Model-Predictive Undercarriage Control for a Pseudo-Omnidirectional, Wheeled Mobile Robot

Christian P. Connette, Stefan Hofmeister, Alexander Bubeck, Martin Hägele, Alexander Verl
Fraunhofer Institute for Manufacturing Engineering and Automation, Stuttgart, Germany

Abstract

A high degree of mobility and flexibility will be a prerequisite for the successful deployment of future service robots. Currently, pseudo-omnidirectional, wheeled mobile robots with independently steered and driven wheels seem to provide a solid compromise between complexity, flexibility and robustness. Yet, such undercarriages are imposed to the risk of actuator fighting and suffer from singular regions within their configuration space.
Within this work a model predictive control (MPC) approach is proposed that addresses both, actuator coordination and singularity avoidance. The control problem is treated within the spherical coordinate representation of the system's velocity space. The MPC approach is simulative and experimentally evaluated w.r.t. the undercarriage of the Care-O-bot® 3 mobile robot (**Figure 1**) and is compared to an earlier developed potential field (PF) based approach.

1 Introduction

1.1 Motivation & Related Work

Future service robot applications will impose high requirements on the employed mobility concepts [1]. While usage in and manipulation of man-made environments require high flexibility, the aspect of commercialization puts a limit to available computational performance and acceptable power consumption. Additionally, aspects as robustness on different, changing undergrounds (carpets or tiles) and modest uneven terrain (e.g. door sills) have to be considered. Lately, pseudo-omnidirectional, wheeled mobile robots whose undercarriages are composed by independently steerable and drivable wheels [2, 3, 4, 5] have emerged as an intermediate-term solution. Such systems present a viable compromise between complexity, robustness and flexibility.
According to the work by Campion et al. [6], a robot with steered standard wheels has 3 degrees-of-freedom (DoF). These DoF are split into the degree of steerability δ_s = 2, associated to the number of independently steerable wheels, and the degree of mobility δ_m = 1, associated to the instantaneously accessible velocity space for the planar motion. Thus, pseudo-omnidirectional mobile robots are able to realize arbitrary velocity and rotational commands, however only after reorienting their wheels. Furthermore, this means that such systems are often overactuated. Usually, they possess four actively steered and driven wheels and thus eight actuators to implement the three DoF. Therefore, it is important to precisely coordinate all motions to reduce actuator fighting [7]. Moreover, such pseudo-omnidirectional, wheeled mobile robots suffer from singular regions within their configuration space [8, 9, 10]. Thuilot et al. solved this problem in [8] by taking into account the singular regions during trajectory planning and control, where they constrained the accessible velocity space of the robot to a region without singular configurations. A similar approach was proposed by Robuffo-Giordano et al. in [9] for the mobile robot Justin. In [10] we sketched an approach that avoids singular configurations by implementing a potential field (PF) [11] based controller. Yet, potential field based approaches are known to be sensitive to local minima [12]. And even though – due to disturbances in practical implementations – the system usually might not stay inside a local minimum for a long time [13], this behaviour can slow down the controller significantly. Moreover, due to the discrete time implementation of the control problem at hand the potential field approach may lead to local oscillations, when the system state centrally approaches the influence region of the repulsive potentials. These properties can be improved by incorporating a predictive horizon into the control scheme.

Within this work a model predictive control (MPC) [14] approach is proposed that addresses both, actuator coordination and singularity avoidance. Therefore, the control problem is treated within the spherical coordinate representation (ρ, φ, θ) of the system's velocity space $(v_{x,r}, v_{y,r}, \omega_r)$ [15], where ρ is associated to the degree of mobility of the system δ_m. The φ and the θ coordinates are associated to the degree of steerability of the system δ_s. Actuator saturation is treated by incorporating the repulsive potential fields applied in [10] into the penalty term within the optimization step of the MPC [16, 17]. The derived algorithm is implemented on the undercarriage (**Figure 1**) of Care-O-bot® 3's mobile base. It is evaluated simulative and experimentally and compared with the performance of a PF based controller.



Figure 1: Mobile Base of Care-O-bot® 3

1.2 Care-O-bot® 3 – Mobile Base

Current and velocity control for the actuators is provided by off-the-shelf motor controllers. The lowest software-layer comprises the control loop for the robot velocities (v_x, v_y, ω) generating the set point values $(\vec{\varphi}_s, \dot{\vec{\varphi}}_d)$ for all motor controllers. It provides an interface for higher level components, for instance, a user interface such as a joy-pad (**Figure 2(a)**), the navigation module (**Figure 2(b)**) which closes the position loop or the arm-control module (**Figure 2(c)**) sending velocity requests to the platform. Therefore, the velocity control loop has to:

1. ensure adherence to the non-holonomic constraints
   (a) identify the valid configuration $(\vec{\varphi}_s, \dot{\vec{\varphi}}_d)$
   (b) derive a valid trajectory $(\vec{\varphi}_s, \dot{\vec{\varphi}}_s, \dot{\vec{\varphi}}_d, \ddot{\vec{\varphi}}_d)$
   (c) respect the actuator limits $(\dot{\vec{\varphi}}_{s,u}, \dot{\vec{\varphi}}_{d,u})$
2. approach the commanded velocities fast
3. compensate the steer/drive-coupling

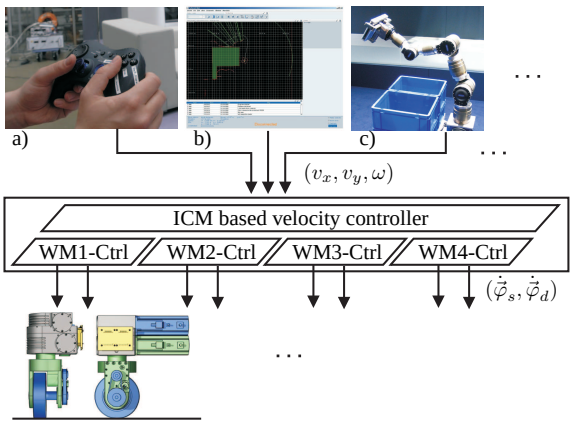


Figure 2: Schematic of Care-O-bot® 3's software-structure. The ICM based velocity controller synchronizes the motion of all wheels. The WMx-controllers synchronize the steer and drive motors of the single wheels.

2 System Kinematics

2.1 State Representation

To decouple the control of the undercarriage from the trajectory-control and to provide a simple interface for commanding velocities, the state space of the undercarriage is reduced to the velocities in the robot coordinate system

$$\vec{t}_r = \vec{g}(\vec{\varphi}_s, \dot{\vec{\varphi}}_d)\,. \tag{1}$$

Within this context, $\vec{t}_r$ is the system's twist-vector, while $\vec{\varphi}_s$ are the directions and $\dot{\vec{\varphi}}_d$ the rotations (associated to drive-motion) of all wheels. Within this work, we focus on a kinematic description of the undercarriage and therefore omit the calculation of wheel-ground contact forces. As proposed in [15], in the we following use the spherical coordinate transform of the twist-vector to represent the system state

$$\vec{t}_r = \begin{pmatrix} \rho_r \\ \varphi_r \\ \theta_r \end{pmatrix} = \begin{pmatrix} \sqrt{v_{x,r}^2 + v_{y,r}^2 + (\omega_r \cdot d_{max})^2} \\ \arctan_2\left(\frac{v_{y,r}}{v_{x,r}}\right) \\ \arctan\left(\frac{\omega_r \cdot d_{max}}{\sqrt{v_{x,r}^2 + v_{y,r}^2}}\right) \end{pmatrix} \tag{2}$$

Here $v_{x,r}$ and $v_{y,r}$ are the robot's linear velocities in the robot coordinate system, ω_r its rotational velocity and d_{max} is a norming factor. Hence, the kinematics equations can be reformulated to calculate steering angles and rotational rates as a function of the spherical twist-vector

$$\left(\vec{\varphi}_s, \dot{\vec{\varphi}}_d\right)^T = \vec{f}\left(\rho_r, \varphi_r, \theta_r\right) \tag{3}$$

$$\left(\dot{\vec{\varphi}}_s, \ddot{\vec{\varphi}}_d\right)^T = \nabla \vec{f} \cdot \left(\dot{\rho}_r, \dot{\varphi}_r, \dot{\theta}_r\right)^T\,. \tag{4}$$

2.2 Input Saturation and State Constraints

This state space representation and the according inverse kinematics equations become singular, when the instantaneous center of motion (ICM) passes through one of the steering axis. In effect, the steering velocity of a wheel grows unbounded, as the ICM moves close to that wheel. However, due to the non-holonomic constraints of the system it is unfeasible to simply constrain the commanded steering velocities $\dot{\vec{\varphi}}_s$ to their maximum values. Doing so destroys the synchronicity of the wheels and leads to actuator conflicts, causing unsteady motions or damaging the actuators.

In [10] a potential-field based controller was applied to avoid the critical regions by representing them as repulsive potentials. However, due to the limited time resolution problems such as oscillations in the vicinity of the repulsive potentials were encountered. The introduction of a predictive horizon, which can be motivated through the MPC formalism, has the potential to remedy these problems.

3 Model Predictive Control

3.1 General Approach

The idea of model predictive control is to solve an optimal control problem for a system

$$\vec{x}_{k+1} = A\vec{x}_k + B\vec{u}_k \,, \tag{5}$$

where $\vec{x}_k$ is the vector of system states, $\vec{u}_k$ the vector of input variables, A the matrix representing the system dynamics and B is representing the influence of the input variables on the system. The MPC approach then derives the input $\vec{u}_k$ such that it optimizes the objective function

$$J = \phi(\vec{x}_N) + \sum_{k=0}^{N-1} \mathcal{L}(\vec{x}_k, \vec{u}_k, k) \tag{6}$$

over a finite time horizon N, by predicting the future development of the system. In this context $\phi(\vec{x}_N)$ penalizes a deviation of goal state and predicted end state and $\mathcal{L}$ is the Lagrangian of the system.

3.2 Control Law Formulation

The system state is composed by the variables $(\rho_r, \varphi_r, \theta_r)^T$ describing the kinematic configuration of the system according to equation (2). To achieve a smooth system behavior an additional integration step is added to the inputs of the system. Therefore, the system is augmented by the additional states $(\dot{\rho}_r, \dot{\varphi}_r, \dot{\theta}_r)^T$. The first variable ρ_r – associated to the system's degree of mobility δ_m or the robot velocity's absolute value – has no influence on the steering commands to be synchronized. Therefore, in the following we omit it in the system state. Hence, the vector $\vec{x}_r$ representing the system state of the pseudo-omnidirectional undercarriage and the input-vector $\vec{u}_r$ become

$$\vec{x}_r = \left(\varphi_r, \theta_r, \dot{\varphi}_r, \dot{\theta}_r\right)^T \,, \tag{7}$$

$$\vec{u}_r = \left(F_{\varphi,r}, F_{\theta,r}\right)^T \,. \tag{8}$$

The system dynamics become

$$\vec{f}(\vec{x_k}, \vec{u_k}) = \begin{pmatrix} x_{1,k} + x_{3,k}\Delta T + \frac{u_{1,k}}{2c_1}\Delta T^2 \\ x_{2,k} + x_{4,k}\Delta T + \frac{u_{2,k}}{2c_2}\Delta T^2 \\ x_{3,k} + u_{1,k}\Delta T \\ x_{4,k} + u_{2,k}\Delta T \end{pmatrix} \tag{9}$$

where c_1 and c_2 are additional integration constants, ΔT is the duration of the discrete time-steps and $A = \partial\vec{f}/\partial\vec{x}$, while $B = \partial\vec{f}/\partial\vec{u}$. Following both constants c_1 and c_2 are set equal to 1, to allow a more convenient writing. A straight forward choice for $\mathcal{L}$ is

$$\mathcal{L} = J^z + J^u + \sum_{i=1}^{2M} J_i^o \,, \tag{10}$$

$$J^z = 1/2 \cdot (\vec{x}_{d,k} - \vec{x}_k)^T Q (\vec{x}_{d,k} - \vec{x}_k), \tag{11}$$

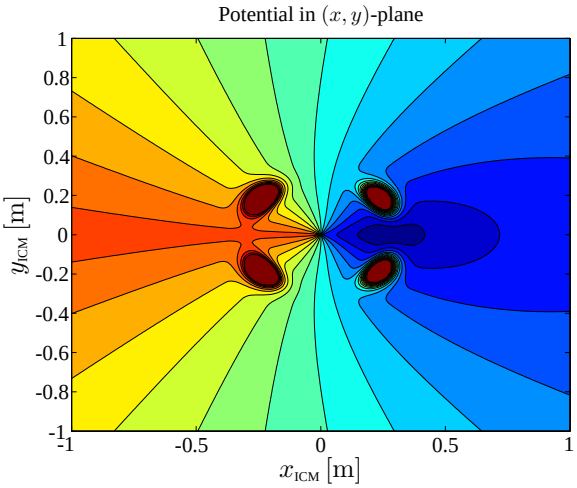
$$J^u = 1/2 \cdot \vec{u}_k^T R \vec{u}_k \,, \tag{12}$$

where J^u and J^z penalize control effort and deviation of current and target state $\vec{x}_{d,k}$ with Q and R being positive semidefinit matrices. The singular regions are incorporated into the objective function via a sum of repulsive potentials

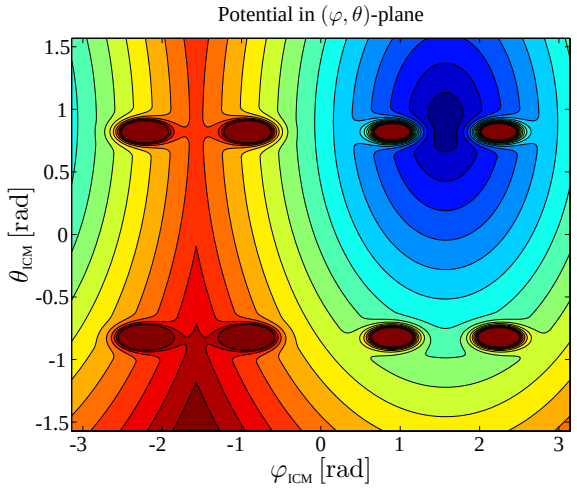
$$J_i^o = \begin{cases} \mu^o \left(\frac{1}{r_i} - \frac{1}{r_0}\right)^2 & \forall \quad r_i \le r_0 \\ 0 & \forall \quad r_i > r_0 \end{cases} \tag{13}$$

$$r_i = \sqrt{\left(\frac{\varphi_r - \varphi_i^o}{a}\right)^2 + \left(\frac{\theta_r - \theta_i^o}{b}\right)^2}, \tag{14}$$

where M is the number of steerable wheels, μ^o is a scaling factor to adjust the gradient of the repulsive potentials, r_0 constrains the region of influence of the potential fields and $(\varphi, \theta)_i^o$ is the position of the i-th wheel's steering axis. A exemplary resulting potential is depicted in **Figure 3**.



(a) Resulting potential field (repulsive and attractive) in Cartesian space



(b) Resulting potential field (repulsive and attractive) in spherical space

Figure 3: Objective function over (φ, θ) resulting from repulsive potentials of all wheels and the attractive potential for a goal configuration at $(\varphi, \theta) = (\frac{1}{2}\pi \text{ rad}, \frac{2}{7}\pi \text{ rad})$, with $k_p = 6$, $\eta = 8$, $r_0 = 1.4$, $a = 0.5$, $b = 0.15$. (dark blue: low potential; dark red: high potential)

3.3 Optimization by Gradient Descent

As proposed in [16, 17] optimization is done by gradient descent based on the Pontryagin minimum principle (PMP) in its discrete formulation. The basic idea of the PMP is to minimize the objective functional J from equation (10) by formulating a Hamiltonian

$$\mathcal{H}_k = \mathcal{L}(\vec{x}_k, \vec{u}_k) + \vec{\lambda}_{k+1}^T \vec{f}(\vec{x}_k, \vec{u}_k) \tag{15}$$

and performing iterated gradient descent by propagating the costate – the Lagrange multipliers $\vec{\lambda}$ – from the predicted end state $\vec{x}_N$ backwards in time

$$\begin{aligned} \vec{\lambda}_k^T &= \frac{\partial \mathcal{H}}{\partial \vec{x}_k} && (16) \\ &= -(\vec{x}_{d,k} - \vec{x}_k)^T Q + \sum_{i=0}^{M} \frac{\partial J_i^o}{\partial \vec{x}_k} + \vec{\lambda}_{k+1}^T \frac{\partial \vec{f}(\vec{x}_k, \vec{u}_k)}{\partial \vec{x}_k} \\ \vec{\lambda}_N^T &= \frac{\partial \phi}{\partial \vec{x}_N} && (17) \end{aligned}$$

over the prediction horizon. To limit the derivative of the original states $(\dot{\varphi}, \dot{\theta})$ of the ICM the optimization is performed w.r.t. a desired state change $(\dot{\varphi}_{d,k}, \dot{\theta}_{d,k})^T$ instead of a desired state. Similar to [11] we calculate a steady state velocity $\dot{\vec{x}}_{eq,k}$ based on the quotient of weighting coefficients in Q

$$\begin{aligned} \dot{\varphi}_{eq,k} &= -q_{11}/q_{33}(\varphi_{d,k} - \varphi_k) && (18) \\ \dot{\theta}_{eq,k} &= -q_{22}/q_{44}(\theta_{d,k} - \theta_k) && (19) \\ \left|\dot{\vec{x}}_{eq,k}\right| &= \sqrt{\dot{\varphi}_{eq,k}^2 + \dot{\theta}_{eq,k}^2}\,. && (20) \end{aligned}$$

The desired state $(\dot{\varphi}_d, \dot{\theta}_d)$ is then derived by constraining the steady-state velocities to a certain limit

$$\begin{aligned} \dot{\varphi}_d &= \min\left(1, \frac{\dot{x}_{max}}{\left|\dot{\vec{x}}_{eq,k}\right|}\right) \cdot \dot{\varphi}_{eq,k} && (21) \\ \dot{\theta}_d &= \min\left(1, \frac{\dot{x}_{max}}{\left|\dot{\vec{x}}_{eq,k}\right|}\right) \cdot \dot{\theta}_{eq,k}\,. && (22) \end{aligned}$$

For the actual optimization process q_{11} and q_{22} are then set to zero. Therefore, the inputs are optimized towards achieving the maximum allowed velocity $\dot{x}_{max}$.
The according inputs $\vec{u}_k$ are then calculated according to

$$\begin{aligned} \vec{u}_k^j &= \vec{u}_k^{j-1} - K\left(\frac{\partial \mathcal{H}}{\partial \vec{u}_k}\right)^T, \text{ with} && (23) \\ \frac{\partial \mathcal{H}}{\partial \vec{u}_k} &= \vec{u}_k^T R + \lambda_{k+1}^T \frac{\partial \vec{f}(\vec{x}_k, \vec{u}_k)}{\partial \vec{u}_k}\,, && (24) \end{aligned}$$

where $\vec{u}_k^j$ is the command-input at step k calculated during the j-th iteration step and K adjusts the step-width during one iteration step. It has to be noted, that the objective functions for the obstacles are not differentiable in its current formulation. Thus, applying the PMP is not guaranteed to deliver an optimal solution.

4 Results

4.1 Simulative Results

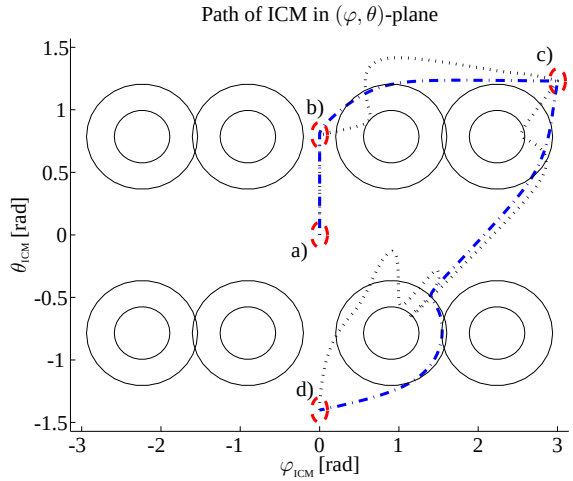
The proposed approach is evaluated in simulation with respect to the specific kinematics of Care-O-bot® 3. The system is simulated with a time step size of $20\,\mathrm{ms}$. The simulation takes into account a transport delay of the measured sizes of $10\,\mathrm{ms}$ and the restrictions on velocity and acceleration of the wheel modules.
Simulation was performed for the implemented MPC based (blue dash-dotted lines in **Figure 4**) as well as for the earlier implemented conventional PF based controller (dotted black lines). To simplify comparison of the results the controllers were tuned to be similar fast. In **Figure 4(b)** one can see, that all targets are reached at approximately the same time for both controllers. For the last set point (reached at about $t = 8\,\mathrm{s}$) the controllers deviate if only their single components are taken into account. Still the total time for both variables is similar. This is due to the fact, that they bypassed the last singular region on different sides (**Figure 4(a)**).
Already on first sight (**Figure 4(a)**) it becomes apparent, that the MPC approach shows a much smoother behavior than the PF controller. The latter one tends to oscillations in the vicinity of the repulsive potentials. Moreover, one can see (**Figure 4(c)**) that the steering velocities associated to the MPC based controller stay significantly lower than those of the PF approach.

4.2 Experimental Results

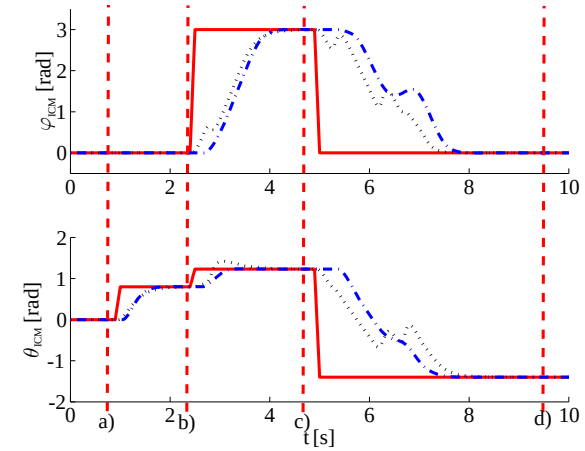
The proposed approach is experimentally evaluated on the Care-O-bot® 3 mobile platform. The control-rate of the system is set to 20 ms. The prediction horizon is 16 timesteps (320 ms) long. The control inputs are generated manually using a joypad.
The results for an experimental run, where the set points were repeatedly set to pass through the singular regions are depicted in **Figure 5**. In **Figure 5(a)** one can see how the controller avoids the critical regions. Accordingly the resulting steer-rate commands stay significantly lower than 10 rad/s (**Figure 5(b)**).

5 Conclusion

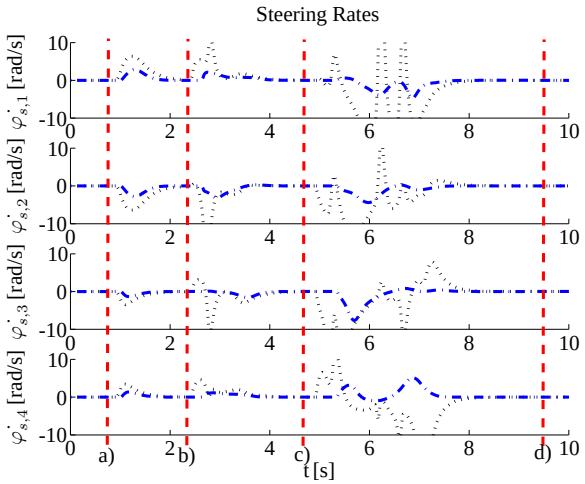
Within this work, a model predictive scheme was applied to the inner-loop control for the undercarriage kinematics of the pseudo-omnidirectional mobile robot Care-O-bot® 3. To ensure adherence to the non-holonomic constraints control was performed in the spherical representation of the velocity space. To ensure avoidance of the system's singular configuration and guarantee limited inputs, the objective function is constituted by the earlier derived potential fields.



(a) Resulting paths of ICM in the (φ, θ)-plane for different controllers



(b) Set point values (red solid) and resulting controlled values of (ρ, φ, θ)

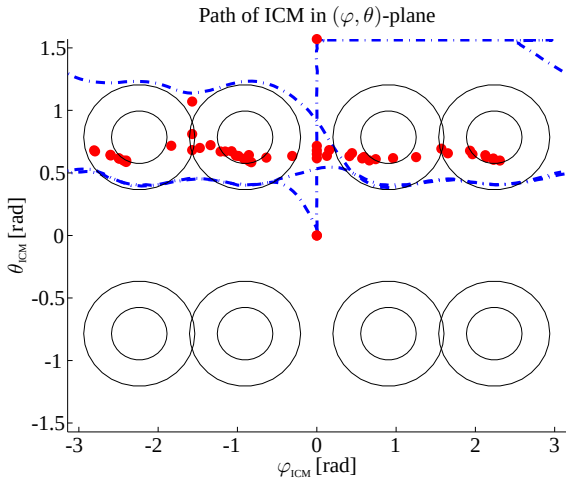


(c) Resulting wheel steering rates in all wheels

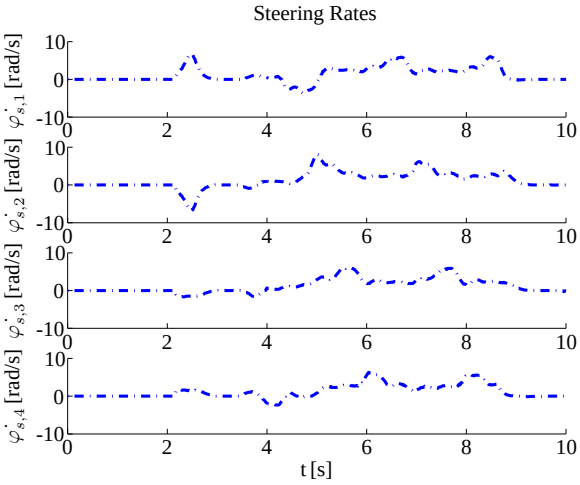
Figure 4: Results for MPC (blue dash-dotted) and PF (red dots) based controller for a sequence of four target configurations (red dashed circles) and a maximum velocity $\dot{x}_{max} = 1\pi\,\mathrm{rad/s}$.

The simulative results show, that the MPC based approach clearly outperforms the PF based controller. While achieving similar fast tracking of the set-point values, the MPC approach shows a much smoother path within the state space and causes significantly lower steering rates. These properties are confirmed by the experimental results obtained with the Care-O-bot® 3.

One disadvantage of the MPC approach is the large number of tunable parameters and the sensitivity of the PMP to parameter changes. For instance, even slight modifications of the parameter K in equation (23) can lead to divergence of the procedure. Moreover, it has to be noted that the proposed approach is not guaranteed to find a global optimum, as the employed objective functions are not continuously differentiable over the full state space.



(a) Desired and Resulting path of ICM in the (φ, θ)-plane



(b) Resulting wheel steering rates in all wheels

Figure 5: Experimental results obtained for MPC based control (blue dash-dotted). The black dotted line in **Figure 5(a)** indicates the desired state-trajectory. The results were obtained for $N = 16$, $K = 0.065$ and $\dot{x}_{max} = 1\pi\,\mathrm{rad/s}$.

6 Outlook

The future plans are twofold: On the one hand, it is planned to investigate our approach in context with people tracking, which requires a high mobility of the robot. On the other hand, it is planned to investigate if further development of the velocity centered optimization approach can improve the convergence by constraining the effective step width during optimization.

References

[1] Graf, B.; Hans, M.; Schraft, R. D.: *Care-O-bot II - Development of a Next Generation Robotic Home Assistant*, Autonomous Robots, Vol. 16, No. 2, 2004, pp. 193-205

[2] Lindemann, R. A.; Bickler, D. B.; Harrington, B. D.; Ortiz, G. M.; Voorhees, C. J.: *Mars Exploration Rover Mobility Development – Mechanical Mobility, Hardware Design, Development and Testing*, IEEE Robotics & Automation Magazine, Vol. 13, No. 2, 2006, pp. 19-26

[3] Fuchs, M.; Borst, C.; Robuffo Giordano, P.; Baumann, A.; Kraemer, E.; Langwald, J.; Gruber, R.; Seitz, N.; Plank, G.; Kunze, K.; Burger, R.; Schmidt, F.; Wimboeck, T.; Hirzinger, G.: *Rollin' Justin - Design Considerations and Realization of a Mobile Platform for a Humanoid Upper Body*, 2009 IEEE International Conference on Robotics and Automation ICRA, Kobe, Japan, 2009, pp. 4131-4137

[4] Wyrobek, K. A.; Berger, E. H.; van der Loos, H. F. M.; Salisbury, J. K.: *Towards a Personal Robotics Development Platform: Rationale and Design of an Intrinsically Safe Personal Robot*, 2008 IEEE International Conference on Robotics and Automation ICRA, Pasadena, USA, 2008, pp. 2165-2170

[5] Reiser, U.; Connette, C. P.; Fischer, J.; Kubacki, J.; Bubeck, A.; Weisshardt, F.; Jacobs, T.; Parlitz, C.; Hägele, M.; Verl, A.: *Care-O-bot 3 - Creating a product vision for service robot applications by integrating design and technology*, 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, St. Louis, USA, 2009, pp. 1992-1998

[6] Campion, G.; Bastin, G.; D'Andréa-Novel, B.: *Structural Properties and Classification of Kinematic and Dynamic Models of Wheeled Mobile Robots*, IEEE Transactions on Robotics and Automation, Vol. 12, No. 1, 1996, pp. 47-62

[7] Lauria, M.; Nadeau, I.; Lepage, P.; Morin, Y.; Giguère, P.; Gagnon, F.; Létourneau, D.; Michaud, F.: *Design and Control of a Four Steered Wheeled Mobile Robot*, IEEE 32nd Anual Conference on Industrial Electronics IECON, Paris, France, 2006, pp. 4020-4025

[8] Thuilot, B.; D'Andréa-Novel, B.; Micaelli, A.: *Modeling and Feedback Control of Mobile Robots Equipped with Several Steering Wheels*, IEEE Transactions on Robotics and Automation, Vol. 12, No. 3, 1996, pp. 375-390

[9] Robuffo Giordano, P.; Fuchs, M.; Albu-Schäffer, A.; Hirzinger, G.: *On the Kinematic Modelling and Control of a Mobile Platform Equipped with Steering Wheels and Movable Legs*, 2009 IEEE International Conference on Robotics and Automation ICRA, Kobe, Japan, 2009, pp. 4080-4087

[10] Connette, C. P.; Parlitz, C.; Hägele, M.; Verl, A.: *Singularity Avoidance for Over-Actuated, Pseudo-Omnidirectional, Wheeled Mobile Robots*, 2009 IEEE International Conference on Robotics and Automation ICRA, Kobe, Japan, 2009, pp. 4124-4130

[11] Khatib, O.: *Real-Time Obstacle Avoidance for Manipulators and Mobile Robots*, The International Journal of Robotics Research, Vol. 5, No. 1, 1986, pp. 90-98

[12] Koren, Y.; Borenstein, J.: *Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation*, 1991 IEEE International Conference on Robotics and Automation ICRA, Sacramento, USA, pp. 1398-1404

[13] Shimoda, S.; Kuroda, Y.; Iagnemma, K.: *Potential Field Navigation of High Speed Unmanned Ground Vehicles on Uneven Terrain*, IEEE International Conference on Robotics and Automation, Barcelona, Spain, 2005, pp. 2839-2844

[14] Rawlings, J. B.: *Tutorial Overview of Model Predictive Control*, IEEE Control Systems Magazine, Vol. 20, No. 3, 2000, pp. 38-52

[15] Connette, C. P.; Pott, A.; Hägele, M.; Verl, A.: *Control of an Pseudo-Omnidirectional, Non-holonomic, Mobile Robot based on an ICM Representation in Spherical Coordinates*, 47th IEEE Conference on Decision and Control CDC, Cancun, Mexico, 2008, pp. 4976-4983

[16] Kim, H. J.; Shim, D. H.: *A flight control system for aerial robots: algorithms and experiments*, Control Engineering Practice, Vol. 11, No. 12, 2003, pp. 1389-1400

[17] Xi, W.; Baras, J. S.: *MPC Based Motion Control of Car-like Vehicle Swarms*, IEEE Mediterranean Conference on Control & Automation, Athens, Greece, 2007, pp. 1-6