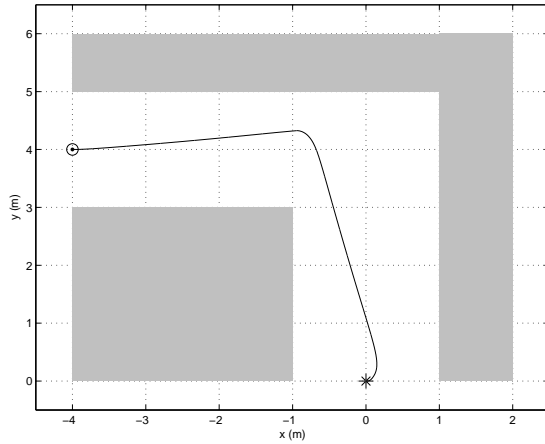

Fig. 8. Trajectory in the XY plane.

Fig. 8 shows the trajectory of the robot in the XY-plane, where the gray polygons stands for the walls of the corridor. It is straightforward to note that the configuration space is non-convex which, for an optimization problem such the one solved here, is unacceptable [16]. This problem can be overcome by splitting the configuration space in two convex half-spaces and applying a *via-point strategy*. Thus, different optimization problems are considered depending on the value of $x(k|k)$. In the case presented here, $x = -1$ defines the limit between the two half-spaces. For each half-space we consider the problem (7)-(11) with different state constraints and reference point $\mathbf{x}_r$ as follows:

$$\text{if } x(k|k) < -1 : \begin{cases} x(k+j|k) \leq 1, \\ 3 \leq y(k+j|k) \leq 5, \\ \mathbf{x}_r = [0 \ 4 \ 0]^T \end{cases}$$

$$\text{if } x(k|k) \geq -1 : \begin{cases} -1 \leq x(k+j|k) \leq 1, \\ y(k+j|k) \leq 5, \\ \mathbf{x}_r = [0 \ 0 \ 0]^T \end{cases}$$

where $\mathbf{x}_r$ is the reference point which the robot must converge to.

C. Comparison with Classical Approaches.

As pointed out in Section I, classical approaches to the control of nonholonomic WMRs include non-smooth and time-varying control laws. In this section a comparative analysis between the NMPC proposed here and some of these approaches are therefore carried out. The goal is once again the origin. For instance, let us consider the time-varying control of [3] and the non-smooth control of [4]. Thus, with a initial condition of $\mathbf{x}_0 = [-1 \ 3 \ 0]^T$ and considering control constraints (12) in the NMPC, we have the results in Figures 9-10, where the continuous line stands for the NMPC, the dashed line stands for the time-varying control law and the dash-dotted line stands for the non-smooth control law.
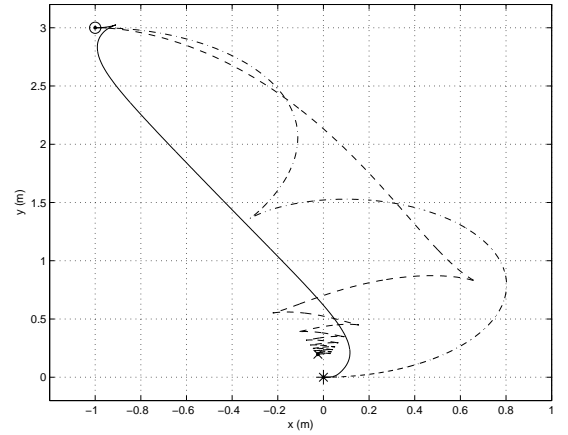

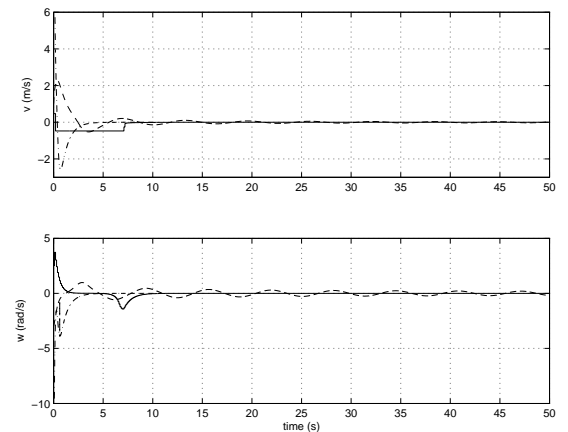Fig. 9. Trajectory in the XY plane.

Fig. 10. Control inputs.

It is straightforward to note the typical features of the time-varying control law: low convergence rate and oscillatory movements. With the use of the non-smooth control, these problems can be avoided. However, the performance achieved with NMPC is even better. Furthermore, note in Fig. 10 that the

classical control laws violate the amplitude control constraints, while with the NMPC proposed here they are respected.

VI. COMPUTATIONAL EFFORT

The use of MPC for real-time control of systems with fast dynamics such as a WMR has been hindered for some time due to its numerical intensive nature. However, with the development of increasingly faster processors the use of MPC in demanding applications can be possible.

In order to evaluate the real-time implementability of the proposed MPC, we consider, as measurement criterion, the number of floating point operations per second (flops). With this aim we consider that the computations run in an Athlon XP 2600+ which is able to perform a peak performance between 576 and 1100 Mflops accordingly to [17], a de-facto standard for floating point performance measurement.

The data presented in Table I refers to the mean value of Mflops along the developed trajectory for the three examples above for a sampling period of $T = 100\ ms$. *Case 1* refers to the NMPC with the original cost function in cartesian coordinates (6); *Case 2* is the NMPC with the cost function (13) of [11]; and *Case 3* is the NMPC with the proposed cost function (14), in polar coordinates.

TABLE I
COMPUTATIONAL EFFORT.

| Horizon | Mflops | | |
	Case 1	Case 2	Case 3
5	6.41	14	8.17
10	114	384	128
12	246	926	434
15	626	6077	1392

The data in Table I provides enough evidence that a standard of-the-shelf computer is able to run a MPC-based controller for a WMR. Note that for $N = 5$ all of the examples are feasible in real-time. However, results presented in Section IV-A show the existence of steady-state error and the results in Section IV-B show poor performance. Thus, among all of them, the approach that performs the best results with respect to state and control trajectories and computational effort is the one proposed in Section V, the NMCP with cost function in polar coordinates.

VII. CONCLUSION

This paper has presented an application of model predictive control to the problem of point stabilization of a nonholonomic wheeled mobile robot. A steady-state error in one of the position variable has been identified and a transformation of the configuration variables into polar coordinates has been proposed and successfully solved the problem. Through some examples, it was shown an important advantage of MPC: it can handle constraints in a straightforward way. The obtained control signals were such that the constraints imposed on the control variables were respected.

Furthermore, considerations regarding the computational effort of the MPC were developed with the purpose of speculating the viability of the proposed technique in a real implementation. It has been shown that with the proposed technique the NMPC can be potentially applied to WMRs in real-time.

REFERENCES

[1] R. W. Brockett, *New directions in applied mathematics.* Springer-Verlag, New York, 1982.

[2] A. M. Bloch and N. H. McClamroch, "Control of mechanical systems with classical nonholonomic constraints," in *Conference on Decision and Control*, Tampa, Florida, 1989, pp. 201–205.

[3] C. Samson and K. Ait-Abderrahim, "Feedback control of a nonholonomic wheeled cart in cartesian space," in *IEEE International Conference on Robotics and Automation*, Sacramento, California, 1991, pp. 1136–1141.

[4] C. Canudas de Wit and O. J. Sørdalen, "Exponential stabilization of mobile robots with nonholonomic constraints," *IEEE Transactions on Automatic Control*, vol. 37(11), pp. 1791–1797, Nov. 1992.

[5] Y. Yamamoto and X. Yun, "Coordinating locomotion and manipulation of a mobile manipulator," *IEEE Transactions on Automatic Control*, vol. 39(6), pp. 1326–1332, Jun. 1994.

[6] R. T. McCloskey and R. M. Murray, "Exponential stabilization of driftless control systems using homogeneous feedback," *IEEE Transactions on Automatic Control*, vol. 42(5), pp. 614–628, May. 1997.

[7] A. Ollero and O. Amidi, "Predictive path tracking of mobile robots. Application to the CMU Navlab," in *5th IEEE International Conference on Advanced Robotics*, Pisa, Italy, 1991, pp. 1081–1086.

[8] J. E. Normey-Rico, J. Gómez-Ortega, and E. F. Camacho, "A Smith-predictor-based generalised predictive controller for mobile robot path-tracking," *Control Engineering Practice*, vol. 7(6), pp. 729–740, Jun. 1999.

[9] J. Gómez-Ortega and E. F. Camacho, "Mobile robot navigation in a partially structured static environment using neural predictive control," *Control Engineering Practice*, vol. 4, no. 12, pp. 1669–1679, Dec. 1996.

[10] X. Yang, K. He, M. Guo, and B. Zhang, "An intelligent predictive control approach to path tracking problem of autonomous mobile robot," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, San Diego, CA, Oct. 1998.

[11] H. V. Essen and H. Nijmeijer, "Non-linear model predictive control of constrained mobile robots," in *European Control Conference*, Porto, Portugal, 2001, pp. 1157–1162.

[12] G. Campion, G. Bastin, and B. D'Andréa-Novel, "Structural properties and classification of kinematic and dynamical models of wheeled mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 1, pp. 47–62, Feb. 1996.

[13] W. F. Lages, "Control and estimation of the position and orientation of mobile robots (in portuguese)," Thesis (D.Sc.), Instituto Tecnológico de Aeronáutica, São José dos Campos, Brazil, 1998.

[14] F. Kühne, "Predictive control of nonholonomic mobile robots (in portuguese)," Dissertation (M.Sc.), Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil, Mar. 2005.

[15] W. F. Lages and E. F. Hemerly, "Smooth time-invariant control of wheeled mobile robots," in *Proceedings of the 13rd International Congress on Systems Science*, Wrocław, Poland, Set. 1998.

[16] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: stability and optimality," *Automatica*, vol. 36(6), pp. 789–814, Jun. 2000.

[17] A. Aburto, "FLOPS C program (double precision) V2.0 18 dec 1992," Dec. 1992, available at ftp://ftp.nosc.mil/pub/aburto/flops. Acessed in Jan. 2005.