PREDICTIVE CONTROL OF DIFFERENTIAL DRIVE MOBILE ROBOT CONSIDERING DYNAMICS AND KINEMATICS

Rahul Sharma K.
Daniel Honc
František Dušek
Department of Process control
Faculty of Electrical Engineering and Informatics, University of Pardubice, Czech Republic
E-mail: rahul.sharma@student.upce.cz,{daniel.honc, frantisek.dusek}@upce.cz

KEYWORDS
Mobile robot, dynamic system modelling simulation, trajectory tracking, predictive control.

ABSTRACT

The paper deals with trajectory tracking of the differential drive robot with a mathematical model governing dynamics and kinematics. Motor dynamics and chassis dynamics are considered for deriving a linear state-space dynamic model. Basic nonlinear kinematic equations are linearized into a successively linearized state-space model. The dynamic and kinematic models are augmented to derive a single state-space linear model. The deviation variables are reference variables which are variables of an ideal robot following a reference trajectory which can be pre-calculated. Reference tracking is achieved by model predictive control of supply voltage of both the drive motors by considering constraints on controlled variables and manipulated variables. Simulation results are provided to demonstrate the performance of proposed control strategy in the MATLAB simulation environment.

INTRODUCTION

Trajectory tracking of mobile robots refers to mobile robot tracking in a predefined time-varying reference trajectory, which is one of the fundamental problems in motion control of mobile robots. In the case of differential drive robots, trajectory tracking has been well studied in the past. The most popular way of trajectory tracking is by considering a linearized dynamic error tracking model with feed forward inputs or a successively linearized model.

Model Predictive Control (MPC) is one of the most popular optimization control strategies in the process industries. It is designed to handle complex, constrained, multivariable control problems. It is an online optimization tool, which will generate optimal control actions required at every time instance minimizing an objective function based on predictions (Camacho and Alba 2004). With the increase in computational power, the MPC is not only limited to slow dynamics processes, where dynamical optimization is easily possible, but also there are new applications for faster systems. For example, MPC control techniques for trajectory tracking of mobile robots as can be seen in (Gu, D. and Hu 2006), (Kuhne et al. 2004) and (Lages et al. 2006). A review of motion control of Wheeled Mobile Robots (WMRs) using MPC can be found in (Kanjanawanishkul 2012). A mobile robot trajectory tracking problem with linear and nonlinear state-space MPC is presented in (Kuhne et al. 2005). An experimental overview of WMR is published in (De Luca et al 2001). Dynamic behavior of a differentially steered robot model, where the reference point can be chosen independently and gives us more general formulation, is published in (Dušek et al. 2011). In our previous work (Sharma et al. 2015), we proposed predictive control of the mobile robot, where the linear and angular velocities are optimally controlled by voltages to the drive motors with constraints on controlled variables, manipulated variable and states (current and wheel speed of the motors).

The most common way (for e.g. Maurovic et al. 2011) of trajectory tracking of mobile robots is by controlling the linear and angular velocities by some advanced controllers and then control the mobile robot's wheel speeds by low level controllers like a PID controller. In this paper, we firstly modelled dynamics of the differential drive robot considering motor dynamics and chassis dynamics. The nonlinear kinematics equations are linearized into a linear time-varying error based model by successive linearization, where state variables are deviations from reference variables. Reference variables are variables of the ideal robot which follows a time-varying reference trajectory. The dynamic and kinematic models are augmented into a discrete time-varying state-space model, whose control inputs are motor control variables and outputs are positions in x and y direction and orientation. Model predictive control is used for trajectory tracking simulation in the MATLAB environment by optimizing a quadratic cost function using quadratic programming.

The main advantage of our approach is that (in contrast to the commonly used WMRs models) we consider dynamics of motors as well, so the controller outputs are motor voltages and the robot can be tracked into a reference trajectory, respecting the physical constraints like currents. Since, in trajectory tracking problem, the future set-points are known, MPC is preferred when

Proceedings 30th European Conference on Modelling and Simulation ©ECMS Thorsten Claus, Frank Herrmann, Michael Manitz, Oliver Rose (Editors)
ISBN: 978-0-9932440-2-5 / ISBN: 978-0-9932440-3-2 (CD)

compared with other control methods and also because of the ability to handle soft and hard constraints.

MATHEMATICAL MODELLING

The differential drive mobile robot is assumed to have two wheels connected with DC series motors and firmly supported by a castor wheel (See figure 1 and 2).

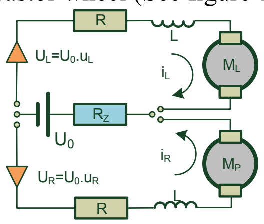

Figure 1: DC Motor Wiring

The mathematical model of the robot, consists of three relatively independent parts. The dynamics of the DC series motor, chassis dynamics (dependency between translational and rotational velocities of the chassis reference point on moments acting to driving wheels), and kinematics (influence of motor speed to translational and rotational velocities).

Dynamics of the mobile robot

The following derivation of the model representing dynamics of the differential drive mobile robot, closely follow the derivations in (Dušek et al. 2011), with some minor notation changes. The Dynamics of the series DC motor can be derived from balancing of voltages (Kirchhoff's law) and balancing of moments. From Kirchhoff's voltage law, we can derive,

$$Ri_L + R_z(i_L + i_R) + L\frac{di_L}{dt} = u_L U_0 - K\omega_L \qquad (1)$$

$$Ri_R + R_z(i_L + i_R) + L\frac{di_R}{dt} = u_R U_0 - K\omega_R \qquad (2)$$

where, K is the back EMF constant, ω_R and ω_L are the right and left motor speeds. u_R and u_L are the control voltages of the right and left motors respectively. All the other parameters are shown in figure 1.

By considering the balance of moments we can derive,

$$J\frac{d\omega_L}{dt} + k_r\omega_L + M_L = Ki_L \qquad (3)$$

$$J\frac{d\omega_R}{dt} + k_r\omega_R + M_R = Ki_R \qquad (4)$$

where J is the moment of inertia of the robot, k_r is the coefficient of rotational resistance. M_L and M_R are the load moments on left and right wheels respectively.

Chassis dynamics is defined with a vector of linear velocity v_B acting on a chassis reference point and with rotation of this vector of angular velocity ω_B (constant for all chassis points). The chassis reference point B is the point of the intersection of the axis joining the wheels and centre of gravity normal projection – see figure 2. Point T is the general centre of gravity – usually it is placed at the centre of the axis joining the wheels.

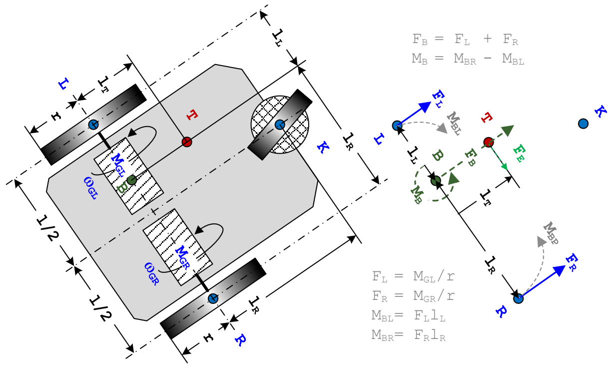

Figure 2: Chassis Scheme and Forces

The chassis dynamics can be expressed by balance of the forces and balance of the moment. Equation (5) is the result of applying balance of forces and Equation (6) from balance of moments.

$$\frac{p_G}{r}M_L + \frac{p_G}{r}M_R - k_v v_B - m\frac{dv_B}{dt} = 0$$

$$M_L + M_R - r_G k_v v_B - r_G m\frac{dv_B}{dt} = 0 \qquad (5)$$

$$-l_L\frac{p_G}{r}M_L + l_R\frac{p_G}{r}M_R - k_\omega\omega_B - \left(J_T + ml_T^2\right)\frac{d\omega_B}{dt} = 0$$

$$-l_L M_L + l_R M_R - k_\omega\omega_B - r_G J_B\frac{d\omega_B}{dt} = 0 \qquad (6)$$

where, p_G is the gear box transmission ratio, k_ω is the resistance coefficient against rotational motion. The rest of the parameters are shown in figure 2.The parameters r_G and J_B are described as,

$$r_G = \frac{r}{p_G} \quad ; \quad J_B = J_T + ml_T^2$$

From the theorems of similar triangles, depicted in figure 3, we can recalculate the peripheral velocities of the wheels v_L, v_R to the linear velocity v_B and angular velocity ω_B at point B as,

$$v_B = \frac{r_G}{l_L + l_R}\left(l_R\omega_L + l_L\omega_R\right) \qquad (7)$$

$$\omega_B = \frac{r_G}{l_L + l_R}\left(-\omega_L + \omega_R\right) \qquad (8)$$

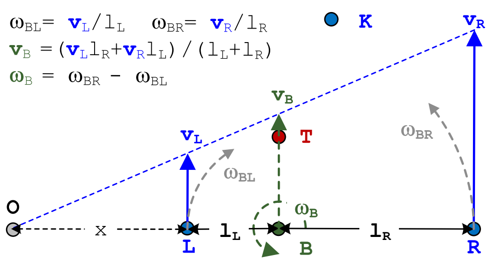

Figure 3: Linear and Angular Velocity Recalculation

These six differential Equations (1)-(6), and two algebraic Equations (7)-(8) containing eight state variables represent a mathematical description of the dynamic behaviour of ideal differentially steered mobile robots with losses linearly dependent on the revolutions or speed. Control signals, u_L and u_R, that control the supply voltages of the motors are input variables.

Calculation of steady-state values for constant engine power voltages are given below. A calculation of steady-state is useful both for the checking of derived equations and for the experimental determination of the values of the unknown parameters. Steady-state in matrix representation is,

$$\begin{bmatrix} R+R_z & R_z & K & 0 & 0 & 0 & 0 & 0 \\ R_z & R+R_z & 0 & K & 0 & 0 & 0 & 0 \\ K & 0 & -k_r & 0 & -1 & 0 & 0 & 0 \\ 0 & K & 0 & -k_r & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -r_G k_v & 0 \\ 0 & 0 & 0 & 0 & -l_L & l_R & 0 & -k_\omega \\ 0 & 0 & l_R & l_L & 0 & 0 & -\frac{l_R+l_L}{r_G} & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & -\frac{l_R+l_L}{r_G} \end{bmatrix} \begin{bmatrix} i_L \\ i_P \\ \omega_L \\ \omega_R \\ M_L \\ M_R \\ v_B \\ \omega_B \end{bmatrix} = \begin{bmatrix} U_L \\ U_R \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{9}$$

The Equation (1-8) can be reduced to a state-space model with four states by introducing the following parameters,

$$a_L = k_r + \frac{k_v l_R r_G^2}{l_L + l_R} \quad ; \quad a_R = k_r + \frac{k_v l_L r_G^2}{l_L + l_R}$$

$$b_L = J + \frac{m l_R r_G^2}{l_L + l_R} \quad ; \quad b_R = J + \frac{m l_L r_G^2}{l_L + l_R}$$

$$c_L = k_r l_L + \frac{k_\omega r_G^2}{l_L + l_R} \quad ; \quad c_R = k_r l_R + \frac{k_\omega r_G^2}{l_L + l_R}$$

$$d_L = J l_L + \frac{J_B r_G^2}{l_L + l_R} \quad ; \quad d_R = J l_R + \frac{J_B r_G^2}{l_L + l_R}$$

$$\begin{aligned} \frac{d\mathbf{x_D}}{dt} &= \mathbf{A_D x_D} + \mathbf{B_D u} \\ \mathbf{y_D} &= \mathbf{C_D x_D} \end{aligned} \tag{10}$$

where,

$$\mathbf{x_D} = \begin{bmatrix} i_L \\ i_R \\ \omega_L \\ \omega_R \end{bmatrix} \; ; \; \mathbf{u} = \begin{bmatrix} u_L \\ u_R \end{bmatrix} \; ; \; \mathbf{y_D} = \begin{bmatrix} v_B \\ \omega_B \end{bmatrix}$$

with matrices $\mathbf{A_D}$, $\mathbf{B_D}$ and $\mathbf{C_D}$ as,

$$\mathbf{A_D} = \begin{bmatrix} -\frac{R+R_z}{L} & -\frac{R_z}{L} & -\frac{K}{L} & 0 \\ -\frac{R_z}{L} & -\frac{R+R_z}{L} & 0 & -\frac{K}{L} \\ \frac{K(d_R + b_R l_L)}{b_L d_R + b_R d_L} & \frac{K(d_R - b_R l_R)}{b_L d_R + b_R d_L} & -\frac{d_R a_L + b_R c_L}{b_L d_R + b_R d_L} & -\frac{d_R a_R - b_P c_R}{b_L d_R + b_R d_L} \\ \frac{K(d_L - b_L l_L)}{b_L d_R + b_R d_L} & \frac{K(d_L + b_L l_R)}{b_L d_R + b_R d_L} & -\frac{d_L a_L - b_L c_L}{b_L d_R + b_R d_L} & -\frac{d_L a_R + b_L c_R}{b_L d_R + b_R d_L} \end{bmatrix}$$

$$\mathbf{B_D} = \begin{bmatrix} \frac{U_0}{L} & 0 \\ 0 & \frac{U_0}{L} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \; ; \; \mathbf{C_D} = \begin{bmatrix} 0 & 0 & \frac{l_R r_G}{l_L + l_R} & \frac{l_L r_G}{l_L + l_R} \\ 0 & 0 & -\frac{r_G}{l_L + l_R} & \frac{r_G}{l_L + l_R} \end{bmatrix}$$

Kinematics of the mobile robot

The following derivations closely follow (Kuhne et al. 2004), despite some notation changes which have been used. Let the global coordinates of the robot be (x_B, y_B), the orientation of the robot be α, and v_B , ω_B are the linear and angular velocities. The kinematic equations of the differential drive mobile robot is given by (Campion et al. 1996),

$$\begin{aligned} \frac{dx_B}{dt} &= v_B \cos\alpha \\ \frac{dy_B}{dt} &= v_B \sin\alpha \\ \frac{d\alpha}{dt} &= \omega_B \end{aligned} \tag{11}$$

This can be represented as a simple model,

$$\dot{\mathbf{x}}_\mathbf{B} = f(\mathbf{x_B}, \mathbf{u_B}) \tag{12}$$

where state variables $\mathbf{x_B}=[x_B \; y_B \; \alpha]^T$ and control inputs $\mathbf{u_B}=[v_B \; \omega_B]^T$.

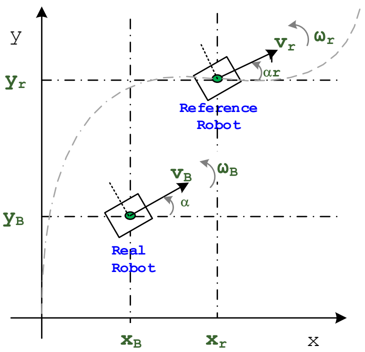

Figure 4: Coordinate System of Real Robot and Reference Robot

A linear model can be derived from the non-linear model, Equations (11), from an error model with respect to the reference robot (see figure 4). A reference robot can be considered as a robot with reference (desired) parameters of the robot to follow a trajectory which can be represented as,

$$\dot{\mathbf{x}}_r = f(\mathbf{x}_r, \mathbf{u}_r) \tag{13}$$

The reference parameters are $[x_r \; y_r \; \alpha_r \; v_r \; \omega_r]$. The linear velocity, orientation angle and angular velocity of the reference robot can be derived from Equation (11) as,

$$v_r(t) = \sqrt{\dot{x}_r(t)^2 + \dot{y}_r(t)^2} \tag{14}$$

$$\alpha_r(t) = arctan2(\dot{y}_r(t), \dot{x}_r(t)) \quad (15)$$

$$\omega_r(t) = \dot{\alpha}_r(t) = \frac{\dot{x}_r(t)\ddot{y}_r(t) - \dot{y}_r(t)\ddot{x}_r(t)}{\sqrt{\dot{x}_r(t)^2 + \dot{y}_r(t)^2}} \quad (16)$$

Applying the Taylor series approximation to Equation (12), around the reference points ($\mathbf{x_r}$, $\mathbf{u_r}$), we can derive,

$$\dot{\mathbf{x}} = f(\mathbf{x_r}, \mathbf{u_r}) + \left.\frac{\partial f(\mathbf{x},\mathbf{u})}{\partial \mathbf{x}}\right|_{\substack{\mathbf{x}=\mathbf{x_r} \\ \mathbf{u}=\mathbf{u_r}}} (\mathbf{x}_B - \mathbf{x}_r) + \left.\frac{\partial f(\mathbf{x},\mathbf{u})}{\partial \mathbf{u}}\right|_{\substack{\mathbf{x}=\mathbf{x_r} \\ \mathbf{u}=\mathbf{u_r}}} (\mathbf{u_B} - \mathbf{u_r})$$

$$\dot{\mathbf{x}} = f(\mathbf{x}_r, \mathbf{u}_r) + f_{\mathbf{x}_r}(\mathbf{x_B} - \mathbf{x}_r) + f_{\mathbf{u_r}}(\mathbf{u_B} - \mathbf{u}_r) \quad (17)$$

Subtracting Equation (17) from Equation (13) gives,

$$\dot{\bar{\mathbf{x}}} = f_{\mathbf{x}_r}\bar{\mathbf{x}} + f_{\mathbf{u_r}}\bar{\mathbf{u}} \quad (18)$$

$\bar{\mathbf{x}}$ is the error vector of state variables and $\bar{\mathbf{u}}$ is the error vector of control variables with respect to the reference robot. The approximation of $\dot{\bar{\mathbf{x}}}$ in Equation (18), by the forward differences gives the following discrete-time linear time-variant (LTV) state-space model:

$$\begin{aligned} \bar{\mathbf{x}}_\mathbf{K}(k+1) &= \bar{\mathbf{A}}_\mathbf{K}(k)\bar{\mathbf{x}}_\mathbf{K}(k) + \bar{\mathbf{B}}_\mathbf{K}(k)\bar{\mathbf{u}}_\mathbf{K}(k) \\ \bar{\mathbf{y}}_\mathbf{K}(k) &= \bar{\mathbf{C}}_\mathbf{K}\bar{\mathbf{x}}_\mathbf{K}(k) \end{aligned} \quad (19)$$

$$\bar{\mathbf{A}}_\mathbf{K}(k) = \begin{bmatrix} 1 & 0 & -v_r(k)sin\alpha_r(k)T \\ 0 & 1 & v_r(k)cos\,\alpha_r(k)T \\ 0 & 0 & 1 \end{bmatrix}$$

$$\bar{\mathbf{B}}_\mathbf{K}(k) = \begin{bmatrix} cos\,\alpha_r(k)T & 0 \\ sin\alpha_r(k)T & 0 \\ 0 & T \end{bmatrix}; \quad \bar{\mathbf{C}}_\mathbf{K} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\bar{\mathbf{x}}_\mathbf{K} = \begin{bmatrix} x_B(k) - x_r(k) \\ y_B(k) - y_r(k) \\ \alpha(k) - \alpha_r(k) \end{bmatrix}; \quad \bar{\mathbf{u}}_\mathbf{K} = \begin{bmatrix} v_B(k) - v_r(k) \\ \omega_B(k) - \omega_r(k) \end{bmatrix}$$

where T is the sampling period and $\bar{\mathbf{x}}_\mathbf{K}$ is deviation state vector which represents the error with respect to the reference robot, and $\bar{\mathbf{u}}_\mathbf{K}$ is associated with the control input. The reference values, v_r, α_r , ω_r are the reference linear velocity, orientation angle and angular velocity respectively which can be calculated from Equations (14-16).

Combined model – LTV

The kinematic model is linearized into a discrete error model. The dynamic model also has to be converted to a discrete error based model for augmenting with a kinematic model. Since the dynamic model is linear time invariant, the error model will be the same as that of Equation (10) but has to be discretized. Let the following be the discrete time state-space dynamic model.

$$\begin{aligned} \dot{\bar{\mathbf{x}}}_\mathbf{D}(k+1) &= \bar{\mathbf{A}}_\mathbf{D}(k)\bar{\mathbf{x}}_\mathbf{D}(k) + \bar{\mathbf{B}}_\mathbf{D}(k)\bar{\mathbf{u}}_\mathbf{D}(k) \\ \bar{\mathbf{y}}_\mathbf{D}(k) &= \bar{\mathbf{C}}_\mathbf{D}(k)\bar{\mathbf{x}}_\mathbf{D}(k) \end{aligned}$$

The matrices $\bar{\mathbf{A}}_\mathbf{D}$, $\bar{\mathbf{B}}_\mathbf{D}$ and $\bar{\mathbf{C}}_\mathbf{D}$ are discretized matrices of the dynamic model (Equation (10)). The state variables and control inputs are deviation variables from the reference points, $\mathbf{x}_{\mathbf{D\_r}}$ and $\mathbf{u}_{\mathbf{D\_r}}$, as,

$$\bar{\mathbf{x}}_\mathbf{D}(k) = \begin{bmatrix} i_L(k) - i_{L\_r}(k) \\ i_R(k) - i_{R\_r}(k) \\ \omega_L(k) - \omega_{L\_r}(k) \\ \omega_R(k) - \omega_{R\_r}(k) \end{bmatrix} \quad \bar{\mathbf{u}}_\mathbf{D}(k) = \begin{bmatrix} u_L(k) - u_{L\_r}(k) \\ u_R(k) - u_{R\_r}(k) \end{bmatrix}$$

This dynamics model and linearized kinematic time-variant model, Equation (19), can be augmented into a single state-space time-variant model with 9 states (currents, wheel speeds, linear and angular velocities and coordinates), two control variables (motor voltage control input) and three outputs (position in x and y direction and orientation measured from x direction).

$$\begin{aligned} \bar{\mathbf{x}}(k+1) &= \bar{\mathbf{A}}(k)\bar{\mathbf{x}}(k+1) + \bar{\mathbf{B}}(k)\bar{\mathbf{u}}(k) \\ \bar{\mathbf{y}}(k) &= \bar{\mathbf{C}}(k)\bar{\mathbf{x}}(k) \end{aligned} \quad (20)$$

where,

$$\bar{\mathbf{A}}(k) = \begin{bmatrix} \bar{\mathbf{A}}_{\mathbf{D}(4\times4)} & \mathbf{0}_{(4\times5)} & \\ \bar{\mathbf{C}}_{\mathbf{D}(2\times4)} & \mathbf{0}_{(2\times5)} & \\ \mathbf{0}_{(3\times4)} & \bar{\mathbf{B}}_{\mathbf{K}(3\times2)} & \bar{\mathbf{A}}_{\mathbf{K}(3\times3)} \end{bmatrix}; \quad \bar{\mathbf{B}}(k) = \begin{bmatrix} \bar{\mathbf{B}}_{\mathbf{D}(4\times2)} \\ \mathbf{0}_{(5\times2)} \end{bmatrix}$$

$$\bar{\mathbf{C}}(k) = \begin{bmatrix} \mathbf{0}_{(3\times6)} & \bar{\mathbf{C}}_{\mathbf{K}(3\times3)} \end{bmatrix}; \quad \bar{\mathbf{x}} = \begin{bmatrix} \bar{\mathbf{x}}_\mathbf{D} & \bar{\mathbf{u}}_\mathbf{K} & \bar{\mathbf{x}}_\mathbf{K} \end{bmatrix}^T; \quad \bar{\mathbf{u}} = \bar{\mathbf{u}}_\mathbf{D}$$

MODEL PREDICTIVE CONTROL

At each sampling time, the model predictive controller generates an optimal control sequence by optimizing a quadratic cost function. The first control action of this sequence is applied to the system. The optimization problem is solved again at the next sampling time using the updated process measurements and a shifted horizon. The cost function formulation depends on the control requirements. The most common cost function is in the form of,

$$J = \sum_{i=1}^{n_y} \sum_{j=1}^{N_2} r_i \left[\hat{y}_i(k+j) - w_i(k+j)\right]^2 + \sum_{i=1}^{n_u} \sum_{j=1}^{N_3} q_i \left[\Delta u_i(k+j-1)\right]^2 \quad (21)$$

Where $\hat{y}_i(k+j)$ is an optimum *j*-step ahead prediction of the system *i*-th output, N_2 is the control error horizon,

N_3 is the control horizon and $w_i(k+j)$ is the future set-point or reference for the i-th controlled variable. The parameters, r_i and q_i are the weighting coefficient for control errors and control increments respectively. $\Delta u_i(k+j-1)$ is the control increment of the i-th input. n_u and n_y are the number of inputs and number of outputs (manipulated and controlled variables).

The cost function consists of two parts, mainly costs due to control error during the control error horizon N_2 and costs to penalize the control signal increments during the control horizon N_3. For simplicity in the following text we consider, $N_2=N_3=N$.

A general discrete-time state-space model is given as,

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\,\mathbf{x}(k) + \mathbf{B}\,\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}\,\mathbf{x}(k) \end{aligned} \tag{22}$$

An incremental state-space model can also be used, if the model input is the control increment $\Delta\mathbf{u}(k)$ instead of $\mathbf{u}(k)$. $\Delta\mathbf{u}(k)=\mathbf{u}(k)-\mathbf{u}(k-1)$

$$\begin{aligned} \underbrace{\begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{u}(k) \end{bmatrix}}_{\mathbf{x}_p(k+1)} &= \underbrace{\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{bmatrix} \mathbf{x}(k) \\ \mathbf{u}(k-1) \end{bmatrix}}_{\mathbf{x}_p(k)} + \underbrace{\begin{bmatrix} \mathbf{B} \\ \mathbf{I} \end{bmatrix}}_{\mathbf{P}} \Delta\mathbf{u}(k) \\ \mathbf{y}(k) &= \underbrace{\begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix}}_{\mathbf{O}} \underbrace{\begin{bmatrix} \mathbf{x}(k) \\ \mathbf{u}(k-1) \end{bmatrix}}_{\mathbf{x}_p(k)} \end{aligned} \tag{23}$$

The predicted output representation of state-space model, in matrix form, is

$$\underbrace{\begin{bmatrix} \hat{\mathbf{y}}(k+1) \\ \hat{\mathbf{y}}(k+2) \\ \vdots \\ \hat{\mathbf{y}}(k+N) \end{bmatrix}}_{\mathbf{Y}} = \underbrace{\begin{bmatrix} \mathbf{OP} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{OMP} & \mathbf{OP} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{OM}^{N-1}\mathbf{P} & \mathbf{OM}^{N-2}\mathbf{P} & \cdots & \mathbf{OP} \end{bmatrix}}_{\mathbf{G}} \underbrace{\begin{bmatrix} \Delta\mathbf{u}(k) \\ \Delta\mathbf{u}(k+2) \\ \vdots \\ \Delta\mathbf{u}(k+N-1) \end{bmatrix}}_{\mathbf{U}} + \underbrace{\begin{bmatrix} \mathbf{OM} \\ \mathbf{OM}^2 \\ \vdots \\ \mathbf{OM}^N \end{bmatrix} \mathbf{x}_p}_{\mathbf{f}}$$

Which can be represented as sum of forced and free responses,

$$\mathbf{Y} = \underbrace{\mathbf{GU}}_{\text{forced response}} + \underbrace{\mathbf{f}}_{\text{free response}} \tag{24}$$

Cost function

The cost function in Equation (21) can be represented in matrix format as,

$$\mathbf{J} = (\mathbf{Y}-\mathbf{W})^{\mathrm{T}}\mathbf{R}(\mathbf{Y}-\mathbf{W}) + \mathbf{U}^{\mathrm{T}}\mathbf{QU} \tag{25}$$

where, $\mathbf{R}$ and $\mathbf{Q}$ are diagonal matrices with diagonal elements r_i and q_i respectively and $\mathbf{W}$ is a column vector of N future set points.

Constraints

In a long range predictive control, the controller has to anticipate constraint violation and correct control actions in an appropriate way. The input constraints are,

$$\begin{aligned} \mathbf{u}_{min} \le \mathbf{u}(i) \le \mathbf{u}_{max}, \quad & i \in \{k, k+N-1\} \\ \mathbf{y}_{min} \le \mathbf{y}(i) \le \mathbf{y}_{max}, \quad & i \in \{k+1, k+N\} \end{aligned} \tag{26}$$

The implementation of MPC with constraints involves the minimization of a quadratic cost function

$$J = \mathbf{U}^T \underbrace{\left(\mathbf{G}^T\mathbf{RG} + \mathbf{Q}\right)}_{\mathbf{H}} \mathbf{U} + 2\underbrace{(\mathbf{f}-\mathbf{W})^T\mathbf{RG}}_{\mathbf{g}^T}\mathbf{U} + \underbrace{(\mathbf{f}-\mathbf{W})^T\mathbf{R}(\mathbf{f}-\mathbf{W})}_{k}$$

Subject to the linear inequalities $\mathbf{AU} \le \mathbf{b}$, which is a Quadratic Programming (QP) problem. The QP problem can be solved e.g. using the function *quadprog* in MATLAB (Honc and Dušek 2013).

Predictive control of mobile robot

The augmented model is an error based model whose state variables are deviations from reference variables. The reference variables can be seen as an ideal robot following a time-varying reference trajectory. These reference velocities v_r, ω_r and orientation angle α_r can be calculated from Equations (14) to (16) from the reference inputs (positional coordinates of the robot - x_r, y_r). The other reference variables $\mathbf{x}_{D\_r}$ and $\mathbf{u}_{D\_r}$ can be pre-calculated from the model, Equation (10), by closed loop control with set-points (as previously calculated) v_r, ω_r and with an initial condition calculated from steady state Equation (9) . The trajectory tracking of the mobile robot is achieved by model predictive control with the linear time-variant model, Equation (20), with a cost function as in Equation (25) considering the constraints, Equation (26). At every time instance, the MPC algorithm will calculate the optimal control inputs (motor voltage control inputs - u_L and u_R). The overall control scheme is shown in figure 5.

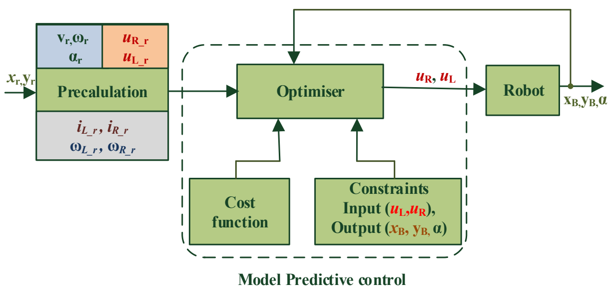

Figure 5: Overall Control Scheme

SIMULATION RESULTS

Chassis parameters and DC motor parameters were chosen as in (Dušek et al. 2011).These values are chosen so that they roughly correspond to the real physical values of the mobile robot. The reference trajectory chosen was an S-shaped trajectory as follows,

$$x_r = sin(2t)$$
$$y_r = 5\,sin\left(t + \frac{\Pi}{2}\right)$$

The mobile robot MPC was simulated in the MATLAB simulation environment with a sample time of 0.1 s and prediction horizon, N=5. The initial position of the robot was chosen to be the same as the reference trajectory points. The weighing matrices are chosen as,

$$\mathbf{Q} = diag(1,10,10)$$
$$\mathbf{R} = diag(10,10)$$

In figure 6, the simulated trajectory is compared with the desired (reference trajectory). The control inputs and reference (calculated by Equation 14 and 16) and simulated linear and angular velocities are shown in figure 7. Figure 8 depicts the wheel speeds and currents. Figure 9 shows the reference orientation (calculated by Equation (15)) and simulated orientation.

Constraints were applied to controlled variables (control voltages to right and left wheel). The constraints of control voltage of the motors were set to [0, 1] since the source voltage is 10 V and no backward motions of motor was assumed. The trajectory was chosen in such a way that we can see the response of the robot when a sudden change of position and orientation to the robot.

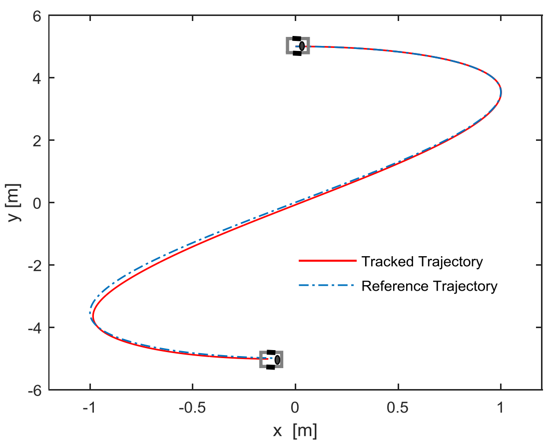

Figure 6: Trajectory Tracking

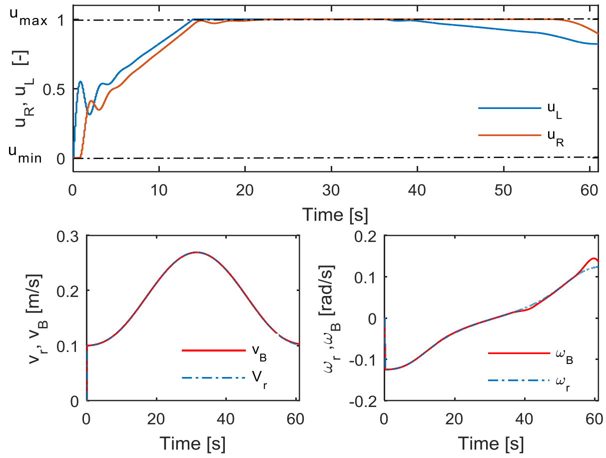

Figure 7: Control Inputs, Linear and Angular Velocity

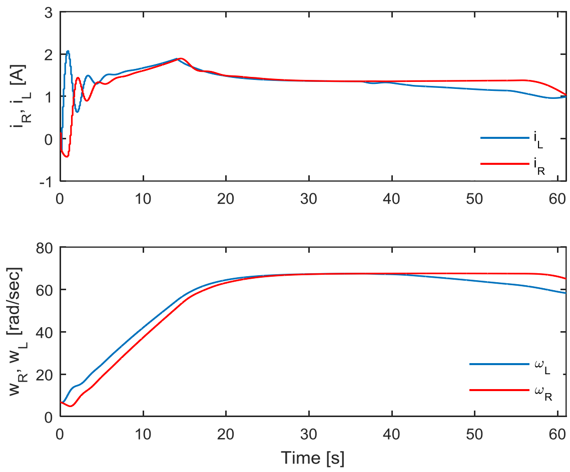

Figure 8: Currents and Wheel Speeds

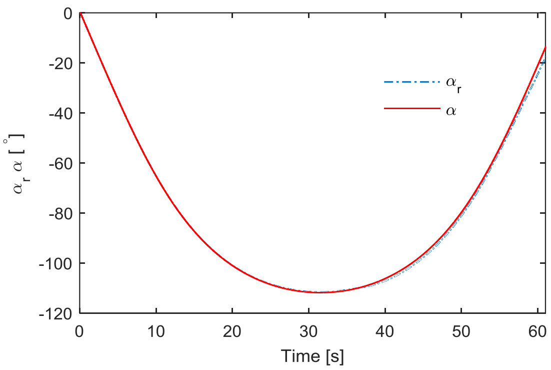

Figure 9: Reference and Simulated Orientation

Since the main objective of the paper was to model and simulate the response, efforts were not made in the control quality (e.g. constraints on state variables, tuning of weighing matrices, steady state error etc.). Control quality can be significantly improved by proper tuning of weighing matrices and/or by choosing an optimal horizon and/or by including a state observer etc.

CONCLUSION

In this paper, a linear time-variant model is derived by considering both the kinematics and dynamics of the mobile robot, which will allow trajectory tracking of mobile robot by controlling the control voltage to the motors. Constraints were considered only for the control variable.

As a future research direction, we are looking to incorporate other issues into our MPC formulation, such as including constraints on wheel speeds and currents, decreasing the computation time etc. Moreover, we expect to finish this controller implementation in a real robot and to conduct real experiments with the mobile robot in various environments. Path planning, obstacle avoidance etc., are other elements we wish to consider.

This research was supported by project SGS_2016_021, Mobile Robot Motion Control with Model Predictive Controller at FEI, University of Pardubice. This support is very gratefully acknowledged.

REFERENCES

Camacho, E.F. and Alba, C.B., 2013. *Model predictive control*. Springer Science & Business Media.

Gu, D. and Hu, H., 2006. Receding horizon tracking control of wheeled mobile robots. *Control Systems Technology, IEEE Transactions on*, *14*(4), pp.743-749.

Kuhne, F., Lages, W.F. and da Silva Jr, J.M.G., 2004. Model predictive control of a mobile robot using linearization. In *Proceedings of mechatronics and robotics* (pp. 525-530).

Lages, W.F. and Alves, J.A.V., 2006, September. Real-time control of a mobile robot using linearized model predictive control. In *Proc. of 4th IFAC Symposium on Mechatronic Systems* (pp. 968-973).

Kanjanawanishkul, K., 2012. Motion control of a wheeled mobile robot using model predictive control: A survey. *KKU Research Journal*, *17*(5), pp.811-837.

Künhe, F., Gomes, J. and Fetter, W., 2005, September. Mobile robot trajectory tracking using model predictive control. In *II IEEE latin-american robotics symposium*.

De Luca, A., Oriolo, G. and Vendittelli, M., 2001. Control of wheeled mobile robots: An experimental overview. In *Ramsete* (pp. 181-226). Springer Berlin Heidelberg.

Dušek, F., Honc, D. and Rozsíval, P., 2011. Mathematical model of differentially steered mobile robot. In *18th International Conference on Process Control, Tatranská Lomnica, Slovakia*.

Honc, D. and Dusek, F., 2013. State-Space Constrained Model Predictive Control. In *ECMS* (pp. 441-445).

Sharma, K.R., Honc, D. and Dušek, F., 2015. Model Predictive Control of Trajectory Tracking of Differentially Steered Mobile Robot. In *Intelligent Data Analysis and Applications* (pp. 85-95). Springer International Publishing.

Campion, G., Bastin, G. e D'Andréa-Novel B., 1996. Structural properties and classification of kinematic and dynamical models of wheeled mobile robots, *IEEE Transactions on Robotics and Automation 12 (1)*: pp.47–62

Maurovic, I., Baotic, M. and Petrovic, I., 2011, July. Explicit model predictive control for trajectory tracking with mobile robots. In *Advanced Intelligent Mechatronics (AIM), 2011 IEEE/ASME International Conference on* (pp. 712-717). IEEE.

AUTHOR BIOGRAPHIES

RAHUL SHARMA K., was born in Kochi, India and went to the Amrita University, where he studied electrical engineering and obtained his M.Tech degree in 2013. He is now doing his Ph.D. studies at the Department of process control, Faculty of Electrical and Informatics, University of Pardubice, Czech Republic. e-mail: `rahul.sharma @student. upce.cz`

DANIEL HONC was born in Pardubice, Czech Republic and studied at the University of Pardubice in the field of Process Control and obtained his Ph.D. degree in 2002. He is the head of the Department of Process Control at the Faculty of Electrical Engineering and Informatics. e-mail: `daniel.honc@upce.cz`

FRANTIŠEK DUŠEK was born in Dačice, Czech Republic and studied at the Pardubice Faculty of Chemical Technology in the field of Automation and obtained his MSc. degree in 1980. He worked for the pulp and paper research institute IRAPA. Now he is the vice-dean of the Faculty of Electrical Engineering and Informatics. In 2001 he became an Associate Professor. e-mail: `frantisek.dusek@upce.cz`