Model Predictive Control without terminal constraints or costs for holonomic mobile robots

Article *in* Robotics and Autonomous Systems · February 2020

DOI: 10.1016/j.robot.2020.103468

CITATIONS
11

READS
763

6 authors, including:

Mohamed W. Mehrez
Zebra Technologies Corporation
30 PUBLICATIONS **265** CITATIONS
SEE PROFILE

Karl Worthmann
Technische Universität Ilmenau
112 PUBLICATIONS **1,283** CITATIONS
SEE PROFILE

Joseph P.V. Cenerini
University of Waterloo
1 PUBLICATION **11** CITATIONS
SEE PROFILE

Mostafa Osman
Delft University of Technology
11 PUBLICATIONS **50** CITATIONS
SEE PROFILE

Model Predictive Control without Terminal Constraints or Costs for Holonomic Mobile Robots

Mohamed W. Mehrez[a], Karl Worthmann[b], Joseph P.V. Cenerini[a], Mostafa Osman[a], William W. Melek[a], Soo Jeon[a]

[a] *Mechanical and Mechatronics Engineering, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada*
[b] *Institute for Mathematics, Technische Universität Ilmenau, Ilmenau, Germany*

Abstract

We investigate Model Predictive Control (MPC) schemes without stabilizing constraints or costs for the set-point stabilization of holonomic mobile robots. Herein, we ensure closed-loop asymptotic stability using the concept of cost controllability. To this end, we derive a growth bound on the finite-horizon value function in terms of the running costs evaluated at the current state, which is then used to determine a stabilizing prediction horizon. In the discrete-time setting, we additionally show that asymptotic stability holds for the shortest possible prediction horizon. Moreover, we deduce a lower bound on the MPC performance on the infinite horizon. Theoretical results are verified by numerical simulations as well as laboratory experiments of stabilizing a holonomic mobile robot to a reference set point.

Keywords:
Motion Control, Holonomic Mobile Robots, Model Predictive Control, Stability Analysis

1. Introduction

Mobile robots are the core components of industrial automation systems especially in mobile manipulation applications. Holonomic mobile robots possess an extra degree of maneuverability when compared to their non-holonomic counterparts; thus, they can achieve independent translational and rotational motions [1, 2]. There are two possible configurations of holonomic mobile robots, i.e. robots with mecanum wheels and robots with omniwheels, see Fig. 1. Both configurations can achieve the same motion profiles, but with different low level kinematics, i.e. the kinematics that govern the relation between the robot's speed and its wheels' speeds.

Motion control objectives of mobile robots include point (posture) stabilization, trajectory tracking, and path-following, see, e.g. [3] for details and [4, 5] for surveys on motion control of non-holonomic mobile robots. For holonomic mobile robots, several controllers are developed for the mentioned control problems; for example, a dynamic trajectory generation controller for point stabilization is shown in [6]. A trajectory tracking controller based on ideal reference velocities is presented in [7]. A backstepping controller for both stabilization and trajectory tracking is introduced in [8]. A path-following controller based on inverse linearized kinematics model is presented in [9]. Moreover, multiple implementations of Model Predictive Control (MPC) to the three control objectives are also considered in the literature, see, e.g. [2, 10, 11, 12, 13]. The previously highlighted controllers, except MPC, are either challenging to tune in order to achieve a satisfactory performance or require a post processing step to consider control saturation limits.

In this paper, we focus on the point-stabilization problem of holonomic mobile robots using MPC *without stabilizing constraints or costs*. In MPC, a cost function, parameterizing the control objective, is minimized on a finite-horizon Optimal Control Problem (OCP). Then, the first portion of the minimizing control is applied to the controlled system before the process is repeated at the following sampling instant [14]. MPC is popular in the control community because of its ability to handle constrained nonlinear systems, see, e.g. [15] for several examples of MPC applications. Ensuring closed-loop asymptotic stability under MPC is one of its central issues. This problem can be solved by adding terminal constraints and/or terminal costs [16, 17]; or by using bounds on the MPC value function to find a stabilizing prediction horizon length [18, 19].

Although several studies employed MPC for the control of holonomic mobile robots, only a few considered the closed-loop asymptotic stability of the system under MPC. In all of these studies, stabilizing terminal conditions were adopted, see, e.g. [13, 10, 20]. MPC schemes *without* stabilizing constraints are easier to design, require less computational effort in implementation, and are the preferred MPC variant in industrial applications when compared to the MPC schemes with stabilizing constraints. Moreover, for a fixed prediction horizon, MPC schemes without stabilizing constraints provide larger stability regions than schemes with stabilizing constraints. Finally, when the control effort in the MPC cost function is largely penal-

ized, schemes without stabilizing conditions lead to better closed-loop performance in contrast to schemes with stabilizing constraints; we refer to [21, Chapter 8] for a more thorough comparison of the variant MPC schemes.

In this paper, the closed-loop asymptotic stability of MPC without stabilizing constraints or costs is analyzed for the regulation control of holonomic mobile robots. In particular, we verify cost controllability, i.e. a sufficient stability condition, which relates the growth of the MPC value function and the running costs uniformly w.r.t. the initial state; for details on the general framework, see [22]. This is achieved by deriving a growth function bounding the MPC value function. This growth function allows to precisely determine the length of the prediction horizon such that asymptotic stability of the desired set-point w.r.t. the MPC closed loop is guaranteed and, in addition, a suboptimality estimate on the infinite horizon can be determined. The latter provides a quantitative measure on the holonomic robot's performance.

Similar to the work of Worthmann et al. [23], an open-loop control maneuver, which stabilizes the considered system to a reference set point is designed and then used to derive a growth function of the MPC value function and calculate a stabilizing prediction horizon length. The special structure of the resulting growth function is used to show that the point stabilization of holonomic mobile robots can be guaranteed to be asymptotically stable, under the employed MPC, by the shortest prediction horizon length. Additionally, we derive an estimate on the MPC performance (suboptimality) index in this case. The analysis is performed in the continuous- and the discrete-time settings; moreover, the connection between the two settings is explicated. We verify the theoretical results by numerical and real-time lab experiments.

The contribution of the paper is threefold: 1) derivation of a growth function for the MPC value function used for the point stabilization of holonomic mobile robots. This function is derived for both continuous- and discrete-time settings. 2) employing the derived growth function to rigorously prove that the shortest possible prediction horizon is sufficient to ensure closed-loop asymptotic stability using the employed MPC scheme. 3) developing a relationship between the growth function for the continuous- and the discrete-time settings, and verifying the theoretical results throughout numerical simulations and experiments.

The paper is organized as follows: in Section 2, the kinematics model of holonomic mobile robots is presented together with the proposed control scheme. The MPC stability results from [24] are revisited in Section 3. Open-loop control maneuvers are designed in Section 4 and a (continuous-time) expression for the value function bound is derived; then, this bound is used to calculate a stabilizing horizon length. In Section 5, results are further investigated in the discrete-time setting, where it is shown that closed-loop asymptotic stability is guaranteed by the shortest prediction horizon; the performance bound of MPC is additionally computed. Simulation and experimental results employing a mecanum wheel holonomic robot are shown in Section 6. Finally, concluding remarks and the outlook of the contribution are discussed in Section 7.

2. Problem Formulation

In this section, we first introduce the notations used in the paper. Second, we present the continuous-time kinematics model of holonomic mobile robots and the set-point stabilization objective. Then, we propose a model predictive control (MPC) scheme without stabilizing constraints or costs.

2.1. Notation

$\mathbb{R}$ and $\mathbb{N}$ denote real and natural numbers, respectively. $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$ represents the non-negative integers and $\mathbb{R}_{\geq 0}$ the non-negative real numbers. A continuous function $\eta : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is said to be of class $\mathcal{K}$ if it is zero at zero and strictly monotonically increasing. If it is, in addition, unbounded it is of class $\mathcal{K}_\infty$. A continuous function $\beta : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is called a $\mathcal{KL}$-function if $\beta(\cdot, t) \in \mathcal{K}_\infty$ for all $t \in \mathbb{R}_{\geq 0}$ and $\beta(r, \cdot)$ is strictly monotonically decaying to zero for each $r > 0$. A function $c : I \to \mathbb{R}$ is said to be *piecewise continuous*, denoted by $\mathcal{PC}(I, \mathbb{R})$, if, for every $a, b \in I$ with $a < b$, the interval $[a, b]$ admits a finite partition $a = t_1 < t_2 < \cdots < t_n = b$, $n \in \mathbb{N}$, such that c is continuous on every subinterval (t_i, t_{i+1}), $i \in \{1, \ldots, n-1\}$, has a right limit $\lim_{t \searrow t_1} c(t)$ at t_1, a left limit $\lim_{t \nearrow t_n} c(t)$ at t_n, and both left and right limit at every t_i, $i \in \{2, 3, \ldots, n-1\}$. The ceil operator $\lceil \cdot \rceil$ is defined as $\lceil x \rceil := \min \{m \in \mathbb{N} : m \geq x\}$ while the floor operator $\lfloor \cdot \rfloor$ is defined as $\lfloor x \rfloor := \max \{m \in \mathbb{N} : m \leq x\}$. For a given vector $\mathbf{x} \in \mathbb{R}^n$, $\|\mathbf{x}\|_A^2$ represents the quadratic form $\mathbf{x}^\top A \mathbf{x}$ with the matrix $A \in \mathbb{R}^{n \times n}$. Finally, we define the infinity norm $\|\mathbf{x}\|_\infty := \max_{i \in \{1,2,\ldots,n\}} |x_i|$, where n is the number of entires in $\mathbf{x}$.

2.2. Kinematics of Holonomic Mobile Robots

The holonomic robot state is described by the state vector $\mathbf{x} = (x_1, x_2, x_3)^\top \in X \subsetneq \mathbb{R}^3$, which consists of the (spatial) position $(x_1, x_2)^\top$ [m,m] and the orientation angle x_3 [rad] of the robot. Additionally, the robot control input is defined by the vector $\mathbf{u} = (u_1, u_2, u_3)^\top \in U \subsetneq \mathbb{R}^3$, where u_1 [m/s], u_2 [m/s], and u_3 [rad/s] are the linear, the lateral, and the angular speeds of the robot, respectively. At time $t \in \mathbb{R}_{\geq 0}$ [seconds], the continuous time kinematics model of a holonomic mobile robot is given by

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)) = \mathcal{R}(x_3(t)) \cdot \mathbf{u}(t), \tag{1}$$

where $f : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}^3$ is an analytic mapping and $\mathcal{R}(x_3) \in \mathbb{R}^{3 \times 3}$ is the following rotation matrix

$$\mathcal{R}(x_3) = \begin{pmatrix} \cos(x_3) & -\sin(x_3) & 0 \\ \sin(x_3) & \cos(x_3) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{2}$$

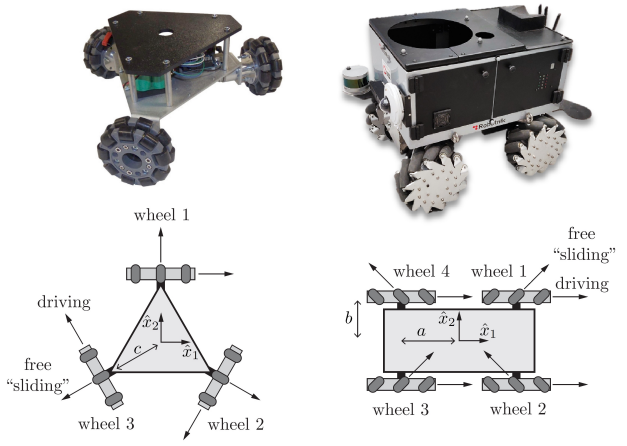

Figure 1: Left: a holonomic mobile robot with omniwheels. Right: a holonomic mobile robot with mecanum wheels (*Summit-XL-Steel* [25]). The driving and the free sliding directions for each wheel type is shown by corresponding vectors. The figure is partially reprinted from [26].

The compact and convex state constraint set X is defined as

$$X := [-\bar{x}_1, \bar{x}_1] \times [-\bar{x}_2, \bar{x}_2] \times [-\pi, \pi],$$

for $\bar{x}_1 > 0$ and $\bar{x}_2 > 0$, with the origin in its interior.

In order to derive the control input set U, special attention should be given to the lower level kinematics. The lower level kinematics model relates the robot's speeds $\mathbf{u}$ and the robot wheels' speeds, i.e. $\omega \in \mathbb{R}^3$ for robots with omniwheels or $\omega \in \mathbb{R}^4$ for robots with mecanum wheels, see Fig. 1. This relation is given by:

$$\omega(t) = \frac{1}{r} \cdot H\mathbf{u}(t), \tag{3}$$

where

$$H = \begin{pmatrix} 1 & 0 & -c \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} & -c \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & -c \end{pmatrix} \text{ and } H = \begin{pmatrix} 1 & -1 & -a-b \\ 1 & 1 & a+b \\ 1 & -1 & a+b \\ 1 & 1 & -a-b \end{pmatrix}$$

for omniwheels robots and mecanum wheels robots, respectively. Here, r [m] is the radius of the robot wheels, and the dimensions a, b and c [m] are shown in Fig. 1, see [26] for the derivation of the matrix H. Now, for a given speed limit on the wheels of either of the holonomic robots, i.e. $\bar{\omega}$, the control input set U can be defined as

$$U := \left\{ \mathbf{u} \in \mathbb{R}^3 \middle| \ \|H\mathbf{u}\|_\infty \leq r\bar{\omega} \right\}. \tag{4}$$

For robots with omniwheels, we have

$$\|H\mathbf{u}\|_\infty := \max \left\{ |u_1 - cu_3|, \left|0.5u_1 + (\sqrt{3}/2)u_2 + cu_3\right|, \left|-0.5u_1 + (\sqrt{3}/2)u_2 - cu_3\right| \right\},$$

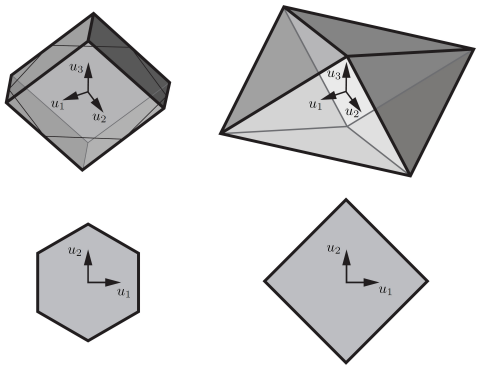

Figure 2: Visualization of the set U for omniwheels robots (left) and mecanum wheels robots (right). Translation only constraints, for $u_3 = 0$, are shown at the bottom. The figure is reprinted from [26].

while for robots with mecanum wheels, we have

$$\|H\mathbf{u}\|_\infty := |u_1| + |u_2| + (a+b)|u_3|.$$

The reason $\|H\mathbf{u}\|_\infty$ reduces to one term for robots with mecanum wheels is the symmetry of the robot configuration around $\hat{x}_1$-axis and $\hat{x}_2$-axis, see Fig. 1. For either of the holonomic robots, the set U is convex with the origin in its interior, see Fig. 2. In summary, we have $(0,0,0)^\top \times (0,0,0)^\top \in \text{int}(X) \times \text{int}(U)$.

For a given initial state $\mathbf{x}_0$, the trajectory generated by the dynamics (1) and the control function $u \in \mathcal{PC}(\mathbb{R}_{\geq 0}, U)$ is denoted by $x(\cdot; \mathbf{x}_0, u)$. Note that (global) existence and uniqueness can be concluded from the global Lipschitz-continuity of the right hand side of the differential equation (1). In the following, we define the admissibility of an input function u.

Definition 1. *For $T \in \mathbb{R}_{>0}$ and $\mathbf{x}_0 \in X$, a control function $u \in \mathcal{PC}([0,T], U)$ is said to be admissible, denoted by $\mathcal{U}_T(\mathbf{x}_0)$, if and only if it satisfies the conditions*

$$\mathbf{u}(t) \in U \quad \text{and} \quad x(t; \mathbf{x}_0, u) \in X \qquad \text{for all } t \in [0,T].$$

Moreover, the set of all admissible control functions $u \in \mathcal{PC}([0,\infty), U)$ satisfying $u|_{[0,T]} \in \mathcal{U}_T(\mathbf{x}_0)$ for all $T > 0$ is denoted by $\mathcal{U}_\infty(\mathbf{x}_0)$.

2.3. Model Predictive Control

For the holonomic mobile robot considered, the control objective is to steer the robot to a desired set point $\mathbf{x}^\star$, which is a (controlled) equilibrium, i.e. $f(\mathbf{x}^\star, 0) = 0$ holds. Without loss of generality the set point is chosen as the origin, i.e. $\mathbf{x}^\star = (0,0,0)^\top$.

We propose to use an MPC scheme to achieve the control objective. To this end, we choose a performance criterion encoded by the quadratic running costs $\ell : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}_{\geq 0}$ defined as

$$\ell(\mathbf{x}, \mathbf{u}) = \|\mathbf{x}\|_Q^2 + \|\mathbf{u}\|_R^2, \tag{5}$$

where Q and R are positive definite weighting matrices with units adjusted such that ℓ is dimensionless. For simplicity, Q and R are chosen as $Q = \mathrm{diag}(q_1, q_2, q_3)$ and $R = \mathrm{diag}(r_1, r_2, r_3)$.

As standard in MPC, the input is determined by repeatedly solving an optimal control problem (OCP). That is, at each time instant $t_k = k\tau$, $k \in \mathbb{N}_0$, with sampling period $\tau > 0$, we solve an OCP on the prediction/optimization horizon $T = N\tau$, $N \in \mathbb{N}_{\geq 2}$, in which the cost functional

$$J_T(\mathbf{x}_k, u) = \int_{t_k}^{t_k+T} \ell\left(x(t; \mathbf{x}_k, u), \mathbf{u}(t)\right) \mathrm{d}t$$

is minimized. Here, the subscript $(\cdot)_k$ indicates that the corresponding variable is measured at the k-th time instant t_k, i.e. $\mathbf{x}_k = \mathbf{x}(t_k)$. The OCP to be solved at the time instants t_k reads:

$$\inf_{u \in \mathcal{U}_T(\mathbf{x}_k)} J_T(\mathbf{x}_k, u) =: V_T(\mathbf{x}_k) \tag{6a}$$

$$\begin{aligned} \text{subject to } & \mathbf{x}(t_k) = \mathbf{x}_k, \\ & \dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)) \quad \forall t \in [t_k, t_k + T) \\ & \mathbf{x}(t) \in X \quad \forall t \in [t_k, t_k + T] \qquad (6b) \\ & \mathbf{u}(t) \in U \quad \forall t \in [t_k, t_k + T) \qquad (6c) \end{aligned}$$

Since the infimum is attained[1], solving OCP (6) results in an optimal control function $u^\star = \mathbf{u}^\star(\cdot; \mathbf{x}_k) : [0, T) \to \mathbb{R}^3$ with $V_T(\mathbf{x}_k) = J_T(\mathbf{x}_k, u^\star)$; $V_T : X \to \mathbb{R}_{\geq 0}$ is the optimal value function. The state and input constraints of the system given by Eq. (1) are enforced by Eq. (6b) and (6c), respectively. Then, for a time shift $\delta = m\tau$, $m \in [1 : N-1]$, we define the MPC feedback law as

$$\mu(t, \mathbf{x}_k) = \mathbf{u}^\star(t - t_k; \mathbf{x}_k), \qquad t \in [t_k, t_k + \delta),$$

i.e. the time shift determines the portion of the control function implemented in an MPC step [28]. δ is also referred to as the control horizon in the literature [19]. Application of the MPC feedback law to system (1) yields the closed-loop trajectory governed by

$$\dot{x}_\mu(t; \mathbf{x}_0) = f(x_\mu(t; \mathbf{x}_0), \mathbf{u}^\star(t - t_\delta; \mathbf{x}_\delta))$$

with $t_\delta := \delta \lfloor t/\delta \rfloor$ and $\mathbf{x}_\delta := x_\mu(t_\delta; \mathbf{x}_0)$, i.e. $t - t_\delta \in [0, \delta)$. The proposed controller is summarized in Algorithm 1.

Recursive feasibility of the proposed MPC scheme summarized in Algorithm 1 is ensured since $u \equiv (0, 0, 0)^\top$ is admissible for arbitrary $\mathbf{x}_k \in X$ implying $\mathcal{U}_T(\mathbf{x}_k) \neq \emptyset$ (i.e. non-empty). However, guaranteeing asymptotic stability of the closed-loop system is not straightforward since we do not impose terminal constraints and/or costs. Hence, our preliminary goal is to ensure that the origin is asymptotically stable, w.r.t. the MPC closed loop, i.e. there exists a function $\beta \in \mathcal{KL}$ satisfying

$$\|x_\mu(t; \mathbf{x}_0)\| \leq \beta(\|\mathbf{x}_0\|, t) \quad \forall\, t \geq 0$$

for all $\mathbf{x}_0 \in X$, where $x_\mu(t; \mathbf{x}_0)$ denotes the MPC closed-loop trajectory.

[1] This can be shown by using standard arguments, see, e.g. [27, Chapter 4]

Algorithm 1 MPC Scheme

Given: Sampling period τ, prediction horizon $T = N\tau$, time shift $\delta = m\tau$, and initial state $\mathbf{x}_0 \in X$

Set: $k = 0$ and $\mathbf{x}_k := \mathbf{x}_0$.

1: Solve the OCP (6) in order to compute a minimizing control function $u^\star \in \mathcal{U}_T(\mathbf{x}_k)$ such that $J_T(\mathbf{x}_k, u^\star) = V_T(\mathbf{x}_k)$ holds.
2: Apply $\mathbf{u}^\star(t)$, $t \in [0, \delta)$, at the plant.
3: Set $\mathbf{x}_{k+1} := x_\mu(t_{k+1}; \mathbf{x}_0) = x(\delta; \mathbf{x}_k, u^\star)$, increment k, and go to step 1.

3. MPC Stability and Performance Bounds

In this section, we recapitulate MPC stability results presented in [24] based on cost controllability – a condition originally introduced in [18], see also [22] for further details. Here, the relaxed Lyapunov inequality

$$V_T(x_\mu(\delta; \mathbf{x})) \leq V_T(\mathbf{x}) - \alpha \int_0^\delta \ell(x_\mu(t; \mathbf{x}), \mu(t; \mathbf{x}))\, \mathrm{d}t \quad \forall\, \mathbf{x} \in X \tag{7}$$

with $\alpha = \alpha_{T,\delta} \in (0, 1]$ was used in order to ensure asymptotic stability of the origin w.r.t. the MPC closed loop without (stabilizing) terminal constraints or costs, see [29] and [30] for relaxed dynamic programming.

The following theorem ensures asymptotic stability of the MPC closed loop and, in addition, allows to estimate the required length of the prediction horizon T.

Theorem 2. *Let the system dynamics and the running costs be given by Eq.* (1) *and Eq.* (5)*, respectively. Moreover, let the cost controllability be satisfied, i.e. there exists a monotonically increasing and bounded function* $B : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ *satisfying*

$$V_t(\mathbf{x}_0) \leq B(t) \cdot \|\mathbf{x}_0\|_Q^2 \qquad \forall\, t \geq 0 \text{ and } \mathbf{x}_0 \in X. \tag{8}$$

Then, for a given time shift $\delta > 0$*, and a prediction horizon* $T > \delta$ *choosen such that the stability condition* $\alpha_{T,\delta} > 0$ *is satisfied with performance index* $\alpha_{T,\delta}$ *defined by*

$$\alpha_{T,\delta} = 1 - \frac{e^{-\int_\delta^T B(t)^{-1}\, \mathrm{d}t} \cdot e^{-\int_{T-\delta}^T B(t)^{-1}\, \mathrm{d}t}}{\left[1 - e^{-\int_\delta^T B(t)^{-1}\, \mathrm{d}t}\right] \left[1 - e^{-\int_{T-\delta}^T B(t)^{-1}\, \mathrm{d}t}\right]},$$

the relaxed Lyapunov inequality (7) *holds. Moreover, the origin is asymptotically stable w.r.t. the MPC closed loop and the performance estimate*

$$V_\infty^{\mu_{T,\delta}}(\mathbf{x}) \leq \alpha_{T,\delta}^{-1} \cdot V_\infty(\mathbf{x}) \tag{9}$$

is satisfied, where $V_\infty^{\mu_{T,\delta}}(\mathbf{x})$ *are the MPC closed-loop costs on the infinite horizon, i.e.*

$$V_\infty^{\mu_{T,\delta}}(\mathbf{x}) := \int_0^\infty \ell(x_{\mu_{T,\delta}}(t; \mathbf{x}), \mu_{T,\delta}(t; \mathbf{x}))\, \mathrm{d}t,$$

where $\mu_{T,\delta}(t, \mathbf{x}) := \mathbf{u}^\star(t - \delta \lfloor t/\delta \rfloor, x_{\mu_{T,\delta}}(\cdot; \mathbf{x}))$ *denotes the MPC closed-loop control.* □

For the proof of Theorem 2, we refer to [24, Theorem 9] noting that the assumptions on the running costs (lower and upper bound given by $\mathcal{K}_\infty$-functions) are trivially satisfied for the running costs (5). Moreover, we emphasize that boundedness of the growth function B satisfying (19) ensures existence of a (sufficiently large) prediction horizon T such that the stability condition $\alpha_{T,\delta} > 0$ is satisfied for arbitrary but fixed time shift δ. Note that the underlying idea is to use the value function V_T as a Lyapunov function.

In conclusion, we have to construct the growth function B and, thus, cost controllability in order to apply Theorem 2. In essence, cost controllability connects the current state $\mathbf{x}$ through the running costs ℓ to the growth of the value function $V_t(\mathbf{x})$ w.r.t. the optimization horizon T. $\alpha_{T,\delta}$ can be interpreted as a performance index (degree of suboptimality) depending on the time shift δ and the prediction horizon T comparing the MPC closed-loop costs with the (optimal achievable) performance for $T = \infty$, cp. Inequality (9).

Since the linearization of the system model (1) at the set point (the origin) is stabilizable (even controllable) and quadratic, positive definite running costs are used, asymptotic stability is clear for a sufficiently long prediction horizon, see, e.g. [31] using the results derived in [32] (linear-quadratic setting for discrete-time systems and [33] where a connection between discrete- and continuous-time systems is established. However, the growth bound derived in the following and, thus, the upcoming quantitative analysis is to the best of our knowledge new and decisive for the application to holonomic mobile robots.

4. Growth Function

In this section, we compute the growth function B satisfying the cost-controllability condition given by Inequality (8) for the considered point-stabilization control problem. Using this growth function B, we, then, determine a prediction horizon T such that the degree of suboptimality $\alpha_{T,\delta}$ is positive and, thus, asymptotic stability using Theorem 2 can be inferred. Similar to [34], we construct the growth function B such that Inequality (8) is satisfied for all $\mathbf{x}_0 \in X$. This is done by constructing an admissible open-loop control function $u_{\mathbf{x}_0} \in \mathcal{U}_\infty(\mathbf{x}_0)$ steering the robot state from $\mathbf{x}_0$ to the reference $\mathbf{x}^\star = (0,0,0)^\top$ in finite time. Second, we compute an upper bound for the (suboptimal) running costs $\ell(x(t;\mathbf{x}_0,u_{\mathbf{x}_0}),\mathbf{u}_{\mathbf{x}_0}(t))$. Third, we derive a function $c \in \mathcal{PC}(\mathbb{R}_{\geq 0},\mathbb{R}_{\geq 0})$ satisfying

$$\frac{\ell(x(t;\mathbf{x}_0,u_{\mathbf{x}_0}),\mathbf{u}_{\mathbf{x}_0}(t))}{\|\mathbf{x}_0\|_Q^2} \leq c(t) \quad \forall\, t \geq 0 \text{ and } \mathbf{x}_0 \in X. \tag{10}$$

Since $\mathbf{u}_{\mathbf{x}_0}(t)$ is defined over a finite time, there exists a time $\bar{t} > 0$ such that $c(t) = 0$ for all $t > \bar{t}$. Thus, the bounded growth function $B(t)$ can be computed by

$$B(t) = \int_0^t c(s)\mathrm{d}s.$$

4.1. Trajectory Generation and Growth Function B

The calculation of the growth function is based on designing an open-loop control maneuver for which the running costs ℓ can be evaluated and the function $c(t)$, from Inequality (10), can be computed. For the initial condition $\mathbf{x}_0 = (x_{0,1}, x_{0,2}, x_{0,3})^\top \in X$, we design an open-loop control maneuver that steers the robot to $\mathbf{x}^\star = (0,0,0)^\top$ in finite time. Since the considered mobile robot is holonomic, we choose the following straight trajectories for each state in X

$$\mathbf{x}(t) = \left(\frac{t_m - t}{t_m}\right)\mathbf{x}_0 \quad \forall\, t \in [0, t_m), \tag{11}$$

where $t_m \in \mathbb{R}_{>0}$ is the time required to perform the maneuver chosen such that the constraints U given by (4) are satisfied for all $\mathbf{x}_0 \in X$. Then, the resulting open-loop control actions required to achieve the maneuver can be calculated by

$$\mathbf{u}_{\mathbf{x}_0}(t) = \mathcal{R}(x_3(t))^{-1}\dot{\mathbf{x}}(t), \qquad t \in [0, t_m), \tag{12}$$

where $\mathcal{R}(\cdot)$ is the rotation matrix given by Eq. (2) and

$$\dot{\mathbf{x}}(t) = -\frac{1}{t_m}\mathbf{x}_0.$$

Now, we assume the following on the weighting matrices Q and R of the running costs defined by Eq. (5)

$$r_1 = r_2, \text{ and } r_i \leq \lambda q_i, \quad i \in \{1,2,3\}. \tag{13}$$

Applying the open-loop control actions in (12), leads to the running costs $\ell(x(t;\mathbf{x}_0,u_{\mathbf{x}_0}),\mathbf{u}_{\mathbf{x}_0}(t))$ given by

$$\left(\frac{t_m - t}{t_m}\right)^2 \sum_{i=1}^{3} q_i x_{0,i}^2 + \frac{1}{t_m^2}\sum_{i=1}^{3} r_i x_{0,i}^2,$$

where the first assumption in (13) is used. We remark that, by squaring and then adding the first two inputs in Eq. (12), we get $u_1^2 + u_2^2 = \dot{x}_1^2 + \dot{x}_2^2$. As a result, employing the second assumption in (13) leads to the estimate

$$\ell(x(t;\mathbf{x}_0,u_{\mathbf{x}_0}),\mathbf{u}_{\mathbf{x}_0}(t)) \leq \left[\left(\frac{t_m - t}{t_m}\right)^2 + \frac{\lambda}{t_m^2}\right] \underbrace{\sum_{i=1}^{3} q_i x_{0,i}^2}_{\|\mathbf{x}_0\|_Q^2},$$

Therefore, Inequality (10) holds with

$$c(t) := \left[\left(\frac{t_m - t}{t_m}\right)^2 + \frac{\lambda}{t_m^2}\right] \qquad \forall t \in [0, t_m). \tag{14}$$

Finally, $c(t) \equiv 0$ on $t \in [t_m, \infty)$. In conclusion, the constructed function c is of class $\mathcal{PC}(\mathbb{R}_{\geq 0},\mathbb{R}_{\geq 0})$, bounded, and integrable on $[0,\infty)$. Hence, it can be used to define $B : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ by $B(t) = \int_0^t c(s)\,\mathrm{d}s$ such that Inequality (8) holds for all $\mathbf{x}_0 \in X$. Note that the calculated

growth function $B(t)$ depends only on the time of the maneuver t_m and the ratio λ. We remark that the maneuvers designed for the non-holonomic mobile robots in the studies [23, 35] can be generally used, here, for the holonomic mobile robots, i.e. by restricting u_2 to zero. However, exploiting the holonomy of the considered system in this paper leads to less conservative estimate on the prediction horizon length as will be seen in the subsequent sections.

4.2. Maneuver Time t_m

The maneuver time $t_m \in \mathbb{R}_{>0}$ is the time required to perform the proposed open-loop control maneuver presented in the previous subsection while the control constraints U, given by Eq. (4), are satisfied. Therefore, t_m is lower bounded by the maneuver time $\bar{t}_m$ corresponding to the initial condition $\mathbf{x}_0 = (\bar{x}_1, \bar{x}_2, \pi)^\top$, i.e. the farthest initial condition in the state set X from the origin $\mathbf{x}^\star = (0,0,0)^\top$. $\bar{t}_m$ can be calculated via the following formula

$$\bar{t}_m = \frac{\max_{x_3(t)\in[0,\pi]} \|t_m \cdot H\mathbf{u}(t)\|_\infty}{r \cdot \bar{\omega}}, \tag{15}$$

where $\mathbf{u}(t)$ is given by Eq. (12). For simplicity, the maximization part in the numerator of Eq. (15) can be performed via a line-search. In the special case of $\bar{x}_1 = \bar{x}_2$, $\bar{t}_m$ reduces to

$$\bar{t}_m = \frac{\bar{x}_1 + \bar{x}_2 + \pi \cdot (a+b)}{r \cdot \bar{\omega}},$$

for a holonomic robot with mecanum wheels. In essence, choosing the maneuver time as $t_m \geq \bar{t}_m$ will satisfy the control constraints for any initial condition $\mathbf{x}_0 \in X$. For three different initial conditions and $t_m = 2$, Fig. 3 shows the open-loop trajectories $\mathbf{x}(t)$ and $\mathbf{u}_{\mathbf{x}_0}(t)$ given by Eq. (11) and Eq. (12), respectively.

4.3. Stabilizing Horizon Length

A prediction horizon length $\hat{T}$ such that a holonomic mobile robot is asymptotically stable under the proposed MPC scheme (Algorithm 1) can now be defined as.

$$\hat{T} = \min\left\{T > 0 \,|\alpha_{T,\delta} > 0\right\},$$

where $\alpha_{T,\delta}$ is calculated via Theorem 2 employing the growth function B derived in Section 4.1. Fig. 4 shows the effect of the ratio λ and the open-loop maneuver time t_m on $\alpha_{T,\delta}$ for different prediction horizon lengths T. As it can be noticed, a shorter stabilizing horizon length $\hat{T}$ can be achieved by reducing the ratio λ. Moreover, increasing the time t_m reduces (improves) the required stabilizing horizon length $\hat{T}$. From this observation, we can conjecture that asymptotic stability can be ensured for the shortest prediction horizon $T = N\tau = 2\tau$, by simply prolonging the maneuver time t_m. This observation will be analyzed in detail in the discrete-time setting in the following section.

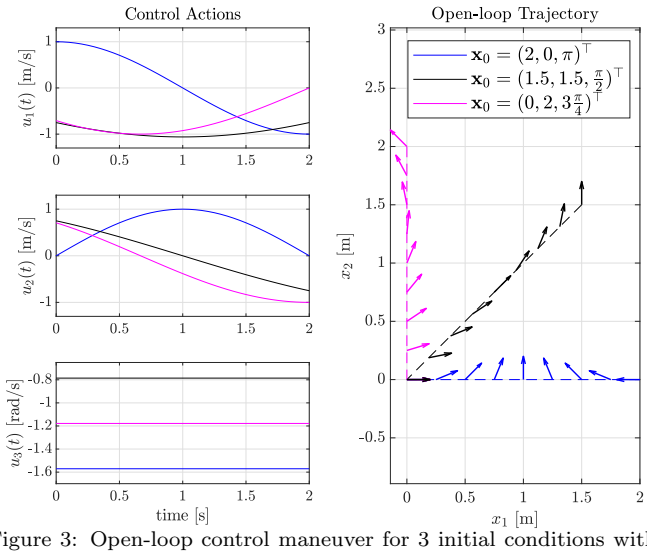

Figure 3: Open-loop control maneuver for 3 initial conditions with $t_m = 2$ [s]. Right: positions and orientations of the robot, along the trajectories, represented by arrows. Left: the corresponding control actions.

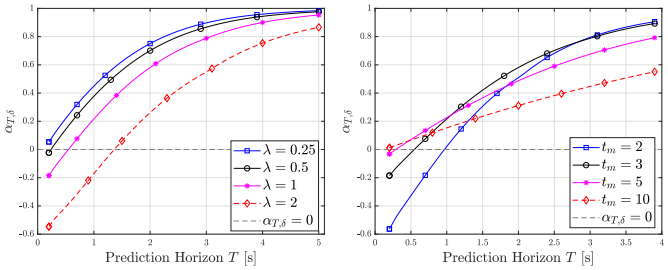

Figure 4: The performance index $\alpha_{T,\delta}$ (with $\delta = 0.1$ [s]) calculated via Theorem 2. Left: the effect of changing the ratio λ on $\alpha_{T,\delta}$ for an open-loop maneuver time $t_m = 2$ [s]. Right: the effect of changing t_m on $\alpha_{T,\delta}$ for $\lambda = 1$.

5. Stability Results in the Discrete Time Setting

In this section, first, we present the discrete time formulation of the MPC Algorithm 1. Second, we show that the closed-loop asymptotic stability of the considered holonomic mobile robot is guaranteed for the shortest prediction horizon length. Here, we note that the basic idea is similar to the one presented in [36, Example 4.3], where an extremely simplified two dimensional model of a holonomic robot is used. However, a direct application of these results is not possible in our context due to the more involved system dynamics given by Eq. (16). Third, we derive a closed-form solution for the MPC performance bound in this case. Finally, we establish a connection between the results of the continuous- and discrete-time settings.

5.1. Discrete Time MPC

As the proposed MPC controller is typically applied in a zero-order hold fashion, we consider the discrete time formulation of the considered problem. Here, model (1) is discretized by the sampling period $\tau > 0$, and assuming piecewise constant control during each sampling interval;

thus, the resulting (*exact*) discrete-time model is written as

$$\mathbf{x}^+ = f_{e,\tau}(\mathbf{x},\mathbf{u}) = \mathbf{x} + \mathcal{R}_e(\tau, x_3, u_3)\mathbf{u}, \tag{16}$$

where the superscript $(\cdot)^+$ denotes the next state and

$$\mathcal{R}_e(\tau, x_3, u_3) = \begin{pmatrix} \frac{\sin(x_3+\tau u_3)-\sin(x_3)}{u_3} & \frac{\cos(x_3+\tau u_3)-\cos(x_3)}{u_3} & 0 \\ \frac{\cos(x_3)-\cos(x_3+\tau u_3)}{u_3} & \frac{\sin(x_3+\tau u_3)-\sin(x_3)}{u_3} & 0 \\ 0 & 0 & \tau \end{pmatrix}.$$

We remark that for $u_3 = 0$ the model given by Eq. (16) reduces to

$$\begin{aligned} \mathbf{x}^+ &= \mathbf{x} + \lim_{u_3 \to 0} \mathcal{R}_e(\tau, x_3, u_3)\mathbf{u} \\ &= \mathbf{x} + \tau \begin{pmatrix} u_1 \cos(x_3) - u_2 \sin(x_3) \\ u_1 \sin(x_3) + u_2 \cos(x_3) \\ 0 \end{pmatrix}. \end{aligned}$$

In analogy to the continuous-time case, a control sequence $u = (\mathbf{u}(0), \mathbf{u}(1), \ldots, \mathbf{u}(N-1)) \in U^N$ of length $N \in \mathbb{N}$ is said to be admissible, for state $\mathbf{x}_0$, denoted by $u \in \mathcal{U}^N(\mathbf{x}_0)$ if the state trajectory

$$x_u(\cdot; \mathbf{x}_0) = (x_u(0; \mathbf{x}_0), x_u(1; \mathbf{x}_0), \ldots, x_u(N; \mathbf{x}_0)),$$

generated iteratively by the model (16) and stemming from $\mathbf{x}_0 =: x_u(0, \mathbf{x}_0)$, satisfies $x_u(n, \mathbf{x}_0) \in X$ for all $n \in \{0, 1, \ldots, N\}$.

Now, we adapt the MPC Algorithm 1 to the discrete-time case. Here, the prediction horizon and the time shift are defined by N and m, respectively, such that $T = N\tau$ and $\delta = m\tau$ hold. Moreover, we define the discrete-time running costs $\ell_\tau(\mathbf{x}, \mathbf{u}) : X \times U \to \mathbb{R}_{\geq 0}$ by

$$\ell_\tau(\mathbf{x}, \mathbf{u}) = \|\mathbf{x}\|_Q^2 + \|\mathbf{u}\|_R^2 \tag{17}$$

in analogy to its continuous-time counterpart ℓ given by Eq. (5). As a result, at a sampling step $k \in \mathbb{N}_0$, the discrete-time cost function $J_N : X \times U^N \to \mathbb{R}_{\geq 0}$ and the corresponding value function $V_N : X \to \mathbb{R}_{\geq 0}$ are

$$\begin{aligned} J_N(\mathbf{x}_k, u) &:= \sum_{n=k}^{k+N-1} \ell_\tau(x_u(n, \mathbf{x}_k), \mathbf{u}(n)) \\ V_N(\mathbf{x}_k) &:= \inf_{u \in \mathcal{U}^N(\mathbf{x}_k)} J_N(\mathbf{x}_k, u) \end{aligned} \tag{18}$$

for $N \in \mathbb{N} \cup \{\infty\}$. For the further analysis in this section, we set $m = 1$. Thus, at the sampling step $k \in \mathbb{N}_0$, the MPC feedback law in the discrete-time setting, denoted by $\mu_N : X \to U$, is given by $\mu_N(k, \mathbf{x}_k) = \mathbf{u}^\star(0, \mathbf{x}_k)$, where $u^\star = u^\star(\cdot, \mathbf{x}_k) \in \mathcal{U}^N(\mathbf{x}_k)$ satisfies $V_N(\mathbf{x}_k) = J_N(\mathbf{x}_k, u^\star)$.

The asymptotic stability of system (16) under the discrete-time MPC scheme without terminal conditions can now be concluded by the following theorem, see, e.g. [19, Theorems 4.2 and 5.3] for details and for the proof of Theorem 3.

Theorem 3. *Let a monotonically increasing and bounded sequence $(\gamma_i)_{i \in \mathbb{N}}$ be given such that the inequality*

$$V_i(\mathbf{x}_0) \leq \gamma_i \cdot \|\mathbf{x}_0\|_Q^2 \qquad \forall\, i \in \mathbb{N} \text{ and } \mathbf{x}_0 \in X \tag{19}$$

holds. Additionally, let the performance index α_N be given by the formula

$$\alpha_N := 1 - \frac{(\gamma_N - 1)\prod_{k=2}^N (\gamma_k - 1)}{\prod_{k=2}^N \gamma_k - \prod_{k=2}^N (\gamma_k - 1)}. \tag{20}$$

Then, if $\alpha_N > 0$ holds, the relaxed Lyapunov inequality

$$V_N(f_{e,\tau}(\mathbf{x}, \mu_N(\mathbf{x}))) \leq V_N(\mathbf{x}) - \alpha_N \ell_\tau(\mathbf{x}, \mu_N(\mathbf{x}))$$

holds for all $\mathbf{x} \in X$ and the MPC closed-loop with prediction horizon N is asymptotically stable. Moreover, the performance bound

$$V_\infty^{\mu_N}(\mathbf{x}) := \sum_{n=0}^{\infty} \ell_\tau(x_{\mu_N}(n, \mathbf{x}), \mu_N(n, \mathbf{x})) \leq \alpha_N^{-1} V_\infty(\mathbf{x}) \tag{21}$$

is guaranteed, where $\mu_N(n, \mathbf{x}) := \mathbf{u}^\star(0, x_{\mu_N}(n, \mathbf{x}))$ and $V_\infty^{\mu_N}(\mathbf{x})$ are the MPC closed-loop control and closed-loop costs on the infinite horizon, respectively. □

We note that the required assumption on the boundedness of $\ell_\tau(\mathbf{x}, \mathbf{u})$ for Theorem 3 holds for the chosen running costs. In essence, Theorem 3 is the discrete time counterpart of Theorem 2, where the time shift δ is set to the sampling time τ, see [33] for more details.

5.2. Growth Bounds γ_i

The growth bound γ_i in Inequality (19) can be obtained using $\gamma_i = \sum_{j=0}^{i-1} c_j$, $i \in \mathbb{N}_{\geq 2}$, where the sequence $c_j \geq 0$, $j \in \mathbb{N}_0$, is summable, i.e. $\sum_{j=0}^{\infty} c_j < \infty$. c_j is calculated such that the inequality

$$\frac{\ell_\tau(x_{u_{\mathbf{x}_0}}(j; \mathbf{x}_0), \mathbf{u}_{\mathbf{x}_0}(j))}{\|\mathbf{x}_0\|_Q^2} \leq c_j, \quad \forall j \in \mathbb{N}_0 \text{ and } \mathbf{x}_0 \in X \tag{22}$$

holds for an admissible sequence of control actions $u_{\mathbf{x}_0} = \mathbf{u}_{\mathbf{x}_0}(j), j \in \mathbb{N}_0$, see, e.g. [23] for more details. The following proposition summarizes the procedure of computing c_j in Inequality (22) and, thus, the growth bound γ_i in Theorem 3.

Proposition 4. *Let the following assumptions*

$$r_1 = r_2, q_1 = q_2, r_1 \leq \lambda q_1/2, \text{ and } r_3 \leq \lambda q_3, \tag{23}$$

hold for the weighting matrices Q and R in Eq. (17). Then, cost controllability, i.e. Inequality (19), holds with $\gamma_i = \sum_{j=0}^{i-1} c_j$, $i \in \mathbb{N}_{\geq 2}$, where c_j is defined by

$$c_j := \left[\left(\frac{k_m - j}{k_m} \right)^2 + \frac{\lambda}{\tau^2 k_m^2} \right] \tag{24}$$

for $j \in \{0, 1, \ldots, k_m - 1\}$ with $k_m \in \mathbb{N}_{\geq 2}$. □

Proof. The sequence c_j can be obtained in a similar procedure as $c(t)$ given by Eq. (14). Thus, we choose the following straight trajectories for each initial state $\mathbf{x}_0 = (x_{0,1}, x_{0,2}, x_{0,3})^\top$ in X to the origin:

$$\mathbf{x}(j) = \left(\frac{k_m - j}{k_m}\right)\mathbf{x}_0,\ \mathbf{x}_0 := \mathbf{x}(0), \quad \forall j \in \{0, \dots, k_m - 1\},$$

where $k_m \in \mathbb{N}_{\geq 2}$ is chosen sufficiently large such that the constraints U given by Eq. (4) are satisfied for all $\mathbf{x}_0 \in X$. In essence, k_m can be calculated as $k_m = \lceil t_m/\tau \rceil$ for t_m introduced in Section 4. Then, the resulting open-loop control actions required to achieve the maneuver can be calculated by

$$\mathbf{u}_{\mathbf{x}_0}(j) = \mathcal{R}_e(\tau, x_3(j), u_3(j))^{-1}\left(-\frac{1}{k_m}\mathbf{x}_0\right), \tag{25}$$

for all $j \in \{0, \dots, k_m - 1\}$, where

$$\mathcal{R}_e(\tau, x_3, u_3)^{-1} = \frac{u_3}{4}\begin{pmatrix} \frac{\sin(x_3+\tau u_3)-\sin(x_3)}{\sin^2(0.5\tau u_3)} & \frac{\cos(x_3+\tau u_3)-\cos(x_3)}{\sin^2(0.5\tau u_3)} & 0 \\ \frac{\cos(x_3)-\cos(x_3+\tau u_3)}{\sin^2(0.5\tau u_3)} & \frac{\sin(x_3+\tau u_3)-\sin(x_3)}{\sin^2(0.5\tau u_3)} & 0 \\ 0 & 0 & \frac{4}{\tau u_3} \end{pmatrix};$$

moreover, later in this proof, we drop the subscript $\mathbf{x}_0$ from all the elements of $\mathbf{u}_{\mathbf{x}_0}(j)$ to keep the presentation technically simple. By inspecting Eq. (25), it is straightforward to obtain $u_3(j)$ as

$$u_3(j) = -\frac{x_{3,0}}{\tau k_m},\ \forall j \in \{0, \dots, k_m - 1\}.$$

Now, applying the open-loop control actions defined by Eq. (25) and using the first assumption in (23) on the weighting matrix R, lead to the following running costs along the resulting open-loop trajectories

$$\ell_\tau(x_{u_{\mathbf{x}_0}}(j; \mathbf{x}_0), \mathbf{u}_{\mathbf{x}_0}(j)) = \left(\frac{k_m - j}{k_m}\right)^2 \sum_{i=1}^{3} q_i x_{0,i}^2 + r_1\left(u_1(j)^2 + u_2(j)^2\right) + r_3\frac{x_{0,3}^2}{\tau^2 k_m^2},$$

where the term $r_1\left(u_1(j)^2 + u_2(j)^2\right)$ can be calculated by squaring then adding the first two equations in (25); thus, we have

$$r_1\left(u_1(j)^2 + u_2(j)^2\right) = \frac{r_1 u_3(j)^2}{4\sin^2(0.5\tau u_3(j))}\left(\frac{x_{0,1}^2}{k_m^2} + \frac{x_{0,2}^2}{k_m^2}\right).$$

Using the Taylor series expansion of $\sin^2(0.5\tau u_3(j))$, we have

$$\sin^2(0.5\tau u_3(j)) \geq \left(\frac{\tau^2 u_3(j)^2}{4} - \frac{\tau^4 u_3(j)^4}{48}\right) = \frac{\tau^2 u_3(j)^2}{4}\left(1 - \frac{\tau^2 u_3(j)^2}{12}\right) > 0,$$

which is valid for all $u_3(j)^2 := (x_{0,3}/\tau k_m)^2 \leq (\pi/\tau k_m)^2$. This leads to the estimate

$$\begin{aligned} r_1\big(u_1(j)^2 + u_2(j)^2\big) &\leq \frac{12 \cdot r_1}{12 - \left(\frac{x_{0,3}^2}{k_m^2}\right)}\left(\frac{x_{0,1}^2}{\tau^2 k_m^2} + \frac{x_{0,2}^2}{\tau^2 k_m^2}\right) \\ &\leq \frac{12 \cdot r_1}{12 - \left(\frac{\pi^2}{k_m^2}\right)}\left(\frac{x_{0,1}^2}{\tau^2 k_m^2} + \frac{x_{0,2}^2}{\tau^2 k_m^2}\right). \end{aligned} \tag{26}$$

The term $12/\left(12 - \frac{\pi^2}{k_m^2}\right)$ has an upper bound of 1.26 for $k_m \geq 2$. Choosing now the weight r_1 as in Assumption (23), leads to the estimate

$$r_1\left(u_1(j)^2 + u_2(j)^2\right) \leq \lambda \cdot q_1\left(\frac{x_{0,1}^2}{\tau^2 k_m^2} + \frac{x_{0,2}^2}{\tau^2 k_m^2}\right). \tag{27}$$

We note that the term $u_3(j)^2/4\sin^2(0.5\tau u_3(j))$ can be defined for $u_3(j) = 0$ as

$$\lim_{u_3(j)\to 0} \frac{u_3(j)^2}{4\sin^2(0.5\tau u_3(j))} = \frac{1}{\tau^2}.$$

Therefore, the estimate (27) holds for $u_3(j) = 0$ too.

As a result, using the assumption $r_3 \leq \lambda q_3$ from Eq. (23), the running costs $\ell_\tau(x_{u_{\mathbf{x}_0}}(j; \mathbf{x}_0), \mathbf{u}_{\mathbf{x}_0}(j))$ can be estimated by

$$\ell_\tau(x_{u_{\mathbf{x}_0}}(j; \mathbf{x}_0), \mathbf{u}_{\mathbf{x}_0}(j)) \leq \left[\left(\frac{k_m - j}{k_m}\right)^2 + \frac{\lambda}{\tau^2 k_m^2}\right]\underbrace{\sum_{i=1}^{3} q_i x_{0,i}^2}_{\|\mathbf{x}_0\|_Q^2},$$

for all $j \in \{0, \dots, k_m\}$. The sequence c_j in Inequality (22) can now be obtained as in Eq. (24), where the $c_j \equiv 0$ for all $j > k_m$. □

Now the growth bound γ_k, $k \in \mathbb{N}_0$ can be computed as

$$\begin{aligned} \gamma_k := \sum_{j=0}^{k-1} c_j &= \frac{1}{k_m^2}\sum_{j=0}^{k-1}\left[j^2 - 2k_m \cdot j + k_m^2 + \frac{\lambda}{\tau^2}\right] \\ &= \frac{1}{k_m^2}\left[\left(k_m^2 + \frac{\lambda}{\tau^2}\right)k + \sum_{j=0}^{k-1} j^2 - 2k_m \cdot \sum_{j=0}^{k-1} j\right]. \end{aligned}$$

Using the sum formulas

$$\sum_{j=0}^{k-1} j = \frac{k(k-1)}{2} \text{ and } \sum_{j=0}^{k-1} j^2 = \frac{k(k-1)(2k-1)}{6},$$

γ_k reduces to

$$\frac{1}{k_m^2}\left[\frac{k^3}{3} - \left(k_m + \frac{1}{2}\right)k^2 + \left(k_m^2 + k_m + \frac{\lambda}{\tau^2} + \frac{1}{6}\right)k\right]. \tag{28}$$

Similar to the continuous-time case, γ_k only depends on the number of steps k_m, the sampling time τ, and the weight ratio λ.

5.3. Asymptotic Stability for the Shortest Prediction Horizon

The shortest possible prediction horizon of discrete-time MPC without terminal constraints or costs is $N = 2$. In this case, one control and the successor state are dominating the cost function given by Eq. (18). As a result, the special structure of γ_k, given by Eq. (28), leads to the following theorem.

Theorem 5. *Given $N = 2$, the weight ratio λ of Assumption (23) and the sampling time τ, there exists a maneuver length $k_m^\star$ such that for all $k_m > k_m^\star$, the performance index $\alpha_N = \alpha_2$ given by (20), is ensured to be positive; thus, the closed-loop asymptotic stability of the considered holonomic mobile robot under MPC without terminal conditions is guaranteed for the shortest possible prediction horizon. Moreover, there exists a maneuver length $k_m > k_m^\star$ such that an upper bound on the performance index α_2, denoted by $\bar{\alpha}_2$, can be computed and, thus, a lower bound on the MPC closed-loop performance is estimated via Inequality (21) as*

$$V_\infty^{\mu_2}(\mathbf{x}) \leq \bar{\alpha}_2^{-1} V_\infty(\mathbf{x}). \tag{29}$$

□

Proof. For $N = 2$, the performance estimate α_N in Eq. (20) reduces to

$$\alpha_2 = 1 - (\gamma_2 - 1)^2. \tag{30}$$

Thus, the asymptotic stability condition can be written as

$$0 < \gamma_2 < 2. \tag{31}$$

From Eq. (28), γ_2 is given by

$$\gamma_2 = \frac{1}{k_m^2}\left[2k_m^2 - 2k_m + \left(\frac{2\lambda}{\tau^2} + 1\right)\right]. \tag{32}$$

Then, as $\gamma_2 > 0$ for all $k_m \geq 1$, the left inequality of (31) is ensured. Moreover, choosing k_m as

$$k_m > k_m^\star := \left\lceil \frac{\lambda}{\tau^2} + \frac{1}{2} \right\rceil \tag{33}$$

satisfies the right inequality of (31).

The upper bound on α_2 can be computed by finding k_m such that $\frac{\partial \alpha_2}{\partial k_m} = 0$. This reduces to finding k_m such that $\frac{\partial \gamma_2}{\partial k_m} = 0$. Differentiating γ_2 given by Eq. (32) with respect to k_m results in

$$\frac{\partial \gamma_2}{\partial k_m} = \frac{2}{k_m^2}\left[1 - \frac{1}{k_m}\left(\frac{2\lambda}{\tau^2} + 1\right)\right].$$

For $\frac{\partial \gamma_2}{\partial k_m} = 0$, we have

$$k_m = \frac{2\lambda}{\tau^2} + 1 > k_m^\star.$$

Using this value, we can then compute an upper bound on α_2 and, thus, a lower bound on the MPC closed-loop performance, via Eq. (30) as

$$\bar{\alpha}_2 := \frac{\lambda/\tau^2 + 1/4}{(\lambda/\tau^2 + 1/2)^2}. \tag{34}$$

□

Essentially, it is shown by Theorem 5 that the closed-loop asymptotic stability can be guaranteed by sufficiently increasing the open-loop maneuver number of steps k_m. Moreover, it can be inferred from the lower bound on the MPC performance $\bar{\alpha}_2$ given by Eq. (34) that the index is always positive and converges to 1 as λ/τ^2 is chosen small. Therefore, by means of Inequality (29), infinite horizon MPC performance can be approached by reducing λ/τ^2 despite using the shortest prediction horizon, i.e. $N = 2$.

We remark that prolonging the open-loop maneuver time/steps leads to more conservative growth bounds $B(t)$ in Eq. (8) and γ_i in Eq. (19), see the definition of both. Therefore, prolonging the open-loop time does not affect the MPC Algorithm 1.

5.4. Connection between Continuous and Discrete Time Results

It is proved in Theorem 5 that closed-loop asymptotic stability of the considered control problem is guaranteed for $N = 2$ for a maneuver time of $k_m > \lceil \lambda/\tau^2 + 0.5 \rceil$. It can also be inferred from Proposition 4 that the sequence c_j in Eq. (24) is a direct discretization of the function $c(t)$ in Eq. (14), but with different assumptions on r_1 and r_2 of the weighting matrix R, i.e. the second assumption in (13) and the third assumption in (23). Now, the estimate given by Inequality(26) can be written as

$$r_1\big(u_1(j)^2 + u_2(j)^2\big) \leq \frac{12 \cdot r_1}{12 - \left(\frac{\pi^2}{\lceil \lambda/\tau^2 + 0.5 \rceil^2}\right)}\left(\frac{x_{0,1}^2}{\tau^2 k_m^2} + \frac{x_{0,2}^2}{\tau^2 k_m^2}\right).$$

Here, we have the limit

$$\lim_{\tau \to 0} \frac{\pi^2}{\lceil \lambda/\tau^2 + 0.5 \rceil^2} = 0,$$

which results in the estimate (27) with the assumptions on the weights r_1 and r_2 given by Assumption (13). Therefore, it can be concluded that as a shorter sampling time τ is employed for the exact discrete-time model given by Eq. (16), the sequence c_j in Eq. (24) resembles the exact discretization of the function $c(t)$ in Eq. (14) with *only* the assumptions on the weighting matrix R introduced in Assumption (13).

We, additionally, remark that the derived sequence c_j in Eq. (24) is also valid for the Euler discretization of the model in Eq. (1) given by

$$\mathbf{x}^+ = \mathbf{x} + \tau \cdot \mathcal{R}(x_3)\mathbf{u}$$

with only the assumptions on the weight made in Assumption (13). This is because, in this case, the resulting control sequence is

$$\mathbf{u}_{\mathbf{x}_0}(j) = \mathcal{R}(x_3(j))^{-1}\left(-\frac{1}{k_m \cdot \tau}\mathbf{x}_0\right),$$

$j \in \{0, \ldots, k_m - 1\}$, which is the exact discretization of the control function given by Eq. (12).

6. Numerical and Experimental Results

In this section, we show the numerical and the experimental results of implementing Algorithm 1. Herein, the kinematic equation of the holonomic robot shown in (16) is used for state prediction. The mecanum wheels robot (*Summit-XL-Steel* [25]) is used in the experiments, see Fig. 1. States are constrained to

$$X := [-2, 2]^2 \times [-\pi, \pi].$$

The input constraints are as described in Eq. (4), where $r = 0.127$ m is the wheel radius, $\bar{\omega} = 12$ rad/s is the maximum wheel speed, $a = 0.223$ m is half the wheel base length and $b = 0.205$ m is half the wheel base width, see Fig. 1 and the specification of the experimental platform [25]. This results in the maneuver time $\bar{t}_m = 3.5$ s, which ensures control constraints satisfaction for all initial conditions in X, by means of Eq. (15).

The simulations as well as the experiments were conducted until the following condition is met

$$||\mathbf{x}(n) - \mathbf{x}^\star|| \leq 0.05,$$

where $\mathbf{x}(n)$ is the state measurement at time step $n \in \mathbb{N}$ and $\mathbf{x}^\star = (0, 0, 0)^\top$ is the reference set point.

6.1. Numerical Results

The simulations were run for a series of eight different initial conditions, which were all then stabilized to the origin. The initial orientation was set to $x_{0,3} = \pi$ for each of the initial conditions, as this is the maximum deviation form the reference orientation $x_3^\star = 0$. Simulations were conducted using MATLAB; the optimal control

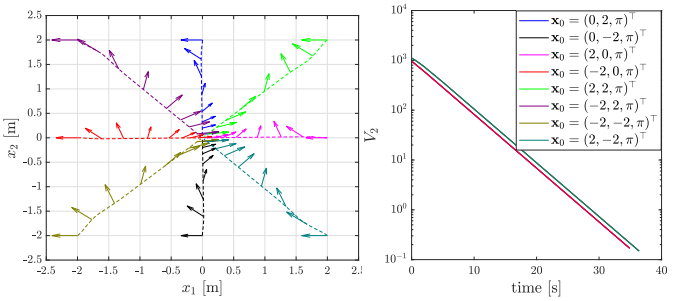

Figure 5: Left: Closed-loop paths for the simulation results for $N = 2$; the dashed line represents the robot's path and the arrow represent its orientation along the path. Right: value function evolution in simulation for $N = 2$.

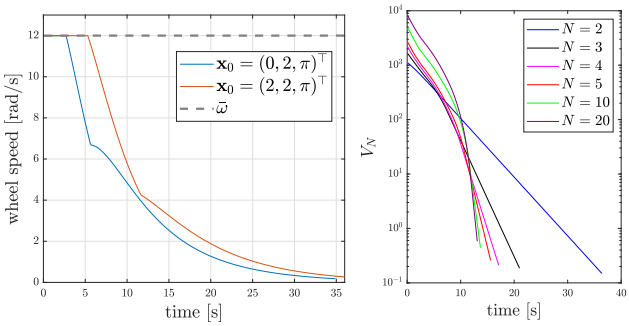

Figure 6: Left: maximum wheel speed for two initial conditions in simulation. Right: evolution of value function at different prediction horizon length N where $\lambda = 0.05$, $\tau = 0.025$ s, and $\mathbf{x}_0 = (2, 2, \pi)^\top$.

problem (OCP) (6) was set up symbolically using CasADi toolbox [37]. Here, the multiple-shooting discretization method [38] is used to convert the OCP to a nonlinear programming problem (NLP). The NLP is then solved using the Interior-Point method implemented in IPOPT [39]. The running costs are defined by the way of using Eq. (17) with $\lambda = 0.05$. Additionally, the prediction horizon is set to $N = 2$ and the sampling time to $\tau = 0.025$ s. Here, the time shift $\delta = m\tau = \tau$; the maneuver length $k_m^\star$ required for asymptotic stability is $k_m^\star = 81$ by means of Theorem 5; and, by means of Eq. (34), we have the performance index $\bar{\alpha}_2 = 0.01238$.

The resulting closed-loop paths can be seen in Fig. 5 (left). It can be noted from these results that the optimal path for the holonomic mobile robot in each case is to take an almost straight path to the origin while the orientation of the robot changes as the robot moves towards the desired position. The maximum wheel speed for two of the initial conditions is shown in Fig. 6 (left). The results confirm that the input constraints are not violated, i.e. Eq.(4) is satisfied for all time steps; the same behavior is observed for the remaining initial conditions.

To show that the system is asymptotically stable, we can observe that the evolution of the value function is monotonically decreasing with time, see Fig. 5 (right). Therefore, we can verify that the conditions for Theorem 3 are met for each initial condition in the simulation. Moreover, as the prediction horizon length is $N = 2$, we can find for each initial condition that $V_2(\mathbf{x}_0)/\ell_\tau^\star(\mathbf{x}_0) \leq \gamma_2 < 2$, see Conditions (19) and (31). In Table 1, we see that this holds for all initial conditions in the simulation.

Through running these simulations, we observed some interesting results of varying parameters such as λ, the

Table 1: $V_2(\mathbf{x}_0)/\ell_\tau^\star(\mathbf{x}_0)$ for each initial condition in simulations

$\mathbf{x}_0^\top$	$\frac{V_2(\mathbf{x}_0)}{\ell_\tau^\star(\mathbf{x}_0)}$	$\mathbf{x}_0^\top$	$\frac{V_2(\mathbf{x}_0)}{\ell_\tau^\star(\mathbf{x}_0)}$
$(0, 2, \pi)$	1.9877	$(2, 2, \pi)$	1.9887
$(0, -2, \pi)$	1.9877	$(-2, 2, \pi)$	1.9887
$(2, 0, \pi)$	1.9877	$(-2, -2, \pi)$	1.9887
$(-2, 0, \pi)$	1.9877	$(2, -2, \pi)$	1.9887

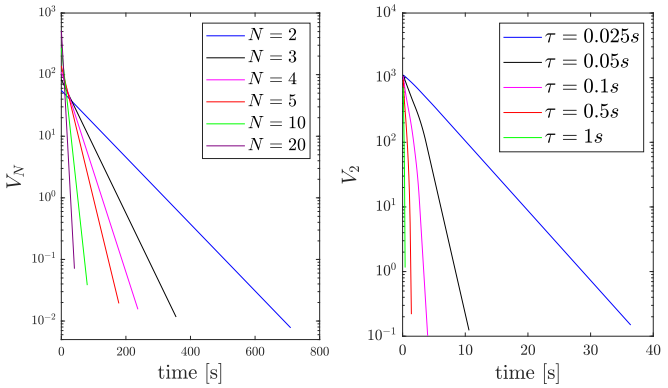

Figure 7: Left: evolution of the value function for different prediction horizon lengths N, where $\lambda = 1$ and $\tau = 0.025$ s. Right: evolution of the value function for different sampling intervals τ, where $\lambda = 0.05$ and $N = 2$. For the two figures, we have $\mathbf{x}_0 = (2, 2, \pi)^\top$.

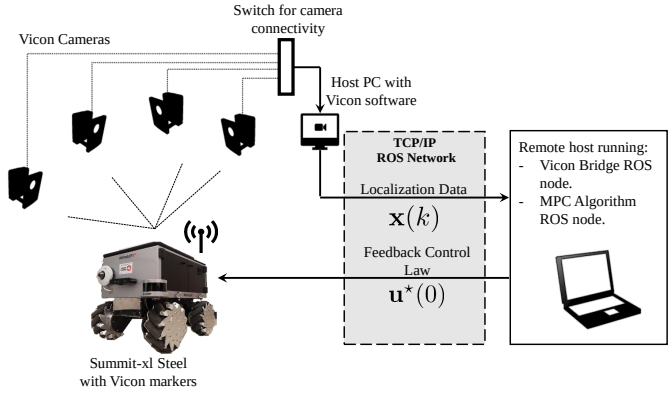

Figure 8: Block diagram of the experimental setup used to validate Algorithm 1.

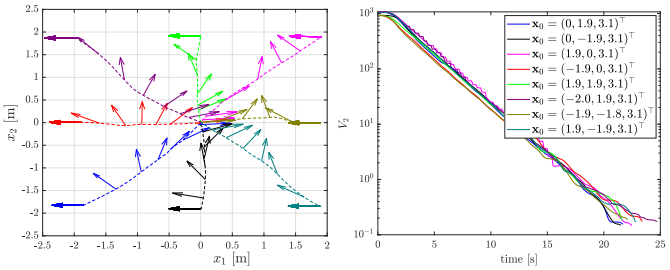

Figure 9: Left: closed-loop paths for the experimental results with prediction horizon length $N = 2$; the dashed line represents the robot's path and the arrow represent its orientation. Right: value function evolution in experiments for $N = 2$.

prediction horizon N, and the sampling time τ on the evolution of the value function V_N. First, a lower λ resulted in a faster convergence of the mobile robot to the set point, which is fairly intuitive; a lower λ means a lower ratio on the weights for the control inputs compared to the weights for the state. This will result in a solution of the OCP (6) with less conservative control actions, and therefore faster convergence of the robot to the set point. What is less intuitive is the effect of the prediction horizon and the sampling time, we explore both here. For these simulations, we used an initial condition of $\mathbf{x}_0 = (2, 2, \pi)^\top$. Fig. 6 (right) shows the effect of the prediction horizon length. Here, we see that larger prediction horizons produce a faster convergence of the robot to the set point. We also see that the evolution of the value function for $N = 2$ is roughly linear in the logarithmic scale, where for longer prediction horizon lengths the value function evolution becomes steeper. Finally, we can also observe diminishing returns in terms of faster convergence to the set point for increasing the prediction horizon; going from $N = 2$ to $N = 3$ reduces convergence time by about 15 s, while going from $N = 10$ to $N = 20$ has a difference of less than 1 s.

Now, we investigate the evolution of the value function for $\lambda = 1$. In Fig. 7 (left), we see the effect of having $\lambda = 1$ for different prediction horizon lengths; convergence to the set point takes significantly longer time for smaller N. Additionally, we can see that the evolution of V_N in each case is nearly linear, not only for $N = 2$. Fig. 7 (right) shows the results of our investigation for the effect of τ on the evolution of the value function V_N. Here, $\lambda = 0.05$ and $N = 2$ are kept constant and τ is varied. There is an apparent tendency that increasing τ reduces the convergence time.

6.2. Experimental Results

Algorithm 1 was validated experimentally using a *Summit-XL-Steel* holonomic mobile platform, see Fig. 1. In the experiments, the mobile platform is stabilized to the origin from eight different initial positions. The experimental platform runs through the Robot Operating System (ROS) [40]; thus, a ROS node was developed in Python to implement Algorithm 1 in which CasADi toolbox was used to symbolically setup the MPC controller with the same settings used in the simulations. Here, the running costs in Eq. (5) is used with $\lambda = 0.05$. Additionally, the prediction horizon was set to $N = 2$ and the sampling time to $\tau = 0.025$ s.

To ensure that the localization of the robot would not affect our results, a *Vicon* motion capture system adopting 12 cameras was used to localize the mobile platform. The position of the robot from the Vicon system was transmitted to ROS using a Vicon bridge [41]. The block diagram of the experimental setup is shown in Fig. 8.

The results for eight different initial conditions can be seen in Fig. 9 (left). The value function V_2 evolution was tracked for each case. Moreover, a comparison of the closed-loop paths in experiments and their simulations counterparts is presented in Fig. 10. In Fig. 9 (right), we see that, in each case, V_2 is monotonically decreasing towards zero until the experiment is stopped, except for few moments for only three initial conditions; this is because of the unmodeled dynamics of the robot and/or floor surface irregularities. The results indicates the (inherent) robustness of the proposed controller.

Additionally, as can be seen in Fig. 10, the closed-

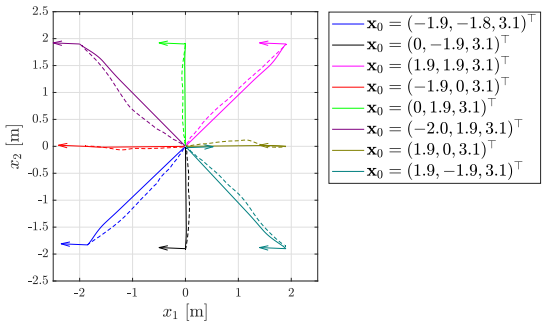

Figure 10: Comparison of the experimental closed-loop paths (dashed line) and simulation closed-loop paths (solid line); initial and final orientations are represented by arrows.

Table 2: $V_2(\mathbf{x}_0)/\ell_\tau^\star(x_0)$ for each initial condition in experiments

$\mathbf{x}_0^\top$	$\frac{V_2(\mathbf{x}_0)}{\ell_\tau^\star(\mathbf{x}_0)}$
$(0.00, 1.91, 3.11)$	1.9876
$(0.00, -1.90, 3.14)$	1.9877
$(1.89, -0.01, 3.14)$	1.9886
$(-1.88, 0.00, 3.13)$	1.9869
$(1.91, 1.90, 3.13)$	1.9875
$(-1.96, 1.87, 3.13)$	1.9877
$(-1.85, -1.83, 3.14)$	1.9857
$(1.89, -1.85, 3.13)$	1.9887

loop paths of the experiments are very similar to that for the simulations especially for the initial conditions on the horizontal and vertical axes. Finally, we verify that the Conditions (19) and (31) are met for each initial condition in the experiments. In Table 2, we see that $V_2(\mathbf{x}_0)/\ell_\tau^\star(\mathbf{x}_0) \leq \gamma_2 < 2$ holds for all initial conditions in the experiments. In summary, experimental results show very good alignment with theory and simulations.

7. Conclusions

In this paper, we rigorously show cost controllability in order to ensure asymptotic stability of a desired set point for holonomic mobile robots using MPC without stabilizing terminal conditions. To this end, we construct a growth function by using suitably designed open-loop control maneuvers. We present the analysis, first, in the continuous-time setting before extending the results to the discrete-time setting. On the one hand, the conducted analysis allows to conclude asymptotic stability for the shortest possible prediction horizon. On the other hand, we also provide a bound on the MPC closed-loop costs in comparison to the infinite-horizon optimal controller. Here, we recover (almost) optimal performance for suitably chosen parameters in the running costs. Results are verified by a series of numerical simulations and real-time lab experiments confirming the theoretical findings; here, the effect of changing the MPC tuning parameters on the closed-loop performance is investigated in detail.

Future work will include the extension of the performed analysis to other interesting control problems, e.g. trajectory (target) tracking [42, 43] and path-following control [34].

Acknowledgment

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), [funding reference numbers PDF-532957-2019 (M.W. Mehrez), STPGP 506987 (S. Jeon)]. K. Worthmann gratefully acknowledges funding from the German research foundation (DFG; grant WO 2056/6-1).

References

[1] R. Siegwart, I. Nourbakhsh, Introduction to Autonomous Mobile Robots, The MIT Press, 2004.
[2] C. Wang, X. Liu, X. Yang, F. Hu, A. Jiang, C. Yang, Trajectory Tracking of an Omni-Directional Wheeled Mobile Robot Using a Model Predictive Control Strategy, Applied Sciences 8 (2) (2018) 231.
[3] T. Faulwasser, Optimization-based Solutions to Constrained Trajectory-tracking and Path-following Problems, Shaker, Aachen, Germany, 2013.
[4] T. P. Nascimento, C. E. T. Dórea, L. M. G. Goncalves, Nonholonomic mobile robots' trajectory tracking model predictive control: a survey, Robotica 36 (5) (2018) 676–696.
[5] M. W. Mehrez, Optimization Based Solutions for Control and State Estimation in Nonholonomic Mobile Robots: Stability, Distributed Control, and Localization, Ph.D. thesis, Memorial University of Newfoundland (2017).
[6] T. Kalmár-Nagy, R. D'Andrea, P. Ganguly, Near-optimal dynamic trajectory generation and control of an omnidirectional vehicle, Robotics and Autonomous Systems 46 (1) (2004) 47 – 64.
[7] A. S. Conceicao, A. P. Moreira, P. J. Costa, Trajectory tracking for omni-directional mobile robots based on restrictions of the motor's velocities, IFAC Proceedings Volumes 39 (15) (2006) 121 – 125, 8th IFAC Symposium on Robot Control.
[8] H.-C. Huang, C.-C. Tsai, Adaptive trajectory tracking and stabilization for omnidirectional mobile robot with dynamic effect and uncertainties, IFAC Proceedings Volumes 41 (2) (2008) 5383 – 5388, 17th IFAC World Congress.
[9] X. Li, A. Zell, Motion Control of an Omnidirectional Mobile Robot, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 181–193.
[10] K. Kanjanawanishkul, A. Zell, Path following for an omnidirectional mobile robot based on model predictive control, in: 2009 IEEE International Conf. on Robotics and Automation, 2009, pp. 3341–3346.
[11] K. Kanjanawanishkul, MPC-Based path following control of an omnidirectional mobile robot with consideration of robot constraints, Advances in Electrical and Electronic Engineering 13 (1) (2015) 54–63.
[12] X. Liu, H. Chen, C. Wang, F. Hu, X. Yang, MPC Control and Path Planning of Omni-Directional Mobile Robot with Potential Field Method, Springer International Publishing, 2018, pp. 170–181.
[13] K. Kanjanawanishkul, X. Li, A. Zell, Nonlinear model predictive control of omnidirectional mobile robot formations, Intelligent Autonomous Systems 10, IAS 2008 (2008) 41–48.
[14] J. B. Rawlings, D. Q. Mayne, M. M. Diehl, Model Predictive Control: Theory, Computation, and Design, 2nd Edition, Nob Hill Publishing, 2018.
[15] M. Zanon, A. Boccia, V. G. S. Palma, S. Parenti, I. Xausa, Direct Optimal Control and Model Predictive Control, Springer International Publishing, Cham, 2017, pp. 263–382.

[16] S. Keerthi, E. Gilbert, Optimal infinite-horizon feedback laws for a general class of constrained discrete-time systems: Stability and moving-horizon approximations, Journal of Optimization Theory and Applications 57 (2) (1988) 265–293.
[17] H. Chen, F. Allgöwer, A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability, Automatica 34 (10) (1998) 1205–1218.
[18] S. Tuna, M. Messina, A. Teel, Shorter horizons for model predictive control, in: Proceedings of the American Control Conference, 2006, pp. 863–868.
[19] L. Grüne, J. Pannek, M. Seehafer, K. Worthmann, Analysis of unconstrained nonlinear MPC schemes with time varying control horizon, SIAM J. Control Optimization 48 (8) (2010) 4938–4962.
[20] H. X. Araújo, A. G. Conceicão, G. H. Oliveira, J. Pitanga, Model predictive control based on LMIs applied to an omnidirectional mobile robot, IFAC Proceedings Volumes 44 (1) (2011) 8171 – 8176, 18th IFAC World Congress.
[21] L. Grüne, J. Pannek, Nonlinear Model Predictive Control, Springer International Publishing, 2017, pp. 45–69.
[22] J.-M. Coron, L. Grüne, K. Worthmann, Model predictive control, cost controllability, and homogeneity, arXiv1906.05112 (2019).
[23] K. Worthmann, M. W. Mehrez, M. Zanon, G. K. I. Mann, R. G. Gosine, M. Diehl, Model predictive control of nonholonomic mobile robots without stabilizing constraints and costs, IEEE Trans. Control Syst. Technol. 24 (4) (2016) 1394–1406.
[24] M. Reble, F. Allgöwer, Unconstrained model predictive control and suboptimality estimates for nonlinear continuous-time systems, Automatica 48 (8) (2012) 1812–1817.
[25] Robotnik, SUMMIT- XL STEEL Datasheet (2018).
URL `www.robotnik.eu`
[26] K. Lynch, F. Park, Modern Robotics: Mechanics, Planning, and Control, Cambridge Univeristy Press, 2017.
[27] E. B. Lee, L. Markus, Foundations of Optimal Control Theory, The SIAM Series in Appl. Mathematics, John Wiley & Sons New York, London, Sydney, 1967.
[28] K. Worthmann, M. W. Mehrez, G. K. Mann, R. G. Gosine, J. Pannek, Interaction of open and closed loop control in MPC, Automatica 82 (2017) 243–250.
[29] L. Grüne, A. Rantzer, On the infinite horizon performance of receding horizon controllers, IEEE Transactions on Automatic Control 53 (9) (2008) 2100–2111.
[30] B. Lincoln, A. Rantzer, Relaxing dynamic programming, IEEE Transactions on Automatic Control 51 (8) (2006) 1249–1260.
[31] M. Reble, F. Allgöwer, Unconstrained Nonlinear Model Predictive Control and Suboptimality Estimates for Continuous-Time Systems, in: Proc. 18th IFAC World Congr., Milano, Italy, 2011, pp. 6733–6738.
[32] A. Boccia, L. Grüne, K. Worthmann, Stability and feasibility of state constrained MPC without stabilizing terminal constraints, Syst. Control Lett. 72(8) (2014) 14–21.
[33] K. Worthmann, M. Reble, L. Grüne, F. Allgöwer, The role of sampling for stability and performance in unconstrained nonlinear model predictive control, SIAM J. Control and Optimization 52 (1) (2014) 581–605.
[34] M. W. Mehrez, K. Worthmann, G. K. Mann, R. G. Gosine, T. Faulwasser, Predictive path following of mobile robots without terminal stabilizing constraints, IFAC-PapersOnLine 50 (1) (2017) 9852–9857.
[35] K. Worthmann, M. W. Mehrez, M. Zanon, G. K. I. Mann, R. G. Gosine, M. Diehl, Regulation of Differential Drive Robots using Continuous Time MPC without Stabilizing Constraints or Costs, IFAC-PapersOnLine: 5th IFAC Conf. Nonlinear Model Predictive Control, Seville, Spain 48 (23) (2015) 129–135.
[36] L. Grüne, K. Worthmann, Distributed Decision Making and Control, no. 417 in Lecture Notes in Control and Information Sciences, Springer Verlag, 2012, Ch. A distributed NMPC scheme without stabilizing terminal constraints.
[37] J. Andersson, A General-Purpose Software Framework for Dynamic Optimization, PhD thesis, Arenberg Doctoral School, KU Leuven, Dept. Elect. Eng., Belgium (October 2013).
[38] D. B. Leineweber, I. Bauer, H. G. Bock, J. P. Schlöder, An efficient multiple shooting based reduced sqp strategy for large-scale dynamic process optimization. part 1: theoretical aspects, Computers and Chemical Engineering 27 (2) (2003) 157 – 166.
[39] A. Wächter, T. L. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, Math. Programming 106 (1) (2006) 25–57.
[40] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng, Ros: an open-source robot operating system, in: ICRA Workshop on Open Source Software, 2009.
[41] M. Achtelik, vicon_bridge, accessed: 2019-08-01.
URL `https://github.com/ethz-asl/vicon_bridge`
[42] T. P. Nascimento, C. E. T. Dórea, L. M. G. Goncalves, Nonlinear model predictive control for trajectory tracking of nonholonomic mobile robots: A modified approach, International Journal of Advanced Robotic Systems 15 (1) (2018) 1–14.
[43] T. P. Nascimento, G. F. Basso, C. E. T. Dórea, L. M. G. Goncalves, Perception-driven motion control based on stochastic nonlinear model predictive controllers, IEEE/ASME Transactions on Mechatronics 24 (4) (2019) 1751–1762.