ZAMM · Z. Angew. Math. Mech. **94**, No. 4, 317 – 330 (2014) / **DOI** 10.1002/zamm.201100133

Stability and performance guarantees for model predictive control algorithms without terminal constraints

Jürgen Pannek[1,*] **and Karl Worthmann**[2,**]

[1] University of the Federal Armed Forces, 85577 Munich, Germany
[2] TU Ilmenau, 98693 Ilmenau, Germany

Received 28 September 2011, revised 26 February 2013, accepted 18 December 2013
Published online 3 February 2014

Key words Asymptotic stability, feedback, model predictive control algorithm, performance, receding horizon control.

A typical bottleneck of model predictive control algorithms is the computational burden in order to compute the receding horizon feedback law which is predominantly determined by the length of the prediction horizon. Based on a relaxed Lyapunov inequality we present techniques which allow us to show stability and suboptimality estimates for a reduced prediction horizon. In particular, the known structural properties of suboptimality estimates based on a controllability condition are used to cut the gap between theoretic stability results and numerical observations.

1 Introduction

Model predictive control (MPC), also termed receding horizon control, is a well established method for optimal control of linear and nonlinear systems [6, 27] and also widely used in industry, cf. [3, 17]. The method generates a sequence of finite horizon optimal control problems in order to approximate the solution of an infinite horizon optimal control problem, the latter being, in general, computationally intractable. Each solution of such a finite horizon control problem yields a sequence of control values of which only the first element is implemented at the corresponding time instant. This procedure is iteratively applicable and generates a closed loop static state feedback. Due to the replacement of the original infinite by a sequence of finite horizon control problems, stability of the resulting closed loop may be lost.

To ensure stability, several modifications of the finite horizon control problems such as stabilizing terminal constraints [20] or a local Lyapunov function [8] as an additional terminal weight have been proposed. Since the construction of suitably chosen terminal costs, e.g. a local control Lyapunov function, is a challenging task, within this work, we focus on the computationally attractive form of the MPC methodology without stabilizing terminal constraints and/or costs. For such a control loop feasibility, stability, and suboptimality has been shown in [14, 30] via a relaxed Lyapunov inequality, see also [26, 28] for the linear case. However, since either a controllability condition or knowledge on the optimal value function of the finite horizon subproblems is required in order to deduce concise bounds on the prediction horizon length, the computation of an appropriate receding horizon invariant initial set satisfying the respective assumptions is also demanding – in particular for nonlinear systems. Furthermore, the obtained results, e.g. performance bounds, may be conservative since stable behavior of the MPC controlled closed loop can be observed in many examples although stability can be guaranteed by means of these theoretical results for a longer prediction horizon only, cf. [29].

Here, our main goal is to reduce this conservatism with respect to the estimated prediction horizon length which is of particular interest since the computational burden of the MPC algorithm grows rapidly with the prediction horizon. Hence, we want to design an MPC algorithm which ensures desired stability properties at runtime for a given initial condition for small prediction horizons. To this end, we also employ a relaxed Lyapunov inequality but do not aim at verifying it at each sampling instant. Our starting point is a result shown in [16]: implementing more than only the first element of the computed open loop sequence of control values and checking the relaxed Lyapunov inequality at the next update time enhances the suboptimality bound from [14] for a large class of control systems. While the idea of utilizing several open loop elements has already been discussed in [9], doing so may be harmful in terms of robustness, cf. [23]. To cover this issue, conditions are presented, which guarantee that the stability condition is maintained and the control loop is closed more often.

[*] E-mail: juergen.pannek@googlemail.com
[**] Corresponding author E-mail: karl.worthmann@tu-ilmenau.de

Additionally, we deduce stability and performance estimates which allow us to violate the relaxed Lyapunov inequality temporarily, much alike the watchdog technique used in nonlinear optimization [7]. To this end, we use an idea similar to [10] and monitor the progress made in the decrease of the value function along the closed loop. Here, we point out that these conditions are checkable at runtime of the algorithm which renders such an approach applicable in practice.

The paper is organized as follows: In the following section the considered MPC problem and a trajectory based stability theorem from [15] are presented. Sect. 3 generalizes this result to an MPC setting which allows for implementing longer parts of the sequence of control values computed at an update time instant. Furthermore, a first algorithm utilizing this weaker condition is derived. To carry over the improvements of the suboptimality bound, an algorithmic approach is presented in Sect. 4. The issue of an exit strategy is addressed in Sect. 5 which allows us to temporarily violate the relaxed Lyapunov inequality and to further improve previously derived suboptimality estimates. Last, conclusions are drawn in Sect. 6. Throughout the work a simple example by the proposed results provides insight into the improvements achieved.

2 Problem setup and preliminaries

Let $\mathbb{N}_0$ denote the natural numbers including zero and $\mathbb{R}_0^+$ the nonnegative real numbers. A continuous function $\gamma : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ is said to be of class $\mathcal{K}_\infty$ if it satisfies $\gamma(0) = 0$, is strictly increasing and unbounded. Furthermore, a continuous function $\beta : \mathbb{R}_0^+ \times \mathbb{R}_0^+ \to \mathbb{R}_0^+$ is of class $\mathcal{KL}$ if it is strictly decreasing in its second argument with $\lim_{t\to\infty} \beta(r,t) = 0$ for each $r > 0$ and satisfies $\beta(\cdot, t) \in \mathcal{K}_\infty$ for each $t \geq 0$.

Let X and U be arbitrary Banach spaces equipped with the metrics $d_X : X \times X \to \mathbb{R}_0^+$ and $d_U : U \times U \to \mathbb{R}_0^+$. In this work nonlinear discrete time control systems of the form

$$x(n+1) = f(x(n), u(n)), \quad x(0) = x_0 \tag{1}$$

for $n \in \mathbb{N}_0$ are considered. $x(n) \in \mathbb{X} \subset X$ and $u(n) \in \mathbb{U} \subset U$ represent the state and control of the system at time instant n. Hence, X and U stand for the state and the control value space, respectively. Since X and U are arbitrary metric spaces, the following results are applicable to discrete time dynamics induced by a sampled – finite or infinite dimensional – system, see, e.g., [2, 18]. Here, using subsets $\mathbb{X}$ and $\mathbb{U}$ allows for incorporating constraints. Throughout this work, the space of control sequences $u : \mathbb{N}_0 \to \mathbb{U}$ is denoted by $\mathbb{U}^{\mathbb{N}_0}$.

For control system (1) we want to construct a static state feedback $u = \mu(x)$ which asymptotically stabilizes the system at a desired equilibrium $x^\star$ via model predictive control. The trajectory generated by the feedback map $\mu : \mathbb{X} \to \mathbb{U}$ is denoted by $x_\mu(n) = x_\mu(n; x_0)$, $n = 0, 1, \ldots$. In order to define asymptotic stability rigorously, the open ball with center x and radius r is denoted by $\mathcal{B}_r(x)$ and the distance from x to the equilibrium $x^\star$ is represented by $\|x\|_{x^\star} := d_X(x, x^\star)$.

Definition 2.1. Let a static state feedback $\mu : \mathbb{X} \to \mathbb{U}$ satisfying $f(x^\star, \mu(x^\star)) = x^\star$ for system (1) be given, i.e., $x^\star$ is an equilibrium for the closed loop. Then, $x^\star$ is said to be (locally) *asymptotically stable* if there exist $r > 0$ and a function $\beta(\cdot,\cdot) \in \mathcal{KL}$ such that, for each $x_0 \in \mathcal{B}_r(x^\star) \cap \mathbb{X}$,

$$x_\mu(n; x_0) \in \mathbb{X} \qquad \text{and} \qquad \|x_\mu(n; x_0)\|_{x^\star} \leq \beta(\|x_0\|_{x^\star}, n) \quad \forall\, n \in \mathbb{N}_0. \tag{2}$$

Without loss of generality existence of a control value $u^\star \in \mathbb{U}$ such that $f(x^\star, u^\star) = x^\star$ holds is assumed throughout this work. The quality of the feedback is evaluated in terms of the infinite horizon cost functional

$$J_\infty(x_0, u) = \sum_{n=0}^{\infty} \ell(x_u(n), u(n)) \tag{3}$$

with continuous stage cost $\ell : X \times U \to \mathbb{R}_0^+$ satisfying $\ell(x^\star, u^\star) = 0$ and $\ell(x, u) > 0$ for all $u \in U$ for each $x \neq x^\star$. The optimal value function corresponding to (3) is denoted by $V_\infty(x_0) = \inf_{\{u \in \mathbb{U}^{\mathbb{N}_0} : x_u(n) \in \mathbb{X}\, \forall n \in \mathbb{N}\}} J_\infty(x_0, u)$. Now, utilizing Bellman's principle of optimality for the optimal value function $V_\infty(\cdot)$ yields the Lyapunov equation

$$V_\infty(x_0) = \inf_{\{u \in \mathbb{U} : f(x_0, u) \in \mathbb{X}\}} \{\ell(x_0, u) + V_\infty(f(x_0, u))\}. \tag{4}$$

In order to avoid technical difficulties, let the infimum of this equality with respect to u be attained, i.e., there exists a feasible control value $u^\star_{x_0}$, i.e. $u^\star_{x_0} \in \mathbb{U}$ with $f(x_0, u^\star_{x_0}) \in \mathbb{X}$, such that $V_\infty(x_0) = \ell(x_0, u^\star_{x_0}) + V_\infty(f(x_0, u^\star_{x_0}))$ holds. $u^\star_{x_0}$ is called the argmin of the right hand side of (4) – albeit uniqueness of the minimizer $u^\star_{x_0}$ is not required. In case of uniqueness the argmin-operator can be understood as an assignment, otherwise it is just a convenient way of writing "$u^\star_{x_0}$ minimizes the right hand side of (4)". Hence, an optimal feedback law on the infinite horizon is defined via

$$\mu_\infty(x(n)) := \operatorname*{argmin}_{\{u \in \mathbb{U} : f(x(n), u) \in \mathbb{X}\}} \{\ell(x(n), u) + V_\infty(f(x(n), u))\} \tag{5}$$

The computation of the control law (5) requires the solution of a Bellman equation which is, in general, very hard to solve. In order to avoid this burden, we utilize a model predictive control approach without terminal constraints and/or costs to approximate the infinite horizon optimal control. Here, we like to point out that, if a terminal cost can be constructed such that the "cost to go" is sufficiently well approximated as in quasi infinite horizon optimal control, see, e.g. [8], its incorporation can contribute to the controller performance. However, if the domain of the terminal cost is too small, then the needed coupling terminal constraint may lead to drawbacks from a performance point of view.

MPC consists of three distinct steps: First, given measurements of the current state x_0, an optimal sequence of control values over a truncated and, thus, finite horizon is computed which minimizes the cost functional

$$J_N(x_0, u) = \sum_{k=0}^{N-1} \ell(x_u(k; x_0), u(k)).$$

Here, $x_u(\cdot; x_0)$ denotes the trajectory emanating from x_0 corresponding to the input signal $u(\cdot)$. As a result, we obtain the open loop control $u_N(\cdot; x_0) = \operatorname{argmin}_{x_u(n) \in \mathbb{X} \text{ for } n \in \{1,2,\dots,N\}} J_N(x_0, u)$ and the corresponding open loop state trajectory

$$x_{u_N}(n+1; x_0) = f(x_{u_N}(n; x_0), u_N(n; x_0)) \tag{6}$$

for $n = 0, \dots, N-1$ with initial value $x_{u_N}(0; x_0) = x_0$. In a second step, the first element of the open loop control is employed in order to define the feedback law $\mu_N(x_0) := u_N(0; x_0)$. Last, the feedback is applied to the system under control revealing the closed loop system

$$x_{\mu_N}(n+1; x_0) = f(x_{\mu_N}(n; x_0), \mu_N(x_{\mu_N}(n; x_0))) \tag{7}$$

for which the abbreviation $x_{\mu_N}(\cdot)$ will be used whenever the initial value x_0 is clear. The closed loop costs with respect to the feedback law $\mu_N(\cdot)$ are given by $V_\infty^{\mu_N}(x_0) = \sum_{n=0}^{\infty} \ell(x_{\mu_N}(n), \mu_N(x_{\mu_N}(n)))$ and $V_N(x_0)$ denotes the optimal value function corresponding to $J_N(x_0, \cdot)$.

In the literature endpoint constraints or a Lyapunov function type endpoint weight are used to ensure stability of the closed loop, see, e.g., [8, 11, 19, 20]. Here, we focus on an MPC version without these modifications. Hence, feasibility of the MPC scheme is an issue that cannot be neglected. In order to exclude a scenario in which the closed loop trajectory runs into a dead end, the following viability condition is assumed.

Assumption 2.2 *For each $x \in \mathbb{X}$ there exists a control value $u_x \in \mathbb{U}$ satisfying $f(x, u_x) \in \mathbb{X}$.*

We like to stress that the computation of such a control invariant set $\mathbb{X}$ is a hard task. However, suitable methods exists, see, e.g. [21]. Furthermore, we refer to [14, 26] for a detailed discussion on feasibility issues in MPC without terminal constraints and/or costs.

In order to be applicable at runtime of the algorithm, we present our conditions in a trajectory based setting, i.e. contrary to Definition 2.1 we suppose conditions to be satisfied along a particular closed loop trajectory only. While such conditions are less restrictive, we stress this difference by calling the closed loop solution to behave like a solution of an asymptotically stable system. According to [15, Proposition 7.6], stable behavior of the closed loop trajectory can be guaranteed for such a controller using relaxed dynamic programming, cf. [22].

Theorem 2.3. *(i) Consider the feedback law $\mu_N : \mathbb{X} \to \mathbb{U}$ and the closed loop trajectory $x_{\mu_N}(\cdot)$ of* (7) *with initial value $x_0 \in \mathbb{X}$. If the value function $V_N : \mathbb{X} \to \mathbb{R}_0^+$ satisfies*

$$V_N(x_{\mu_N}(n)) \geq V_N(x_{\mu_N}(n+1)) + \alpha \ell(x_{\mu_N}(n), \mu_N(x_{\mu_N}(n))) \qquad \forall\, n \in \mathbb{N}_0 \tag{8}$$

for some $\alpha \in (0, 1]$, then the performance estimate

$$\alpha V_\infty(x_{\mu_N}(n)) \leq \alpha V_\infty^{\mu_N}(x_{\mu_N}(n)) \leq V_N(x_{\mu_N}(n)) \leq V_\infty(x_{\mu_N}(n)) \tag{9}$$

holds for all $n \in \mathbb{N}_0$.
(ii) If, in addition, there exist $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$ such that $\alpha_1(\|x\|_{x^\star}) \leq V_N(x) \leq \alpha_2(\|x\|_{x^\star})$ and $\ell(x, u) \geq \alpha_3(\|x\|_{x^\star})$ hold for $x = x_{\mu_N}(n) \in \mathbb{X}$, $n \in \mathbb{N}_0$, then there exists $\beta \in \mathcal{KL}$ which only depends on $\alpha_1, \alpha_2, \alpha_3$, and α such that (2) *holds, i.e., $x_{\mu_N}(\cdot)$ behaves like a trajectory of an asymptotically stable system.*

Within Theorem 2.3 the relaxed Lyapunov inequality (8) is the key assumption. From the literature, it is well-known that this condition is satisfied for sufficiently long horizons N, cf. [1, 12, 19], and that a suitable N may be computed via methods described in [16, 24, 30]. The suboptimality index α in this inequality can be interpreted as a lower bound for the rate of convergence.

Remark 2.4 (Performance measure)**.** Inequality (9) yields a performance bound for the MPC closed loop in comparison to the optimal costs on the infinite horizon. This assessment of the achieved closed loop performance is particularly appealing for $\alpha \notin \{0, 1\}$ since MPC with terminal cost yields either (approximately) optimal performance, i.e. $\alpha \approx 1$ or solely stability $\alpha > 0$ without rigorous bounds on the degree of suboptimality.

3 Shortening the prediction horizon by weakening the relaxed Lyapunov inequality

For the described MPC setting we want to guarantee stability and a certain lower bound on the degree of suboptimality with respect to the infinite horizon optimal control law $\mu_\infty(\cdot)$. Yet, at the same time the optimization horizon N shall be as small as possible. These goals oppose each other since it is known from the literature, see, e.g., [1, 12, 19], that stability can only be guaranteed if the optimization horizon is sufficiently long. Keeping N small, however, is important from a practical point of view since the horizon length is the dominating factor regarding the computational burden. Here, motivated by theoretical results deduced in [16], our goal is to improve the stability and suboptimality bounds from [15, Proposition 7.6] and, thereby, to allow the use of smaller prediction horizons.

Note that Condition (8) is a sufficient, yet not a necessary condition. In particular, even if stability and the desired performance Estimate (9) cannot be guaranteed via Theorem 2.3, stable and satisfactory behavior of the closed loop may still be observed, even in the linear case, as illustrated by the following example.

Example 3.1. Let $\mathbb{X} = X = \mathbb{R}^2$, $\mathbb{U} = U = \mathbb{R}$ be given and consider the linear control system $x(n+1) = Ax(n) + Bu(n)$ with quadratic stage cost function $\ell(x, u) := x^\top Qx + u^\top Ru$ from [28] with

$$A = \begin{pmatrix} 1 & 1.1 \\ -1.1 & 1 \end{pmatrix}, \qquad B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \qquad Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \text{and} \quad R = \begin{pmatrix} 1 \end{pmatrix}.$$

For the given system, the optimal finite horizon costs $V_N(x_0) = x_0^T P_N x_0$ be can be computed via the Riccati difference equation using the initial condition $P_1 = Q$ according to

$$P_{j+1} = A^\top \left[P_j - P_j B \left(B^\top P_j B + R \right)^{-1} B^\top P_j \right] A + Q.$$

Then, the resulting open loop control law $u_N(\cdot, x)$ is given by

$$u_N(k; x) = -\left(B^\top P_{N-k} B + R \right)^{-1} B^\top P_{N-k} A x_{u_N}(k; x),$$

$k = 0, 1, \ldots, N-2$, and $u_N(N-1; x) = 0$, cf. [24]. Now, we implement the Riccati feedback in the usual receding horizon fashion $\mu_N(x(n)) := u_N(0; x(n))$ with $N = 3$ and evaluate the suboptimality bound from Theorem 2.3 along the closed loop. As displayed in Fig. 1 (left) a typical trajectory tends towards the origin quickly. However, Fig. 1 (right) shows that for the set of initial values $\mathcal{X} := \{(\cos(2\pi k/k_{\max}), \sin(2\pi k/k_{\max})) \mid k = 1, 2, \ldots, k_{\max}\}$ with $k_{\max} = 2^7$ there exists a nonempty subset of initial values for which we obtain $\alpha < 0$ in (8). Hence, stability cannot be deduced from Theorem 2.3. Here, we like to mention that stability can be guaranteed for all initial values $x_0 \in \mathcal{X}$ by means of Theorem 2.3 if we choose $N = 4$. Note that in [24, Sect. 6] it has been shown in a similar manner that the closed loop is stable if $N \geq 5$. We like to point out that the approach presented in [24] for the unconstrained linear quadratic case exploits the connection between stability properties of the system reflected by the solution of the corresponding Riccati equation and the monotonicity of the cost function, see also [5].

Motivated by the fact that the computational effort grows rapidly with respect to the prediction horizon N, it is not desirable to choose N larger than necessary to guarantee the relaxed Lyapunov inequality (8) to hold. Our goal in this work is to develop an algorithm which allows us to check whether stability and performance in the sense of Theorem 2.3 can be guaranteed for a prediction horizon which is smaller than the prediction horizon needed in order to ensure (8). In order to further motivate this approach, Example 3.1 is considered again.

Example 3.2. Instead of repeatedly considering only the first MPC step to evaluate α via (8), we define the respective quantity for every second MPC step via

$$\alpha_{N,2}(x_0) := \frac{V_N(x_0) - V_N(x_{\mu_N}(2; x_0))}{\sum_{n=0}^{1} \ell(x_{\mu_N}(n; x_0), \mu_N(x_{\mu_N}(n; x_0)))}$$

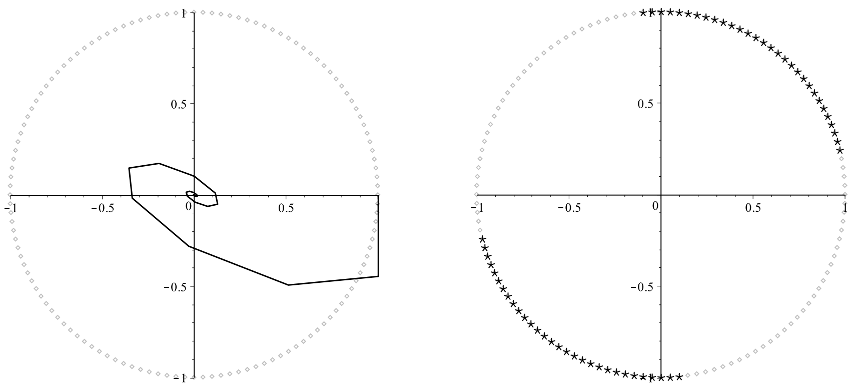

Fig. 1 Left: Illustration of the used set of initial values (grey) together with a typical closed loop solution (black). Right: Illustration of the subset of initial values (black) of $\mathcal{X}$ (grey) for which $\alpha \geq 0$ in (8) does not hold.

and obtain $\alpha_{N,2}(x_0) > 0$ for each $x_0 \in \mathcal{X}$. Hence, defining $\alpha := \inf_{x_0 \in \mathcal{X}} \alpha_{N,2}(x_0) > 0$ yields

$$V_N(x_0) \geq V_N(x_{\mu_N}(2;x_0)) + \alpha \left(\sum_{n=0}^{1} \ell(x_{\mu_N}(n;x_0), \mu_N(x_{\mu_N}(n;x_0))) \right), \tag{10}$$

i.e., the relaxed Lyapunov inequality with a positive suboptimality index α after two steps. Indeed, this conclusion holds true for all $x \in \mathbb{X} = X$ which can be shown by using the MPC feedback computed in Example 3.1. Hence, despite our conditions to be trajectory based, for this example asymptotic stability in the sense of Definition 2.1 can be concluded.

In particular, Example 3.2 shows that a generalized type of relaxed Lyapunov inequality similar to (10) may hold after implementing several controls despite the fact that the central assumption (8) of Theorem 2.3 is violated. Checking the relaxed Lyapunov inequality (8) less often may help in order to ensure desired stability properties of the resulting closed loop. We aim at designing a strategy which ensures – a priori and at runtime of the corresponding MPC algorithm – that a relaxed Lyapunov inequality is fulfilled after $m \in \{1, 2, \ldots, N-1\}$ steps.

Our first attempt is motivated by an observation from [16, Sect. 7]. In this reference estimates for the suboptimality degree α for a set of initial conditions are deduced and the following fact has been proven: if more than one element of the computed sequence of control values is applied, then the suboptimality estimate α is increasing (up to a certain point). To incorporate this idea into our MPC scheme, we first need to extend our notation. The list $\mathcal{S} = (\sigma(0), \sigma(1), \ldots) \subseteq \mathbb{N}_0$ is introduced, which is assumed to be in ascending order, in order to indicate time instants at which the control sequence is updated. The closed loop solution at time instant $\sigma(n)$ is denoted by $x_n = x_{\mu_N}(\sigma(n))$. Furthermore, the abbreviation $m_n := \sigma(n+1) - \sigma(n)$, i.e., the time between two MPC updates, is used. Hence,

$$x_{\mu_N}(\sigma(n) + m_n) = x_{\mu_N}(\sigma(n+1)) = x_{n+1} \tag{11}$$

holds. This enables us – in view of Bellman's principle of optimality – to define the closed loop control

$$\mu_N^{\mathcal{S}}(\cdot;x_n) := \operatorname*{argmin}_{\{u \in \mathbb{U}^{m_n} : x_u(k;x_n) \in \mathbb{X} \text{ for } k = 1,2,\ldots,m_n\}} \left\{ V_{N-m_n}(x_u(m_n;x_n)) + \sum_{k=0}^{m_n-1} \ell(x_u(k;x_n), u(k)) \right\}. \tag{12}$$

Describing the fact shown in [16, Sect. 7] more precisely, a lower bound on the degree of suboptimality α_{N,m_n} relative to the horizon length N and the number of controls to be implemented m_n can be obtained. This bound allows for measuring the tradeoff between the infinite horizon cost induced by the MPC feedback law $\mu_N^{\mathcal{S}}(\cdot;\cdot)$ similar to Theorem 2.3, i.e.

$$V_\infty^{\mu_N^{\mathcal{S}}}(x_0) := \sum_{n=0}^{\infty} \sum_{k=0}^{m_n-1} \ell\left(x_{\mu_N^{\mathcal{S}}}(k;x_n), \mu_N^{\mathcal{S}}(k;x_n) \right), \tag{13}$$

and the infinite horizon optimal value function $V_\infty(x_0)$ evaluated at x_0. We point out that the results shown in [16] ensure stability for a set of initial values. Hence, this approach may lead to a conservative performance estimate at least with respect to parts of the state space. Here, we extend (8) to an m-step relaxed Lyapunov inequality which is similar to [16] but applied in a trajectory based setting. Note that the controllability condition, which was used in order to derive these results, is, in general, difficult to check if state constraints have to be taken into account.

Proposition 3.3. *Let a list* $\mathcal{S} = (\sigma(n))_{n\in\mathbb{N}_0} \subseteq \mathbb{N}_0$ *such that* $m_n = \sigma(n+1) - \sigma(n) \in \{1, 2, \ldots, N-1\}$ *holds be given. Consider the open loop system* $x_{u_N}(\cdot;\cdot)$ *of* (6)*, the feedback law* $\mu_N^{\mathcal{S}}(\cdot;\cdot)$*, the closed loop trajectory* x_n*,* $n \in \mathbb{N}_0$*, of* (11) *with initial value* $x_0 \in \mathbb{X}$ *and a fixed* $\overline{\alpha} \in (0, 1]$*. If there exists a function* $V_N : \mathbb{X} \to \mathbb{R}_0^+$ *satisfying*

$$V_N(x_n) \geq V_N(x_{n+1}) + \overline{\alpha} \sum_{k=0}^{m_n-1} \ell(x_{u_N}(k; x_n), u_N(k; x_n)) \tag{14}$$

with $m_n \in \{1, \ldots, N-1\}$ *for all* $n \in \mathbb{N}_0$*, then*

$$\overline{\alpha} V_\infty(x_0) \leq \overline{\alpha} V_\infty^{\mu_N^{\mathcal{S}}}(x_0) \leq V_N(x_0) \leq V_\infty(x_0) \tag{15}$$

holds for $\mu_N^{\mathcal{S}}(\cdot;\cdot)$ *given by* (12) *for all* $n \in \mathbb{N}_0$.

Proof. Reordering (14), we obtain $\overline{\alpha} \sum_{k=0}^{m_n-1} \ell(x_{u_N}(k; x_n), u_N(k; x_n)) \leq V_N(x_n) - V_N(x_{n+1})$. Summing over all the first $n \in \mathbb{N}_0$ time instants yields

$$\overline{\alpha} \sum_{i=0}^{n} \sum_{k=0}^{m_i-1} \ell(x_{u_N}(k; x_i), u_N(k; x_i)) \leq V_N(x_0) - V_N(x_{n+1}) \leq V_N(x_0).$$

Hence, by definition of $\mu_N^{\mathcal{S}}(\cdot;\cdot)$ in (12) and $V_\infty^{\mu_N^{\mathcal{S}}}(\cdot)$ in (13), taking n to infinity implies the second inequality in (15). The first and the last inequality in (15) follow by the definition of the value functions V_N and V_∞ which concludes the proof. □

An implementation which aims at guaranteeing a fixed lower bound of the degree of suboptimality $\overline{\alpha}$ in the sense of Proposition 3.3 may take the following form:

```
Algorithm 3.4 Given state $x_0 := x$, $n = 0$, list $\mathcal{S} = (n)$, $N \in \mathbb{N}_{\geq 2}$ and $\overline{\alpha} \in (0,1]$

  (I) Set $j := 0$ and compute $u_N(\cdot; x_n)$ and $V_N(x)$. Do
        (a) Set $j := j + 1$, compute $V_N(x_{u_N}(j; x_n))$
        (b) Compute $\alpha = \{\overline{\alpha} \mid \overline{\alpha}$ maximally satisfies (14) with $m_n := j\}$
        (c) If $\alpha \geq \overline{\alpha}$: Set $m_n := j$ and goto (II)
        (d) If $j = N - 1$: Set $m_n$ according to exit strategy and goto (II)
      while $\alpha < \overline{\alpha}$
 (II) For $j = 1, \ldots, m_n$ do
          Implement $\mu_N^{\mathcal{S}}(j-1; x_n) := u_N(j-1; x_n)$
(III) Set $\mathcal{S} := (\mathcal{S}, back(\mathcal{S}) + m_n)$, $x_{n+1} := x_{\mu_N^{\mathcal{S}}}(m_n; x_n)$, $n := n + 1$ and goto (I)
```

Here, we adopted the programming notation *back* which allows for fast access to the last element of a list. Note that $\mathcal{S}$ is built up during runtime of the algorithm and not known in advance. Hence, $\mathcal{S}$ is always ordered.

Remark 3.5. If (14) is not satisfied for $j \leq N-1$, an exit strategy has to be used since the performance bound $\overline{\alpha}$ cannot be guaranteed. In order to cope with this issue, there exist remedies, e.g., one may increase the prediction horizon and repeat Step (I). While local validity of (14) can be ensured for sufficiently large N, extending the prediction horizon typically results in prolonged computing times. Therefore, if realtime guarantees are required, a respective bound for the initial length of the horizon needs to be satisfied. Additionally, the proof of Proposition 3.3 cannot be applied in this context due to the prolongation of the horizon. Yet, it can be replaced by estimates from [10, 15] for varying prediction horizons to obtain a result similar to (15). Alternatively, one may continue with the algorithm. If the exit strategy does not have to be called again, the algorithm guarantees the desired performance for x_n instead of x_0, i.e., from that point on.

Utilizing Algorithm 3.4 the following result shows asymptotically stable behavior of the computed state trajectory:

Theorem 3.6. *Suppose a control system* (1) *with initial value* $x_0 \in \mathbb{X}$ *and* $\overline{\alpha} \in (0, 1]$ *to be given and apply Algorithm* 3.4*. Assume that for each iterate* $n \in \mathbb{N}_0$ *condition* $\alpha \geq \overline{\alpha}$ *in Step (Ic) of Algorithm* 3.4 *is satisfied for some* $j \in \{1, \ldots, N-1\}$*. Then, the closed loop trajectory corresponding to the closed loop control* $\mu_N^{\mathcal{S}}(\cdot;\cdot)$ *resulting form Algorithm* 3.4 *satisfies the performance estimate* (15)*. If, in addition, the conditions of Theorem* 2.3*(ii) hold, then* $x_{\mu_N^{\mathcal{S}}}(\cdot)$ *behaves like a trajectory of an asymptotically stable system.*

Proof. The algorithm constructs the set $\mathcal{S}$. Since Step (Ic) is satisfied for some $j \in \{1, \ldots, N-1\}$ the assumptions of Proposition 3.3, i.e., Inequality (14) for $x_n = x_{\mu_N^{\mathcal{S}}}(\sigma(n))$, are satisfied. Hence, by Proposition 3.3, the performance Estimate (15) follows for the control sequence $\mu_N^{\mathcal{S}}(\cdot;\cdot)$. Last, similar to the proof of Theorem 2.3 given in [15, Theorem 7.6], standard direct Lyapunov techniques can be applied to conclude asymptotically stable behavior of the closed loop trajectory $x_{\mu_N^{\mathcal{S}}}(\cdot)$. □

Example 3.7. Consider Example 3.1 in the context of Theorem 3.6. Recall that using the Riccati based open loop control law $u_N(\cdot; x(n))$ in the standard MPC fashion $\mu_N(x(n)) := u_N(0; x(n))$ for horizon length $N = 3$ together with Theorem 2.3, there exist initial values for which $\alpha > 0$ cannot be guaranteed. Now, if we use Algorithm 3.4 instead, i.e. we allow for $m_n > 1$, then we obtain $\alpha > \overline{\alpha} = 0$ for each initial value $x_0 \in X$. In Fig. 2 we illustrated the impact of changing m_n on the difference $V_N(x_n) - V_N(x_{n+1})$ for $N = 3$. In particular, Fig. 2 (left) shows that there exist regions in the state space where the value function is increasing whereas $V_3(\cdot)$ in Fig. 2 (right) is always decreasing, i.e. all trajectories converge towards the origin.

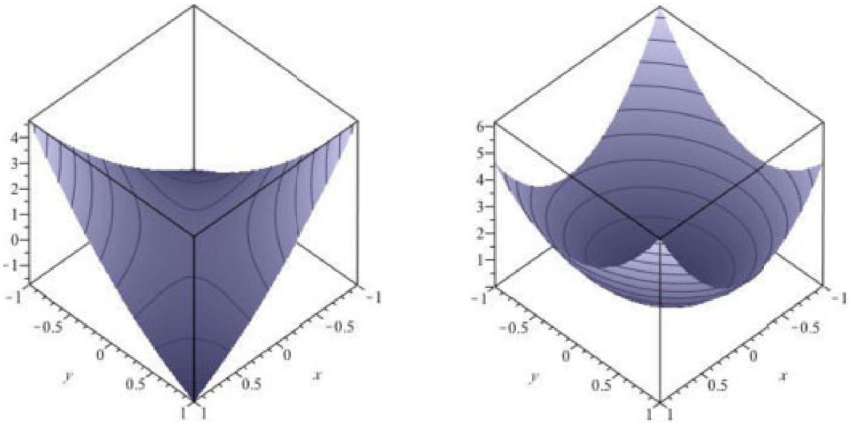

Fig. 2 Left: Difference $V_3(x_n) - V_3(x_{n+1})$ with $m_n = 1$. Right: Difference $V_3(x_n) - V_3(x_{n+1})$ for with $m_n = 2$.

Using $\alpha_1(\|x\|) = \alpha_3(\|x\|) = \|x\|^2 = \min_{u \in U} \ell(x,u) \leq V_N(x) = x^T P_N x \leq \overline{\lambda}(P)\|x\|^2 = \alpha_2(\|x\|)$ where $\overline{\lambda}$ denotes the largest Eigenvalue of the matrix P_N ensures the assumptions of Theorem 2.3(ii). Hence, applying Theorem 3.6 enables us to conclude asymptotic stability of the closed loop generated by Algorithm 3.4.

Example 3.7 indicates that checking the relaxed Lyapunov inequality less often may allow us to maintain stability for a reduced prediction horizon. We point out that in Example 3.7 condition (14) has been ensured in advance, i.e., before implementing a control input at the plant. Algorithm 3.4 may vary the number of open loop control values to be implemented during runtime at each MPC iteration. In particular, the system may stay in open loop for more than one sampling period in order to guarantee (14) to hold. Such a procedure may lead to severe problems in terms of robustness, cf. [4,23]. Hence, from a practical point of view it is preferable to close the control loop as often as possible, i.e., $m_n = 1$ for all $n \in \mathbb{N}_0$. Furthermore, for many applications stable behavior of the closed loop is observed for $m_n = 1$ even if stability cannot be guaranteed via (8), cf. Example 3.2. In the following, we deduce conditions which allow the control loop to be closed more often compared to Algorithm 3.4 while maintaining stability.

4 Robustification by closing the loop more often

If $m_n > 1$ is required in Step (I) of the proposed algorithm in order to ensure (14), the following methodology which has been proposed in [25, Sect. 4] allows us to prove the following: Given a certain condition, the degree of suboptimality $\overline{\alpha}$ in (14) is maintained for the updated control

$$\hat{u}_N(k; x_n) := \begin{cases} u_N(k; x_n), & k \leq j-1 \\ u_N(k-j; x_{u_N}(j; x_n)), & k \geq j \end{cases} \tag{16}$$

with $j \in \{1, \ldots, m_n - 1\}$ and where $\sigma(n) + j$ is added to the list $\mathcal{S}$. The theoretical foundation of such a method is given by the following result:

Proposition 4.1. *Let the open loop system $x_{u_N}(\cdot;\cdot)$ of* (6) *with initial value $x_n \in \mathbb{X}$ be given and inequality* (14) *hold for $u_N(\cdot;x_n)$, $\overline{\alpha} \in (0,1]$, and $m_n \in \{2,\ldots,N-1\}$. If the inequality*

$$\begin{aligned} &V_N(x_{u_N}(m_n-j;x_{u_N}(j;x_n))) - V_{N-j}(x_{u_N}(j;x_n)) \\ &\le (1-\overline{\alpha})\sum_{k=0}^{j-1}\ell(x_{u_N}(k;x_n),u_N(k;x_n)) - \overline{\alpha}\sum_{k=j}^{m_n-1}\ell(x_{u_N}(k-j;x_{u_N}(j;x_n)),u_N(k-j;x_{u_N}(j;x_n)))\end{aligned} \tag{17}$$

holds for some $j \in \{1,\ldots,m_n-1\}$, then the control sequence $u_N(\cdot;x_n)$ can be replaced by (16) *and the lower bound on the degree of suboptimality $\overline{\alpha}$ is locally maintained.*

Proof. In order to show the assertion, we need to show (14) for the modified control sequence (16). Reformulating (17) by shifting the running costs associated with the unchanged control to the left hand side of (17) we obtain

$$V_N(x_{u_N}(m_n-j;x_{u_N}(j;x_n))) - V_N(x_{u_N}(0;x_n)) \le -\overline{\alpha}\sum_{k=0}^{m_n-1}\ell(x_{\hat{u}_N}(k;x_n),\hat{u}_N(k;x_n))$$

which is equivalent to the relaxed Lyapunov inequality (14) for the updated control $\hat{u}_N(\cdot;x_n)$. □

Utilizing Proposition 4.1 in Algorithm 3.4 we see that only Steps (II) and (III) need to be changed and may take the following form:

```
Algorithm 4.2 (Modification of Algorithm 3.4)

(II) Set $\hat{n} := 1$ and $s := back(\mathcal{S})$. For $j = 1,\ldots,m_n$ do
    (a) Implement $\mu_N^{\mathcal{S}}(j-1;x_n) := u_N(j-1;x_n)$
    (b) Compute $u_N(\cdot;x_{u_N}(j;x_n))$, construct $\hat{u}_N(\cdot;x_n)$ according to (16) and compute $V_N(x_{\hat{u}_N}(m_n;x_n))$
    (c) If condition (17) holds:
            Update $u_N(\cdot;x_n) := \hat{u}_N(\cdot;x_n)$. If $j < m_n$ set $\mathcal{S} := (\mathcal{S}, s+j)$, $x_{n+\hat{n}} := x_{\mu_N^{\mathcal{S}}}(j;x_n)$ and $\hat{n} = \hat{n}+1$.
(III) Set $\mathcal{S} := (\mathcal{S}, s+m_n)$, $x_{n+\hat{n}} := x_{\mu_N^{\mathcal{S}}}(m_n;x_n)$, $n := n+\hat{n}$ and goto (I)
```

Due to the principle of optimality, the value of $V_{N-j}(x_{u_N}(j;x_n))$ in (17) is known in advance from $V_N(x_n)$. Hence, only $u_N(\cdot;x_{u_N}(j;x_n))$ and $V_N(x_{\hat{u}_n}(m_n;x_n))$ have to be computed. This result has to be checked with $V_N(x_n)$ for all $j \in \{1,\ldots,m_n-2\}$. Hence, the updating instant $\sigma(n)$ has to be kept in mind. We also like to stress the fact that condition (17) allows for a less fast decrease of energy along the closed loop, i.e., the case $V_N(x_{\hat{u}_N}(m_n;x_n)) \ge V_N(x_{u_N}(m_n;x_n))$ is not excluded in general which is illustrated by the following example.

Example 4.3. Consider Example 3.1 and the initial values $\tilde{x}_0 = (0,1)^T$ with prediction horizon $N = 3$. If we apply Algorithm 4.2 we obtain $V_N(\tilde{x}_0) = 5.109994744$ and $V_N(x_{\mu_N}(2;\tilde{x})) = 2.83461176$ which yields $\alpha = 0.5136$. If Algorithm 3.4 is employed we obtain $V_N(x_{u_N}(2;\tilde{x})) = 2.827656536$ which implies $\alpha = 0.5144$. Hence, Algorithm 4.2 accepts also deteriorations as long as the desired suboptimality degree is still maintained.
Taking $\overline{x}_0 = (1,0)^T$ as initial value shows that the impact of Algorithm 4.2 may also improve these key figures: In this case Algorithm 4.2 provides $V_N(\overline{x}) = 4.08117251$, $V_N(x_{\mu_N}(2;\overline{x})) = 0.96290399$ with $\alpha = 0.7733$ whereas Algorithm 3.4 gives us $V_N(x_{u_N}(2;\overline{x})) = 1.22718283$ with $\alpha = 0.7470$. For a further comparison we refer to the numerical experiments in the ensuing section.

Next, a counterpart to Theorem 3.6 based on Algorithm 4.2 instead of Algortihm 3.4 is established. This results allows us to verify that modifying and, thus, robustifying the algorithm still leads to the desired stability and performance properties. To this end, Proposition 4.1 is applied iteratively to show asymptotically stable behavior of the state trajectory generated by Algorithm 3.4:

Theorem 4.4. *Let a control system* (1) *with initial value $x_0 \in \mathbb{X}$ and $\overline{\alpha} \in (0,1]$ be given. Furthermore, suppose Algorithm* 3.4 *with the modification of Algorithm* 4.2 *is applied. Assume that the condition $\alpha \ge \overline{\alpha}$ in Step (Ic) of Algorithm* 3.4 *is satisfied for some $j \in \{1,\ldots,N-1\}$ for each iterate $n \in \mathbb{N}_0$. Then the closed loop trajectory corresponding to the closed loop control $\mu_N^{\mathcal{S}}(\cdot;\cdot)$ resulting from the used algorithm satisfies the performance estimate* (15) *from Proposition* 3.3*. If, in addition, the conditions of Theorem* 2.3*(ii) hold, then $x_{\mu_N^{\mathcal{S}}}(\cdot)$ behaves like a trajectory of an asymptotically stable system.*

Proof. The list $\mathcal{S}$ constructed by the algorithm contains all time instants at which the sequence of control values is updated by the MPC feedback law $\mu_N^{\mathcal{S}}(\cdot;\cdot)$. Hence, different to the basic Algorithm 3.4 the list $\mathcal{S}$ may not only contain time instants at which the relaxed Lyapunov inequality (14) holds.

Consider a second list $\widetilde{\mathcal{S}} = (\tau(n))_{n\in\mathbb{N}_0}$ constructed analogously to $\mathcal{S}$ from Algorithms 3.4, 4.2 but which is not updated in the modified Step (II) of Algorithm 4.2. Since Step (Ic) is satisfied for some $j \in \{1,\ldots,N-1\}$, Inequality (14) with $x_n = x_{\mu_N^{\mathcal{S}}}(\tau(n))$ and $m_n = \tau(n+1) - \tau(n)$ is satisfied. Using Proposition 4.1 ensures that this inequality is maintained despite of updates carried out by Algorithm 4.2. Hence, by Proposition 3.3, the performance estimate (15) follows for the control sequence $\mu_N^{\mathcal{S}}(\cdot;\cdot)$. Again, similar to the proof of Theorem 2.3 given in [15, Theorem 7.6], standard direct Lyapunov techniques can be used to show asymptotically stable behavior of the closed loop $x_{\mu_N^{\mathcal{S}}}(\cdot)$. □

An important conclusion of Proposition 4.1 and Theorem 4.4 is the following:

Corollary 4.5. *Consider the open loop system $x_{u_N}(\cdot;\cdot)$ of* (6), *the feedback law $\mu_N^{\mathcal{S}}(\cdot;\cdot)$, the closed loop trajectory x_n, $n \in \mathbb{N}_0$, of* (11) *with initial value $x_0 \in \mathbb{X}$ and a fixed $\overline{\alpha} \in (0,1]$ to be given. Moreover, suppose inequality* (14) *to hold for $u_N(\cdot;x_n)$, $\overline{\alpha} \in (0,1]$, and $m_n \in \{2,\ldots,N-1\}$. If* (17) *holds for all $j \in \{1,\ldots,m_n-1\}$ and all $n \in \mathbb{N}_0$, then* (9) *holds, that is the standard MPC feedback $\mu_N(\cdot)$ can be applied. If, in addition, the conditions of Theorem* 2.3*(ii) hold, then $x_{\mu_N}(\cdot)$ behaves like a trajectory of an asymptotically stable system.*

Proof. Follows directly from Theorem 4.4. □

The following example illustrates the impact of the modified algorithm, but it also indicates that the lower bound may still not be tight.

Example 4.6. In Example 3.7 stability can be shown if one allows for implementing more than than one element of the open loop sequence of control values, i.e., $m_n \geq 1$. Now we use Algorithm 4.2 instead of Algorithm 3.4, i.e. we allow for longer control horizons m_n and try to verify whether the control loop can be closed more often. And indeed, we obtain $\alpha > \overline{\alpha} = 0$ for each initial value $x_0 \in \mathcal{X}$ with $m_n = 1$, that is we can show stability for standard MPC according to Corollary 4.5. Moreover, taking Example 3.7 into account, this assertion holds for all $x \in X = \mathbb{R}^2$.

Yet, if the suboptimality bound is slightly increased to $\overline{\alpha} = 0.01$, the condition $\alpha \geq \overline{\alpha}$ in Step (Ic) is not satisfied for any $j \in \{1,\ldots,N-1\}$ for trajectories emanating from a subset of $\mathcal{X}$, cf. Fig. 3.

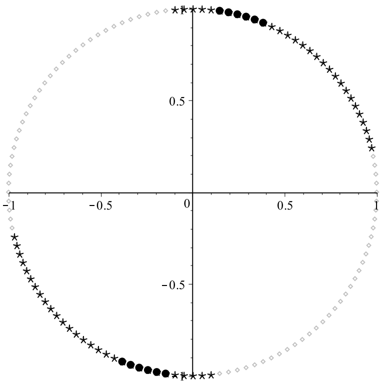

Fig. 3 Illustration of initial values (grey) where $\alpha < 0$ in (8) ($\star$) whereas $\alpha < \overline{\alpha} = 0.01$ in Step (Ic) of Algorithm 3.4 together with Algorithm 4.2 for at least one iterate n ($\bullet$).

Intuitively, one could guess that the performance of the Riccati based feedback law in our example is better than $\overline{\alpha} = 0.01$. In the following section, we address this issue indirectly as an outcome of an exit strategy required in Step (Id) of Algorithm 3.4, see also Remark 3.5.

5 Acceptable violations of relaxed Lyapunov inequality

Until now we have supposed that our relaxed Lyapunov inequality (14), i.e.,

$$\rho_i = \rho_i(\overline{\alpha}) := V_N(x_i) - V_N(x_{u_N}(m_i;x_i)) - \overline{\alpha} \sum_{k=0}^{m_i-1} \ell(x_{u_N}(k;x_i), u_N(k;x_i)) \geq 0, \tag{18}$$

is satisfied for some $m_i \in \{1,2,\ldots,N-1\}$. It is well known that (14) and, thus, $\rho_i \geq 0$ holds for sufficiently long horizon N, see [1, 12, 16, 19]. However, this is not necessarily true for short horizons N – even if the closed loop shows asymptotically stable behavior. Our basic Algorithm 3.4 allows us to cope with such a case via an exit strategy in Step (Id). As outlined in Remark 3.5, the length of the prediction horizon could be sufficiently increased in order to deal with this issue – a remedy which should be avoided due to its high computational costs. Here, we pursue a different approach based on the following result which uses an idea similar to the watchdog technique in nonlinear optimization, see [7].

Theorem 5.1. *Let a list $\mathcal{S} = (\sigma(0), \sigma(1), \ldots) \subseteq \mathbb{N}_0$ be given such that the sequence $(m_i)_{i\in\mathbb{N}_0}$ defined by $m_i := \mathcal{S}(i+1) - \mathcal{S}(i)$ is contained in $\{1, 2, \ldots, N-1\}$. In addition, for optimization horizon N and initial state $x_0 \in \mathbb{X}$, let the closed loop trajectory $(x_i)_{i\in\mathbb{N}_0}$ be generated by the feedback law $\mu_N^{\mathcal{S}}(\cdot;\cdot)$ according to* (11)*. Consider the open loop system $x_{u_N}(\cdot;\cdot)$ of* (6)*, $\overline{\alpha} \in (0,1]$, and the sequence $(\rho_i)_{i\in\mathbb{N}_0}$ from* (18) *to be given. Furthermore, suppose there exist $\alpha_1(\cdot)$, $\alpha_2(\cdot) \in \mathcal{K}_\infty$ such that $\alpha_2(\|x_i\|_{x^\star}) \geq V_N(x_i)$ and $\ell^\star(x_i) \geq \alpha_1(\|x_i\|_{x^\star})$ hold for all x_i, $i \in \mathbb{N}_0$. Then, convergence of $(s_n)_{n\in\mathbb{N}_0}$ with $s_n := \sum_{i=0}^n \rho_i$ ensures the convergence $x_i \to x^\star$ for i tending to infinity, i.e., $x_{\mu_N^{\mathcal{S}}}(\cdot)$ behaves like a trajectory of an asymptotically stable system. Furthermore, we have $V_N(x_i) \to 0$ for i approaching infinity.*

Proof. Plugging the definition of ρ_i into s_n yields

$$s_n = \sum_{i=0}^n \rho_i = \sum_{i=0}^n \left(V_N(x_i) - V_N(\underbrace{x_{u_N}(m_i;x_i)}_{=x_{i+1}}) - \overline{\alpha} \sum_{k=0}^{m_i-1} \ell(x_{u_N}(k;x_i), u_N(k;x_i)) \right)$$
$$= V_N(x_0) - V_N(x_{n+1}) - \sum_{i=0}^n \sum_{k=0}^{m_i-1} \ell(x_{u_N}(k;x_i), u_N(k;x_i)).$$

Since $V_N(x_{n+1}) \geq 0$ and $\ell(\cdot,\cdot)$ is positive the convergence of the sequence $(s_n)_{n\in\mathbb{N}_0}$ yields boundedness of each subtrahend. Using this assertion for the sum of the stage costs in combination with positivity of $\ell(\cdot,\cdot)$ and, thus, monotonicity of this summand with respect to the index i, implies that each summand of the stage costs $\sum_{k=0}^{m_i-1} \ell(x_{u_N}(k;x_i), u_N(k;x_i))$ tend to zero for i approaching infinity. Then, the assertion with respect to the closed loop trajectory can be concluded by

$$\|x_{\mu_N^{\mathcal{S}}}(n;x_0)\|_{x^\star} = \|x_{u_N}(k;x_i)\|_{x^\star} \leq \alpha_1^{-1}(\ell^\star(x_{u_N}(k;x_i))) \leq \alpha_1^{-1}(\ell(x_{u_N}(k;x_i), \mu_N^{\mathcal{S}}(k;x_{u_N}(k;x_i))))$$

for $n = \sigma(i) + k$. The latter also implies the assertion $V_N(x_i) \to 0$ in view of $\alpha_2(\|x_i\|_{x^\star}) \geq V_N(x_i)$. □

Theorem 5.1 ensures asymptotic stability but does not guarantee the desired performance specification. We like to point out that ρ_i, $i \in \mathbb{N}_0$, does not have to be positive. Indeed, even $V_N(\cdot)$ may increase along the closed loop trajectory before it finally converges to zero.

Analyzing the sequence $(s_n)_{n\in\mathbb{N}_0}$ and its limit more carefully leads to the following corollary which allows us to generalize Algorithm 3.4 by incorporating knowledge of the sequence $(s_n)_{n\in\mathbb{N}_0}$.

Corollary 5.2. *Let $\overline{\alpha} \in (0,1]$ be given and $s_n \geq 0$ hold for some $n \in \mathbb{N}_0$ with s_n from Theorem* 5.1*. Then, the closed loop trajectory $x_{\mu_N^{\mathcal{S}}}(\cdot)$ satisfies the following relaxed Lyapunov inequality at x_{n+1}*

$$V_N(x_{n+1}) + \overline{\alpha} \sum_{i=0}^n \sum_{k=0}^{m_i-1} \ell(x_{u_N}(k;x_i), u_N(k;x_i)) \leq V_N(x_0). \tag{19}$$

If $\lim_{n\to\infty} s_n \geq 0$ holds, then the suboptimality Estimate (15) *is satisfied.*

Proof. We obtain the stated relaxed Lyapunov inequality directly by inserting the definition of ρ_i from (18) into s_n and using the equivalence of the open and closed loop control $u_N(\cdot;x_i)$, $\mu_N^{\mathcal{S}}(\cdot;x_i)$ from (12) which allows us to replace $x_{u_N}(m_i;x_i)$ by x_{i+1}. In order to show (15) we have to establish $\overline{\alpha} V_\infty^{\mu_N^{\mathcal{S}}}(x_0) \leq V_N(x_0)$. To this end, we define

$$\tilde{s}_n := V_N(x_0) - \overline{\alpha} \sum_{i=0}^n \sum_{k=0}^{m_i-1} \ell(x_{u_N}(k;x_i), u_N(k;x_i)) = s_n + V_N(x_{u_N}(m_i;x_i)).$$

The fact that the range of $V_N(\cdot)$ and $\ell(\cdot,u)$ is contained in $\mathbb{R}_0^+$ for all $u \in U$ ensures that $(\tilde{s}_n)_{n\in\mathbb{N}_0}$ is a monotonically decreasing sequence which satisfies $\tilde{s}_n \geq s_n$ for all $n \in \mathbb{N}_0$. Since $\tilde{s}_n < 0$ for $n \in \mathbb{N}_0$ contradicts the positivity of $\lim_{n\to\infty} s_n$, we conclude $\tilde{s}_n \geq 0$ for all $n \in \mathbb{N}_0$. Hence, $(\tilde{s}_n)_{n\in\mathbb{N}_0}$ is monotonically decreasing and bounded and, thus, converges to $\tilde{s} \geq \lim_{n\to\infty} s_n \geq 0$. This yields

$$V_N(x_0) \geq \lim_{n\to\infty} \overline{\alpha} \sum_{i=0}^n \sum_{k=0}^{m_i-1} \ell(x_{u_N}(k;x_i), u_N(k;x_i)) = \overline{\alpha} V_\infty^{\mu_N^{\mathcal{S}}}(x_0),$$

i.e., the desired assertion. □

The key idea of Corollary 5.2 in comparison to Theorem 2.3(i) and Proposition 3.3 is to allow for intermediate increases within (8) or (14) for certain time instants n which corresponds to $\rho_n < 0$. Such a behavior is typical if the system is not minimal phase with respect to the cost functional and has to be accounted for if the cost functional cannot be adapted appropriately. We point out that the conditions of Theorem 5.1 and Corollary 5.2 with respect to $\lim_{n\to\infty} s_n$, unlike conditions (8) or (14), cannot be checked at runtime. Still, while the maximal $\overline{\alpha}$ satisfying (8) or (14) is locally a lower bound on the degree of suboptimality, cf. (14), the knowledge of s_n allows us to compute such a bound for a horizon of length $\sum_{i=0}^{n} m_i$, cf. (19). Note that the latter bound uses the stage costs as weighting factors.

Corollary 5.3. *Consider a feedback law $\mu_N^{\mathcal{S}}(\cdot;\cdot)$ with sequence $(m_n)_{n\in\mathbb{N}}$, $m_n \in \{1,2,\ldots,N-1\}$ for all $n \in \mathbb{N}$ as well as the corresponding closed loop trajectory x_n, $n \in \mathbb{N}_0$, of* (11) *with initial value $x_0 \in \mathbb{X}$ to be given. Furthermore, suppose $\overline{\alpha} \in (0,1]$ to be fixed and $(s_n)_{n\in\mathbb{N}_0}$ to be defined as in Theorem* 5.1. *Then we have the following:*
(i)

$$\{\overline{\alpha} \mid \overline{\alpha} \textit{ maximally satisfies } (19)\} = \frac{V_N(x_0) - V_N(x_{n+1})}{V_N(x_0) - V_N(x_{n+1}) - s_n}\overline{\alpha}. \tag{20}$$

(ii) If $(s_n)_{n\in\mathbb{N}_0}$ converges with $\lim_{n\to\infty} s_n = \theta$, then (15) *holds with degree of suboptimality $\alpha = \overline{\alpha}V_N(x_0)/(V_N(x_0) - \theta)$.*

Proof. To prove (i) we first reformulate the definition of s_n to obtain

$$\sum_{i=0}^{n}\sum_{k=0}^{m_i-1} \ell(x_{u_N}(k;x_i), u_N(k;x_i)) = \frac{V_N(x_0) - V_N(x_{n+1}) - s_n}{\overline{\alpha}} \tag{21}$$

where we used the equivalence of the open and closed loop control $u_N(\cdot;x_i)$, $\mu_N^{\mathcal{S}}(\cdot;x_i)$ from (12) to replace $x_{u_N}(m_i;x_i)$ by x_{i+1} for $i = 0,\ldots,n$. To obtain $\{\overline{\alpha} \mid \overline{\alpha}$ maximally satisfies (19)$\}$ we consider (19) to hold as an equality and solve for $\overline{\alpha}$. Now, we can use (21) to substitute the resulting denominator which gives us (20).

In order to obtain assertion (ii), we use the following fact shown in the proof of Theorem 5.1: If $(s_n)_{n\in\mathbb{N}_0}$ converges, then $V_N(x_i) \to 0$ as $i \to \infty$. Hence, taking n to infinity in (20) shows assertion (ii). □

Corollaries 5.2 and 5.3 give rise to a possible exit strategy in Step (Id): If s_n remains positive, then asymptotic stability and the desired performance bound can be guaranteed if the control loop is closed. One possible implementation of an algorithm based on Corollary 5.2 is the following:

```
Algorithm 5.4 (Extension of Algorithm 3.4)
Given state $x_0 := x$, $n = 0$, list $\mathcal{S} = (n)$, $N \in \mathbb{N}_{\geq 2}$, $\overline{\alpha} \in (0,1]$ and $s_n = 0$

(I) Set $j := 0$ and compute $u_N(\cdot; x_n)$ and $V_N(x)$. Do
    (a) Set $j := j + 1$ and compute $V_N(x_{u_N}(j; x_n))$
    (b) Compute $\rho_n$ from (18) and set $\rho_n(j) := \rho_n$
    (c) If $\rho_n(j) \geq 0$ holds: Set $m_n = j$ and goto (II)
    (d) If $j = N - 1$:
            If $s_n + \max_{j\in\{1,\ldots,N-1\}} \rho_n(j) < 0$: Print warning, set $m_n := 1$ and goto (II)
            Else: Set $m_n := \min\{j^* \mid j^* = \operatorname{argmin}_{j\in\{1,\ldots,N-1\}} -\rho_n(j)\}$, $\rho_n := \rho_n(m_n)$ and goto (II)
    while $\rho_n < 0$
(II) For $j = 1, \ldots, m_n$ do
        Implement $\mu_N^{\mathcal{S}}(j - 1; x_n) := u_N(j - 1; x_n)$
(III) Set $\mathcal{S} := (\mathcal{S}, back(\mathcal{S}) + m_n)$, $s_{n+1} := s_n + \rho_n(m_n)$, $x_{n+1} := x_{\mu_N^{\mathcal{S}}}(m_n; x_n)$, $n := n + 1$ and goto (I)
```

If $s_{n+1} = s_n + \max_{j\in\{1,\ldots,N-1\}} \rho_n(j) < 0$ the performance Estimate (15) cannot be guaranteed. Hence, even if $\lim_{n\to\infty} s_n \geq 0$ holds, the exit strategy cannot be used since this knowledge is not at hand at time instant n. Furthermore, we like to point out that Algorithm 5.4 coincides with Algorithm 3.4 for $n = 0$, i.e., the first MPC step. Note that this is the only time instant at which we may increase the optimization horizon N such that the presented stability proofs still hold. Hence, we can repeat Step (I) of the algorithm in order to ensure the desired relaxed Lyapunov inequality. In contrast to that, reducing N may be done at runtime of the proposed algorithms.

Remark 5.5. Using the definition of ρ_i in (18) and $V_N(x) \geq V_{N-k}(x)$ for $k \in \{0, \dots, N-2\}$ we obtain

$$\hat{\rho}_i := V_N(x_i) - V_{N-k}(x_{u_N}(m_i; x_i)) - \overline{\alpha} \sum_{k=0}^{m_i-1} \ell(x_{u_N}(k; x_i), u_N(k; x_i)) \geq \rho_i(\overline{\alpha}) = \rho_i$$

for $k \in \{0, \dots, N-2\}$. Hence, the telescope sum argument used in the proofs of Proposition 3.3 and Theorem 5.1 still holds if the horizon length N decreases monotonically along the closed loop.

Remark 5.5 gives rise to the following strategy: First, a large optimization horizon is chosen to avoid the startup problem. Then, the horizon can be reduced gradually along the closed loop provided the relaxed Lyapunov inequality (14) holds for the reduced horizon for all future time instants. In context of Theorem 5.1 one has to guarantee that s_n stays positive along the closed loop in order to reduce the optimization horizon. Note that an algorithm based on a quantity representing slack from proceeding steps has also been designed in [10, Theorem 1] in the context of varying optimization horizons.

Remark 5.6. For $\overline{\alpha} > 0$ an additional exit strategy is the following: if $0 < \alpha < \overline{\alpha}$ holds for suboptimality index α computed from Corollary 5.3(i), then stability or a certain performance bound may still be ensured. Additionally, Corollary 5.2 allows for checking whether the originally desired suboptimality estimate $\alpha \geq \overline{\alpha}$ is guaranteed again at a later time instant.

In order to check whether the control loop can be closed more often without loosing stability or violating the lower bound on the degree of suboptimality $\overline{\alpha}$, Corollaries 5.2 and 5.3 are employed directly. Note that this is possible since the sequence $(s_n)_{n \in \mathbb{N}_0}$ gives us an absolute value with respect to the desired decrease along the closed loop – contrary to Propositions 3.3 and 4.1 which have to be interpreted in terms of the stage costs $\ell(\cdot, \cdot)$. One possible implementation of the update check is the following:

Algorithm 5.7 (Modification of Algorithm 5.4)

```
 (II) Set $\hat{n} := 1$ and $s :=$ back($\mathcal{S}$). For $j = 1, \dots, m_n$ do
      (a) Implement $\mu_N^{\mathcal{S}}(j-1; x_n) := u_N(j-1; x_n)$
      (b) Compute $u_N(\cdot; x_{u_N}(j; x_n))$, construct $\hat{u}_N(\cdot; x_n)$ according to (16) and compute $V_N(x_{\hat{u}_N}(m_n; x_n))$
      (c) Compute $\rho_n$ from (18) with $u_N(\cdot; x_n)$ replaced by $\hat{u}_N(\cdot; x(n))$ and set $\rho_n(j) := \rho_n$
      (d) If $s_{n+\hat{n}-1} + \max_{j \in \{1,\dots,N-1\}} \rho_n(j) \geq 0$ holds:
              Update $u_N(\cdot; x_n) := \hat{u}_N(\cdot; x_n)$.
              If $j < m_n$: Set $\mathcal{S} := (\mathcal{S}, s+j)$, $s_{n+\hat{n}} := s_{n+\hat{n}-1} + \rho_n(j)$, $x_{n+\hat{n}} := x_{\mu_N^{\mathcal{S}}}(j; x_n)$ and $\hat{n} = \hat{n} + 1$.
(III) Set $\mathcal{S} := (\mathcal{S}, s+m_n)$, $s_{n+\hat{n}} := s_{n+\hat{n}-1} + \rho_n(m_n)$, $x_{n+\hat{n}} := x_{\mu_N^{\mathcal{S}}}(m_n; x_n)$, $n := n + \hat{n}$ and goto (I)
```

Example 5.8. Consider Example 3.1 one last time. As we have seen in Example 3.7 and 4.6 stability of the closed loop can be shown for initial values $x_0 \in \mathcal{X}$ by means of Propositon 3.3 for $m_n \geq 1$ and by Corollary 4.5 for $m_n = 1$. In Example 4.6 it has also been shown that one cannot guarantee the lower performance bound $\overline{\alpha} = 0.01$ by Theorem 4.4. Yet, we would expect a better performance of the Riccati based feedback law. And indeed, using Algorithm 5.7 together with Corollary 5.3 we obtain $\alpha = 0.52307$ for standard MPC ($m_n = 1$) with horizon length $N = 3$. In Fig. 4 the values of α resulting from Proposition 3.3 and Corollary 5.3 are displayed for different horizons N showing the improvement of Corollary 5.3. For reasons of completeness, we also displayed the performance results from [24, Sect. 6]. Note that while the latter hold for the entire state space $X = \mathbb{R}^2$, our results are only exact up to discretization accuracy. Still, the improvement of suboptimality bounds is significant and allows us to reduce the optimization horizon from $N = 5$ as shown in [24] to $N = 3$.
Unfortunately, we observe $s_{n+1} = s_n + \max_{j \in \{1,\dots,N-1\}} \rho_n(j) < 0$ along the closed loop for exactly the same initial values for which condition $\alpha \geq \overline{\alpha}$ in Step (Ic) of Algorithm 3.4 is not satisfied for at least one iterate n, cf. Fig. 4 (right).

Remark 5.9. The proposed algorithms and theoretic results in this paper are designed in a trajectory based manner. Therefore, these methods are not suited in order to ensure asymptotic stability or a desired suboptimality degree for a set of initial values $\mathcal{X} \subset \mathbb{X}$ – in contrast to the approaches presented in [13, 24]. Yet, incorporating the conditions presented in Proposition 4.1 or Corollary 5.2 in the methodology proposed in [13, Sect. 4] is possible. This topic will be subject to future research.

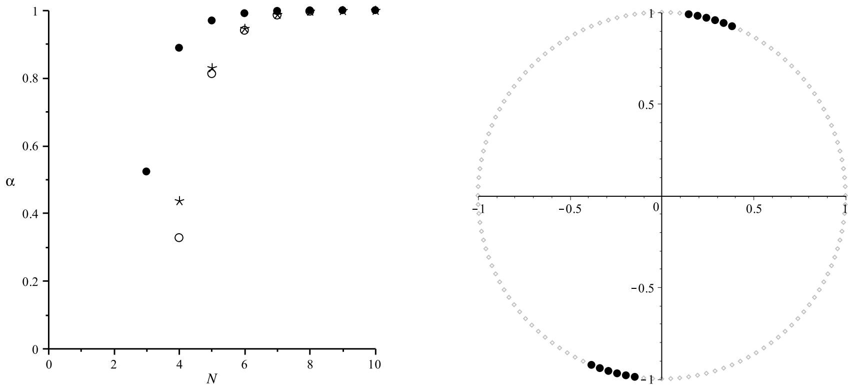

Fig. 4 Left: Illustration suboptimality bounds from [24, Sect. 6] (○), $\overline{\alpha}$ from Proposition 3.3 ($\star$) and α from Corollary 5.2 (●) for increasing horizon length N. Right: Illustration of initial values for which $s_{n+1} = s_n + \max_{j \in \{1,\ldots,N-1\}} \rho_n(j) < 0$ holds for at least one iterate n (●).

6 Conclusions

An algorithm based approach has been presented which ensures stability of the MPC closed loop without terminal constraints and/or costs. In particular, the proposed methodology allows for deducing stability and performance bounds for comparatively small prediction horizons. These results are based on structural properties of a relaxed Lyapunov inequality for the open loop which are computed a priori. Conditions which guarantee that this Lyapunov inequality is maintained despite closing the control loop at additional time instants have been derived in order to robustify the outcome of the corresponding algorithm. Last, a further improvement has been achieved by incorporating an accumulated quantity in the presented algorithms which reflects previous decrease in terms of the value function of the MPC problem. Doing so yields an exit strategy which often resolves problems occuring within our basic algorithms if the prediction horizon is chosen too small. Furthermore, enhanced performance estimates are obtained.

Acknowledgements This work was supported by the DFG Priority Project 1305 "Control Theory of Digitally Networked Dynamical Systems", grant number Gr1569/12, and the Leopoldina Fellowship Programme LPDS 2009-36.

References

[1] M. Alamir and G. Bornard, Stability of a truncated infinite constrained receding horizon scheme: the general discrete nonlinear case, Automatica **31**(9), 1353–1356 (1995).

[2] K. Worthmann, M. Reble, L. Grüne, and F. Allgöwer, The role of sampling for stability and performance in unconstrained nonlinear model predictive control, SIAM Journal on Control and Optimization, to appear (2014).

[3] T. Badgwell and S. Qin, A survey of industrial model predictive control technology, Control Eng. Pract. **11**, 733–764 (2003).

[4] R. Berber, Methods of Model Based Process Control, NATO ASI Series, Series E, Applied Sciences Vol. 293 (Kluwer Academic Publishers, 1995).

[5] R. Bitmead, M. Gevers, and V. Wertz, Adaptive Optimal Control: The Thinking Man's GPC., International Series in Systems and Control Engineering (Prentice-Hall, New York, 1990).

[6] E. Camacho and C. Bordons, Model Predictive Control, Advanced Textbooks in Control and Signal Processing, Vol. 24 (Springer-Verlag, London, 2004).

[7] R. Chamberlain, M. Powell, C. Lemarechal, and H. Pedersen, The watchdog technique for forcing convergence in algorithms for constrained optimization, Algorithms for constrained minimization of smooth nonlinear functions, Math. Program. Stud. **16**, 1–17 (1982).

[8] H. Chen and F. Allgöwer, Nonlinear Model Predictive Control Schemes with Guaranteed Stability, in: Nonlinear Model Based Process Control (Kluwer Academic Publishers, Dordrecht, 1999), pp. 465–494.

[9] G. De Nicolao and R. Scattolini, Stability and Output Terminal Constraints in Predictive Control, in: Advances in Model-Based Predictive Control, edited by D. Clarke (Oxford University Press, Oxford, 1994), pp. 105–121.

[10] P. Giselsson, Adaptive Nonlinear Model Predictive Control with Suboptimality and Stability Guarantees, in: Proceedings of the 49th Conference on Decision and Control (IEEE, Atlanta, GA, 2010), pp. 3644–3649.

[11] K. Graichen and A. Kugi, Stability and incremental improvement of suboptimal MPC without terminal constraints, IEEE Trans. Autom. Control **55**(11), 2576 –2580 (2010).

[12] G. Grimm, M. Messina, S. Tuna, and A. Teel, Model predictive control: for want of a local control Lyapunov function, all is not lost, IEEE Trans. Autom. Control **50**(5), 546–558 (2005).

[13] L. Grüne, Analysis and design of unconstrained nonlinear MPC schemes for finite and infinite dimensional systems, SIAM J. Control Optim. **48**, 1206–1228 (2009).

[14] L. Grüne, NMPC Without Terminal Constraints, in: Proceedings of the IFAC Conference on Nonlinear Model Predictive Control 2012 (NMPC'12) (University of Bayreuth, Bayreuth, 2012), pp. 1 – 13.

[15] L. Grüne and J. Pannek, Nonlinear Model Predictive Control: Theory and Algorithms, 1st edition, Communications and Control Engineering (Springer, Berlin, Heidelberg, 2011).

[16] L. Grüne, J. Pannek, M. Seehafer, and K. Worthmann, Analysis of unconstrained nonlinear MPC schemes with varying control horizon, SIAM J. Control Optim. **48**(8), 4938–4962 (2010).

[17] L. Grüne, S. Sager, F. Allgöwer, H. Bock, and M. Diehl, Predictive Planning and Systematic Action – On the Control of Technical Processes, in: Production Factor Mathematics, edited by M. Grötschel, K. Lucas, and V. Mehrmann (Springer, Berlin, Heidelberg, 2010), pp. 9–37.

[18] K. Ito and K. Kunisch, Receding horizon optimal control for infinite dimensional systems, ESAIM Control Optim. Calc. Var. **8**, 741–760 (2002).

[19] A. Jadbabaie and J. Hauser, On the stability of receding horizon control with a general terminal cost, IEEE Trans. Autom. Control **50**(5), 674–678 (2005).

[20] S. Keerthi and E. Gilbert, Optimal infinite-horizon feedback laws for a general class of constrained discrete-time systems: stability and moving-horizon approximations, J. Optim. Theor. Appl. **57**(2), 265–293 (1988).

[21] E. C. Kerrigan and J. M. Maciejowski, Invariant Sets for Constrained Nonlinear Discrete-Time Systems with Application to Feasibility in Model Predictive Control, in: Proc. 39th IEEE Conference on Decision and Control (IEEE, Orlando, 2000).

[22] B. Lincoln and A. Rantzer, Relaxing dynamic programming, IEEE Trans. Autom. Control **51**(8), 1249–1260 (2006).

[23] L. Magni and R. Scattolini, Robustness and Robust Design of MPC for Nonlinear Discrete-Time Systems, in: Assessment and Future Directions of Nonlinear Model predictive control, edited by R. Findeisen, F. Allgöwer, and L. T. Biegler, Lecture Notes in Control and Inform. Sci. Vol. 358 (Springer, Berlin, 2007), pp. 239–254.

[24] V. Nevistić and J. A. Primbs, Receding Horizon Quadratic Optimal Control: Performance Bounds for a Finite Horizon Strategy, in: Proceedings of the European Control Conference (IEEE, Brussels. 1997).

[25] J. Pannek and K. Worthmann, Reducing the Prediction Horizon in NMPC: An Algorithm Based Approach, in: Proceedings of the 18th IFAC World Congress (IFAC, Milan, Italy, 2011), pp. 7969–7974.

[26] J. Primbs and V. Nevistić, Feasibility and stability of constrained finite receding horizon control, Automatica **36**, 965–971 (2000).

[27] J. B. Rawlings and D. Q. Mayne, Model Predictive Control: Theory and Design (Nob Hill Publishing, Madison, 2009).

[28] J. Shamma and D. Xiong, Linear nonquadratic optimal control, IEEE Trans. Autom. Control **42**(6), 875–879 (1997).

[29] R. Soeterboek, Predictive Control: A Unified Approach (Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1992).

[30] S. E. Tuna, M. J. Messina, and A. R. Teel, Shorter Horizons for Model Predictive Control, in: Proceedings of the American Control Conference (IEEE, Minneapolis, Minnesota, 2006).