2019 Latin American Robotics Symposium (LARS), 2019 Brazilian Symposium on Robotics (SBR) and 2019 Workshop on Robotics in Education (WRE)

# Model Predictive Control for Omnidirectional Small Size Robot with Motor and Non-Slipping Constraints

Felipe C. R. Pinheiro[1] , Marcos R. O. de A. Máximo[2] and Takashi Yoneyama[3]

***Abstract*— The Robocup Small Size Soccer League is a widely known Robotics league around the world. One important competition done in Robocup is the Small Size Soccer League (SSL). The path following problem is important to score goals in this competition. It is also necessary a speed control algorithm embedded in the low level. The teams mostly use embedded linear speed controllers. However, the robot has constraints not considered by linear control theory. The main goal of this paper is to derive a mathematical description of the robot's model and propose a Model Predictive Controller that considers these constraints, getting near to an optimal approach for the problem. The speed controller achieved shows a system that uses the maximum acceleration available throughout the transient. Besides this, the controller obtained slows down the system to avoid slippages. The slippages could drastically reduce the robot's performance when following paths. Finally, the linear physical model obtained for the constraints opens up opportunities for implementing faster algorithms using e.g. Multi Parametric Programming.**

## I. Introduction

The ITAndroids robotics team participates regularly in robotic related competitions. Our team started to partake in the Small Size Robot Soccer League, that organizes games between two teams of 6 robots (Div B), with a maximum size of 18 cm in diameter and 15 cm height, in a field of 9 m x 6 m. The winner is the team who makes more goals [8]. The physical setup provides a top view of the field via computer vision processing. With the preprocessed image, it is possible to recognize the poses of the teammates, as well as the adversary.

Our Small Size robot has 4 omnidirectional wheels [9]. The movement of the wheels is controlled by an embedded processor that commands the motors. Up to the date, the embedded processor used the PID algorithm. However, this algorithm does not consider the coupling between the outputs of the controlled system, and therefore does not have an optimal performance.

The team has experimented with model predictive control, such as [5] and [6] with promising results for Very Small Size robots, which have additional constraints when comparing to SS robots [7]. In the case of the standard small size robots, there are constraints in terms of the maximum voltage that can be applied to the motors. Besides this, slippages are undesirable so that the maximum acceleration demanded by the robot is also constrained. In fact, this constraint is associated with the maximum friction force between the wheel and the floor. Linear controllers, such as the traditional PID, are not able to reason about constraints. The main objective of this article is to propose a speed model predictive controller (MPC) in which the robot dynamics and the constraints are adequately represented and taken into account.

To the best of our knowledge, the main state of the art works related have not been considering non slipping constraints for omnidirectional four wheeled robots [1]. Other authors such as [2] and [3] attempted to solve related problems considering friction side effects. On their work, the MPC design was used to handle only voltage limit constraints. The system model considered includes dry and viscous friction. Whereas the article [4] presents the non-slipping constraints obtained for the omnidirectional robot system. However, the model obtained do not include the motor dynamics. Besides this, the constraints derived are of non-linear form, making infeasible the optimization problem embedded in the linear MPC quadratic programming algorithm. This article contributes with an alternative method to yield an optimization problem with linear constraints, that can be solved by means of linear MPC techniques. As the control system requires a fast response, non-linear MPC techniques would be inconvenient, since it has a larger computacional cost. Moreover, linear constraints facilitate the physical analysis.

The Section II presents the model of the system, the Section III shows the MPC methodology, the Section IV depicts results for the adopted case study, the Section V provides some conclusions and indicates some further works and finally the Section VI with acknowledgements. For this particular paper the system model derived can be extended to any four wheeled omnidirectional robot following the physical conditions described in the Section II.

## II. System Modelling

In a higher abstraction level, a robot is represented by a state vector which describes its position in the field, as well as its orientation, according to a chosen reference system. However, in a low abstraction level, a robot is viewed in physical terms, focusing in entities such as the input $\mathbf{u}$, the voltage applied to each one of the motors, and the output $\mathbf{y}$, the angular velocity of each wheel. This section is devoted to the construction of a mathematical description of the robot.

[1]Felipe Pinheiro is with Aeronautics Institute of Technology, São José dos Campos, DCTA, 50, fcrpfelipe96@gmail.com

[2]Marcos Máximo with the Autonomous Computational Systems Lab (LABSCA) , Aeronautics Institute of Technology, São José dos Campos, DCTA, 50, maximo.marcos@gmail.com

[3]Takashi Yoneyama with the Control and Systems Department , Aeronautics Institute of Technology, São José dos Campos, DCTA, 50, takashi@ita.br

978-1-7281-4268-5/19/$31.00 ©2019 IEEE
DOI 10.1109/LARS-SBR-WRE48964.2019.00011

### A. Conventions

In the global reference frame, $\{x_g, y_g\}$, the robot is represented by the vector of states $\mathbf{x} = [x\ y\ \psi]^T$, in which $x$ and $y$ are cartesian coordinates of a point in the robot (typically, the point is the intersection between the lines along the wheel's axes) and $\psi$ is the robot's orientation taking as the reference the robot's front side, defined by the presence of devices: *kicker*, *chipper* and *dribbler*. Besides, a coordinate system $\{x_r, y_r\}$ attached to the robot with its origin at the robot's reference point and rotated in a way that the axis $x_r$ is aligned with the front [9]. With this setup, $v$, $v_n$ and $\omega$ are considered the front, side and angular speeds, respectively. The conventions described so far are depicted in the Figure 1.

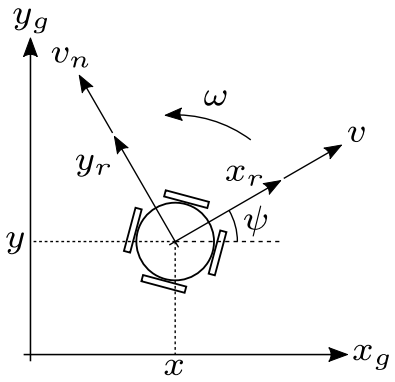


Fig. 1. Conventions for other Cartesian coordinate systems.

With respect to the wheels, the following conventions, schematically shown in the Figure 2, is adopted: the wheels are numbered in a anticlockwise way, starting with the axis $x_r$ and assigning the number 1 to the first wheel. The linear velocities convention follows the same rationale. Finally, the convention for the angular velocity is done in a way that is consistent with the corresponding linear velocity, i.e. $v_i = \omega_i r_i$ and to the i$^{th}$ wheel, in which $v_i$, $\omega_i$ and $r_i$ represent the wheel's linear velocity, the wheel's angular velocity and the i$^{th}$ wheel's radius, respectively.

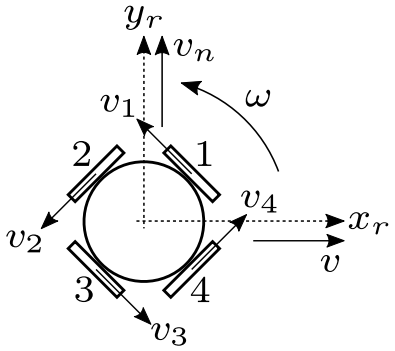


Fig. 2. Wheels' conventions.

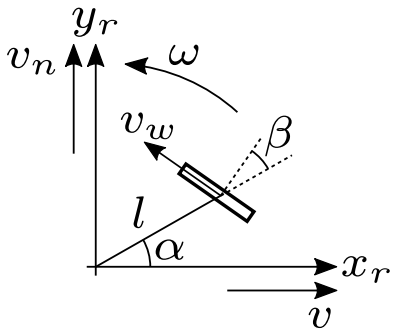


Fig. 3. Non-slipping condition to omnidirectional wheel.

### B. Kinematic constraints

It is important to ensure that the robot is not slipping. As the robot's wheels are omnidirectional, they do not impose any constraint in the normal direction, in which the *rollers* are free to spin. However, in the tangential direction, for non-slipping purposes, there should not exist any relative velocity between the wheel and the ground in the contact point.

Consider an omnidirectional wheel with a distance $l$ from the reference, and positioned according to the angles $\alpha$ and $\beta$, as shown in Figure 3. In that case, if the reference point is the intersection between the lines along the wheels' axes (if it exists), then $\beta = 0$ to any wheel. Given $v$ and $v_n$, the linear speed on the origin, in the directions $x_r$ and $y_r$, respectively, the non slip condition on the wheel can be written as [10]:

$$v_w = \omega_w r = -\sin(\alpha+\beta)\, v + \cos(\alpha+\beta)\, v_n + l\cos(\beta)\, \omega \tag{1}$$

in which $v_w$ and $\omega_w$ are the linear and angular speeds of the wheel, respectively. In the case of our robot, there are 4 wheels, thus the non-slipping constraints can be represented as:

$$\underbrace{\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}}_{\mathbf{v}_w} = \begin{bmatrix} \omega_1 r_1 \\ \omega_2 r_2 \\ \omega_3 r_3 \\ \omega_4 r_4 \end{bmatrix}$$

$$= \underbrace{\begin{bmatrix} -\sin(\alpha_1+\beta_1) & \cos(\alpha_1+\beta_1) & l_1\cos(\beta_1) \\ -\sin(\alpha_2+\beta_2) & \cos(\alpha_2+\beta_2) & l_2\cos(\beta_2) \\ -\sin(\alpha_3+\beta_3) & \cos(\alpha_3+\beta_3) & l_3\cos(\beta_3) \\ -\sin(\alpha_4+\beta_4) & \cos(\alpha_4+\beta_4) & l_4\cos(\beta_4) \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{bmatrix} v \\ v_n \\ \omega \end{bmatrix}}_{\mathbf{v}_r} \tag{2}$$

in which the matrix $\mathbf{M}$ maps the velocity (omnidirectional) of the robot in wheels' speed. The inverse mapping is sometimes required. As the matrix $\mathbf{M}$ is not square, the inverse mapping can be obtained by means of the Moore-Penrose's pseudo-inverse [13]:

$$\mathbf{v}_r = \mathbf{M}^{+}\mathbf{v}_w. \tag{3}$$

Therefore, regardless of the configuration of the wheels, it is always possible to determine the mappings between the robot's velocity and wheels' velocities. In the case of our robot, the wheels' axes intersect in a single point and all the wheels have the same radius $r$ and dist $l$ from the origin. The Table I summarizes the wheels' angle parameters for the our robot.

TABLE I
ANGLE PARAMETERS FOR OUR ROBOT.

| Wheel | $\alpha$ | $\beta$ | $\sin(\alpha+\beta)$ | $\cos(\alpha+\beta)$ | $\cos\beta$ |
|---|---|---|---|---|---|
| 1 | 45° | 0 | $\sqrt{2}/2$ | $\sqrt{2}/2$ | 0 |
| 2 | 135° | 0 | $\sqrt{2}/2$ | $-\sqrt{2}/2$ | 0 |
| 3 | 225° | 0 | $-\sqrt{2}/2$ | $-\sqrt{2}/2$ | 0 |
| 4 | 315° | 0 | $-\sqrt{2}/2$ | $\sqrt{2}/2$ | 0 |

Therefore, the coordinate mappings for our robot are:

$$\begin{bmatrix} v \\ v_n \\ \omega \end{bmatrix} = r \begin{bmatrix} -\frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} \\ \frac{1}{4l} & \frac{1}{4l} & \frac{1}{4l} & \frac{1}{4l} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} \tag{4}$$

### C. Dynamic Equations

For deriving the dynamic equations, the Lagrangian Mechanics formulation is used. In the derivation considered, we assume that the non slipping conditions are satisfied. The rotation angle of the wheels, i.e. $\mathbf{q} = [\theta_1\ \theta_2\ \theta_3\ \theta_4]^T$, are chosen as generalized coordinates and the Lagrange equations become:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial L}{\partial \mathbf{q}} = \tau \tag{5}$$

in which $L$ is the system's lagrangian

$$L = T - U, \tag{6}$$

and $\tau$ is the generalized force vector. In the expression (6), $T$ and $U$ are the system kinetic and potential energy. In the present case, the single potential energy is due to the gravity, but as the robot moves on a plane, it is possible to choose the potential's reference in a way that $U = 0$. On the other hand, the kinetic energy is given by:

$$T = \frac{1}{2} m \mathbf{v}_{CoM}^T \mathbf{v}_{CoM} + \frac{1}{2} I_{CoM} \omega^2, \tag{7}$$

in which $m$ is the robot's mass, $I_{CoM}$ is the inertia with the center of mass (CoM) of the robot as a reference and $\mathbf{v}_{CoM}$ is the robot's CoM speed. In the case in which the CoM coincides with the reference point ($x_{CoM} = y_{CoM} = 0$), one has that:

$$\underbrace{\begin{bmatrix} I_d & I_{wc} & I_{sc} & I_{wc} \\ I_{wc} & I_d & I_{wc} & I_{sc} \\ I_{sc} & I_{wc} & I_d & I_{wc} \\ I_{wc} & I_{sc} & I_{wc} & I_d \end{bmatrix}}_{\mathbf{H}_\tau} \underbrace{\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \\ \dot{\omega}_4 \end{bmatrix}}_{\dot{\mathbf{v}}_w} = \underbrace{\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix}}_{\tau} \tag{8}$$

in which:

$$I_d = \frac{mr^2}{4} + \frac{I_{CoM} r^2}{16 l^2} \tag{9}$$

$$I_{wc} = \frac{I_{CoM} r^2}{16 l^2} \tag{10}$$

$$I_{sc} = \frac{I_{CoM} r^2}{16 l^2} - \frac{mr^2}{4} \tag{11}$$

### D. Inclusion of the motor and wheel models

The kinetic energies from the motor and wheel were not included in the Lagrangian, so that it is necessary to discount them from $\tau$, the momentum necessary to accelerate the motor and wheel. For simplicity, the following analysis is carried out for the $i^{th}$ wheel to be generalized later. The torque generated by the motor $i$ to accelerate the motor's rotor and the wheel, denoted $\tau$, satisfies:

$$\tau_e = J_w \dot{\omega}_i + B_w \omega_i + \tau_i \tag{12}$$

$$\tau_e = N \eta \tau_g \tag{13}$$

$$\tau_g = K j_i - J_m N \dot{\omega}_i - B_m N \dot{\omega}_i \tag{14}$$

in which $\tau_g$ and $\tau_e$ are the torques before and after the reduction, respectively; $J_m$ and $J_w$ are the inertia from the motor and wheel, respectively; $B_m$ and $B_w$ are the viscous friction coefficients from the motor and wheel, respectively; $K$ is the motor torque constant; $N$ is the reduction factor; and $\eta$ is the efficiency of the reduction system. Isolating $\tau_i$ as a function of all these terms, we obtain:

$$\begin{aligned} \tau_i &= N\eta \left(K j_i - J_m N \dot{\omega}_i - B_m N \dot{\omega}_i\right) - J_w \dot{\omega}_i - B_w \omega_i \Rightarrow \\ \tau_i &= N\eta K j_i - \underbrace{\left(N^2 \eta J_m + J_w\right)}_{J_{eq}} \dot{\omega}_i - \underbrace{\left(N^2 \eta B_m + B_w\right)}_{B_{eq}} \omega_i \Rightarrow \\ \tau_i &= N\eta K j_i - J_{eq} \dot{\omega}_i - B_{eq} \omega_i \end{aligned} \tag{15}$$

Substitution of (15) into (8) yields:

$$\underbrace{\left(\mathbf{H}_\tau + J_{eq} \mathbf{I}_4\right)}_{\mathbf{H}_j} \underbrace{\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \\ \dot{\omega}_4 \end{bmatrix}}_{\dot{\mathbf{v}}_w} + \underbrace{B_{eq} \mathbf{I}_4}_{\mathbf{C}_j} \underbrace{\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix}}_{\mathbf{v}_w} = \underbrace{N\eta K}_{k_j} \underbrace{\begin{bmatrix} j_1 \\ j_2 \\ j_3 \\ j_4 \end{bmatrix}}_{\mathbf{j}}, \tag{16}$$

where $\mathbf{I}_4$ is the 4x4 identity matrix.

In this case, observe that neglecting the couplings, the global inertia (seen from the output) that the motor must accelerate is

$$I'_d = \frac{mr^2}{4} + \frac{I_{CoM} r^2}{16 l^2} + N^2 \eta J_m + J_w \tag{17}$$

Equation (16) represents the robot's dynamics, having the motor current as input. The motor current is a function of the voltage on the motor terminals $V$, the back electromotive

force $V_{backemf}$ and the motor resistance $R$. The inductance effects of the motor winding are neglected, since this value is usually very low. The model for the motor is thus given by

$$j_i = \frac{V_i - V_{backemf}}{R} = \frac{V_i - KN\omega_i}{R}, \tag{18}$$

and substituting (18) in (16):

$$\mathbf{H}_j \dot{\mathbf{v}}_w + \mathbf{C}_j \mathbf{v}_w = \frac{N\eta K}{R} \begin{bmatrix} V_1 - KN\omega_1 \\ V_2 - KN\omega_2 \\ V_3 - KN\omega_3 \\ V_4 - KN\omega_4 \end{bmatrix} \tag{19}$$

The equations can now be grouped as

$$\underbrace{\mathbf{H}_j}_{\mathbf{H}_V} \underbrace{\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \\ \dot{\omega}_4 \end{bmatrix}}_{\dot{\mathbf{v}}_w} + \underbrace{\left( B_{eq} + \frac{N^2\eta K^2}{R} \right) \mathbf{I}_4}_{\mathbf{C}_V} \underbrace{\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix}}_{\mathbf{v}_w} = \underbrace{\frac{N\eta K}{R}}_{k} \underbrace{\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix}}_{\mathbf{u}} \tag{20}$$

which can be rewritten in a compact form

$$\begin{aligned} \mathbf{H}\dot{\mathbf{v}}_w + \mathbf{C}\mathbf{v}_w = k\mathbf{u} \Rightarrow \mathbf{H}\dot{\mathbf{v}}_w = -\mathbf{C}\mathbf{v}_w + k\mathbf{u} \Rightarrow \\ \dot{\mathbf{v}}_w = \underbrace{-\mathbf{H}^{-1}\mathbf{C}}_{\mathbf{A}}\mathbf{v}_w + \underbrace{k\mathbf{H}^{-1}}_{\mathbf{B}}\mathbf{u} \Rightarrow \dot{\mathbf{v}}_w = \mathbf{A}\mathbf{v}_w + \mathbf{B}\mathbf{u}. \end{aligned} \tag{21}$$

### *E. Constraints*

The controlled system has a first constraint on the manipulated variable, in this case, the voltage applied to the motors' terminals. This voltage can not surpass the maximum voltage supplied $V_m$, i.e.,

$$\begin{bmatrix} -V_m \\ -V_m \\ -V_m \\ -V_m \end{bmatrix} \leq \mathbf{u} \leq \begin{bmatrix} V_m \\ V_m \\ V_m \\ V_m \end{bmatrix}. \tag{22}$$

In order to avoid slipping, the maximum torque supplied by the motor to the robot is constrained by the maximum friction torque $\tau_{max}$, directly related to the maximum friction force given by

$$f_{max} = \mu n, \tag{23}$$

in which $\mu$ is the friction coefficient between the wheel and ground, and $n$ the normal force between the wheel and ground. A relation between the friction forces under the 4 wheels and the normal force generated by the contact of the robot with the ground can be found in [4]

$$\underbrace{\begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{bmatrix}}_{\mathbf{n}} = \underbrace{\frac{h}{2l} \begin{bmatrix} 0 & 1 & 0 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ 1 & 0 & -1 & 0 \end{bmatrix}}_{\mathbf{N}} \underbrace{\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}}_{\mathbf{f}} + \underbrace{\frac{1}{4} \begin{bmatrix} mg \\ mg \\ mg \\ mg \end{bmatrix}}_{\mathbf{M}}, \tag{24}$$

in which $h$ is the CoM height above the ground. The direction of each friction force $f_i$ of each wheel is considered parallel to $v_i$.

The friction forces are also related to the acceleration parallel to $v$, $v_n$ and $\omega$. If the robot's CoM is in the geometric center of the robot, the friction forces will have a distance of $l$ to the CoM. Summing all the friction components in $x$, $y$ and the torques in the $z$ axis, the following equations are obtained

$$\begin{bmatrix} a \\ a_n \\ \alpha \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{\sqrt{2}}{2m} & -\frac{\sqrt{2}}{2m} & -\frac{\sqrt{2}}{2m} & \frac{\sqrt{2}}{2m} \\ -\frac{\sqrt{2}}{2m} & -\frac{\sqrt{2}}{2m} & \frac{\sqrt{2}}{2m} & \frac{\sqrt{2}}{2m} \\ \frac{l}{I_z} & \frac{l}{I_z} & \frac{l}{I_z} & \frac{l}{I_z} \end{bmatrix}}_{\mathbf{O}} \mathbf{f}, \tag{25}$$

in which $I_z$ is the robot's moment of inertia in the $z$ axis. Substituting (25) and (21) in (4),

$$\begin{aligned} \mathbf{f} = \mathbf{O}^+ r\mathbf{M}\dot{\mathbf{v}}_\omega \Rightarrow \\ \mathbf{f} = \underbrace{\mathbf{O}^+ r\mathbf{M}\mathbf{A}}_{\mathbf{F}}\mathbf{v}_\omega + \underbrace{\mathbf{O}^+ r\mathbf{M}\mathbf{B}}_{\mathbf{E}}\mathbf{u}. \end{aligned} \tag{26}$$

Substituting (26) and (24) into (23), we obtain

$$\begin{aligned} \mathbf{f} \leq \mu\mathbf{n} \Rightarrow \\ \mathbf{f} \leq \mu\mathbf{N}\mathbf{f} + \mu\mathbf{M} \Rightarrow \\ (\mathbf{I}_4 - \mu\mathbf{N})(\mathbf{E}\mathbf{u} + \mathbf{F}\mathbf{v}_w) \leq \mu\mathbf{M}, \end{aligned} \tag{27}$$

which is also valid for negative friction forces like in

$$\begin{aligned} \mathbf{f} \geq -\mu\mathbf{n} \Rightarrow \\ (\mathbf{I}_4 + \mu\mathbf{N})(\mathbf{E}\mathbf{u} + \mathbf{F}\mathbf{v}_w) \geq -\mu\mathbf{M}. \end{aligned} \tag{28}$$

It is important to notice that the constraints in (27), (28) and (22) are linear.

### *F. Robot parameters*

The parameters used in the equations of the model can be found in the Table II.

TABLE II
ROBOT PARAMETERS.

| | | | |
|---|---|---|---|
| $m$ | 0.71 kg | $r$ | 25 mm |
| $I_{CoM}$ | 0.0092 kgm$^2$ | $J_m$ | 9.25 kgmm$^2$ |
| $l$ | 85 mm | $B_m$ | 8.12 kg mm$^2$/s |
| $J_\omega$ | 32.8 kg mm$^2$ | $B_\omega$ | 0.0467 kg mm$^2$/s |
| $N$ | 3 | $\eta$ | 0.94 |
| $K$ | 25.5 N mm/A | $R$ | 1.24 Ω |
| $\mu$ | 0.60 | | |

## III. MODEL BASED PREDICTIVE CONTROL

In order to optimize a cost function while inequality constrains are satisfied, the idea is to use MPC. Since the constraints involved in this problem are linear, a classical MPC using quadratic programming [11] [12] suffices.

The discretized linear state equation are obtained by approximating the original model equations in 21 by keeping the first order terms of the Taylor expansion to yield

$$\begin{aligned} \mathbf{x}_{k+1} = \mathbf{A}_d\mathbf{x}_k + \mathbf{B}_d\mathbf{u}_k \\ \mathbf{y}_k = \mathbf{C}_d\mathbf{x}_k. \end{aligned} \tag{29}$$

The state disturbances $\mathbf{p}$ are accommodated using an expanded state representation

$$\mathbf{x}_k = \begin{bmatrix} (\mathbf{v}_w)_k \\ (\mathbf{p})_k \end{bmatrix}, \tag{30}$$

in which $T$ is the closed loop sample time. Hereafter, $kT$ will be replaced by $k$, if the meaning is clear from the context. The sample time, $T = 0.01s$ is the one currently used in the actual robot. For the predictions, the state space equations are

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k} &= \mathbf{A}_d \mathbf{x}_k + \mathbf{B}_d \hat{\mathbf{u}}_{k|k} \\ \hat{\mathbf{x}}_{k+2|k} &= \mathbf{A}_d \hat{\mathbf{x}}_{k+1|k} + \mathbf{B}_d \hat{\mathbf{u}}_{k+1|k} \\ &\vdots \\ \hat{\mathbf{x}}_{k+N|k} &= \mathbf{A}_d \hat{\mathbf{x}}_{k+N-1|k} + \mathbf{B}_d \hat{\mathbf{u}}_{k+N-1|k}, \end{aligned} \tag{31}$$

in which $\mathbf{A}_d$ and $\mathbf{B}_d$ are the model matrices in the discrete state space. Considering the output prediction vector, given by $\hat{\mathbf{y}}_{k+i|k} = \mathbf{C}_d \hat{\mathbf{x}}_{k+i|k}$ the control input $\mathbf{u}_k$ is required to minimize the quadratic cost function $J(\hat{\mathbf{y}}, \hat{\mathbf{u}})$ subject to the linear inequalities dependent on the voltage and friction constraints. The cost function is, therefore, of quadratic form and weights the cost of deviating from the reference $\mathbf{r}$ against the cost of spending $\mathbf{u}$.

$$J(\hat{\mathbf{y}}, \hat{\mathbf{u}}) = (\hat{\mathbf{y}} - \mathbf{r})^T \mathbf{R}_{\text{qp}} (\hat{\mathbf{y}} - \mathbf{r}) + \hat{\mathbf{u}}^T \mathbf{Q}_{\text{qp}} \hat{\mathbf{u}}, \tag{32}$$

in which $\mathbf{R}_{\text{qp}}$ and $\mathbf{Q}_{\text{qp}}$ are matrices with values to be chosen by the designer. The complete problem is the following quadratic programming optimization problem

$$\begin{aligned} \min_{\hat{\mathbf{u}}} \quad & J(\hat{\mathbf{y}}, \hat{\mathbf{u}}) \\ \text{subject to} \quad & \mathbf{A}_{qp} \hat{\mathbf{u}} + \mathbf{C}_{qp} \hat{\mathbf{x}} \leq \mathbf{b}_{qp}. \end{aligned} \tag{33}$$

The constraints obtained in (22), (27) and (28) can be written in the matrix form $\mathbf{A}_{qp} \hat{\mathbf{u}} + \mathbf{C}_{qp} \hat{\mathbf{x}} \leq \mathbf{b}_{qp}$.

### A. State Observer

A state observer can be implemented following the predictor-corrector formulation. The prediction of "future" values of $\mathbf{x}$, given the control signals $\mathbf{u}$, are obtained using

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{A}_d \hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}_d \mathbf{u}_{k-1}, \tag{34}$$

followed by a posterior correction, given the measurement $\mathbf{y}_k$, as in

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{A}_d^{-1} \mathbf{L} (\mathbf{y}_k - \hat{\mathbf{y}}_k). \tag{35}$$

## IV. Results

For the simulations depicted in this session, the YALMIP [14] framework was used to write the optimization problem.

### A. System with no disturbance

For this first result, a fixed prediction horizon of $\mathcal{N} = 100$, and an time interval of $t = 2$ s was considered for controlling the wheels' speed. The reference angular speed on the wheels was set at $t = 0$ s to $\mathbf{r} = [50\ 25\ -25\ -50]^T$ rad/s. After $t = 0.5$ s, the reference angular speed was set to $\mathbf{r} = [25\ 12.5\ -12.5\ -25]^T$ rad/s. The Figure 4 shows the obtained step response in closed loop. The dashed lines shows the step response for a closed loop system without the friction constraints (27) and (28). The continuous lines show the step response for a closed loop system with the constraints.

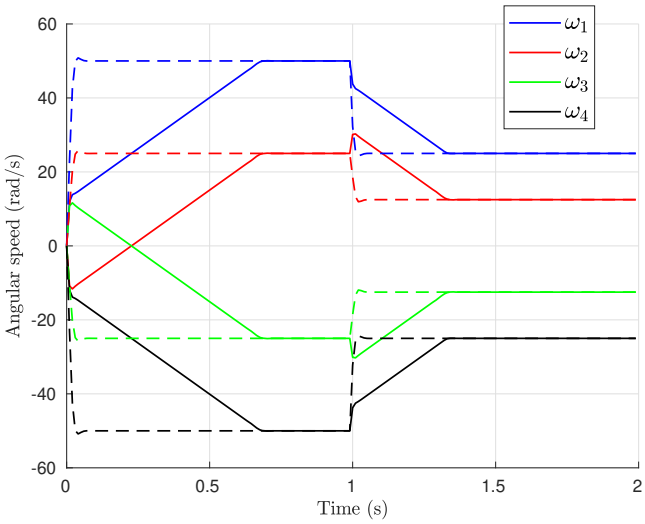


Fig. 4. Angular speed controlled.

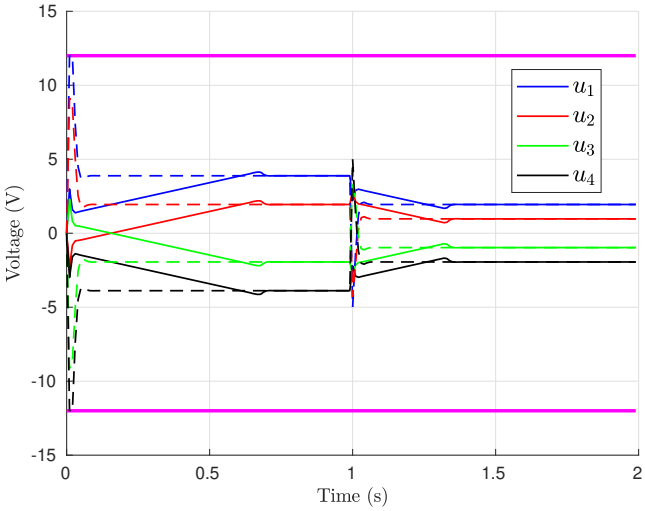


Fig. 5. Voltage applied to motors' terminals. The color magenta is representing the voltage limits.

Therefore, the friction constraints slows down the system response to avoid slippages. In the case of a real robot, fast control responses lead to slippages, which could drastically reduce the robot's performance when following paths. The Figure 5 shows the control signal variation with time to the corresponding step response, and the dashed plot shows the input for the system with no friction constraints. It is another evidence of the range of voltage variation limitation associated to the constraint.

### B. System with disturbance

For this case, a disturbance is included in the input signal and the same simulation time adopted is $t = 1$ s. The disturbance will be active up to $t < 0.5$ s and for the

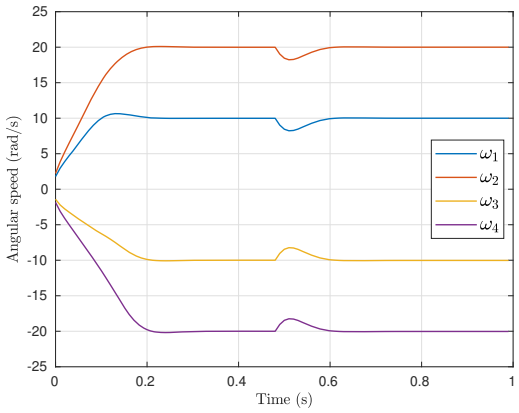


Fig. 6. Angular speed controlled.

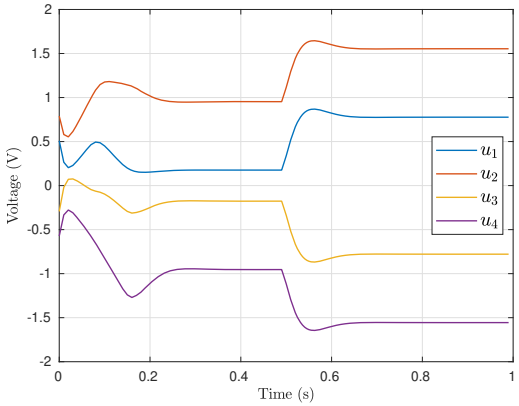


Fig. 7. Voltage applied to motors' terminals.

remaining time there will be no disturbance. Considering a constant disturbance in the input voltage as the vector $\mathbf{p}_k = [0.5\ 0.5\ -0.5\ -0.5]^T$,$t < 0.5$s, the result is shown in the figures 6 and 7. The state observer estimates the perturbation and the system can still converge to the reference speed used ($\mathbf{r} = [10\ 20\ -10\ -20]^T$ rad/s ). After the disturbance ceases the controller does not respond so fast, keeping the friction constraints satisfied.

## V. Conclusion and Further Work

### A. Conclusions

In this paper we contributed with a linear model for the robots and its physical constraints. This linear version of the model it's a cornerstone for implementing linear classical MPCs. Linear MPCs are more feasible in real time problems due to its reduced computation cost.

Although, the results show that under the presence of the constraints, the performance of the controller is reduced, but the non-slipping conditions will be satisfied. These conditions are important for trajectory planning, since the robot strategy could command the robot to go to a particular position, avoiding obstacle collisions during the path. Slippages could make the robot go out of the planned path.

### B. Further work

A real time implementation of the MPC obtained in this paper could be done in the future, using also Multi-Parametric Programming (MPP) techniques. It is possible to make a benchmarking between the two approaches, comparing computation costs and other trade-offs. Besides this, the current analysis was done under the presence of disturbances. We are not considering the effects of measurement noises and model mismatches. For the first case, a Kalman Filter [15] could be used to enhance the work. For the latter, linear matrix inequalities could be used to obtain a robust controller, reducing the side effects of mismatches, but eventually with more conservative performance.

## VI. Acknowledgements

We would like to acknowledge ITAndroids' sponsors and supporters Altium, Intel, ITAEx, Mathworks, Metinjo, Micropress, Polimold, Rapid, Solidworks, ST Microelectronics, Wildlife Studios and Virtual Pyxis. They provided fundamental support and resources to this paper.

## References

[1] Harasim, Patryk and Trojnacki, Maciej. (2016). State of the Art in Predictive Control of Wheeled Mobile Robots. Journal of Automation, Mobile Robotics and Intelligent Systems. 10. 34-42. 10.14313/JAMRIS 1-2016/5.

[2] A. G. S. Conceicao, C. E. T. Dorea and J. C. L. B. Sb., "Predictive Control of an Omnidirectional Mobile Robot with Friction Compensation," 2010 Latin American Robotics Symposium and Intelligent Robotics Meeting, Sao Bernardo do Campo, 2010, pp. 30-35. doi: 10.1109/LARS.2010.22

[3] J. C. L. Barreto S., A. G. S. Conceio, C. E. T. Drea, L. Martinez and E. R. de Pieri, "Design and Implementation of Model-Predictive Control With Friction Compensation on an Omnidirectional Mobile Robot," in IEEE/ASME Transactions on Mechatronics, vol. 19, no. 2, pp. 467-476, April 2014. doi: 10.1109/TMECH.2013.2243161

[4] O. Purwin and R. D'Andrea, "Trajectory generation for four wheeled omnidirectional vehicles," Proceedings of the 2005, American Control Conference, 2005., Portland, OR, USA, 2005, pp. 4979-4984 vol. 7. doi: 10.1109/ACC.2005.1470795

[5] Máximo, M.R.O.A. (2015). Model predictive controller for trajectory tracking by differential drive robot with actuation constraints. SBAI.

[6] Okuyama, I.F., Pinto, S.C., and Máximo, M.R.O.A. (2017). trajectory optimization for a differential drive soccer robot. SBAI.

[7] Okuyama, I. Franzoni, "Trajectory planning considering acceleration limits for an autonomous soccer player" ( Bachelor Thesis), 2017.

[8] Robocup, "Laws of the RoboCup Small Size League 2016".

[9] Aloysio L. , B. Mirahy , D. Barros, E. Moreira, J. do Nascimento and V. Loures, "ITAndroids Small Size Team Description Paper", Sistema Olimpo, 2018.

[10] Siegwart, R., Nourbakhsh, I.R., and Scaramuzza, D. (2011). Introduction to Autonomous Mobile Robots. The MIT Press, 2nd edition.

[11] Maciejowski, J.M. (2002). Predictive control with constraints. Prentice-Hall, Harlow

[12] Camacho, E.F. and Bordons, C. (2004). Model predictive control. Springer-Verlag, London, 2 edition.

[13] Penrose, Roger (1955). "A generalized inverse for matrices". Proceedings of the Cambridge Philosophical Society.

[14] Löfberg, J.,"YALMIP : A Toolbox for Modeling and Optimization in MATLAB", In Proceedings of the CACSD Conference. Taipei, Taiwan, 2004

[15] Hemerly, E.M. (2000). Controle por computador de sistemas dinâmicos. Edgard Blücher, São Paulo, 2 edition.